

Trabajo Fin de Master
Máster en Organización Industrial y Gestión de
Empresas

**Optimización de la mano de obra en una línea mixta
de ensamblado. Aplicación en la industria del
automóvil.**

Autor: Álvaro Planes Rodríguez

Tutor: Pedro Luís Gonzalez Rodríguez

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

Sevilla, 2019



Trabajo Fin de Máster
Máster en Organización Industrial y Gestión de Empresas

Optimización de la mano de obra en una línea mixta de ensamblado. Aplicación en la industria del automóvil.

Autor:

Álvaro Planes Rodríguez

Tutor:

Pedro Luís Gonzalez Rodríguez

Profesor titular

Dpto. de Organización Industrial y Gestión de Empresas I

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2019

Trabajo de Fin de Máster: Optimización de la mano de obra en una línea mixta de ensamblado. Aplicación en la industria del automóvil.

Autor: Álvaro Planes Rodríguez

Tutor: Pedro Luís Gonzalez Rodríguez

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2019

El Secretario del Tribunal

Resumen

El objetivo de este trabajo es estudiar la minimización de la fuerza de trabajo en una línea de ensamblado de modelo mixto y el desarrollo de una aproximación heurística para resolverlo. La heurística se prueba en un caso industrial real y en diferentes instancias teóricas. El trabajo es una mejora del proceso propuesto por un artículo de investigación.

Abstract

The objective of this work is to study the workforce minimization in a mixed model assembly line and the development of a heuristic approach to solve it. The heuristic is tested in a real case industry as well as in different theoretical instances. The work is an improvement of the procedure proposed in a research article.

Resumen	vii
Abstract	ix
Índice	xxi
Índice de Figuras	xiv
Glosario	xvi
Notación	xviii
1. Introducción, objeto del trabajo y revisión de la literatura	17
1.1. Introducción	17
1.2. Objeto del trabajo	18
1.3. Estructura del Trabajo	19
2. Revisión de la literatura	21
2.1. Introducción a la revisión	21
2.2. Estado del arte	22
3. Descripción del problema y desarrollo de la heurística de asignación	27
3.1. Caso industrial a estudiar	27
3.2. Diseño y desarrollo de la heurística de asignación de trabajadores.	31
4. Experimentación	37
5. Análisis de los resultados	43
5.1. Presentación de los resultados	43
5.2. Análisis de los resultados	47
6. Conclusiones	55
Referencias	57
Anexo	61

ÍNDICE DE FIGURAS

Figura 1. Línea de ensamblado de automóviles Ford a principios del siglo XX.	10
Figura 2. Línea de ensamblado completamente automatizada.	10
Figura 3. Línea de ensamblado mixta de motores de automóvil.	11
Figura 4. Diferencia entre producción de tipo push y pull	13
Figura 5. Comparativa revisión literatura y posicionamiento del objeto de estudio del TFM.	15
Figura 6. Cabecera artículo “Workforce minimization for a mixed-model assembly line” de Battaia <i>et al.</i> (2014)	19
Figura 7. Secuencia de producción definida para las tres posibles versiones de los productos existentes. V8, V10 y V12. (Figura del artículo de Battaia <i>et al.</i> (2014))	20
Figura 8. Ejemplo de asignación de tareas a una estación de ensamblado.	20
Figura 9. Movimiento de trabajadores de una estación a otra para ayudar a completar la tarea. (Figura del artículo de Olga Battaia <i>et al.</i> (2014))	21
Figura 10. Ejemplo de organización de una secuencia inicial en una parte de la línea y la distribución de los trabajadores.	21
Figura 11. Ejemplo de organización de una secuencia optimas en una parte de la línea y la distribución de los trabajadores.	22
Figura 12. Diagrama de flujo de la búsqueda local simple	24
Figura 13. Diagrama de flujo de un recorrido simulado.	24
Figura 14. Diagrama de flujo de la heurística desarrollada	25
Figura 15. Listado de tiempos agrupados por estación del caso industrial.	25
Figura 16. Listado de tiempo de las operaciones realizadas en una estación de ensamblaje.	27
Figura 17. Ejemplo de generación de tiempos en Excel.	29
Figura 18. Tiempos generados para 40 estaciones.	30
Figura 19. Tiempos generados para 60 estaciones.	30
Figura 20. Tiempos generados para 80 estaciones	31
Figura 21. Resultados del caso industrial.	32
Figura 22. Información de ejecución del caso industrial.	33
Figura 23. Resultados del caso de 40 estaciones	35
Figura 24. Información de ejecución del caso de 40 estaciones.	35
Figura 25. Resultados del caso de 60 estaciones	36
Figura 26. Información de ejecución del caso de 60 estaciones.	36
Figura 27. Resultados del caso de 80 estaciones	37
Figura 28. Información de ejecución del caso de 80 estaciones.	37
Figura 29. Figura comparativa de los casos.	38
Figura 30. Evolución de la heurística en términos de operarios usados en la secuencia.	39

Figura 31. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso industrial.	39
Figura 32. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 40 estaciones.	40
Figura 33. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 60 estaciones.	41
Figura 34. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 80 estaciones.	41
Figura 35. Tiempo en segundos invertido por estación	41
Figura 36. Takt time por paquete de estaciones y diferencia con takt time de la línea para caso industrial.	42
Figura 37. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 40 estaciones.	42
Figura 38. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 60 estaciones.	43
Figura 39. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 80 estaciones.	43
Figura 40. Tiempo invertido en resolver los casos industriales.	46

Notación

Takt Time

Ritmo de producción objetivo de la línea

Modelo Mixto

Varios modelos de producto en una misma línea

1 INTRODUCCIÓN, OBJETO DEL TRABAJO Y REVISIÓN DE LA LITERATURA

1. 1. Introducción

Una línea de ensamblado es un proceso de fabricación en el cual las partes, que de forma general son intercambiables, son añadidas a una base para conseguir un producto final de forma que avanza la materia prima por la línea de estación de trabajo en estación hasta conseguir el producto final.



Figura 1. Línea de ensamblado de automóviles Ford a principios del siglo XX.

Mediante el movimiento mecánico del producto semiensamblado a lo largo de la línea y moviendo el producto de una estación a otra se consigue que el producto final sea ensamblado más rápido y con menor mano de obra necesaria, al contrario que el modelo tradicional que implica mover las piezas hacia el producto que va a ser ensamblado.

Las líneas de ensamblado son diseñadas para la organización secuencial de trabajadores, herramientas o máquinas y partes del producto final. El movimiento de los trabajadores es minimizado al máximo posible. Todas las piezas que son instaladas en el producto final son servidas a la línea a través de maquinaria auxiliar, no dependiendo del operario de montaje de la línea. Los trabajadores son asignados una serie de operaciones lo más simple posible, que se repetirán de forma cíclica.



Figura 2. Línea de ensamblado completamente automatizada.

Las líneas de ensamblado tienen varios principios fundamentales: El posicionamiento de las herramientas y las personas en la secuencia de operación de montaje se hace de tal forma que cada componente viaje el mínimo posible durante el proceso de montaje.

Se deben usar posicionadores de forma que las piezas ensambladas y herramientas se depositen siempre en el mismo sitio, de la forma más accesible posible, y de forma que este avance hacia la siguiente estación de ensamblado por sí misma. Y, por último, la entrega de piezas de una estación a otra debe cumplir unos criterios de conveniencia de entrega en términos de distancia y temporales.

Una línea de ensamble de modelos mixtos es una línea de producción en la cual se puede producir una variedad de modelos diferentes de los mismos productos en las mismas estaciones. En cada estación de trabajo se realizan un cierto conjunto de operaciones de trabajo de ensamble, garantizando que las estaciones cuentan con la suficiente flexibilidad para poder realizar sus distintas operaciones en los diferentes modelos.



Figura 3. Línea de ensamblado mixta de motores de automóvil.

Este tipo de líneas son típicamente usadas para lograr el ensamble final de automóviles, aviones, y electrónica de consumo. El balanceo de líneas de modelos mixtos es igual que el balanceo en una línea convencional, mientras que la carga de materia prima a la línea mixta es más compleja debido a que cada modelo tiene un contenido diferente de trabajo, lo cual genera diferentes tiempos de montaje en las estaciones.

El éxito de las líneas de ensamblado reside en la alta optimización del uso de los recursos humanos, es por ello por lo que gran parte de las líneas de fabricación en serie son modeladas y estudiadas de acuerdo a introducir todo tipo de parámetros para conseguir los modelos más acordes a la realidad, y el estudio de investigación operativa para garantizar un funcionamiento correcto y eficiente de la misma, es donde tiene cabida este trabajo de fin de master.

1. 2. Objeto del trabajo

El objeto de este trabajo consiste en el estudio del problema de optimización de la mano de obra en una línea de ensamblado mediante el desarrollo de una heurística en el lenguaje de programación Python para la minimización de la mano de obra por estación y en total de la línea. Se toma como base el caso industrial propuesto en el artículo “Workforce minimization for a mixed-model assembly line” de Olga Battaia *et al.* (2014), partiendo de la misma batería de datos. Se fija como objetivos específicos del trabajo:

- Objetivo específico 1: Revisar el estado del arte y sacar conclusiones del mismo.
- Objetivo específico 2: Plantear el caso de estudio y desarrollar una heurística capaz de resolver el problema planteado, con el objetivo de minimizar el número de trabajadores en la línea de ensamblado.
- Objetivo específico 3: Comparación de los resultados del trabajo con los resultados obtenidos en el caso expuesto en la revisión de la literatura.
- Objetivo específico 4: Testear el funcionamiento en instancias similares de mayor tamaño para evaluar su funcionamiento.

1.3. Estructura del Trabajo

En el primer apartado del trabajo se va a desarrollar partiendo de una explicación de los conceptos básicos de ensamblado y en que consiste explícitamente un modelo mixto de línea de ensamblado, e indicando los objetivos habituales.

En el segundo apartado se hace una completa revisión de la literatura, estableciendo un cuadro comparativo y enmarcando el trabajo propuesto con la relación a la literatura existente.

En el tercer apartado, se describe de forma amplia el problema a estudiar basado en el caso industrial real en la industria automovilística. De forma detallada se expone el problema y condiciones de contorno para tener una idea concisa del tema que se pretende abordar. A partir de esto, se explica la heurística generada para la resolución del problema.

En el cuarto apartado se exponen el estudio computacional del caso industrial y los demás casos ficticios basados en el caso real para testear el funcionamiento adecuado para cualquier escala de la heurística, exponiendo la batería de datos inicial.

En el quinto apartado, se analizarían los resultados obtenidos y se establecen comparativas entre los diferentes casos industriales y fases del proceso de búsqueda de la solución más óptima.

En el sexto y último apartado se sacarán unas conclusiones generales del resultado, y de la cumplimentación de los diferentes objetivos marcados para el trabajo.

2 REVISIÓN DE LA LITERATURA

2.1. Introducción a la revisión

Entre todos los retos afrontados en la fabricación moderna, donde los altos niveles de demanda y calidad son los pilares del mercado, la reducción de costes toma la dirección de búsqueda hacia la minimización de la mano de obra utilizada en estas. La investigación operativa esta en el camino de intentar alcanzar el objetivo de dar respuestas al dimensionamiento y organización de los recursos para producir en masa en las líneas de ensamblado de modelo mixto.

Buena muestra de la búsqueda de la máxima optimización de recursos productivos es el uso de las diferentes técnicas de equilibrado de líneas existente, que tienen como objetivo la distribución eficiente de los trabajadores y tareas a lo largo de una línea de ensamblado, así como garantizar la distribución uniforme de de carga de trabajo entre los operarios. Estos sistemas están basados y tienen su referencia en las líneas de producción a pulso, en las cuales la cadena funciona de manera perfectamente coordinada con el mercado, donde solo se produce a petición del cliente, y este el que crea la orden de fabricación y tira de la línea productiva.

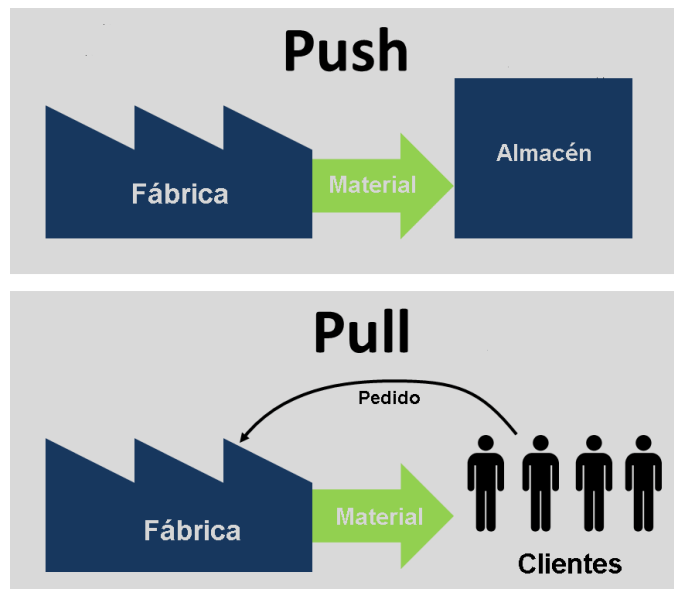


Figura 4. Diferencias entre producción de tipo push y pull.

Para la optimización eficiente de la carga de trabajo en una línea productiva, se dividen el total de operaciones por el número de estaciones en la línea. Después, se asigna el número de recursos productivos humanos a la estación que van a realizar las tareas de la misma. En las líneas de ensamblado manuales de modelo mixto existe la posibilidad de producir modelos variados de un mismo producto, siendo estas variantes o modelos.

Una vez este trabajo se sitúa en el campo de la asignación eficiente de recursos a una línea de producción en serie, se realiza un extensivo análisis de la literatura en este campo. Se pueden encontrar varias líneas de investigación definidas:

- Tomando como criterio la optimización del uso de recursos y costes
- Según niveles de habilidad y agilidad de los trabajadores
- Según si se organiza en múltiples periodos y los trabajadores son fijos o temporales.

2.2. Estado del arte

Se puede diferenciar entre tendencias en la línea de investigación para la organización y optimización de líneas de ensamblado que se puede organizar diferenciando las diferentes restricciones aplicadas al estudio del problema que se afronta en la investigación operativa de las líneas de ensamblado. A continuación, se diferencia según el tipo de restricción tenido en cuenta en cada uno de los estudios llevados a cabo:

Una de estas restricciones se centra en la función objetivo, más concretamente en el criterio de optimización de la misma. En la misma se pueden encontrar diversos enfoques ya sea marcando como objetivo el coste, el tiempo, el número de operarios, que de forma general todos tienden a indicar de una forma u otra el funcionamiento más eficaz de la línea de ensamblado.

El criterio de optimización más utilizado es el que reduce el uso de recursos. Como por ejemplo en el caso de los problemas de equilibrado de una línea de ensamblado con asignación de recursos productivos humanos está siendo estudiado en profundidad actualmente. Ver por ejemplo Miralles *et al.* (2008), Blum y Miralles (2011), Moreira y Costa (2013) y Mutlu *et al.*, (2013).

El problema de asignación de la fuerza productiva con un criterio de minimización total del coste, en el cual un trabajador altamente cualificado puede sustituir a uno no cualificado, pero no viceversa, ha sido estudiado por Özgüven y Sungur (2013) y Sungur y Yavuz (2014).

La asignación de recursos productivos humanos no idénticos a las máquinas de la línea en U usando como criterio de minimización de número de operarios, teniendo que el número de recursos productivos humanos se ha minimizado, Nakade y Nishiwaki (2008). El estudio de una heurística para el problema de asignación de recursos productivos humanos para una línea en U de ensamblado por Shewchuk (2008).

