



Programa de Doctorado en Estadística, Optimización y
Matemática Aplicada

**Optimización de la secuencia
de pick and place de un robot
colaborativo bimanual en una
línea de producción
industrial**

Jorge Borrell Méndez

Director de la tesis

Dr. José Vicente Segura Heras

Codirector de la tesis

Dr. Carlos Pérez Vidal

Universidad Miguel Hernández de Elche

La presente Tesis Doctoral, titulada “Optimización de la secuencia de pick and place de un robot colaborativo bimanual en una línea de producción industrial”, se presenta bajo la modalidad de **tesis convencional con el/los siguiente/s indicios de calidad:**

(Referenciar los artículos/publicaciones/libros/capítulos de libro/...)

- Borrell Méndez J., Perez-Vidal C., Segura Heras J. V. and Pérez-Hernández J. J., 2020, “Robotic Pick-and-Place Time Optimization: Application to Footwear Production”, IEEE Access, vol. 8, pp. 209428-209440. DOI: <https://doi.org/10.1109/ACCESS.2020.3037145>
- Borrell Méndez J, Cremades D, Nicolas F, Perez-Vidal C, Segura-Heras JV. Conceptual and Preliminary Design of a Shoe Manufacturing Plant. Applied Sciences. 2021; 11(22):11055. DOI: <https://doi.org/10.3390/app112211055>
- Borrell Méndez J., Perez-Vidal C., and Segura Heras J. V., Optimization of the pick and place sequence of a bimanual collaborative robot in an industrial production line, International Journal of Advanced Manufacturing Technology DOI: <https://doi.org/10.1007/s00170-023-12922-9>



El Dr. *Jose Vicente Segura heras*, director, y el Dr. *Carlos Pérez Vidal*, codirector de la tesis doctoral titulada “Optimización de la secuencia de pick&place de un robot colaborativo bimanual en una línea de producción industrial”

INFORMAN:

Que D. *Jorge Borrell Méndez* ha realizado bajo nuestra supervisión el trabajo titulado “Optimización de la secuencia de pick&place de un robot colaborativo bimanual en una línea de producción industrial” conforme a los términos y condiciones definidos en su Plan de Investigación y de acuerdo al Código de Buenas Prácticas de la Universidad Miguel Hernández de Elche, cumpliendo los objetivos previstos de forma satisfactoria para su defensa pública como tesis doctoral.

Lo que firmo/firmamos para los efectos oportunos, en Elche a 11 de Enero de 2024

Director de la tesis

Dr. José Vicente Segura Heras

Codirector de la tesis

Dr. Carlos Pérez Vidal



El Dr. Domingo Morales González, Coordinador del Programa de Doctorado en Estadística, Optimización y Matemática Aplicada.

INFORMA:

Que D. Jorge Borrell Méndez ha realizado bajo la supervisión de nuestro Programa de Doctorado el trabajo titulado "Optimización de la secuencia de pick&place de un robot colaborativo bimanual en una línea de producción industrial" conforme a los términos y condiciones definidos en su Plan de Investigación y de acuerdo al Código de Buenas Prácticas de la Universidad Miguel Hernández de Elche, cumpliendo los objetivos previstos de forma satisfactoria para su defensa pública como tesis doctoral.

Lo que firmo para los efectos oportunos, en Elche a 11 de Enero de 2024

Prof. Dr. Domingo Morales González

Coordinador/a del Programa de Doctorado en Estadística, Optimización y Matemática Aplicada

A Papá

Siempre nos protegerás

“Hasta el infinito y más allá”

Agradecimientos

Con esta tesis se acaba una de las grandes etapas de mi vida, la cual ha traído grandes cambios tanto a nivel profesional como personal. Han sido años de cambio, pero si hubiese una palabra que definiese perfectamente esta etapa sería lucha. Porque no ha habido un día que no haya luchado, contra lo que tocase en ese momento, me han educado así, que hay que afrontar lo que venga y siempre de la mejor manera posible. Siempre vamos a tener problemas, lo que nos tenemos que preocupar es que esos problemas sean mejores. Esta forma de vivir y ser, me la enseñó una de las personas más importantes en mi vida, en gran parte él es uno de los responsables de que yo haya tanto iniciado como concluido mi tesis. Mi padre me ha enseñado a pelear y a luchar, a levantarme una y otra vez, habilidades que considero fundamentales en investigación, en este mundo tan duro y competitivo no puedes venirte abajo fácilmente. Considero que al final no hay mayor agradecimiento hacia una persona que te ha dado tanto que seguir peleando por tu camino y por tu vida, seguir convirtiéndote en esa persona de la cual estaría tremendamente orgullosa y ese es mi objetivo día tras día.

Podría continuar en esta línea pero tampoco sería algo natural en mí, soy una persona concisa que no le da mil vueltas siempre a lo mismo. Por extensión, y con la misma responsabilidad de que haya empezado y terminado la tesis la tiene la otra parte del tandem que han conformado, en gran medida, mi forma de ser y que siempre me ha dejado decidir mi camino y tomar mis propias decisiones, aconsejándome activamente, mi madre. Y por último pero no menos importante, la cuarta pata de la mesa, la princesa de la casa, que también en gran medida ha formado parte de esta aventura y ha demostrado lo que he recalcado desde el principio, la gran capacidad de lucha y sacrificio que tenemos. Porque entre los cuatro formamos un gran equipo, un equipo perfectamente equilibrado, que es capaz de enfrentarse a cualquier cosa. Estemos donde estemos, siempre estaremos juntos.