La resolución mediante pseudo algoritmos de la minimización de trabajadores en una línea de ensamblado con ritmo fijo y el modelado del problema NP, centrado en un problema de asignación de recursos para un caso particular (ver Delorme *et al.*, 2018).

El estudio del problema que distingue dentro de la fuerza de trabajo los permanentes y los temporales fue estudiado. Introduciendo grupos de tareas incompatibles que imposibilitado el cambio de recursos productivos humanos entre estaciones. Se fijó como objetivo la minimización del número de recursos productivos humanos temporales (ver Corominas *et al.*, 2008).

En un artículo de investigación se trata la conversión de una línea de ensamblado convencional en un sistema de fabricación por células, en el cual se tomó como función objetivo la minimización del número de recursos productivos humanos (ver Yu *et al.*, 2013).

El uso heurísticas basadas en búsquedas de tipo proyección para resolver el problema de asignación de recursos productivos humanos con un número fijo de recursos productivos humanos y fijando como objetivo de maximización el ritmo o tasa de producción (ver Borba y Ritt, 2014).

En el caso que atañe al estudio realizado, adquiere una nueva orientación, dejando definida las tareas a realizar en cada una de las estaciones y modificando el número de recursos productivos humanos asignados a las estaciones, permitiéndose moverse entre estaciones de forma dinámica.

El problema de cómo distribuir la carga de trabajo entre las diferentes estaciones ha sido ampliamente estudiado. Debido a las desigualdades de los tiempos de proceso para las diferentes variantes de producto, el balance perfecto es bastante complicado de encontrar. Esto resulta en tiempos de esperas e ineficiencia en el uso de la fuerza productiva (ver Venkatesh, 2008).

Otra de las líneas de investigación ampliamente estudiadas y tratadas en la literatura referente a las líneas de ensamblado se refiere a las restricciones impuestas por los niveles de habilidad y agilidad de los trabajadores en una línea de ensamblado. En la literatura se trata la diversidad del conjunto productivo incluyendo casos como la discapacidad, etc.

Numerosas publicaciones en el campo de la asignación de la fuerza productiva o de planificación de la misma toman como base que los recursos productivos humanos tienen diferentes niveles de habilidad, lo que influye en el tiempo de finalización de las tareas asignadas, y por tanto en el rendimiento de la línea productiva. Algunos artículos que han estudiado la situación en esta línea son: Nakade y Ohno (1999), Miralles *et al.* (2007), Niemi (2009), Araújo *et al.* (2012), Othman *et al.* (2012), Fowler *et al.* (2008) y Costa y Miralles (2009).

Lo más avanzado en el campo de los problemas de planificación de la fuerza productiva asume que los recursos productivos humanos tienen diferentes niveles de habilidad (ver De Bruecker *et al.*, 2015).

El problema de líneas de ensamblado con recursos productivos humanos convencionales y con discapacidad ha sido estudiado. En este propone una serie de heurísticas para minimizar el número de estaciones, mientras se procura integrar en la línea de ensamblado un número determinado de recursos productivos humanos con discapacidad (ver Moreira *et al.*, 2015).

Una heurística para la planificación de recursos productivos humanos con alta formación e idénticos en sistemas de producción síncronos se propone. Se fija como criterio de minimización, incluyendo en la función objetivo los costes de la fuerza productiva y los asociados al horizonte temporal de planificación (ver Vairaktarakis *et al.*, 2002).

El problema de asignación de recursos productivos humanos con altas habilidades y permanentes junto con recursos productivos humanos de bajas habilidades y temporales fue estudiado. En este problema, cada tipo de recursos productivos humanos en cada periodo de tiempo es una variable de decisión, que tiene como objetivo minimizar el salario total pagado y los costes de contratación y despido (ver Techawiboonwong *et al.*, 2006).

Otro de los retos a modelar que se encuentran en la literatura refiere al efecto del aprendizaje en el rendimiento de los trabajadores, siendo una parte esencial a la hora de determinar la duración de las tareas, que debe ir disminuyendo a lo largo del tiempo de operación de la línea de ensamblado.

Un gran número de trabajos estudian el efecto del aprendizaje y la agilización de los recursos productivos humanos con el tiempo como Grosse y Glock (2012), Grosse *et al.* (2013), Boenzi *et al.* (2015) y Hewitt *et al.* (2015).

En algunos casos concretos estudiados en la literatura puede ser interesante suponer un rendimiento completamente igual para cualquier tipo de operario, haciendo que la el tiempo de operación solo dependa del número de trabajadores que estén realizando dicha operación.

Se pueden encontrar investigaciones en la cual se asume que el rendimiento de los recursos productivos humanos realizando la misma tarea es el mismo para todos, y que el tiempo de realización de una tarea depende solamente del número de recursos productivos humanos asignados a esa tarea (ver Akagi *et al.*, 1983, Wilson (1986) y Lutz y Davis (1994)).

El estudio del trabajo en parejas en una línea de ensamblado manual referida a un caso industrial específico, donde el aprendizaje de la pareja está siendo observado (ver Peltokorpi *et al.*, 2018).

Se puede diferenciar los estudios que tienen a la línea de la organización en múltiples periodos y en función de si los trabajadores son fijos o temporales, haciendo que la adaptación de los recursos humanos a las variaciones en la línea tenga una serie de restricciones en el coste, penalizando de manera mayor el despido de trabajadores de tipo fijo respecto a temporales, asimilando a la situación real de la empresa.

La asignación de tareas en paralelo de predecesores idénticos a distintos recursos productivos humanos fue introducida en planificación con el nombre de tareas maleables (ver Blazewicz *et al.*, 2006, Fan *et al.*, 2012, Jansen y Zhang (2012) y Sadykov (2012)).

Por último, es esencial nota que algunos artículos en esta línea de investigación tienden a dirigirse según el método de resolución usado para encontrar la solución del mismo.

Un algoritmo de heurística constructiva para un problema de múltiples periodos de asignación de recursos productivos humanos en una línea de ensamblado de automóviles de modelo mixto fue propuesto por Karabak *et al.*, 2011.

Una programación matemática del modelo y dos metaheurísticas para la asignación de trabajadores de carácter idéntico. Una extensiva investigación respecto a la literatura fue propuesta por Battaia *et al.*, 2013.

Tipo	Trabajo	Autor	Año
Literatura	Revisión Literatura equilibrado de líneas	Battaia y Dolgui	2013
Problema	Asignación de tareas entre estaciones	Venkatesh	2008
	Asignación de tareas en paralelo de predecesor idéntico (tareas maleables)	Blazewicz	2006
		Fan	2012
		Jansen y Zhang	2012
		Sadykov	2012
		Delorme <i>et al.</i>	2018
	Minimización numero trabajadores en línea de producción con avance fijo.	Delorme <i>et al.</i>	2018
	Equilibrado de líneas	Dolgui y Proth	2010
	Optimización línea con asignación de trabajadores usando takt time como criterio	Miralles	2008
		Blum y Miralles	2011
		Moreira y Costa	2013
		Mutlu	2013
		Özgüven y Sungur	2013
		Sungur y Yavu	2014
		Moreira	2015
	Asignación de trabajadores en una línea en U.	Shewchuk	2008
	Efecto del aprendizaje y agilización de trabajadores.	Grosse y Glock	2012
		Grosse	2013
		Boenzi	2015
		Hewitt	2015
Efecto del rendimiento del trabajador en las tareas.		Akagi	1983
		Wilson	1986
		Lutz y Davis	1994
Heurística	Heurística tipo beam con trabajadores fijos y objetivo minimizar temporales.	Borba y Ritt	2014
	Asignación de trabajadores permanentes y temporales. Minimización de temporales.	Corominas	2008
	Heurística constructiva asignación trabajadores línea ensamblado automóviles mixta	Karabak	2011
	Asignación trabajadores permanentes y hábiles con temporales e inhábiles. Objetivo minimizar salario total pagado y costes contratación/despido.	Techawiboonwong	2006
	Heurística trabajadores alta formación sistemas síncronos. Minimizar mano obra.	Vairaktarakis	2002
	Desarrollo de dos metaheurísticas para una línea de ensamblado.	Peltokorpi <i>et al.</i>	2018
	Optimización de fabricación en células. Objetivo minimización número trabajadores.	Yu	2013
	Heurística asignación trabajadores en línea ensamblado modelo mixto. Objetivo minimizar número trabajadores.	Battaia, Delorme y Dolgui	2014
	Metaheurística asignación trabajadores en línea ensamblado modelo mixto. Objetivo minimizar número trabajadores.	TFM	2019

Figura 5. Comparativa de la revisión de la literatura y líneas de investigación.

En diferentes colores se marcan las líneas de investigación que sigue cada artículo, teniendo en naranja los que toman como criterio la optimización del uso de recursos y costes, en morado los que según niveles de habilidad y agilidad de los trabajadores se organizan, y por último en blanco los que se organiza en múltiples periodos y los trabajadores son fijos o temporales.

En líneas generales, se puede apreciar un gran interés por el tema planteado, ya que son numerosos los trabajos en temas similares, siendo muchos de ellos recientes. Se observa también que de forma genérica, todos los estudios tienden a intentar optimizar el número de trabajadores totales en la línea, ya sea cuantificando el resultado directamente midiendo su número, o asociándolos a unos costes que derivarían de su uso. Por otra parte, se tienen en cuenta factores como la habilidad o el rendimiento de los trabajadores. Otro de los aspectos a valorar es si la planificación es multiperiodo o si los trabajadores son fijos o temporales.

En el presente trabajo se va a centrar en el análisis de un caso “Workforce minimization for a mixed-model assembly line” planteado por Battaña *et al.*, 2014 y posteriormente descrito en el capítulo 3, para el que se propodnrá un método de resolución.

3 DESCRIPCIÓN DEL PROBLEMA Y DESARROLLO DE LA HEURÍSTICA DE ASIGNACIÓN

3.1 Caso industrial a estudiar

La idea de desarrollar un algoritmo que optimice el número de trabajadores en una línea de ensamblado de modelo mixto surge del estudio “Workforce minimization for a mixed-model assembly line” de Battaïa *et al.*, 2014, donde se proponen diferentes heurísticas para la asignación de tareas y operarios de la manera más eficiente. El problema se describe de la siguiente forma:



Workforce minimization for a mixed-model assembly line in the automotive industry



Olga Battaïa^a, Xavier Delorme^a, Alexandre Dolgui^{a,*}, Johannes Hagemann^b, Anika Horlemann^b, Sergey Kovalev^{c,1}, Sergey Malyutin^a

^a LIMOS, UMR CNRS 6158, Ecole des Mines de Saint-Etienne, 158 cours Fauriel, 42023 Saint-Etienne Cedex 2, France

^b MBtech Group GmbH & Co. KGaA, Germany

^c INSEEC Business School ECE Lyon, 19 Place Tolozan, 69001 Lyon, France

ARTICLE INFO

Article history:
Received 15 April 2014
Accepted 28 May 2015
Available online 19 June 2015

Keywords:
Assembly line
Workforce dimensioning
Worker assignment
Scheduling
Optimization
Integer linear programming
Heuristics

ABSTRACT

A paced assembly line consisting of several workstations is considered. This line is intended to assemble products of different types. The sequence of products is given. The sequence of technological tasks is common for all types of products. The assignment of tasks to the stations and task sequence on each station are known and cannot be modified, and they do not depend on the product type. Tasks assigned to the same station are performed sequentially. The processing time of a task depends on the number of workers performing this task. Workers are identical and versatile. If a worker is assigned to a task, he/she works on this task from its start till completion. Workers can switch between the stations at the end of each task and the time needed by any worker to move from one station to another one can be neglected. At the line design stage, it is necessary to know how many workers are necessary for the line. To know the response to this question we will consider each possible takt and assign workers to tasks so that the total number of workers is minimized, provided that a given takt time is satisfied. The maximum of minimal numbers of workers for all takt will be considered as the necessary number of workers for the line. Thus, the problem is to assign workers to tasks for a takt. We prove that this problem is NP-hard in the strong sense, we develop an integer linear programming formulation to solve it, and propose conventional and randomized heuristics.

© 2015 Elsevier B.V. All rights reserved.

Figura 6. Cabecera artículo “Workforce minimization for a mixed-model assembly line” de Battaïa *et al.*, 2014

Se considera una línea de ensamblado de varias estaciones con un ritmo de producción fijo. Esta línea fue diseñada para poder producir diferentes variaciones de producto, en el caso industrial que se estudia, son motores de coches de 8, 10 y 12 cilindros, los cuales suponen una variación en los tiempos de montaje, pero no variación en las operaciones, por esto se la secuencia tecnológica de tareas está definida y no sufre variaciones respecto a las diferentes variaciones de motor que existen.

La secuencia de productos está previamente definida y no esta sujeta a variación, se considera una restricción a cumplir.



Figura 7. Secuencia de producción definida para las tres posibles versiones de los productos existentes. V8, V10 y V12. (Figura del artículo de Olga Battaia *et al.*)

La asignación de tareas a cada una de estaciones de subensamblado y ensamblado y el orden de la secuencia de las mismas esta previamente definido tecnológicamente y no es posible modificarlas, además no depender de la versión del motor que se está fabricando.

2.1 Camshaft drive assembly		
118	VM 02.1	Screwing the double Wheel
119	VM 02.1	Insertion of the sleeve and the axle
120	VM 02.1	Fitting of O-Ring and plugs

Figura 8. Ejemplo de asignación de tareas a una estación de ensamblado.

Las tareas asignadas a la misma estación son realizadas de forma secuencial, no siendo posible adelantar o atrás operaciones secuenciadas en una estación de trabajo, ni el movimiento entre ellas. El tiempo de realización de cada una de las operaciones asignadas a las estaciones depende del número de trabajadores asignados a esa estación, siendo inversamente proporcional al número de trabajadores asignados a la misma.

Este estudio emplea la hipótesis en la que se consideran los trabajadores igualmente de capacitados, eficientes y competentes para la realización de cada una de las operaciones que componen la línea de ensamblado, suponiendo el principio de ser trabajadores idénticos y versátiles, siendo posible el intercambio o asignación de trabajadores a cualquier estación de ensamblado.

Se parte del principio de que un trabajador asignado a una operación, la realiza de principio a fin, sin ser posible interrumpirla a medias. Una vez finalizada la operación a realizar, el operario puede moverse de una estación a otra, suponiendo que el tiempo de traslado de las estaciones más lejanas entre si es menor de unos 30 segundos, se supone que el tiempo de movimiento es despreciable, debido a la morfología de la línea de ensamblado. En el proceso de diseño de la línea, es necesario saber cuántos trabajadores se necesitan en línea.

Para la definición de los tiempos de las actividades, se parte de la batería de datos recibida para el caso industrial y en los otros casos propuestos de la generada a partir de esta. Este estudio permite que el tiempo de actividad de una operación sea atendido por más de un operario, produciéndose una disminución del tiempo de operación total proporcionalmente al número de operarios que la atiendan, con un límite de hasta 4 operarios por operación.

$$\text{Tiempo actividad} = \frac{\text{Tiempo operación}}{\text{Número de operarios}}$$

Se toma en consideración el número de trabajadores asignados para minimizar el número total de operarios en la línea, asegurando que se cumple el takt time definido. El máximo del número mínimo de trabajadores para el takt definido será considerado el número mínimo de trabajadores para la línea.

De ahí, se encuentra la necesidad de encontrar el mínimo número de trabajadores cumpliendo el takt previamente definido y realizando todo el número de operaciones en todas las estaciones.

En el artículo anteriormente citado utiliza un problema MIP diseñado para representar la situación real de la línea con la producción actual en curso. Se hace un profundo estudio de la complejidad del problema que se afronta, y debido a su magnitud se comprueba que es NP, por lo que se opta por el desarrollo de heurísticas concretas especialmente diseñadas para el caso industrial a estudiar, y se hace una extensa comparativa entre ellas, eligiendo la más eficiente en términos de numero total de trabajadores.

El estudio de una nueva heurística se centra a la creación de una nueva línea de ensamblado para los tres tipos de motores de 8,10 y 12 cilindros. En las fases de diseño, se requiere conocer la dimensión de la mano de obra que será asignada a la línea para poder contratar al personal necesario. Se busca que este número sea lo menor posible para disminuir los costes de mano de obra, de ahí la necesidad de este estudio y que a su vez cumplan los requisitos del contorno del problema.

Las estaciones de la línea de ensamblado están conectadas de forma unidireccional. La secuencia de estaciones de ensamblado que serán visitadas por los motores en proceso de montaje y la secuencia de tareas necesarias para procesar hasta que el motor se convierte en un producto final son idénticas para todas las variantes del motor.

Sin embargo, los tiempos de proceso dependen de la variante de motor. Para el mismo número de trabajadores asignados a una tarea, ciertas tareas de los motores de 12 cilindros son más duraderas que las de los motores de 10 o 8 cilindros, mientras que otras tienen exactamente la misma duración independientemente de la variante del motor a fabricar.

La línea de ensamblado tiene un ritmo, por lo que la suma de los tiempos de las tareas en cualquiera de las líneas de las estaciones de la línea no puede exceder un determinado takt time que será fijado para la línea.

Para hacer un cálculo del takt time de la línea, se utiliza información de la siguiente Figura:

Tiempo de producción disponible	
Días por año	230
Turnos por día	2
Duración del turno (horas)	8

Figura 9. Tiempo de producción disponible deseado para la línea de ensamblado.

El volumen de producción y las ratios de demanda de cada motor son conocidos y detallados en la siguiente Figura:

Volumen de producción anual y ratios de variantes	
Volumen anual total (motores/año)	1450
V8	75%
V10	20%
V12	5%

Figura 10. Volumen total de todas las variantes y porcentaje de cada variante

Teniendo en cuenta las Figuras 3 y 4 que son los datos con los que se ha diseñado la línea, esta se prevé que produzca un total de 1450 motores de los 3 tipos de variantes conjuntamente al año, por lo que el takt time teniendo en cuenta el tiempo que la línea estará disponible, se obtiene el takt time:

$$\text{Takt time} = \frac{230 \frac{\text{días}}{\text{año}} * 2 \frac{\text{turnos}}{\text{día}} * 8 \frac{\text{horas}}{\text{turno}}}{1450 \frac{\text{motores}}{\text{año}}} = 2.5 \frac{\text{horas}}{\text{motor}}$$

Por preferencias del caso industrial, se opta por tener una variante de motores igualmente distribuidos en la secuencia de fabricación. Por ello el ciclo de fabricación se ha adaptado para cumplir la distribución igualada de producción de las diferentes variantes de fabricación. Se puede apreciar una representación gráfica de esta en la Figura 1. De forma análoga, se puede obtener la secuencia productiva para todo un año, siendo esta una repetición de la secuencia de 20 motores.

De forma general, se estudiará sobre este ciclo de 20 motores ya que cualquier aplicación para un ciclo idéntico, pero de mayor número, que serán repeticiones de la misma, se podrá aplicar lo estudiado.

Para el caso industrial que se estudia, existen 11 estaciones de ensamblado en la línea. Se define como primer estado de la línea la secuencia definida de la Figura 1. De forma cíclica al ritmo del takt time de la línea, irán abandonando y entrando en el orden del ciclo definido.

En caso de falta de mano de obra en una de las estaciones, un operario puede moverse para ayudar a completar una tarea de otra estación a tiempo. En la Figura 5 se puede apreciar el movimiento que realiza un operario de un motor ya terminado, marcado en verde, moverse a una estación adyacente para apoyar a completar las tareas a tiempo en otra estación, demarcado en color rojo el motor sin acabar.

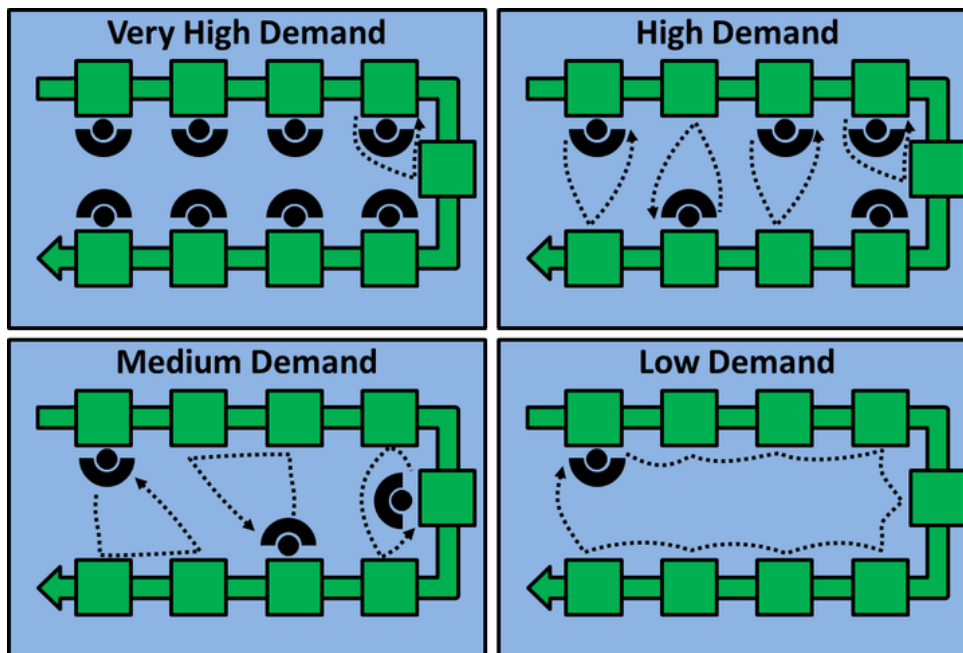


Figura 11. Movimiento de trabajadores de una estación a otra para ayudar a completar la tarea.

La distancia máxima del caso industrial en dos estaciones es de 11 metros, y cualquier trabajador puede moverse entre ellas en menos de 10 segundos, que supone menos de un 0,1% del takt time, de ahí que se tome como válida la suposición de tiempos despreciables de traslado entre estaciones de la línea.

En el caso particular de estudio, esta línea de ensamblado de motores de automóviles cuenta con estaciones de subensamblado en determinadas estaciones, las cuales son usadas para la realización de tareas de preparación necesarias para la estación que le precede. Estas son consideradas como una estación más en términos de análisis, y por lo tanto debe cumplir con el criterio de adherirse al takt time.

Para la línea de ensamblado del caso industrial que se estudia existen un total de 170 tareas a realizar en 11 estaciones de ensamblado y 17 estaciones de subensamblado, que es el recorrido que debe seguir un motor para completar su montaje completo.

Debido a que en términos de análisis computacional y de organización no existe distinción alguna posible entre estaciones de ensamblado y subensamblado, consideramos que la línea de forma pragmática tiene 28 estaciones en total.

El problema y por lo que se estudia este caso se basa en encontrar una solución al problema de encontrar una asignación de trabajadores que sea lo mínima posible para elevar el ahorro de costes al máximo, mientras que se cumple el takt time de 2.5 horas o 150 minutos.

La solución propuesta por la empresa en la cual se hace el estudio de esta línea propone 29 trabajadores como mínimo para cumplir todas las tareas y no sobrepasar la restricción del takt. La solución tras aplicación de la heurística más eficiente propuesta en el artículo de Battaia *et al.*, 2014, es de 19 trabajadores. En la heurística desarrollada se alcanza la cota inferior teórica de 16 trabajadores para el caso industrial propuesto.

3.2. Diseño y desarrollo de la heurística de asignación de trabajadores.

En este apartado se centra en mostrar y explicar de forma detallada las diferentes heurísticas que se van a aplicar al caso industrial, y su homólogo de mayor número de estaciones. La necesidad de desarrollar una nueva heurística reside en investigar si es posible mejorar los resultados obtenidos para el caso industrial específico, haciendo uso de heurística de carácter general aplicada, diseñada y adaptada a las necesidades del problema. Se pretende comparar la eficiencia de resolución para diferentes tamaños de línea de ensamblado.

Se estudia como función objetivo de la heurística mediante la búsqueda de la mejor combinación de operarios a las diferentes estaciones de forma que el número total de trabajadores en la línea de ensamblado sea el menor posible, cumpliendo el criterio de que en todas las estaciones se realizan todas las tareas, en un tiempo menor que el takt time de la línea.

Para ello se ha desarrollado una heurística que combina una primera heurística específica de asignación de trabajadores para este problema, después a la secuencia de asignación se le realiza una búsqueda local con la posibilidad de hacer una estrategia de reparación en caso de obtenerse una solución no válida, y, por último, se realiza un recocido simulado en el que de igual forma se permite la reparación en caso de secuencia no válida una vez.

Se trata de determinar de manera conjunta el número de estaciones y el número de operarios por estación, donde como resultado se obtienen las tareas a realizar en cada uno de los paquetes de trabajo y el número de trabajadores que les corresponde a ese mismo paquete de estaciones.

El resultado de la heurística se refleja en un vector de carácter numérico y de tamaño máximo el número de estaciones totales de la línea, donde se reflejará el paquete de trabajadores al que debe asignarle una serie de estaciones de forma secuencial hasta que incumpla el número teórico del takt time de la línea.

La secuencia de inicio se obtiene mediante la aplicación de una heurística específicamente diseñada para el caso industrial. Esto genera beneficios en términos del punto de partida de búsqueda para las heurísticas posteriores, ya que, si de primeras se le entrega una secuencia inicial válida y con una eficiencia básica pero aceptable, el primer escalón, y uno de los más complejos, que es buscar el espacio adecuado de exploración de soluciones válidas.

Esta consiste en la generación de un vector solución, el cual se rellena de ceros en su longitud total equivalente al número de estaciones de la línea. A partir de ese punto, se realiza un bucle que recorre todas las posiciones del vector, y va insertando el número máximo de trabajadores posibles limitado por el caso industrial, en este caso cuatro trabajadores, de forma que el vector de asignación se va rellenando de la siguiente forma:

$$x_0 = (0,0,0,0, 0, \dots, 0)$$

$$x_0 = (4,0,0,0, 0, \dots, 0)$$

$$x_0 = (4,4,0,0, 0, \dots, 0)$$

La secuencia se inicializa en x_0 , y a partir de este momento se va generando los vectores en orden, antes de ser introducidos en la heurística.

De tal forma que, en cada iteración de rellenado, se comprueba con la función de evaluación si esta secuencia generada es válida, es decir cumple con todas las restricciones del problema. De ser así, la heurística inicial se detiene y entrega como mejor secuencia la que obtenga el mínimo número de paquetes de cuatro trabajadores que cumpla los requisitos de la línea.

A continuación, se muestra una representación gráfica de un instante inicial de la secuencia que se introduce en la heurística en una parte concreta de la línea, y la situación final de la misma zona de la línea con el número de trabajadores que la línea a determinado que cumple el takt y es más eficiente.

Se toma como secuencia inicial de carácter general $x_0 = (4,4,0,0, 0, \dots, 0)$ y se representa a continuación:

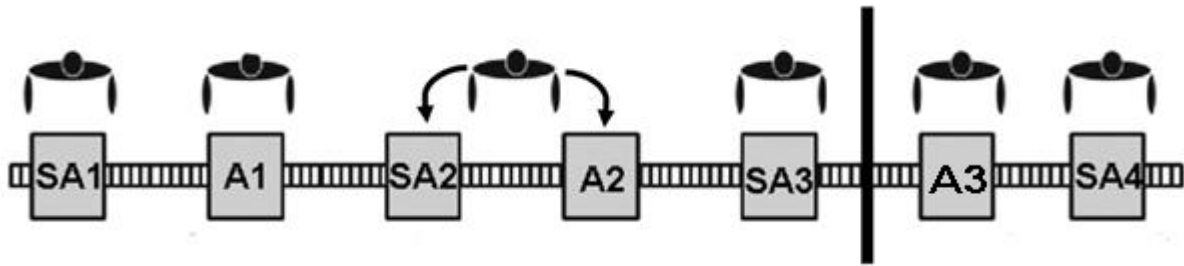


Figura 12. Ejemplo de organización de una secuencia inicial en una parte de la línea y la distribución de los trabajadores.

En ella se puede apreciar como en el primer paquete de estaciones de trabajo se asigna 4 operarios para 5 estaciones, teniendo un operario repartido entre una de ellas, en el siguiente paquete que no se representa en su totalidad también se asigna 4 trabajadores para un paquete de 6 estaciones.

Ahora se va a representar una secuencia obtenida por la heurística $x_0 = (2,4,3,2, 0, \dots, 0)$ con la siguiente representación:

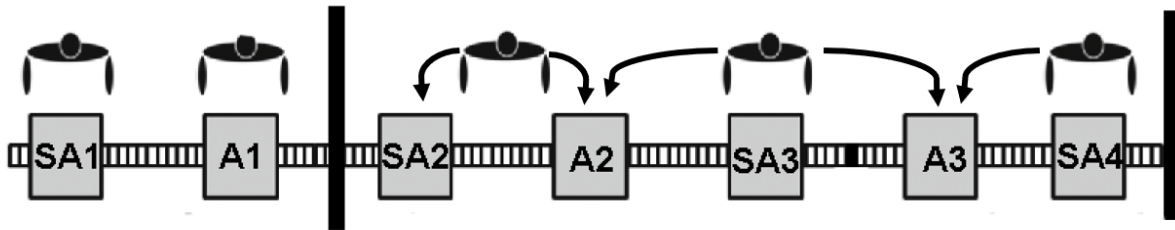


Figura 13. Ejemplo de organización de una secuencia eficiente en una parte de la línea y la distribución de los trabajadores.

Como se puede apreciar, la eficiencia en la distribución de operarios para las diferentes operaciones a realizar en la línea de ensamblado permitiendo la mutabilidad más habitualmente permite reducir el número de operarios en uno para este tamaño de la línea, siendo este efecto acentuado con el tamaño de la línea.

En este caso, el primer trabajo de paquete está asignado a dos operarios y a 2 estaciones, mientras que el segundo se puede asignar a 3 operarios para 5 estaciones, suponiendo una mejora en la eficiencia de la distribución de los recursos.