El último año de la tesis, como no podía ser de otra manera ha sido un poco turbulento, no teníamos ni suerte ni salían las cosas como esperábamos, eso sí, siempre he estado rodeado de gente que me hacía crecer, que me apoyaba en todo lo que

hacía, pero los últimos meses fueron de consolidación, terminé de rodearme de gente increíble, con una personita que brilla por ella misma y que hace que todo sea más fácil. Ahora tengo una base muy sólida y creo que ahora sí que va a venir toda la suerte y el trabajo bien hecho que no hemos tenido durante la tesis.

Por último, agradecer a mis tutores, Carlos Pérez y Jose Vicente Segura, la oportunidad que me han dado en el desarrollo de esta tesis, siempre a mi lado y pendientes para sacar esta investigación adelante, luchando juntos (otra vez sale la lucha en esta etapa) contra las adversidades que nos hemos encontrado, siempre intentando encontrar la mejor solución y a mi compañero de laboratorio Fran, por todo el trabajo que hemos hecho juntos.

Contenido

Índice de figuras	x
Índice de tablas	xii
Resumen	xv
Abstract	xvii
1 Introducción	2
1.1 Automatización en el sector industrial del calzado	3
1.2 Sistemas multi-robot	7
1.3 Optimización de la trayectoria	11
1.4 Estructura de la memoria presentada	13
2 Objetivos	14
3 Materiales y Métodos	18
3.1 Descripción del sistema	18
3.1.1 Características del robot colaborativo y el procesamiento de imágenes	22
3.1.2 Generación de secuencias	24
3.2 Análisis Estadístico de las secuencias de montaje	25
3.3 Problema de programación lineal entera y binaria (BILP)	26
3.3.1 Modelo de 4 piezas	26
3.3.2 Modelo de n piezas	31
4 Resultados	34
4.1 Diseño de una fábrica de calzado de inyección de polímero.	34

4.2	Análisis de las secuencias de montaje	40
4.2.1	Valores ausentes, colisiones y asociaciones	41
4.2.2	Resultados de cada secuencia	44
4.2.3	Selección de la secuencia adecuada	48
4.3	Optimización de la trayectoria de montaje	52
4.3.1	Coste computacional	56
4.3.2	Generalización para n piezas	58
5	Discusión	62
6	Conclusiones	66
7	Conclusions	68
8	Líneas futuras	70
	Bibliografía.	73
	Glosario	81
	Anexo 1 - Código en Python del BILP	83

Índice de figuras

Figura 1.1	Pasos para fabricar un par de zapatos.	4
Figura 1.2	Máquina rotativa DESMA	5
Figura 2.1	Flujograma del sistema de automatización. Resaltada la etapa en la que se va a centrar la tesis.	15
Figura 3.1	Sistema prototipo real	20
Figura 3.2	Modelos de zapatos	21
Figura 3.3	IRB 14000 YuMi.	22
Figura 3.4	Plano de las dimensiones del IRB 14000 YuMi.	23
Figura 3.5	Diagrama con todas las posibilidades.	28
Figura 3.6	Secuencia con bucle.	31
Figura 4.1	Diagrama de clases del marco conceptual de referencia.	36
Figura 4.2	Bloques lógicos y configuración de la transición entre las dos fases. Estos bloques están asociados al proceso de inyección directa.	38
Figura 4.3	Diagrama de bloques del proceso de inyección directa.	39
Figura 4.4	Representación espacial del sistema de inyección directa.	40
Figura 4.5	Sistema simulado en Robotstudio.	41
Figura 4.6	Valores ausentes y número de colisiones.	42
Figura 4.7	Asociación entre cada grupo de secuencias.	43

Figura 4.8	Distribución del tiempo medio para cada secuencia.	44
Figura 4.9	Distribución mejores secuencias.	47
Figura 4.10	Árbol de decisión condicional con el modelo de las seis mejores secuencias.	49
Figura 4.11	Árbol de decisión condicional con el modelo de las tres mejores secuencias.	51
Figura 4.12	Soluciones óptimas para diferentes posiciones de montaje. Fila superior para el modelo de zapato del Fabricante 1. Fila inferior para el modelo de zapato del Fabricante 2.	53
Figura 4.13	Diagrama de cajas y gráfico de densidad para la distribución de distancias recorridas con cada fabricante.	55
Figura 4.14	Diagrama de cajas del coste computacional en segundos.	57
Figura 4.15	Coste computacional en función del número de bandejas.	57
Figura 4.16	Soluciones óptimas para distintos números de piezas, 4, 5, 6 y 7 respectivamente. La línea azul corresponde al brazo rojo y la línea azul al brazo izquierdo.	59
Figura 4.17	Distancia óptima media y coste computacional para distinto número de piezas.	60
Figura 5.1	Comparativa entre el modelo de árbol de decisión y el modelo BILP.	64