Una vez que se obtiene una secuencia inicial aceptable, se le aplican las heurísticas de búsqueda local primero y recocido simulado después, que se detallan a continuación. El pseudo código de la heurística de búsqueda local:

1. Se genera una solución inicial (s)
2. Mientras se pueda mejorar, se mantiene la condición en *TRUE*
3. Se ejecuta el bucle
 - a. Se encuentra una solución alternativa ($N(s) = s'$)
 - b. Si s' mejor que s entonces se guarda como nueva mejor solución $s = s'$
4. En caso contrario.
5. *es_posible_mejorar* se le asigna el valor *FALSE*
6. Se repite el bucle mientras (*es_posible_mejorar*) = *TRUE*

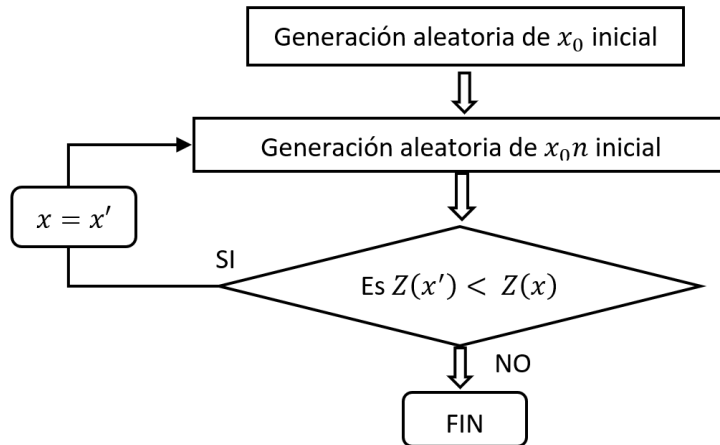


Figura 14. Diagrama de flujo de la búsqueda local simple

Una vez sales terminada la búsqueda local, se ejecuta el recocido simulado con la mejor secuencia que hayas encontrado de la búsqueda local. A continuación, se detalla el pseudocódigo del recocido simulado:

1. Seleccionar la mejor solución del vector x_0 a ser optimizado.
2. Inicializar parámetros de Temperatura T , Constante de Boltzmann K y factor de reducción C
3. Mientras el criterio de finalización del bucle no se cumple:
 - a. Para cada nueva solución posible generada x_{0n} :
 - i. Si mejora a x_0 la nueva solución se almacena como nueva $x_0 = x_{0n}$
 - ii. En caso contrario, se calcula un numero aleatorio con la fórmula de enfriamiento:
 1. Si número aleatorio está por encima del valor establecido, se almacena como nueva x_0
 2. En caso contrario, no se guarda.
4. Se decremento la condición del criterio del bucle.

A continuación, se muestra el diagrama de flujo de un recocido simulado:

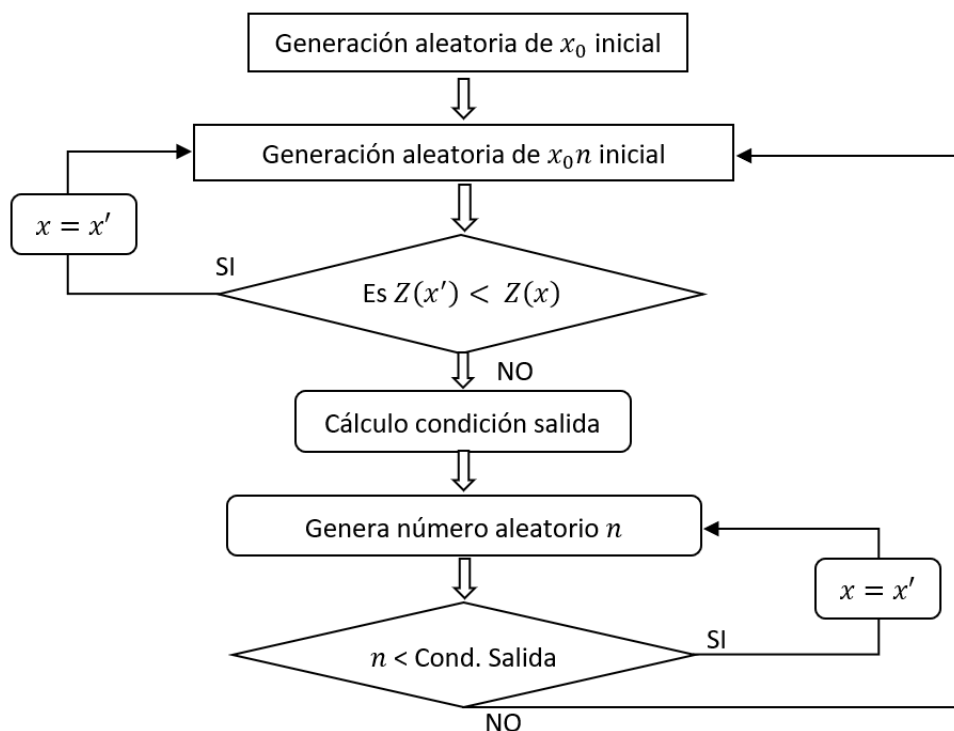


Figura 15. Diagrama de flujo de un recocido simulado.

A continuación, se deja detallado el pseudocódigo implantado para el desarrollo de la búsqueda de soluciones en el conjunto de todas las posibles soluciones:

- 1) Generación de la secuencia inicial con una heurística específica. Esta secuencia se definirá como el vecino inicial el cual será introducido en la Búsqueda Local.
- 2) En un bucle de cinco iteraciones se realizan las siguientes heurísticas:
 - I. Búsqueda Local: Se hace una búsqueda generando un nuevo vecino a partir del introducido de la secuencia inicial, bajando una unidad de manera aleatoria del vector siempre que sea posición distinta de cero. En caso de no ser válida, se le aplica la función reparadora.
 - a) Función reparadora: En una posición aleatoria de la secuencia anterior, se aumenta en una unidad siempre que sea posición que valga menos de cuatro. Se evalúa si es válida y si es mejor que la anterior solución se definirá como el nuevo vecino, y se vuelve a aplicar una Búsqueda Local a esta nueva secuencia generada.
 - II. Recocido Simulado: Una vez que acaba la Búsqueda Local, lo mejor que salga de la Búsqueda Local y las iteradas reparaciones, en el recocido simulado se genera un vecino que, en una posición aleatoria del vector, se sube o baja de forma aleatoria una unidad, y menor de cuatro.
 - a) Función reparadora: En una posición aleatoria de la secuencia anterior, se aumenta en una unidad siempre que sea posición que valga menos de cuatro. Se evalúa si es válida y si es mejor que la anterior solución se definirá como el nuevo vecino, y se vuelve a aplicar el Recocido Simulado a esta nueva secuencia generada.
- 3) Se guarda la mejor secuencia posible y se obtienen los operarios que deben ser asignados a cada estación.

Se aporta detallado un diagrama de flujo del funcionamiento de la heurística en la figura 16.

Con esto queda completamente detallada la heurística que se ha desarrollado para la búsqueda de soluciones del problema de asignación de trabajadores de forma eficiente a una línea de ensamblado de automóviles de modelo mixto, en el cual se producen diferentes variantes de un producto en la misma línea de ensamblaje.

Existen diferentes funciones a lo largo del código con el objeto de facilitar computacionalmente el análisis de las secuencias generadas por las heurísticas. A continuación, se detallan el funcionamiento de cada una:

- Función Evaluadora

Esta función recibe como entrada una secuencia con la asignación de operarios, una batería de tiempos y otra de estaciones de la línea con sus nombres, que deben ser ambas de las mismas dimensiones, ya que ambas baterías de datos son complementarias y definen la secuencia de asignación.

Se generan diccionarios para registrar el Takt time de cada una de las estaciones, otro para saber en qué paquete de trabajadores se le ha asignado a cada una de las estaciones de la línea.

Se crea una bandera para verificar que todas las estaciones son asignadas con la secuencia introducida. Se recorre en un bucle de longitud el número total de estaciones, en el cual en cada iteración se realiza otro bucle con la longitud de las estaciones que aún quedan por asignar, y se van asignando hasta que incumple la limitación de del takt time. Si al finalizar ambos bucles se han asignado todas las estaciones, se asigna la bandera como secuencia válida y se calcula el número total de operarios utilizados por la secuencia.

La función devuelve el número total de operarios que utiliza la secuencia de asignación, si la secuencia es válida, el takt time y el paquete al que se ha asignado a cada una de las estaciones

- Función Reparadora

Esta función recibe como entrada una secuencia con la asignación de operarios, y en una posición aleatoria que sea distinta de cuatro, aumenta en una unidad esa posición aleatoria en la secuencia de asignación.

- Función Ceros al final

Esta función recibe el vector de asignación de operarios y envía los ceros del mismo al final.

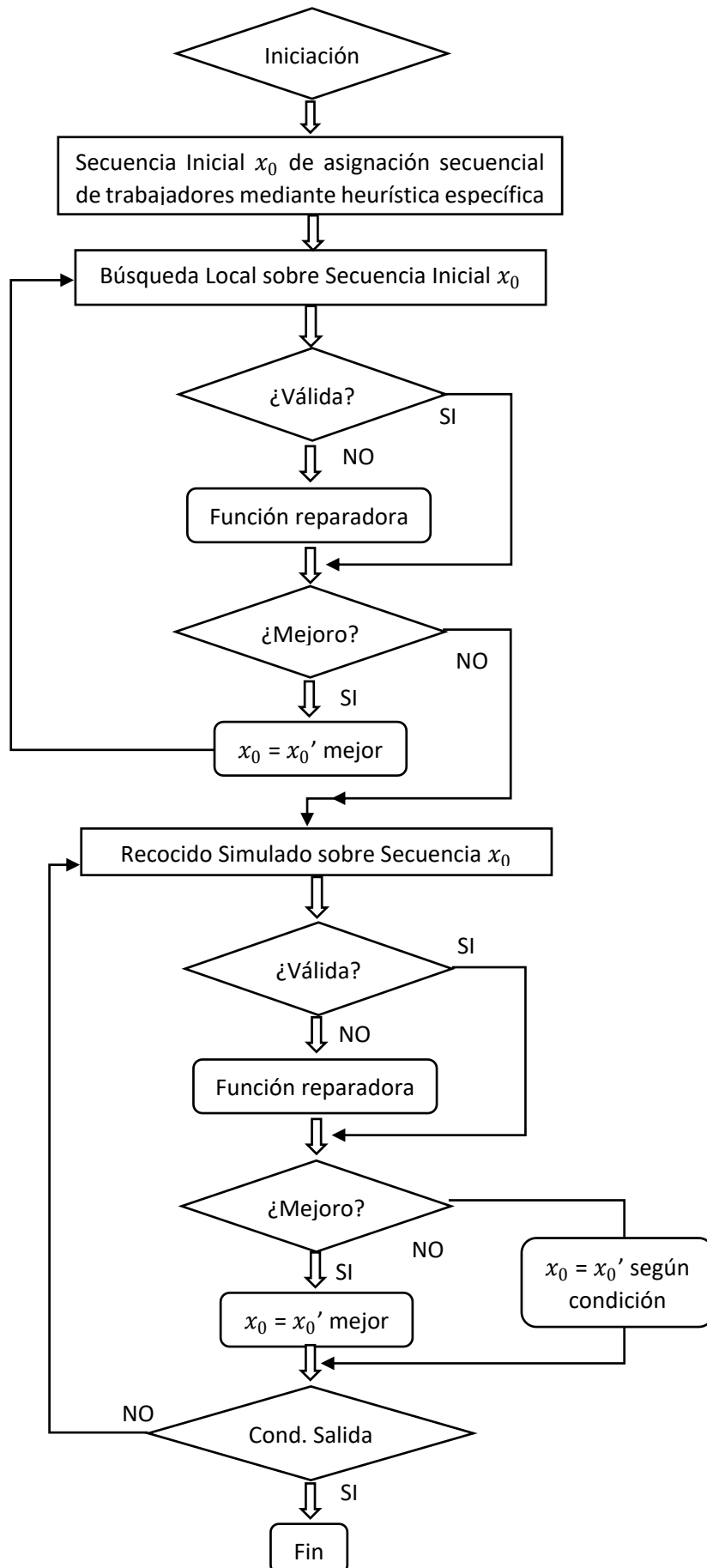


Figura 16. Diagrama de flujo de la heurística desarrollada

4 EXPERIMENTACIÓN

De acuerdo con las heurísticas desarrolladas para la resolución del problema planteado por el caso industrial, se requiere una batería de datos del proveedor del caso industrial para poder obtener soluciones con las cuales constatar la validez de la solución proporcionada por el modelo.

Debido a que uno de los objetivos de a validar durante este trabajo es la comprobación de si el caso es aplicable para una línea de ensamblado mixta de mayor tamaño, es por ello por lo que se desarrolla mediante el uso de la estadística una batería de datos para 40, 60 y 80 estaciones basadas en la distribución estadística de los tiempos de ensamblado en cada una de las estaciones de fabricación.

Se parte de un archivo Excel donde se facilita el tiempo de cada una de las operaciones que componen una tarea de montaje de cada uno de los subsistemas que componen el motor. Para poder tratar los datos con el lenguaje de programación Python, estos datos se simplifican por estaciones y subestaciones, agrupando la suma total de tiempos de cada una de estas.

Como en términos matemáticos no existe diferenciación entre estación y subestación, se obtiene una lista en Excel con el siguiente formato, donde se diferencia el nombre de la estación, y su tiempo total de operación. Se muestra a continuación el caso industrial.

Estaciones	Tiempo(Min)	Estaciones	Tiempo(Min)
SA1	74	SA6	6,8
A1	174	A6	171,6
SA2	57,6	A7	289,2
A2	161,4	A8	194,4
SA3	33,2	SA9	42
A3	151,6	A9	177,6
SA4	24	SA10	29,6
A4	156	A10	176,4
SA5	86	A11	122
A5	153,6	Total Time	2281

Figura 17. Listado de tiempos agrupados por estación del caso industrial.

Se puede apreciar respecto al documento original, que se ha simplificado de manera que se la unidad significativa sea el tiempo total de la estación, y no las operaciones ya que esto supone una simplificación en la cantidad total de variables de decisión, mejorando los tiempos de resolución.

A continuación, se muestra el formato original de tiempos para un subsistema concreto:

No.	Stat	Task description	Equipment	Variant 1			
				processing time (min) * 1,2 in * human resources	processing time (min) * 1,2	processing time in min	processing time in seg
1.2 Pistons - Liner - Connecting rod							
110	VM 01.2	Insertion of the bearing bush into the connecting rod	force fitting support	3,33	3,33	2,78	167
111	VM 01.2	Connection of the piston and the connecting rod	force fitting support, snap ring pliers	8,00	4,00	3,33	200
112	VM 01.2	Fitting of the sealing ring with the liner	support equipment	2,00	2,00	1,67	100
113	VM 01.2	Fitting of the piston ring with the linear	Piston ring pliers	6,00	6,00	5,00	300
114	VM 01.2	Assembly of the piston fixture	hand tool	6,00	6,00	5,00	300
115	VM 01.2	Insertion of the piston and the connecting rod in the liner		6,00	6,00	5,00	300
116	VM 01.2	Removal of the connecting rod cap	hand tool	2,00	2,00	1,67	100
117	VM 01.2	Assembly of the connecting rod head fixture	hand tool	4,00	4,00	3,33	200

Figura 18. Listado de tiempo de las operaciones realizadas en una estación de ensamblaje.

Para obtener una batería de datos de gran tamaño que contase con más estaciones que las del caso industrial, para poder testear la heurística en un entorno industrial de mayor tamaño de línea. Para que se asemejase lo máximo posible al caso industrial, se hace un estudio estadístico en el cual de los tiempos por estación se obtiene una media y desviación estándar de la batería de datos.

Utilizando la fórmula de Excel ALEATORIO() se consigue un número aleatorio entre 0 y 1, la cual la media se situaría en 0,5 en los números aleatorios generados, y en la media establecida de la batería de datos, en el caso del caso de estudio, 151.6 minutos por estación, con una desviación estándar de ± 79.2 minutos.

Teniendo generados la secuencia de número aleatorios, se genera el valor absoluto de una Normal con media y desviación la obtenida del caso industrial usando la fórmula de Excel =ABS(DISTR.NORM.INV(probabilidad, media, desviación estándar)), el cual genera unos datos para los casos de soporte de gran tamaño para el testeo de la heurística que siguen el patrón y forma del caso industrial original. A continuación se muestra un ejemplo de generación de tiempos:

	Aleatorio <1	Normal Aleatorio
Media	0.917538607	261.5
151.6	0.763476198	208.4
Desv. Est.	0.260475363	100.8
79.2	0.71346484	196.2

Figura 19. Ejemplo de generación de tiempos en Excel.