Índice de tablas

Tabla 3.1	Secuencia de combinación entre los brazos en el pick	24
Tabla 3.2	Notación del modelo BILP.	27
Tabla 4.1	Características de la secuencia óptima para el Fabricante 1 y Tiempos de fabricación por producto. Esta información está asociada a la inyección directa y al proceso de montaje y pegado.	37
Tabla 4.2	Características de la secuencia óptima para el Fabricante 1 y Tiempos de fabricación por producto. Esta información está asociada a la inyección directa y al proceso de montaje y pegado.	40
Tabla 4.3	Principales parámetros para el tiempo de montaje con cada secuencia del modelo del Fabricante 1.	45
Tabla 4.4	Principales parámetros para el tiempo de montaje con cada secuencia del modelo del Fabricante 2.	46
Tabla 4.5	Ajuste y validación en árbol de seis secuencias.	50
Tabla 4.6	Ajuste y validación en árbol de tres secuencias.	51
Tabla 4.7	Resumen resultados.	52
Tabla 4.8	Datos de la secuencia óptima.	54
Tabla 4.9	Resumen del coste computacional en segundos.	56

Optimización de la secuencia de pick and place en una línea de producción industrial

Jorge Borrell Méndez

Resumen

Esta tesis se centra en la optimización de las tareas de pick-and-place realizadas por un robot colaborativo de doble brazo, en un entorno particular de la industria de fabricación de calzado. El robot tiene que identificar las piezas de un zapato colocadas en una bandeja, y recogerlas y colocarlas en un molde de calzado para su posterior procesamiento. Las piezas del zapato llegan a la bandeja en posiciones y ángulos aleatorios (patrones) y pueden recogerse en distinto orden. La optimización de estas tareas permitirían aumentar la rapidez en el montaje de cada unidad y por tanto, mejorar la producción de zapatos. Para ello se desarrolla un modelo de árbol de decisión y un modelo matemático basado en programación lineal entera binaria (BILP). El primero nos permite identificar una secuencia fija para el montaje del calzado, mientras que el segundo encuentra la secuencia óptima para recoger y colocar las piezas del zapato en el molde, con lo que se minimiza el tiempo de recogida y de toma de decisiones. Las dos soluciones propuestas se prueban con dos modelos de zapato de 3 piezas unitarias, uno de entrenamiento y otro de validación, que comprenden un total de 500 bandejas. Los resultados obtenidos se comparan entre si. El análisis de los resultados de este estudio muestra que el BILP presenta ventajas para la planificación del movimiento de tareas en un entorno complicado con múltiples trayectorias y potencial colisión entre brazos. Se generaliza el modelo BILP para zapatos con n piezas de montaje, confirmando la robustez del modelo para cualquier número de piezas.

Abstract

This thesis focuses on the optimisation of pick and place tasks performed by a dual-arm collaborative robot in a specific environment of the shoe manufacturing industry. The robot has to identify, pick and place pieces of a shoe placed on a tray into a shoe mould for further processing. The shoe pieces arrive on the tray in random positions and angles (patterns) and can be picked up in different orders. Optimising these tasks would increase the speed of assembly of each unit and therefore improve the production of shoes by developing a decision tree model and a mathematical model based on binary integer linear programming (BILP). The former allows us to identify a fixed sequence for shoe assembly, while the latter finds the optimal sequence for picking up and placing the shoe pieces in the mould, thus minimising picking and decision time. The two proposed solutions are tested on two shoe models of 3 pieces, one for training and one for validation, with a total of 500 trays. The results obtained are compared. The analysis of the results of this study shows that the BILP has advantages for task motion planning in a complicated environment with multiple trajectories and potential collisions between arms. The BILP model is generalised for shoes with n assembly pieces, confirming the robustness of the model for any number of pieces.

Capítulo 1

Introducción

Los procesos de fabricación, trasladados en su momento a países emergentes por el bajo coste de la mano de obra, están volviendo a localizarse en entornos cercanos. Para compensar los costes de estos traslados las empresas están apostando por la automatización de procesos y el análisis de una ingente cantidad de datos generados en cada uno de los mismos. De esta forma esperan compensar, junto al ahorro en el transporte, los costes laborales de producción.

Nuestro entorno industrial está centrado en el sector del calzado, del que Elche es un referente a nivel nacional. Muchas de las empresas de este tejido industrial trasladaron su producción a países como China, y en estos momentos están volviendo a fabricar donde comenzaron su andadura. Esto ha motivado el interés de algunas de ellas por colaborar con la universidad para establecer alianzas que permitan el desarrollo de proyectos de I+D+I. Esta tesis doctoral es el resultado de uno de esos acuerdos, y por ese motivo está catalogada como doctorado industrial.

La introducción a la temática de la tesis se ha estructurado en tres secciones. En la primera se abordará la situación actual de la automatización en la industria, centrándose en particular en la industria del calzado. Se analizarán los sistemas y tecnologías de automatización utilizados en esta industria destacando las ventajas y limitaciones de cada uno de ellos. Se examinarán los procesos de producción y las operaciones que se realizan, identificando las áreas donde la automatización podría tener un mayor impacto en términos de eficiencia, calidad y seguridad.

En la segunda se evalúa el papel de la robótica multisistema. Se explorará cómo esta tecnología puede ayudar a mejorar la flexibilidad y la adaptabilidad de los sistemas de automatización, permitiendo la realización de múltiples tareas en una sola máquina. Se examinarán las diferentes configuraciones de estos sistemas y se

analizarán los retos y oportunidades que presentan en el contexto de la industria del calzado.