A continuación se muestran los ejemplos generados de tiempos para los casos de las estaciones de 40, 60 y 80, de forma que se pueda apreciar la homogeneidad de los tiempos:

Estaciones	Tiempo(Min)	Estaciones	Tiempo(Min)	Estaciones	Tiempo(Min)
SA1	139,0	A7	40,1	SA14	210,4
A1	202,5	SA8	308,6	A14	118,2
SA2	153,7	A8	65,8	SA15	177,4
A2	128,0	SA9	87,5	A15	198,8
SA3	156,3	A9	246,2	SA16	218,5
A3	1,5	SA10	14,0	A16	163,7
SA4	195,9	A10	130,4	SA17	267,4
A4	184,3	SA11	116,5	A17	137,8
SA5	130,9	A11	163,7	SA18	118,1
A5	240,3	SA12	140,9	A18	131,0
SA6	44,4	A12	144,8	SA19	206,9
A6	22,3	SA13	181,5	A19	97,4
SA7	304,1	A13	57,7	SA20	241,1
				A20	140,5

Figura 20. Tiempos generados para 40 estaciones.

Estaciones	Tiempo(Min)	Estaciones	Tiempo(Min)	Estaciones	Tiempo(Min)
SA1	60,7	SA11	158,0	SA21	204,9
A1	237,3	A11	133,3	A21	172,3
SA2	158,0	SA12	150,7	SA22	197,7
A2	173,0	A12	284,0	A22	59,0
SA3	364,4	SA13	137,7	SA23	130,0
A3	141,6	A13	161,2	A23	47,0
SA4	59,1	SA14	186,6	SA24	219,8
A4	294,6	A14	179,2	A24	138,2
SA5	154,3	SA15	141,6	SA25	180,2
A5	101,6	A15	160,5	A25	75,9
SA6	117,2	SA16	89,3	SA26	129,5
A6	240,5	A16	201,7	A26	203,3
SA7	138,6	SA17	295,3	SA27	135,7
A7	249,4	A17	207,7	A27	207,8
SA8	214,3	SA18	78,4	SA28	183,3
A8	132,9	A18	208,9	A28	210,4
SA9	173,7	SA19	220,9	SA29	119,4
A9	226,7	A19	57,6	A29	135,7
SA10	1,9	SA20	210,1	SA30	131,2
A10	223,5	A20	112,9	A30	190,5

Figura 21. Tiempos generados para 60 estaciones.

Estaciones	Tiempo(Min)	SA14	177,7	A27	145,3
SA1	74,3	A14	218,6	SA28	178,3
A1	132,8	SA15	140,2	A28	84,4
SA2	101,1	A15	277,5	SA29	97,6
A2	111,8	SA16	177,4	A29	166,8
SA3	187,1	A16	159,4	SA30	234,6
A3	123,2	SA17	162,4	A30	109,0
SA4	57,4	A17	38,0	SA31	94,7
A4	79,4	SA18	149,8	A31	68,3
SA5	45,2	A18	147,9	SA32	59,5
A5	166,5	SA19	67,2	A32	25,0
SA6	114,6	A19	110,2	SA33	175,8
A6	202,9	SA20	193,0	A33	190,9
SA7	249,1	A20	33,2	SA34	57,7
A7	45,5	SA21	123,5	A34	242,5
SA8	115,6	A21	261,6	SA35	177,9
A8	157,3	SA22	148,7	A35	113,0
SA9	51,5	A22	217,1	SA36	84,5
A9	201,1	SA23	142,4	A36	110,7
SA10	272,3	A23	252,5	SA37	71,4
A10	144,7	SA24	274,6	A37	82,3
SA11	96,9	A24	128,5	SA38	155,0
A11	23,9	SA25	184,4	A38	71,6
SA12	260,4	A25	169,3	SA39	315,0
A12	4,0	SA26	102,6	A39	145,9
SA13	204,4	A26	145,1	SA40	231,2
A13	87,8	SA27	24,1	A40	167,5

Figura 22. Tiempos generados para 80 estaciones.

Por último, una vez se tienen todas las hojas de Excel con las baterías de datos adecuadamente formateadas, desde el código desarrollado se leen los nombres de las estaciones y sus tiempos correspondientes, para posteriormente ser almacenados en diccionarios de Python, los cuales tendrán como claves el nombre de las estaciones y como valor temporal el tiempo que se le haya determinado. Por último se define la variable para el takt time, que tendrá un valor fijo y constante, y queda completamente definido tanto teóricamente como computacionalmente la entrada de datos del trabajo.

5 ANÁLISIS DE LOS RESULTADOS

5.1 Presentación de los resultados

En este apartado se van a dar conocer las secuencias de asignación de trabajadores obtenidas de los diferentes casos propuestos mediante la heurística específicamente desarrollada para este tiempo de línea de ensamblado de modelo mixto.

A continuación, se va a mostrar un informe detallado del resultado de obtenido por la heurística para el caso industrial real:

Sec. Inicial: [4, 4, 4, 4, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Numero de operarios en secuencia inicial: 20
Mejor ruta Recocido Simulado: [2, 3, 4, 2, 3, 2, 0, 0, 0, 0, 0, 0, 0]
Numero de operarios en secuencia óptima: 16
¿Es una secuencia válida?: Si
El Takt del paquete 1 es de 124 min con las estación/es SA1,A1, y 2 operario/s.
El Takt del paquete 2 es de 142 min con las estación/es SA2,A2,SA3,A3,SA4, y 3 operario/s.
El Takt del paquete 3 es de 143 min con las estación/es A4,SA5,A5,SA6,A6, y 4 operario/s.
El Takt del paquete 4 es de 144 min con las estación/es A7, y 2 operario/s.
El Takt del paquete 5 es de 147 min con las estación/es A8,SA9,A9,SA10, y 3 operario/s.
El Takt del paquete 6 es de 149 min con las estación/es A10,A11, y 2 operario/s.
Numero operarios usados: 16

Figura 23. Resultados del caso industrial.

El tiempo para la obtención de esta secuencia fue de 37 segundos. Mediante el estudio de los parámetros de datos para el caso industrial se ha llegado a las siguientes conclusiones:

Tiempo total línea (min)	2281
Takt Time de la línea	150
Mínimo teórico de operarios	15,2

Figura 24. Información de ejecución del caso industrial.

De forma análoga, se aportan las diferentes secuencias de asignación de operarios obtenidas por la heurística para cada uno de los casos industriales teóricos generados para testear la validez de la búsqueda en entornos productivos de mayor tamaño, en este caso siendo para 40, 60 y 80 estaciones respectivamente:

Para 40 estaciones se ha determinado como mejor ruta posible:

Sec. Ini: [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 0, 0, 0]
Numero de operarios en secuencia inicial: 48
Mejor ruta: [4, 4, 4, 3, 4, 3, 4, 4, 4, 3, 4, 3, 1, 0, 0]
Secuencia válida?: Si
Numero operarios usados: 45
El Takt del paquete 1 es de 123 min con las estación/es SA1,A1,SA2, y 4 operario/s.
El Takt del paquete 2 es de 120 min con las estación/es A2,SA3,A3,SA4, y 4 operario/s.
El Takt del paquete 3 es de 149 min con las estación/es A4,SA5,A5,SA6, y 4 operario/s.
El Takt del paquete 4 es de 122 min con las estación/es A6,SA7,A7, y 3 operario/s.
El Takt del paquete 5 es de 115 min con las estación/es SA8,A8,SA9, y 4 operario/s.
El Takt del paquete 6 es de 130 min con las estación/es A9,SA10,A10, y 3 operario/s.
El Takt del paquete 7 es de 141 min con las estación/es SA11,A11,SA12,A12, y 4 operario/s.
El Takt del paquete 8 es de 141 min con las estación/es SA13,A13,SA14,A14, y 4 operario/s.
El Takt del paquete 9 es de 148 min con las estación/es SA15,A15,SA16, y 4 operario/s.
El Takt del paquete 10 es de 143 min con las estación/es A16,SA17, y 3 operario/s.
El Takt del paquete 11 es de 148 min con las estación/es A17,SA18,A18,SA19, y 4 operario/s.
El Takt del paquete 12 es de 112 min con las estación/es A19,SA20, y 3 operario/s.
El Takt del paquete 13 es de 140 min con las estación/es A20, y 1 operario/s.
Numero operarios usados: 45

Figura 25. Resultados del caso de 40 estaciones.

El tiempo para la obtención de esta secuencia fue de 111 segundos. Mediante el estudio de los parámetros de datos para el caso industrial se ha llegado a las siguientes conclusiones:

Tiempo total línea (min)	6028,1
Takt Time de la línea	150
Mínimo teórico de operarios	40,2

Figura 26. Información de ejecución del caso de 40 estaciones.

Para 60 estaciones se ha determinado como mejor ruta posible:

Sec. Ini: [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Numero de operarios en secuencia inicial: 80
Secuencia válida?: Si
Mejor ruta total: [4, 4, 2, 3, 4, 4, 3, 4, 4, 4, 4, 4, 3, 4, 4, 4, 3, 3, 3, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Numero de operarios en secuencia inicial: 72
Secuencia válida?: Si
El Takt del paquete 1 es de 114 min con las estación/es SA1,A1,SA2, y 4 operario/s.
El Takt del paquete 2 es de 134 min con las estación/es A2,SA3, y 4 operario/s.
El Takt del paquete 3 es de 100 min con las estación/es A3,SA4, y 2 operario/s.
El Takt del paquete 4 es de 149 min con las estación/es A4,SA5, y 3 operario/s.
El Takt del paquete 5 es de 149 min con las estación/es A5,SA6,A6,SA7, y 4 operario/s.
El Takt del paquete 6 es de 149 min con las estación/es A7,SA8,A8, y 4 operario/s.
El Takt del paquete 7 es de 134 min con las estación/es SA9,A9,SA10, y 3 operario/s.
El Takt del paquete 8 es de 128 min con las estación/es A10,SA11,A11, y 4 operario/s.
El Takt del paquete 9 es de 143 min con las estación/es SA12,A12,SA13, y 4 operario/s.
El Takt del paquete 10 es de 131 min con las estación/es A13,SA14,A14, y 4 operario/s.
El Takt del paquete 11 es de 148 min con las estación/es SA15,A15,SA16,A16, y 4 operario/s.
El Takt del paquete 12 es de 145 min con las estación/es SA17,A17,SA18, y 4 operario/s.
El Takt del paquete 13 es de 143 min con las estación/es A18,SA19, y 3 operario/s.
El Takt del paquete 14 es de 146 min con las estación/es A19,SA20,A20,SA21, y 4 operario/s.
El Takt del paquete 15 es de 139 min con las estación/es A21,SA22,A22,SA23, y 4 operario/s.
El Takt del paquete 16 es de 146 min con las estación/es A23,SA24,A24,SA25, y 4 operario/s.
El Takt del paquete 17 es de 136 min con las estación/es A25,SA26,A26, y 3 operario/s.
El Takt del paquete 18 es de 114 min con las estación/es SA27,A27, y 3 operario/s.
El Takt del paquete 19 es de 131 min con las estación/es SA28,A28, y 3 operario/s.
El Takt del paquete 20 es de 144 min con las estación/es SA29,A29,SA30,A30, y 4 operario/s.
Numero operarios usados: 72

Figura 27. Resultados del caso de 60 estaciones.

El tiempo para la obtención de esta secuencia fue de 421 segundos. Mediante el estudio de los parámetros de datos para el caso industrial se ha llegado a las siguientes conclusiones:

Tiempo total línea (min)	9910,7
Takt Time de la línea	150
Mínimo teórico de operarios	66,1

Figura 28. Información de ejecución del caso de 60 estaciones.

Para 80 estaciones se ha determinado como mejor ruta posible:

Sec. Ini: [4, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Numero de operarios en secuencia inicial: 92
Secuencia válida?: Si
Mejor ruta total: [3, 3, 4, 4, 4, 4, 2, 3, 4, 4, 4, 3, 4, 4, 4, 3, 4, 4, 3, 4, 4, 3, 4, 0, 0, 0, 0, 0, 0, 0]
Número de operarios usados: 83
El Takt del paquete 1 es de 140 min con las estación/es SA1,A1,SA2,A2, y 3 operario/s.
El Takt del paquete 2 es de 149 min con las estación/es SA3,A3,SA4,A4, y 3 operario/s.
El Takt del paquete 3 es de 132 min con las estación/es SA5,A5,SA6,A6, y 4 operario/s.
El Takt del paquete 4 es de 141 min con las estación/es SA7,A7,SA8,A8, y 4 operario/s.
El Takt del paquete 5 es de 131 min con las estación/es SA9,A9,SA10, y 4 operario/s.
El Takt del paquete 6 es de 132 min con las estación/es A10,SA11,A11,SA12,A12, y 4 operario/s.
El Takt del paquete 7 es de 146 min con las estación/es SA13,A13, y 2 operario/s.
El Takt del paquete 8 es de 132 min con las estación/es SA14,A14, y 3 operario/s.
El Takt del paquete 9 es de 148 min con las estación/es SA15,A15,SA16, y 4 operario/s.
El Takt del paquete 10 es de 127 min con las estación/es A16,SA17,A17,SA18, y 4 operario/s.
El Takt del paquete 11 es de 137 min con las estación/es A18,SA19,A19,SA20,A20, y 4 operario/s.
El Takt del paquete 12 es de 128 min con las estación/es SA21,A21, y 3 operario/s.
El Takt del paquete 13 es de 127 min con las estación/es SA22,A22,SA23, y 4 operario/s.
El Takt del paquete 14 es de 131 min con las estación/es A23,SA24, y 4 operario/s.
El Takt del paquete 15 es de 146 min con las estación/es A24,SA25,A25,SA26, y 4 operario/s.
El Takt del paquete 16 es de 104 min con las estación/es A26,SA27,A27, y 3 operario/s.
El Takt del paquete 17 es de 131 min con las estación/es SA28,A28,SA29,A29, y 4 operario/s.
El Takt del paquete 18 es de 147 min con las estación/es SA30,A30,SA31,A31,SA32,A32, y 4 operario/s.
El Takt del paquete 19 es de 141 min con las estación/es SA33,A33,SA34, y 3 operario/s.
El Takt del paquete 20 es de 133 min con las estación/es A34,SA35,A35, y 4 operario/s.
El Takt del paquete 21 es de 143 min con las estación/es SA36,A36,SA37,A37,SA38,A38, y 4 operario/s.
El Takt del paquete 22 es de 104 min con las estación/es SA39, y 3 operario/s.
El Takt del paquete 23 es de 136 min con las estación/es A39,SA40,A40, y 4 operario/s.
Numero operarios usados: 83

Figura 29. Resultados del caso de 80 estaciones.

El tiempo para la obtención de esta secuencia fue de 710 segundos. Mediante el estudio de los parámetros de datos para el caso industrial se ha llegado a las siguientes conclusiones:

Tiempo total línea (min)	11194,1
Takt Time de la línea	150
Mínimo teórico de operarios	74,6

Figura 30. Información de ejecución del caso de 80 estaciones.

5.2. Análisis de los resultados

Una vez se han mostrado los resultados obtenidos por la heurística específicamente desarrollada para el caso industrial, y posteriormente aplicada a los diferentes casos con más estaciones generados a partir de la batería de datos del original. La primera comparativa que se establece se refiere a la comparación de los cuatro casos generados en diferentes ámbitos, para ellos se adjunta la siguiente tabla con el estudio:

	Operarios Secuencia Inicial	Operarios Secuencia Final	Operarios Mínimos teóricos	Operarios Artículo Publicado	Reducción Respecto Sec. Inicial	Diferencia Respecto Min. Teórico
Caso Industrial	20	16	16	19	4	0
40 estaciones	48	45	41	N/A	3	4
60 estaciones	80	72	67	N/A	8	5
80 estaciones	92	83	75	N/A	9	8

Figura 31. Figura comparativa de los casos.