Por último, se estudiará la optimización de trayectorias en el problema de montaje de un zapato. En particular se analizará como se pueden mejorar los movimientos de los robots y maquinarias automatizadas, de manera que los procesos de producción sean más eficientes y precisos. Se explorarán las técnicas de optimización de trayectorias más relevantes y se analizarán cómo se pueden aplicar en la industria del calzado para obtener mejoras significativas en la calidad del producto y la productividad.

1.1. Automatización en el sector industrial del calzado

La fabricación de calzado no ha sufrido cambios significativos a lo largo de los años. A pesar del reciente uso de maquinaria, el proceso es principalmente artesanal, muy laborioso (unas 80 operaciones diferentes) y especialmente complejo, por lo que el coste de la mano de obra es una de las partes más importantes de la estructura de costes del calzado [1]. La Figura 1.1 muestra algunas de las tareas que se realizan habitualmente en la industria del calzado. Los principales pasos que se realizan son los siguientes:

- Corte: Se cortan las diferentes piezas (de cuero, lona sintética, ante, textil, . . .) que formarán la parte superior del calzado (Figura 1.1(A5)). Además, las piezas cortadas se preparan para los siguientes procesos haciendo marcas de guía o uniones temporales.
- Costura: Las piezas superiores se cosen con una máquina de coser (Figura 1.1(B5-A4)).
- Montaje: La parte superior se prepara colocando el contrafuerte del talón, el ribete y la puntera si es necesario. Algunas piezas se preforman y se montan utilizando una horma (Figura 1.1(B4-C1)).
- Fijación (o inyección) de la suela: La suela se fija a la parte superior (método de montaje y pegado) o se inyecta directamente a la parte superior (método de inyección directa). En la Figura 1.1 se muestra el método de montado y pegado (F3, A1, B1).

- Acabado: En este último paso el producto adquiere su aspecto final. Englobaría los procesos de limpieza, colocación de plantillas, cordones, etc. El producto se embala y se almacena (Figura 1.1(D1-I1)).

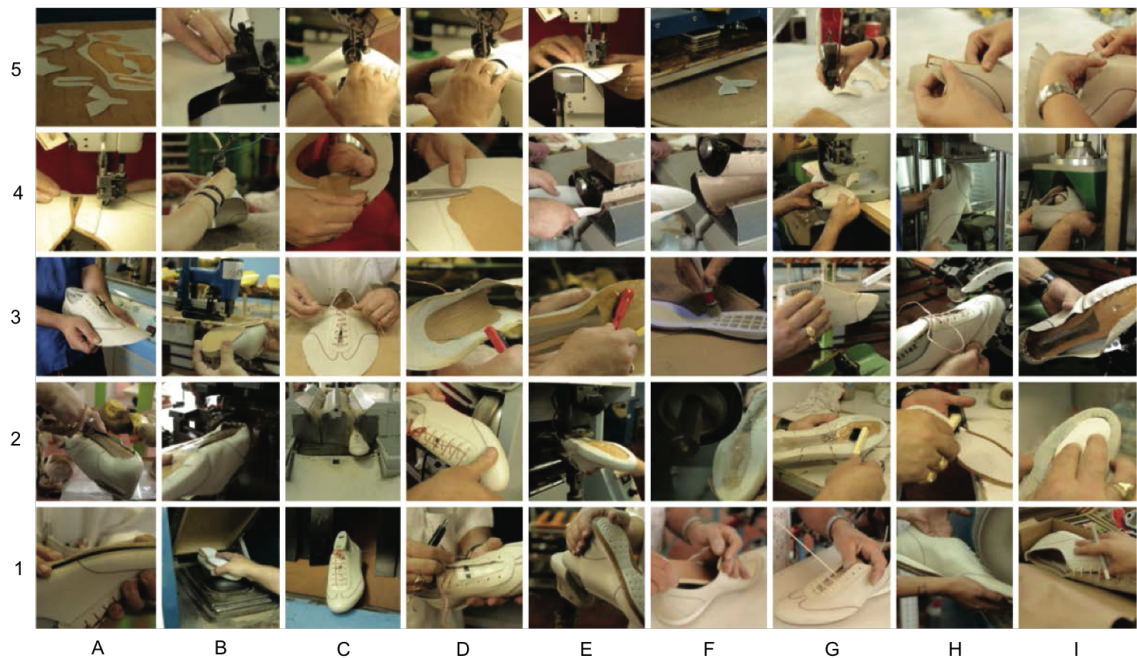


Figura 1.1: Pasos para fabricar un par de zapatos. Cortesía de Simplicity Works Europe.

La fijación o inyección de la suela es el paso clave del proceso de producción y puede determinar el éxito o fracaso del producto final. La elección de la técnica de fijación depende de varios factores, como el tipo de zapato, el material de la suela o la cantidad de pares de zapatos que se van a producir

El método más común de fijación se denomina montaje y pegado. Esta técnica implica pegar la suela a la parte superior del zapato utilizando una capa de adhesivo. El proceso se puede realizar manualmente mediante un pincel impregnado de adhesivo o mediante una máquina o robot para garantizar una aplicación uniforme del adhesivo en toda la superficie. El pegado de la suela y el empeine se realiza mediante una máquina de prensa con base deformable que asegura una unión perfecta entre los dos elementos.