En la tabla se pueden encontrar las siguientes columnas:

- Número de operarios necesarios generados en la secuencia inicial por la heurística desarrollada la cual va asignando trabajadores en paquetes de 4 hasta que se cumplen que todas las operaciones se realizan y que todas que cumplen el criterio del takt time.
- Número de operarios necesarios generados en la secuencia final obtenida por la heurísticas específicamente desarrollada para el caso industrial.
- Número mínimo de operarios de forma teórica calculada con el tiempo total de operaciones a asignar en toda la línea productiva, que variara en cada uno de los casos generados, dividido entre el takt time de la línea. Este supone una referencia como límite inferior, que en muchas ocasiones no es posible alcanzar debido a la limitación del reparto de operaciones a través de las distintas estaciones de ensamblado y subensamblado.
- Operarios utilizados en el artículo. Este punto solo aplica para el caso industrial real, y se toma como referencia la solución obtenida de 19 operarios como referencia para mejorar y comparar si la heurística desarrollada es eficiente y encuentra mejores soluciones.

- Reducción respecto a secuencial inicial. Mediante este punto se trata de establecer un indicador de comparación de eficiencia entre la heurística inicial y la heurística, de acuerdo a poder comprobar que margen de mejora tiene la heurística. Para todos los casos se encuentra que existe una mejoría de un entorno del 10% al 20% respecto al número total de operarios necesarios, lo que dependiendo de la escala del problema pueden suponer desde 3 hasta 9 trabajadores de reducción en la línea productiva dependiendo del tamaño de la misma.
- Diferencia respecto al mínimo teórico. Con esta medida se puede comprobar como de lejos se está respecto de la situación teórica mejor. En este caso, en el caso industrial se produce la mejor situación posible teórica, cosa que de normal general no se consigue como se puede apreciar en las demás casos.

Se puede apreciar, que independientemente del caso, la metaheurística mejora sustancialmente los valores iniciales de operarios, quedándose en todos los casos cerca proporcionalmente de los mínimos teóricos, lo que garantiza la optimalidad de la búsqueda.

Otro punto de referencia para la optimalidad de la solución encontrada es la comparativa con el caso industrial que toma de referencia este estudio, en el cual establece el mínimo de operarios para el caso industrial de 19 operarios, siendo la solución encontrada por la metaheurística de 16 operarios, estando en la misma dimensión de rangos y mejorándolo en términos de trabajadores totales.

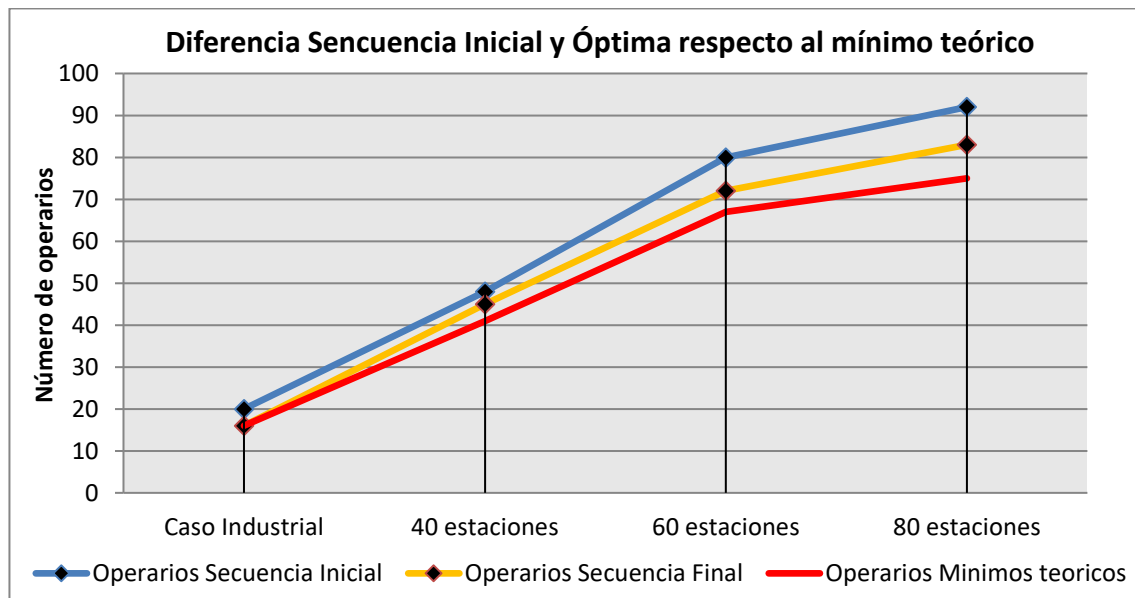


Figura 32. Evolución de la heurística en términos de operarios usados en la secuencia.

En la gráfica representada se puede apreciar como la diferencia entre el mínimo teórico y la solución óptima encontrada por la heurística se va ampliando proporcionalmente al número de operarios.

A continuación, se muestra la evolución temporal de la heurística para los diferentes casos industriales, donde se puede apreciar en el eje X el tiempo en segundos empleado por el método para encontrar la solución más óptima mediante una secuencia y en el eje Y el número de operarios que usa esa secuencia.

Con estas representaciones se puede apreciar la evolución característica de la heurística típica del recocido simulado, donde se va mejorando o quedándose igual en este caso debido a que las soluciones son enteras, y cuando llega un momento de estancamiento donde acaba enfriándose la búsqueda y se detiene.

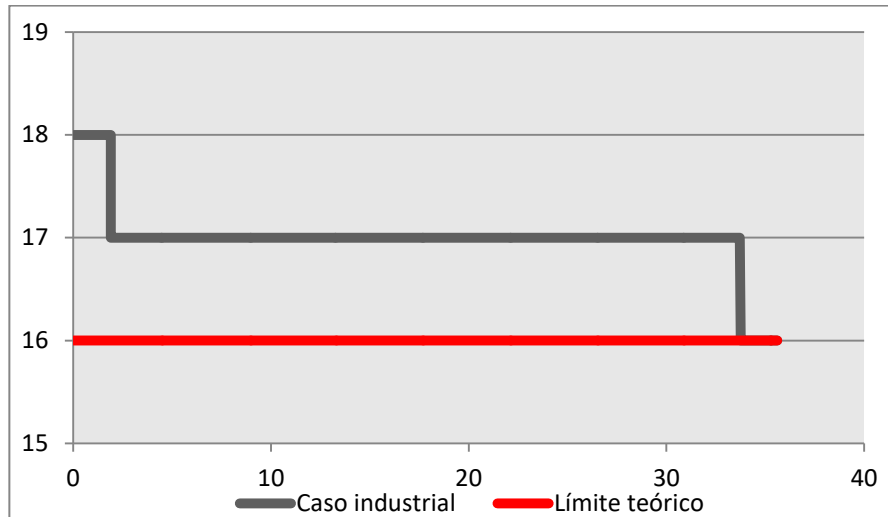


Figura 33. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso industrial.

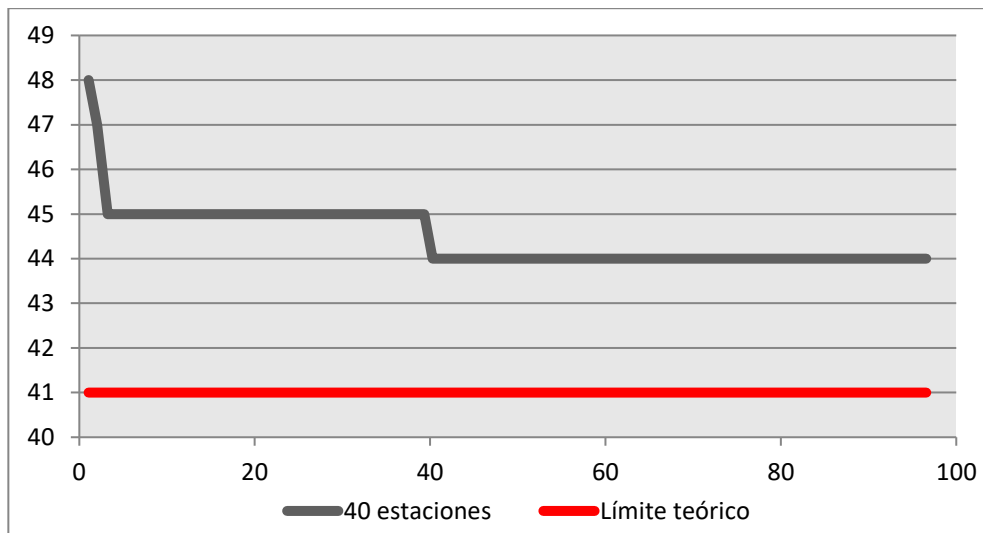


Figura 34. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 40 estaciones.

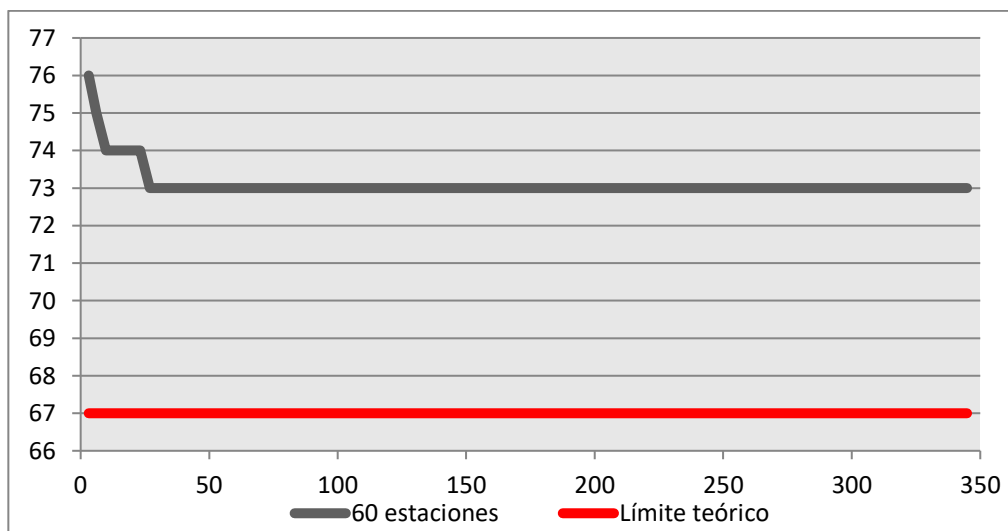


Figura 35. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 60 estaciones.

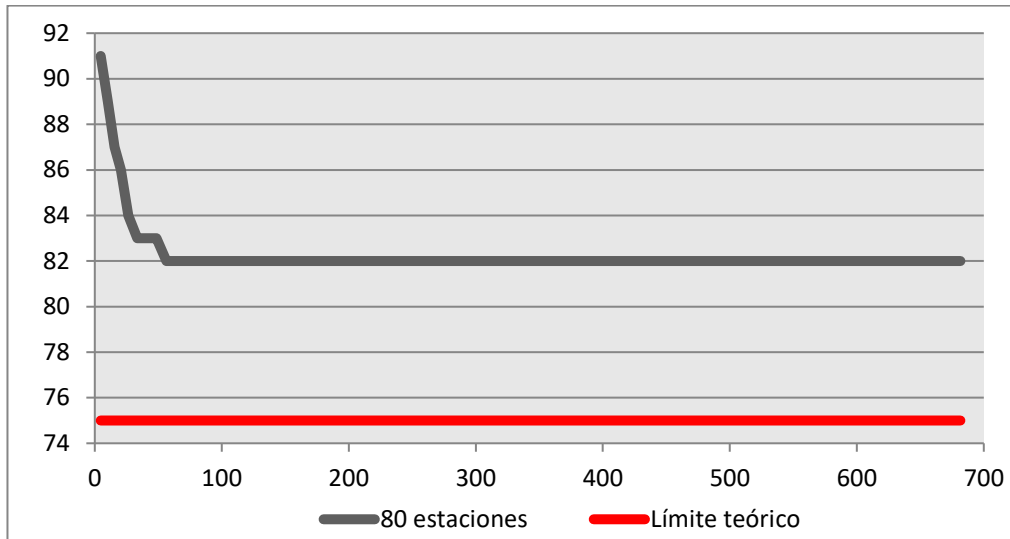


Figura 36. Evolución de la heurística en términos de operarios usados en la secuencia a lo largo del tiempo para el caso de 80 estaciones.

De media general, se obtiene la siguiente tabla del tiempo medio de búsqueda de solución en función del número de estaciones totales:

	Tiempo Usado (segundos)	Tiempo/estación (segundos)
Caso Industrial	37	1,85
40 estaciones	111	2,775
60 estaciones	421	7,02
80 estaciones	710	8,875

Figura 37. Tiempo en segundos invertido por estación.

En términos temporales, se aprecia que cuando menor es el número de estaciones, y por consecuencia, el número de operaciones total y operarios, de forma genérica la heurística tarda en asignar los operarios a una estación una media de 1,85 segundos por estación. Esta va aumentando exponencialmente a medida que el caso a resolver aumenta en su número de estaciones, ya que la combinatoria de posibilidades crece, y por lo tanto la complejidad de búsqueda y comprobación de las soluciones crece.

Comparativa de diferencias entre el takt de los paquetes y el takt máximo de la línea de 150 minutos para los cuatro casos industriales.

Para el caso industrial real con la secuencia más óptima:

Caso industrial	Takt Secuencia	Takt Máximo	Margen/Paquete
Paquete 1	124	150	26
Paquete 2	142	150	8
Paquete 3	143	150	7
Paquete 4	144	150	6
Paquete 5	147	150	3
Paquete 6	149	150	1
Margen Total	849	900	51

Figura 38. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para el caso industrial.

Para el caso con 40 estaciones con la secuencia más óptima:

Caso industrial	Takt Secuencia	Takt Máximo	Margen/Paquete
Paquete 1	123	150	27
Paquete 2	120	150	30
Paquete 3	149	150	1
Paquete 4	122	150	28
Paquete 5	115	150	35
Paquete 6	130	150	20
Paquete 7	141	150	9
Paquete 8	141	150	9
Paquete 9	148	150	2
Paquete 10	143	150	7
Paquete 11	148	150	2
Paquete 12	112	150	38
Paquete 13	140	150	10
Margen Total	1732	1950	218

Figura 38. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 40 estaciones.

Para el caso con 60 estaciones con la secuencia más óptima:

Caso industrial	Takt Secuencia	Takt Máximo	Margen/Paquete
Paquete 1	114	150	36
Paquete 2	134	150	16
Paquete 3	100	150	50
Paquete 4	149	150	1
Paquete 5	149	150	1
Paquete 6	149	150	1
Paquete 7	134	150	16
Paquete 8	128	150	22
Paquete 9	143	150	7
Paquete 10	131	150	19
Paquete 11	148	150	2
Paquete 12	145	150	5
Paquete 13	143	150	7
Paquete 14	146	150	4
Paquete 15	139	150	11
Paquete 16	146	150	4
Paquete 17	136	150	14
Paquete 18	114	150	36
Paquete 19	131	150	19
Paquete 20	144	150	6
Margen Total	2723	3000	277

Figura 38. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 60 estaciones.