Una vez pegadas la suela y la parte superior del zapato, se debe reactivar el adhesivo mediante un proceso de calor en una cabina especializada. La aplicación uniforme del adhesivo es fundamental para asegurar una sujeción perfecta, y se debe garantizar que se logre una temperatura óptima para reactivar el mismo. Una vez que se ha reactivado, se debe unir la suela al empeine con precisión para que ambos queden perfectamente centrados. Esto se logra mediante el uso de una máquina de prensa con base deformable que se ajusta según el tipo y la forma del zapato. Éste es un proceso tradicional y antiguo como se describe en [2].

El método alternativo que actualmente está ganando terreno debido a la eficacia y la calidad del producto se denomina proceso de inyección directa [3],[4]. En este método, la suela se inyecta directamente sobre la parte superior creando una fuerte unión entre los materiales, lo que hace que el calzado sea más duradero. Las operaciones de corte y cosido en la parte superior se realizan exactamente igual que en el proceso de montaje y pegado, pero cuando la parte superior está preparada, se coloca en una máquina de inyección para unir la misma de forma permanentemente a la suela. Normalmente, el proceso de inyección directa utiliza una máquina rotativa (véase la Figura 1.2) con múltiples estaciones de moldeo para permitir la colocación de la parte superior montada. Se pueden realizar una o dos inyecciones, y el proceso de curado se lleva a cabo en función del movimiento de la máquina. El resultado final es una suela resistente, flexible y duradera, que se une directamente al empeine del zapato. La principal ventaja de la inyección directa es la rapidez del proceso, lo que

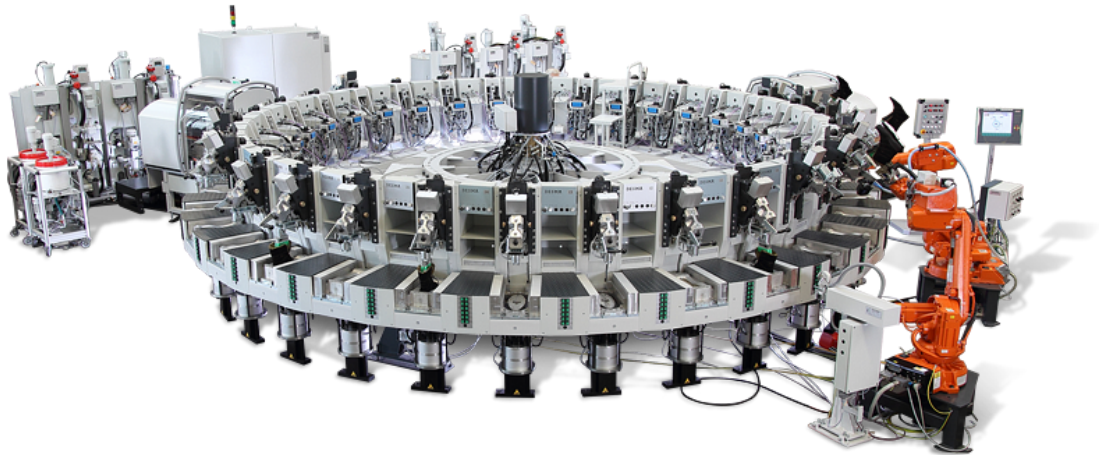


Figura 1.2: Máquina rotativa DESMA equipada con dos inyectores y células robotizadas. Cortesía de DESMA Schuh-maschinen GmbH.

permite producir grandes cantidades de pares de zapatos en poco tiempo. Además, se requiere menos mano de obra y se produce menos desperdicio. En ambos procesos, se utilizan robots para lijar la piel base con el fin de aumentar la adherencia de la suela, aplicar cola para el pegado, o recoger las piezas para el molde [5]-[7].

El coste económico de la máquina de inyección rotativa es significativamente superior al de la maquinaria equivalente para realizar el proceso de montaje y pegado, pero tiene dos grandes ventajas. Por un lado, la productividad de la inyección directa es mayor, por otro, la calidad del producto a comercializar justifica un incremento en el precio del mismo, lo que facilita el retorno de la inversión realizada en maquinaria. En relación con el coste de la mano de obra, los salarios en algunos países de Europa del Este e incluso en algunas zonas de Portugal ya son competitivos en comparación con el coste medio de la mano de obra en China (aunque no por el momento en Vietnam). Esto hace que la producción de calzado en Europa sea una opción cada vez más atractiva para las empresas de producción.

Estos procesos en la industria del calzado llevan al uso de diversas tecnologías:

- La tecnología de tricotado 3D [8] permite producir, con una sola máquina, una parte superior de zapato completa y sin costuras, lista para ser conectada directamente a la suela. Esto permite eliminar completamente el corte, la costura y el termofijado dependientes de la mano de obra, la máquina de tejer es cargada con fibras de poliéster, nailon o elastano.
- 3DPrinting [9], esta tecnología también se utiliza para imprimir algunas piezas del zapato en plástico, pudiendo generar piezas más complejas y personalizadas para cada caso.
- 3DBonding [10] se caracteriza por inyectar polímero en un molde para fabricar el zapato. Esta tecnología permite unir diferentes piezas de material inyectando un polímero en un conjunto de canales dentro de un molde. Esta tecnología permite reducir el material utilizado, tanto para el polímero inyectado como para el textil o el cuero, ya que las piezas no se superponen entre sí. Durante el proceso de fabricación, muchas de las operaciones tradicionales, como el cosido, se eliminan, de esta manera también se reducen los tiempos y costes de producción en esta fase.