Para el caso con 80 estaciones con la secuencia más óptima:

Caso industrial	Takt Secuencia	Takt Máximo	Margen/Paquete
Paquete 1	140	150	10
Paquete 2	149	150	1
Paquete 3	132	150	18
Paquete 4	141	150	9
Paquete 5	131	150	19
Paquete 6	132	150	18
Paquete 7	146	150	4
Paquete 8	132	150	18
Paquete 9	148	150	2
Paquete 10	127	150	23
Paquete 11	137	150	13
Paquete 12	128	150	22
Paquete 13	127	150	23
Paquete 14	131	150	19
Paquete 15	146	150	4
Paquete 16	104	150	46
Paquete 17	131	150	19
Paquete 18	147	150	3
Paquete 19	141	150	9
Paquete 20	133	150	17
Paquete 21	143	150	7
Paquete 22	104	150	46
Paquete 23	135	150	15
Margen Total	3085	3450	365

Figura 39. Takt time por paquete de estaciones y su diferencia con el takt time de la línea para 80 estaciones.

Se puede apreciar de forma en las diferentes columnas de las tablas:

- El paquete de trabajadores que se ha asociado a cada una de las estaciones (para saber que estaciones corresponden a cada paquete se encuentra en la tabla de soluciones de cada caso).
- Tiempo de operación de ese paquete, siendo esta el resultado de la suma total de operaciones de las estaciones asignadas a ese paquete dividido entre el número de operarios.
- El Takt Time de la línea para tener de referencia la cota superior de los paquetes asignados.
- La diferencia entre el Takt Time de la línea o cota superior y el tiempo de operación de ese paquete.

Se puede apreciar de forma rápida que la diferencia entre el Takt Time de la línea o cota superior y el tiempo de operación del paquete estará en un tono verde en caso de ser lo más cercano a cero, mientras que tomará un tono amarillo o incluso rojo en el caso de que la diferencia en minutos sea más grande, indicando esta ineficiencia en la asignación de los paquetes.

Hay que tener en cuenta que la asignación de los trabajadores, y por consecuencia los paquetes están fuertemente limitada por las operaciones de las estaciones, es por ello que pese a encontrar eficiencia a la hora de la resolución de la secuencia de asignación, existen ineficiencias respecto al límite teórico con las que se debe convivir.

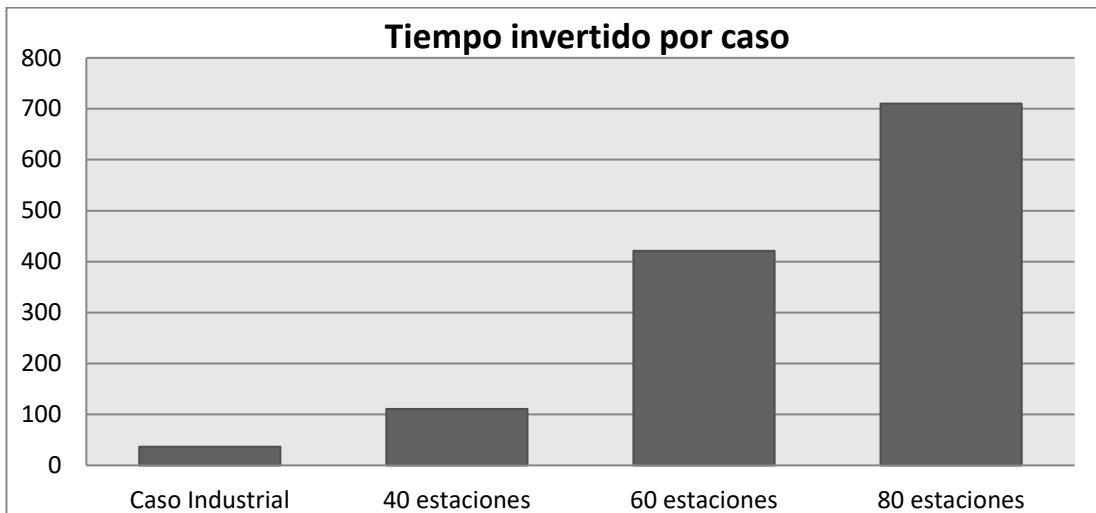


Figura 40. Tiempo invertido en resolver los casos industriales.

Como parámetros para la parte de la heurística que implica un recocido simulado, se ha tomado como referencia de temperatura máxima 100 grados, de temperatura mínima o condición de salida 1 grado, con una velocidad de enfriamiento de 0.0001.

6 CONCLUSIONES

Creo que se ha alcanzado el objetivo general planteado al inicio del proyecto, ya que se ha logrado de manera satisfactoria cumplir los objetivos específicos. Concretamente:

- Objetivo específico 1: Revisar el estado del arte y sacar conclusiones del mismo.

Se puede obtener como conclusión sobre la revisión del estado del arte que el tema planteado es de gran interés, ya que son numerosos los trabajos en temas similares. Dentro de las diferentes restricciones que se presuponen en las diferentes líneas de investigación se puede encontrar las que atienden al criterio de optimización, a los niveles de habilidad y agilidad, cuantificar el efecto del aprendizaje, el rendimiento de los operarios y la temporalidad de los mismos.

- Objetivo específico 2: Plantear el caso de estudio y desarrollar una heurística capaz de resolver el problema planteado, con el objetivo de minimizar el número de trabajadores en la línea de ensamblado.

En este caso, la heurística ha cumplido con el cometido de, tomando un paquete de datos formateados en Excel, indicando el número de estaciones, sus nombres y tiempos de operación para cada una de estas, crear paquetes de estaciones, asignando el número de operarios más adecuados que minimicen el takt total de ese paquete de estaciones.

- Objetivo específico 3: Comparación de los resultados del trabajo con los resultados obtenidos en el caso expuesto en la revisión de la literatura.

Se ha conseguido para el caso industrial, que coincide con el estudiado en el artículo una reducción de 3 operarios respecto a la heurística propuesta, suponiendo un total de 16 operarios para la línea, frente a los 19 propuestos en el artículo, luego se da por satisfecho el poder comparar con respecto a otros trabajos la eficacia de la heurística.

- Objetivo específico 4: Testear el funcionamiento en instancias similares de mayor tamaño para evaluar su funcionamiento.

De acuerdo a cumplir con este objetivo, se crearon los casos industriales a partir de la batería de datos del caso original, respetando la media y desviación típica de cada una de las estaciones, para generar unos datos lo más similares posible a la realidad de 40, 60 y 80 estaciones.

Tras aplicar la heurística a estos casos, se puede apreciar que las soluciones obtenidas se mantienen en un porcentaje del 10% de desviación respecto al mínimo teórico, garantizando el buen funcionamiento de la misma para problemas de mayor escala. En términos de tiempos, para encontrar la mejor solución en escalas mayores, el tiempo aumenta exponencialmente, acorde al aumento exponencial de las posibles secuencias solución.

Esto supone que la asignación de operarios en una línea de fabricación puede ser resuelto utilizando la heurística desarrollada mientras que las condiciones de contornos fijadas que suponen que los operarios se pueden mover de una estación a otra en un tiempo despreciable y se pueden retomar o incorporar a actividades que ya están en curso.

Es necesario tener en cuenta que, debido a la morfología de generación de posibles soluciones de los casos industriales, la componente aleatoria es introducida a la hora de disminuir en una unidad la secuencia base que recibe la heurística. Esto proporciona el punto positivo de evitar soluciones en mínimos locales, algo a lo que tiende la primera parte de la heurística que consiste en una búsqueda local.

Como ampliación futura sobre la misma línea de investigación, se podría considerar seguir integrando en la heurística restricciones que han sido consideradas en otros trabajos, como los mencionados en la revisión de la literatura, incluyendo: tener en cuenta la formación y experiencia de los operarios, rendimiento, limitar el movimiento de operarios, o aspectos como la eventualidad de los contratos, entre otros.

REFERENCIAS

1. Akagi, F. , Osaki, H. , Kikuchi, S. , 1983. A method for assembly line balancing with more than one worker in each station. *Int. J. Prod. Res.* 21(5), 755–770.
2. Araújo, F. F. B. , Costa, A. M. , Miralles, C. , 2012. Two extensions for the ALWABP: parallel stations and collaborative approach. *Int. J. Prod. Econ.* 140(1), 483–495.
3. Battaia, O. , Delorme, X. , Dolgui, A., Workforce minimization for a mixed-model assembly line, 2014.
4. Battaia, O. , Dolgui, A. , 2013. A taxonomy of line balancing problems and their solution approaches. *Int. J. Prod. Econ.* 142(2), 259–277.
5. Blazewicz, J. , Kovalyov, M. Y. , Machowiak, M. , Trystram, D. , Weglarz, J. , 2006. Preemptable malleable task scheduling problem. *IEEETrans. Comput.* 55(4), 486–490.
6. Blum, C. , Miralles, C. , 2011. On solving the assembly line worker assignment and balancing problem via beam search. *Comput. Oper. Res.* 38(1), 328–329.
7. Boenzi, F. , Mossa, G. , Mummolo, G. , Romano, V. A. , 2015. Workforce aging in production systems: modelling and performance evaluation. *ProcediaEng.* 100, 1108–1115.
8. Borba, L. , Ritt, M. , 2014. A heuristic and a branch-and-bound algorithm for the assembly line worker assignment and balancing problem. *Comput. Oper. Res.* 45, 87–96.
9. Costa, A. M. , Miralles, C. , 2009. Job rotation in assembly lines employing disabled workers. *Int. J. Prod. Econ.* 120(2), 625–632.
10. Corominas, A. , Pastor, R. , Plans, J. , 2008. Balancing assembly line with skilled and unskilled workers. *Omega*36(6), 1126–1132.
11. De Bruecker, P. , VandenBergh, J. , Beliën, J. , Demeulemeester, E. , 2015. Workforce planning incorporating skills: state of the art. *Eur. J. Oper. Res.* 243(1), 1–16.
12. Delorme, X. , Dolgui, A., Kovalev, S. , Kovalyov, M. , Minimizing the number of workers in a paced mixed-model assembly line, *European Journal of Operational Research*, Volume 272, Issue 1, 2019, Pages 188-194,
13. Dolgui, A. , Proth, J-M. , 2010. *Supply Chain Engineering: Useful Methods and Techniques*. Springer, London, UK. Fan, L. , Zhang, F. , Wang, G. , Liu, Z. , 2012.
14. Fan, L. , An effective approximation algorithm for the Malleable Parallel Task Scheduling problem. *J. ParallelDistrib. Comput.* 72, 693–704.
15. Fowler, J. W. , Wirojanagud, P. , Gel, E. S. , 2008. Heuristics for workforce planning with worker differences. *Eur. J. Oper. Res.* 190(3), 724–740.
16. Garey, M. R. , Johnson, D. S. , 1979. *Computers and Intractability: A Guide to the NP- Completeness*. W. H. Freeman&Company, NewYork, USA.
17. Grosse, E. H. , Glock, C. H. , 2015. The effect of worker learning on manual order picking processes. *Int. J. Prod. Econ.* [http://dx. doi. org/10. 1016/j. ijpe. 2014. 12. 018](http://dx.doi.org/10.1016/j.ijpe.2014.12.018), in press. Grosse, E. H. , Glock, C. H. , Jaber, M. Y. , 2013.
18. Grosse, E. H. , The effect of worker learning and forgetting on storage reassignment decisions in order picking systems. *Comput. Ind. Eng.* 66(4), 653–662.
19. Hewitt, M. , Chacosky, A. , Grasman, S. E. , Thomas, B. W. , 2015. Integer programming techniques for solving non-linear workforce planning models with learning. *Eur. J. Oper. Res.* 241(3), 942–950.

20. Ighravwe, D. E. , Oke, S. A. , 2014. A non-zero integer non-linear programming model for maintenance workforce sizing. *IntJ. Prod. Econ.* 150, 204–214.
21. Jansen, K. , Zhang, H. , 2012. Scheduling malleable tasks with precedence constraints. *J. Comput. Syst. Sci.* 78, 245–259.
22. Karabak, F. , Güner, N. D. , Satir, B. , Kandiller, L. , Gürsoy, I. , 2011. An optimization model for worker assignment of a mixed model vehicle production assembly line under worker mobility. In: *Proceedings of the 41st International Conference on Computers & Industrial Engineering*, pp. 483–490.
23. Kursat, M. , U-shaped assembly line worker assignment and balancing problem: A mathematical model and two meta-heuristics, *Computers & Industrial Engineering*, Volume 112, 2017, Pages 246-263,
24. Lutz, C. M. , Davis, K. R. , 1994. Development of operator assignment schedules: a DSS approach. *Omega* 22(1), 57–67.
25. Miralles, C. , García-Sabater, J. P. , Andrés, C. , Cardós, M. , 2007. Advantages of assembly lines in sheltered work centres for disabled. *Acasestudy. IntJ. Prod. Econ.* 110(1–2), 187–197.
26. Miralles, C. , García-Sabater, J. P. , Andrés, C. , Cardós, M. , 2008. Branch and bound procedures for solving the assembly line worker assignment and balancing problem: application to sheltered work centres for disabled. *Discret. Appl. Math.* 156, 357–367.
27. Moreira, M. C. O. , Costa, A. M. , 2013. Hybrid heuristics for planning job rotation schedules in assembly lines with heterogeneous workers. *IntJ. Prod. Econ.* 141 (2), 552–560.
28. Moreira, M. C. O. , Miralles, C. , Cost, A. M. , 2015. Model and heuristics for the assembly line worker integration and balancing problem. *Comput. Oper. Res.* 54, 64–73.
29. Mutlu, O. , Polat, O. , Supciller, A. A. , 2013. An iterative genetic algorithm for the assembly line worker assignment and balancing problem of type-II. *Comput. Oper. Res.* 40(1), 418–426.
30. Nakade, K. , Nishiwaki, R. , 2008. Optimal allocation of heterogeneous workers in a U-shaped production line. *Comput. Ind. Eng.* 54(3), 432–440.
31. Nakade, K. , Ohno, K. , 1999. An optimal worker allocation problem for a U-shaped production line. *Int. J. Prod. Econ.* 60–61(1), 353–358.
32. Niemi, E. , 2009. Worker allocation in make-to-order assembly cells. *Robot. Comput. -Integr. Manuf.* 25(6), 932–936.
33. Othman, M. , Bhuiyan, N. , Gouw, G. J. , 2012. Integrating workers' differences into workforce planning. *Comput. Ind. Eng.* 63(4), 1096–1106.
34. Özgüven, C. , Sungur, B. , 2013. Integer programming models for hierarchical workforce scheduling problems including excess off-days and idle labour times. *Appl. Math. Model.* 37(22), 9117–9131.
35. Sadykov, R. , 2012. A dominant class of schedules for malleable jobs in the problem to minimize the total weighted completion time. *Comput. Oper. Res.* 39, 1265–1270.
36. Shewchuk, J. P. , 2008. Worker allocation in lean U-shaped production lines. *Int. J. Prod. Res.* 46(13), 3485–3502.
37. Sungur, B. , Yavuz, Y. , 2015. Assembly line balancing with hierarchical worker assignment. *J. Manuf. Syst.* <http://dx.doi.org/10.1016/j.jmsy.2014.08.004>, inpress.
38. Techawiboonwong, A. , Yenradee, P. , Das, S. K. , 2006. A master scheduling model with skilled and unskilled temporary workers. *IntJ. Prod. Econ.* 103(2), 798–809.
39. Vairaktarakis, G. L. , Cai, X. , Lee, C. -Y. , 2002. Workforce planning in synchronous production systems. *Eur. J. Oper. Res.* 136(1), 551–572.
40. Venkatesh, J. , 2008. Evaluation of performance measures for representing operational objectives of a mixed model assembly line balancing problem. *Int. J. Prod. Res.* 46, 6367–6388.
41. Wilson, J. M. , 1986. Formulation of a problem involving assembly lines with multiple manning of

- workstations. *Int. J. Prod. Res.* 24(1), 59–63.
42. Yu, Y. , Tang, J. , Sun, W. , Yin, Y. , Kaku, I. , 2013. Reducing workers by converting assembly line into a pure cell system. *IntJ. Prod. Econ.* 145(2), 799–806.
43. Peltokorpi, J., Differences between worker pairs in manual assembly: a case study, *Procedia Manufacturing*, Volume 25, 2018, Pages 535-542,
44. https://en.wikipedia.org/wiki/Assembly_line
45. <https://www.coursehero.com/file/pdclfm/Una-l%C3%ADnea-de-ensamble-de-modelos-mixtos-es-una-l%C3%ADnea-de-producci%C3%B3n-manual-capaz/>
46. <https://www.cardinalevw.com/blog/volkswagen-announces-india-engine-assembly-project/>
47. <https://www.autotribute.com/38052/understanding-automotive-assembly-lines/>
48. <https://money.cnn.com/gallery/technology/2015/04/29/ford-factory-assembly-line-robots/3.html>
49. Apuntes asignatura Modelado y Sistemas de Gestión. Master en Organización Industrial y Gestión de empresas. Alejandro Escudero Santana. Pablo Aparicio Ruiz.
50. <https://www.allaboutlean.com/line-layout-strategies-part-1-the-big-picture/u-line-flexibility/>

ANEXOS

Anexo I. Código del caso industrial en Python

```
import time
import matplotlib.pyplot as plt
from random import randrange
import random
import math
import xlrd

start_time = time.time()
#LECTURA DE DATOS Y CREACIÓN DE DICIONARIOS
workbook=xlrd.open_workbook('DatosTFM.xlsx')
sheet=workbook.sheet_by_name('Hoja1')
Estaciones=[i for i in range(1, 20)]
NombreEstaciones={}
Tiempos={}
TiemposV1={}
TiemposV2={}
TiemposV3={}
Operarios={}
for i in Estaciones:
    Operarios[i]=0
    NombreEstaciones[i]=sheet.cell(i, 0).value
    Tiempos[i]=float(sheet.cell(i, 1).value)
    TiemposV1[i]=float(sheet.cell(i, 2).value)
    TiemposV2[i]=float(sheet.cell(i, 3).value)
    TiemposV3[i]=float(sheet.cell(i, 4).value)
TaktLimite=150
#####
#FUNCION DE EVALUACION PARA UN SECUENCIA DE OPERARIOS DADA
def EvaluacionFO(Operarios, Tiempos, Estaciones):
    Takt={}
    EstaciónTakt={} #Te dice que estaciones estan en cada takt
    EstacionesEliminar=Estaciones.copy()
    NombreEstacionesEliminar=NombreEstaciones.copy()
    k=1
    TodoBien="No"
```

```
for i in range(1, 20):
    Takt[i]=0
    EstaciónTakt[i]="
t=1
bandera=0
for i in range(0, len(Operarios)):
    bandera=0
    for j in EstacionesEliminar:
        m=j
        if Operarios[i]==0:
            break
        while Takt[k]+Tiempos[m]/Operarios[i] < TaktLimite and bandera==0:
            Takt[k]+=Tiempos[m]/Operarios[i]
            EstaciónTakt[k]=EstaciónTakt[k] + NombreEstaciones[m] + ", "
            EstacionesEliminar.pop(0)
            NombreEstacionesEliminar.pop(t)
            t=t+1
            m=m+1
            if len(EstacionesEliminar)==0:
                break
        bandera=1
    if len(EstacionesEliminar)==0:
        TodoBien="Si"
        break
    k=k+1
Total=0
for n in range(0, len(Operarios)):
    Total=Total+Operarios[n]
return(Total, TodoBien, Takt, EstaciónTakt)
#####
#FUNCION REPARADORA
def Reparadora(Operarios):
    posicaleatoria=randrange(len(Operarios))
    while Operarios[posicaleatoria]==4:
        posicaleatoria=randrange(len(Operarios))
    subeobaja=1    #0 disminuye y 1 aumenta la posición
    Operarios[posicaleatoria]=Operarios[posicaleatoria]+subeobaja
FO, Validete, toli, tolui=EvaluacionFO(Operarios, Tiempos, Estaciones)
```

```

#while Validete!='Si':
# Operarios=Reparadora(Operarios)
# FO, Validete, toli, tolu=EvaluacionFO(Operarios, Tiempos, Estaciones)
return(Operarios)
#####
#####
#Funcion ceros al fondo, te envia los ceros al fondo, la posicion de los numero no afecta en este problema
def Cerosalfondo(Operarios):
    return [nonZero for nonZero in Operarios if nonZero!=0] + [Zero for Zero in Operarios if Zero==0]
#####
# Introduciendo una Secuencia Inicial Aleatoria y Valida
#print("Secuencia aleatoria \n")
#Operarios=[2, 1, 2, 3, 2, 4, 2, 4, 2, 1, 1, 0, 0, 0, 0]
#TotalOperarios, Valida, Taktaso, Estaciónaso=EvaluacionFO(Operarios, Tiempos, Estaciones)
#print("Numero de operarios en secuencial:", TotalOperarios)
#print("Secuencia válida?:", Valida)
#OperariosUsados=0
#for k in range(1, len(Taktaso)):
# if Taktaso[k]>0:
#     print("El Takt del paquete %d es de %d min con las estación/es %s y %d operario/s. " %(k, Taktaso[k],
Estaciónaso[k], Operarios[k-1]))
#     OperariosUsados=OperariosUsados+Operarios[k-1]
#print("Numero operarios usados: %d" %(OperariosUsados))
#####
#####
#SE CREA UNA RUTA INICIAL QUE VA ASIGNANDO OPERARIOS
#Se asigna el maximo numero de operarios hasta que se cumpla el takt y la secuencia sea válida.
print("\n")
print("Heuristica inicial \n")
Operarios=[0] * len(NombreEstaciones) #Se crea un vector de 0 vacio de tamaño lo que sea.
for i in range(0, len(Operarios)):
    Operarios.insert(i, 4)
    Operarios.pop(-1)
    TotalOperarios, Valida, Taktaso, Estaciónaso=EvaluacionFO(Operarios, Tiempos, Estaciones)
    if Valida=='Si':
        OperariosSecIni=Operarios.copy()
        break
print("Numero de operarios en secuencia inicial:", TotalOperarios)
print("Secuencia válida?:", Valida)

```

```
print("Secuencia:", Operarios)
OperariosUsados=0
for k in range(1, len(Taktaso)):
    if Taktaso[k]>0:
        print("El Takt del paquete %d es de %d min con las estación/es %s y %d operario/s. " %(k, Taktaso[k],
Estaciónaso[k], Operarios[k-1]))
        OperariosUsados=OperariosUsados+Operarios[k-1]
print("Numero operarios usados: %d" %(OperariosUsados))
print("\n")
#####
###LOCAL SEARCH con GENERACION UNITARIA ALEATORIA###
#Realizamos una Local Search del copon
def LocalSearch(Operarios, Tiempos, Estaciones): #Se define la funcion de busqueda local
    SecProb=Operarios.copy()          #Se copia secuencia de entreda para evitar que se modifique la lectura
del bucle for
    SecActual=SecProb.copy()          #Se copia la secuencia de prueba para hacer de secuencia Actual
    ValActual, tt, yy, ss=EvaluacionFO(Operarios, Tiempos, Estaciones)#Se evalua la secuencia Actual
    Vecino=SecProb.copy()
    ValNueva=ValActual
    while True:                        #Se realiza el bucle hasta que se deja de mejorar
        #Vecindario=[]                #Se crea una lista vacia para los vecinos de la secuencia introducida
        posicaleatoria=randrange(len(Vecino))
        subeobaja=-1                    #0 disminuye y 1 aumenta la posición
        while Vecino[posicaleatoria]==0:
            posicaleatoria=randrange(len(Vecino))
        print("La poscion", posicaleatoria, "baja en ", subeobaja)
        Vecino[posicaleatoria]=Vecino[posicaleatoria]+subeobaja
        print("Vecino generado:", Vecino)
        juju, Validin, toletus, tolaiyer=EvaluacionFO(Vecino, Tiempos, Estaciones)
        print("Es valida:", Validin)
        if Validin=='No':
            Vecino=Reparadora(Vecino)
            juju, Validin, toletus, tolaiyer=EvaluacionFO(Vecino, Tiempos, Estaciones)
            Vecino=Cerosalfondo(Vecino)
            print("Genero un vecino aleatorio", Vecino)
            FO, Validete, toli, tolui=EvaluacionFO(Vecino, Tiempos, Estaciones)
            print("Vecino reparado valido?:", Validete)
            if Validete=='Si' and FO>0 and FO<=ValNueva:                #En el caso de mejorar y de ser valida, FO mayor
que 0 en evaluacion entrar
```



```
    SecNueva=Vecino.copy()          #Se guarda el mejor vecino de esta iteracion
    ValNueva=FO                     #El nuevo valor es el de la función objetivo
    print("Vecino aleatorio es igual de bueno que el anterior")
    subeobaja=randrange(-1, 1, 2)
    Vecino
    ValNueva=ValActual              #La evaluación de la nueva es la de la secuencia actual
    #for i in Vecindario:           #Para todos los vecinos
        FO, Validete, toli, toli=EvaluacionFO(Vecino, Tiempos, Estaciones)      #Se calcula su valor de la
funcio objetivo
        #print(Vecino, Validete, FO, ValNueva)
        if Validete=='Si' and FO>0 and FO<=ValNueva:      #En el caso de mejorar y de ser valida, FO mayor
que 0 en evaluacion entrar
            SecNueva=Vecino.copy()          #Se guarda el mejor vecino de esta iteracion
            ValNueva=FO                     #El nuevo valor es el de la función objetivo
            print("Se genera una mejor")
            print("Sec.Nueva:", SecNueva, "FO Sec.Nueva:", ValNueva) #Se imprime esta nueva mejor solucion
y su evaluacion
        if ValNueva<ValActual:           #Dentro de este vecindario, si el mejor vecino es mejor que el absoluto
            SecActual=SecNueva.copy()      #Guardamos este vecino como el mejor
            ValActual=ValNueva            #Guardamos el valor de la FO
            Vecindario=[]                 #Limpiamos el vecindario para la siguiente iteracion
    else:
        print("--- %s seconds ---" % (time.time() - start_time))
        break
    return SecNueva                    #Devolvemos la mejor secuencia y valor de la FO
#####
#SE LLAMA A LA LOCAL SEARCH
print("Local Search \n")
print("Sec. Ini. introducida en LS:", OperariosSecIni)
Rutaso=LocalSearch(OperariosSecIni, Tiempos, Estaciones)
TotalOperarios, Valida, Taktaso, Estacionaso=EvaluacionFO(Rutaso, Tiempos, Estaciones)
Operarios=Rutaso.copy()
print("\n")
print("Numero de operarios en secuencia inicial:", TotalOperarios)
print("Secuencia válida?:", Valida)
print("Secuencia:", Operarios)
OperariosUsados=0
for k in range(1, len(Taktaso)):
    if Taktaso[k]>0:
```

```
print("El Takt del paquete %d es de %d min con las estación/es %s y %d operario/s. " %(k, Taktaso[k],
Estaciónaso[k], Operarios[k-1]))
OperariosUsados=OperariosUsados+Operarios[k-1]
print("Numero operarios usados: %d" %(OperariosUsados))
print("\n")
#CosteFinal=FuncionEvaluadora(Rutaso)
#print("Coste Final", CosteFinal)
#####
#RECOSIDO SIMULADO
def RecocidoSimulado(Operarios, Tiempos, Estaciones):
    TempMax= 100 #Temperaturas maximas y minimas
    TempMin = 1
    #NumIteraciones = 2 #Numero de iteracions de la temperatura
    alpha=0.0001 #Alpha velocidad de enfriamiento
    Temperatura = TempMax #Guardamos temperatura como maxima
    SolActual=Operarios.copy()
    ValActual, Validete, tolete, tolai=EvaluacionFO(SolActual, Tiempos, Estaciones)
    SolMejor=SolActual.copy()
    ValMejor=EvaluacionFO(SolMejor, Tiempos, Estaciones)
    Tiempaso={}
    Valoraso={}
    k=1
    Vecino=Operarios.copy()
    while Temperatura > TempMin:
        posicaleatoria=randrange(len(Vecino))
        subeobaja=-1 #0 disminuye y 1 aumenta la posición
        while Vecino[posicaleatoria]==0:
            posicaleatoria=randrange(len(Vecino))
        print("La poscion", posicaleatoria, "varia en ", subeobaja)
        Vecino[posicaleatoria]=Vecino[posicaleatoria]+subeobaja
        Vecino=Cerosalfondo(Vecino)
        print("Vecino generado normal:", Vecino)
        juju, Validin, toletus, tolaiyer=EvaluacionFO(Vecino, Tiempos, Estaciones)
        print("Es valida:", Validin)
        if Validin=='No':
            VecinoRep=Reparadora(Vecino)
            juju, Validin, toletus, tolaiyer=EvaluacionFO(VecinoRep, Tiempos, Estaciones)
            VecinoRep=Cerosalfondo(VecinoRep)
```

```
print("Vecino reparado", VecinoRep)
FO, Validete, toli, tolui=EvaluacionFO(VecinoRep, Tiempos, Estaciones)
print("Vecino reparado valido?:", Validete)
if Validete=='Si' and FO>0 and FO<=ValActual:      #En el caso de mejorar y de ser valida, FO mayor
que 0 en evaluacion entrar
    Vecino=VecinoRep.copy()      #Se guarda el mejor vecino de esta iteracion
    ValNueva=FO      #El nuevo valor es el de la función objetivo
    print("Vecino aleatorio es igual de bueno que el anterior")
    order=list(range(len(Vecino)))
random.shuffle(order)
    for i in range(len(Vecino)):
        SolNueva = Vecino.copy()
        ValNueva, Validete, tolete, tolai=EvaluacionFO(SolNueva, Tiempos, Estaciones)
        if Validete=='Si':
            delta = ValNueva - ValActual
            if ValNueva==0:
                break
            else:
                if delta<0:
                    ValActual = ValNueva
                    SolActual = SolNueva.copy()
                    ValMejor = ValActual
                    SolMejor = SolNueva.copy()
                    print("Numero de operarios:", ValActual)
                    print("Secuencia:", SolActual)
                    Tiempaso[k]=time.time() - start_time
                    Valoraso[k]=ValMejor
                    k+=1
                    break
                else:
                    a=random.random()
                    if a < math.exp(-delta/Temperatura):
                        ValActual = ValNueva
                        SolActual = SolNueva.copy()
                        print("Numero de operarios:", ValActual)
                        print("Secuencia:", SolActual)
                        break
Temperatura = Temperatura / (1+alpha * Temperatura)
```

```
print("----- T=", Temperatura)
print("\n")
return SolMejor
#####
#SE LLAMA AL RECOCIDO SIMULADO
Rutaso=RecocidoSimulado(Operarios, Tiempos, Estaciones)
print("Mejor ruta Recocido Simulado:", Rutaso)
TotalOperarios, Valida, Taktaso, Estaciónaso=EvaluacionFO(Rutaso, Tiempos, Estaciones)
Operarios=Rutaso.copy()
print("\n")
print("Numero de operarios en secuencia inicial:", TotalOperarios)
print("Secuencia válida?:", Valida)
print("Secuencia:", Operarios)
OperariosUsados=0
for k in range(1, len(Taktaso)):
    #if Operarios[k]>0:
    if Taktaso[k]>0:
        print("El Takt del paquete %d es de %d min con las estación/es %s y %d operario/s. " %(k, Taktaso[k],
Estaciónaso[k], Operarios[k-1]))
        OperariosUsados=OperariosUsados+Operarios[k-1]
print("Numero operarios usados: %d" %(OperariosUsados))
print("\n")
```