Esta última tecnología permite la automatización del proceso de montaje mediante la robotización de tareas de pick and place. Las piezas llegan en una bandeja en posiciones aleatorias. Un robot con uno o dos brazos las coloca en el molde en el que se inyecta el polímero y en pocos segundos el zapato está completado.

Existe una tendencia creciente a la automatización de la industria actual buscando una mejora en diferentes variables del proceso productivo. La automatización en las líneas de fabricación permite mejorar el tiempo de producción, la precisión en tareas complejas o la repetitibilidad. En el caso de la industria del calzado todavía se tiene una automatización baja en comparación con otros sectores como puede ser el automovilístico o el del juguete. Los últimos años se han realizado una gran cantidad de esfuerzos para automatizar este sector y hacerlo más competitivo.

La implementación de tecnologías avanzadas, como robots y sistemas de visión artificial, ha permitido mejorar la eficiencia de los procesos de fabricación del calzado, reducir los costos de producción y mejorar la calidad de los productos finales.

Sin embargo, aún existen desafíos importantes que deben ser abordados para lograr una automatización completa en la industria del calzado. Uno de ellos es la complejidad de los diseños de calzado, lo que hace que la automatización de ciertas tareas, como el ensamblaje de las diferentes partes del calzado, sea más difícil. Además, la falta de estandarización en los procesos y la gran variedad de materiales utilizados en la fabricación también son desafíos importantes que deben ser abordados para lograr una automatización completa y eficiente.

1.2. Sistemas multi-robot

El aumento de la complejidad de las tareas industriales y la búsqueda de la minimización del tiempo ha dado lugar a adaptar los diseños de los robots de la forma más parecida posible a la que desarrollan sus actividades los seres humanos. En la industria, se espera que los robots antropomórficos de tamaño humano sustituyan a los operarios sin más cambios en la tarea industrial. En consecuencia, el uso de manipuladores antropomórficos o de doble brazo es un campo cada vez mayor en la investigación de nuevas tecnologías. La investigación y la historia de los manipuladores simples y duales se estudia con detalle en algunos artículos que se enumeran a continuación.

Un ejemplo del aumento de la complejidad de las tareas es el transporte de piezas flexibles y la obtención de un mejor posicionamiento de las mismas. En [11] se propone un esquema de manipulación basado en aprendizaje profundo para la soldadura automática de Printed Circuit Board (PCB) flexibles. El controlador propuesto permite que el robot contacte automáticamente con la PCB flexible primero, para luego llevarla a la posición deseada con la retroalimentación visual, y finalmente, soldarla en la máquina preparada a tal efecto.

En [12] se desarrollan los movimientos óptimos en el tiempo de manipuladores

robóticos en presencia de obstáculos; en [13] se presenta un robot submarino y en [14] se muestra el humanoide espacial Robonaut.

El uso de robots manipuladores es un tema ampliamente estudiado en los últimos años para optimizar y automatizar tareas industriales, pero cuando se diseña una tarea con un manipulador de doble brazo, se introduce una complejidad en el sistema.

Cuando se utiliza un robot de doble brazo existen dos estrategias principales para llevar a cabo tareas de pick and place. La primera y más sencilla es detener uno de los brazos mientras el otro realiza la tarea [15]. Sin embargo, esta estrategia puede resultar menos productiva, ya que implica tiempos de espera y, en consecuencia, puede ralentizar el proceso de producción. Por lo tanto, la estrategia más efectiva es mover ambos brazos al mismo tiempo, lo que permite reducir los tiempos de espera y, por ende, aumentar la productividad.

En [16] se realizó un estudio sobre el estado del arte de los manipuladores de doble brazo. El problema en el uso de robots de doble brazo o multisistema radica en algunas características que en un solo brazo no son tan importantes, y que a continuación se enumeran:

- **Coordinación:** Es una característica esencial para obtener tareas con éxito. Uno de los factores que motivan el uso de estas configuraciones multisistema es la flexibilidad de las piezas. En este caso, la coordinación entre ambos brazos para mover la pieza es esencial e incluye aspectos como la distancia o la orientación, para asegurar el éxito del pick and place. En [17] se propone una solución para la coordinación de robots de doble brazo. Los autores presentan un algoritmo capaz de coordinar automáticamente un manipulador de doble brazo durante la tarea. Otros autores [18] proponen un algoritmo de control de movimiento en tiempo real, basado en el control predictivo no lineal, que tiene en cuenta evitar colisiones estáticas y dinámicas. En este algoritmo cada robot es un agente en un juego no cooperativo, en el que se optimiza su propio movimiento y tiene en cuenta el movimiento previsto de los agentes circundantes.
- **Trayectoria:** Cuando un único robot trabaja solo en el entorno se tiene en cuenta para optimizar la trayectoria evitando obstáculos. Pero con una configuración multisistema, hay otros elementos en el espacio de trabajo que se mueven a lo largo del tiempo. Por lo tanto, el cálculo de la trayectoria debe hacerse en tiempo real, teniendo en cuenta la posición de los otros sistemas en cada momento. Para una buena optimización de la trayectoria, el algoritmo debe ser capaz de recalcular la trayectoria para evitar colisiones. En [19] los

autores presentan un método de control adaptativo para sistemas robóticos de doble brazo para realizar tareas bimanuales bajo incertidumbres de modelado. Se desarrolla una técnica de control filtrado de comandos para el seguimiento de la trayectoria y el control de la fuerza de contacto.

- **Restricciones:** Cuando una pieza es transportada por dos brazos hay que tener en cuenta algunas limitaciones y el espacio de trabajo de ambos brazos. Por ejemplo, tanto la distancia entre pinzas, como la orientación, son indispensables para garantizar el éxito de la tarea, es decir, que durante toda la trayectoria el robot sujete la pieza. Otros autores [20] desarrollan un sistema de restricciones del espacio de trabajo en línea para un grupo de robots móviles, cada uno de los cuales tiene asignada una especificación lógica temporal lineal. Se propone una estrategia de coordinación de movimiento multirobot totalmente distribuida. Para cada robot, la estrategia de coordinación del movimiento consta de tres capas. Una capa fuera de línea calcula previamente el área de frenado de cada región del espacio de trabajo, el sistema de transición controlado y una función potencial. Una capa de inicialización genera una trayectoria inicial segura y una capa de coordinación en línea resuelve los conflictos cuando se produce alguno.

La coordinación de dos brazos en movimiento aumenta la complejidad de la programación. Esta estrategia implica una mayor cantidad de colisiones potenciales entre los mismos, lo que requiere un cuidadoso diseño y planificación de las trayectorias de los brazos para evitar colisiones y garantizar la seguridad del sistema. Por lo tanto, los robots de doble brazo deben ser programados para coordinar sus movimientos de forma óptima, lo que aumenta la complejidad del sistema. A pesar de esto, la reducción del tiempo de la tarea que se logra con la estrategia de mover ambos brazos simultáneamente es un beneficio significativo que justifica el aumento en la complejidad de la programación.

Los sistemas robóticos pick and place han sido objeto de investigación en numerosos trabajos. En [21] los autores estudian el problema de secuenciación de operaciones en máquinas y lo aplican a un sistema controlado en una gran instalación de ensamblaje de tarjetas electrónicas mediante heurísticas de enrutamiento. En [22] se desarrolla una función de triple objetivo con el enfoque del pick and place dinámico de Chebyshev, con el fin de optimizar las máquinas secuenciales de pick and place y minimizar los tiempos de montaje de los robots, los movimientos de los alimentadores y las tablas de PCB.

El problema también se puede abordar utilizando un algoritmo iterativo híbrido

de búsqueda local [23], donde se puede encontrar rápidamente un óptimo relativo. En [24], los autores desarrollan una estrategia de control iterativo de aprendizaje (ILC) en trayectorias no repetitivas, aplicada a manipuladores robóticos, en este caso para un robot delta. El desarrollo de este controlador ILC se deriva de forma general para controladores de segundo orden en sistemas no lineales de multientrada y multisalida (MI- MO) expuestos a perturbaciones externas y el modelo se comprueba en operaciones de pick and place. En la misma línea de investigación sobre robots paralelos, [25] presenta un modo deslizante difuso para maximizar la velocidad en operaciones de pick and place.

En [26] se desarrolla un sistema modular de secuenciación de tareas robóticas y coordinación de movimientos para sistemas multibrazo. El módulo para resolver Robotic Task Sequencing Problems (RTSP) de doble brazo ha sido desarrollado considerando el problema generalizado de RTSPs multibrazo. Utiliza un algoritmo basado en la agrupación para desacoplar la tarea en varios subproblemas que se resuelven de forma independiente. Otros estudios se centran en la planificación de secuencias de tareas orientadas al ensamblaje, en [27] se presenta una planificación de secuencia de tareas de doble brazo basada en restricciones ambientales mediante simulación de Monte Carlo, el Modelo de Mezcla Gaussiana y la definición de funciones binarias para desarrollar la tarea.

En [28] se proporciona una visión general de los sistemas de pick and place utilizados en la industria, incluyendo robots bimanuales. Se examinan varios enfoques de control y planificación de movimientos para este tipo de tarea, y se identifican las limitaciones y desafíos actuales. En [29] se centran en la manipulación de objetos mediante manos robóticas, incluyendo el uso de robots bimanuales en tareas de pick and place. Se discuten diferentes estrategias de agarre y se analiza cómo la utilización de varias manos robóticas puede mejorar la eficiencia y la precisión de la tarea.

Varias líneas de investigación desarrollan algoritmos para incorporar multi-robot en tareas industriales. En [30] se presenta una estrategia que se basa en un algoritmo de optimización por enjambre de partículas. Esta estrategia busca mejorar la utilización de recursos de multirobots heterogéneos minimizando el tiempo de multitarea y manteniendo el equilibrio de carga de los recursos robóticos, encontrando de este modo una solución casi óptima para la programación colaborativa.

En [31] se desarrolla un método de optimización de trayectorias de referencia para robots de doble brazo en tareas industriales. Ambos brazos trabajan en el mismo espacio de trabajo, por lo tanto, cada brazo tiene que tener en cuenta la posición del otro brazo, considerándolo como un obstáculo y cambiando su trayectoria si es necesario. En [32] se presenta una investigación sobre una solución para realizar la

optimización en manipulación robótica en presencia de obstáculos.

1.3. Optimización de la trayectoria

La optimización de trayectorias es un tema clave en el ámbito de la robótica, y es de vital importancia para lograr una planificación de movimientos eficiente y efectiva. La planificación de trayectorias implica la determinación de la secuencia de movimientos que deben seguir los robots para completar una tarea específica teniendo en cuenta una variedad de restricciones, como la geometría del entorno, los límites de los motores y los tiempos de respuesta.

La optimización de trayectorias se enfoca en la búsqueda de itinerarios que minimicen el tiempo necesario para completar una tarea o maximicen la eficiencia energética del robot, sin comprometer la seguridad y precisión del movimiento. Los algoritmos de optimización de trayectorias pueden clasificarse en dos tipos: algoritmos basados en heurísticas y algoritmos basados en optimización matemática. Los primeros se basan en una serie de reglas simples, como la búsqueda aleatoria, o la reducción de velocidad en curvas cerradas, para generar trayectorias eficientes. Los segundos emplean técnicas matemáticas avanzadas, como la optimización no lineal y la programación dinámica, para determinar trayectorias óptimas.

Los enfoques actuales en la optimización de trayectorias incluyen el uso de redes neuronales, sistemas de control adaptativo y la integración de sistemas avanzados de sensores. Además, el uso de técnicas de aprendizaje automático para la generación de trayectorias se está convirtiendo en una herramienta cada vez más importante para la optimización de las mismas.

La optimización del tiempo en la planificación de trayectorias de manipuladores robóticos es una de las características de los mismos más estudiadas. En la literatura se ha propuesto una amplia gama de soluciones a este problema. En [33] se propone un modelo matemático para formular la operación pick and place en la industria alimentaria. El algoritmo húngaro se ha empleado para optimizar la distancia total recorrida por el robot. La capacidad del algoritmo para aumentar la utilización del proceso se ha demostrado con la ayuda del software de simulación desarrollado. En [34], los autores propusieron un método para implementar los pares óptimos en lazo abierto a través de un sistema convencional de control lineal realimentado.

En [35], [36] y [37], se presenta una solución para un problema similar con un algoritmo codicioso para la planificación de la velocidad de avance y un cálculo de aproximaciones de trayectorias suaves de tiempo mínimo. En [38] y [39] se introduce otro enfoque basado en el desarrollo del Problema del Agente Viajero (TSP) y un

Algoritmo Genético (GA), mientras que en [40] y [41] sólo utilizan un GA. En otros trabajos, es el robot quien aprende la trayectoria óptima. En [42] se propone un novedoso marco de aprendizaje híbrido por imitación (HIL) en el que los métodos de Clonación de Comportamiento (BC) y Clonación de Estado (SC) se combinan de forma mutuamente complementaria para mejorar la eficiencia de aprendizaje de las tareas de manipulación robótica.

En [43] se desarrolla un algoritmo de optimización, llamado Best Uniform Algorithm (BUA), para resolver el problema. Este algoritmo es un meta-heurístico que devuelve un conjunto de soluciones aleatorias de todo el espacio de búsqueda. La mejor solución se utiliza para generar otras soluciones, minimizando la distancia entre las posiciones de recogida y las posiciones de colocación.

En [44] los autores proponen un método combinado en tiempo real, basado en simulaciones y algoritmos genéticos, para optimizar la trayectoria en robots colaborativos. En este enfoque el algoritmo genético establece unos parámetros que son simulados y se selecciona el mejor. Otros enfoques como [45] se centran en optimizar el tiempo de transporte mediante trayectorias optimizadas. En [26], el módulo de optimización de la asignación se presenta como un método heurístico para distribuir un conjunto de tareas entre los sistemas robóticos asignando la configuración más adecuada a cada tarea.

La gran mayoría de las investigaciones presentadas estudian el problema centrándose en un manipulador de un solo brazo o en un conjunto de puntos donde la secuencia no es tan importante, como en [25], donde el problema TSP se resuelve considerando una velocidad constante. En esta investigación, la configuración del sistema es diferente. La tarea a optimizar es un pick and place, por lo tanto, hay puntos que deben realizarse uno tras otro, cuando se recoge una pieza, el siguiente punto es obligatoriamente el punto de entrega de esa pieza y no puede ir a otro punto. Además, el sistema se basa en un robot de doble brazo. La optimización del tiempo tiene que considerar la coordinación de ambos brazos para evitar colisiones y, en algunas situaciones, ambos brazos transportan la misma pieza.

El problema de la evaluación del rendimiento y la optimización de las tareas de pick and place en tiempo real no está muy estudiado en la literatura. En [46] desarrollan un algoritmo para realizar una tarea de control segura en tiempo real con procesamiento de imágenes. En [47] cada robot realiza en tiempo real las operaciones de pick and place asignadas para maximizar la tasa de rendimiento.

1.4. Estructura de la memoria presentada

Esta tesis doctoral se ha estructurado en 8 capítulos, incluyendo la introducción a la temática objeto de estudio presentada hasta ahora. En el Capítulo 2 se definen los objetivos definidos al principio de la investigación. En el Capítulo 3 se describe el problema y recogen los principales aspectos metodológicos que se han tenido en cuenta para la resolución del mismo. En el cuarto Capítulo se muestran todos los resultados obtenidos, empezando por el diseño de una fábrica de calzado de inyección de polímero que permite contextualizar el problema y, justificar porque nos centramos en la técnica de inyección directa, en lugar de en la técnica de montaje y pegado <https://doi.org/10.3390/app112211055>.

En el segundo subapartado resolvemos el problema de pick and place mediante la fijación de tres secuencias de recorrido preprogramadas que son elegidas por el robot, una vez analizada la posición de las piezas en la bandeja de montaje <https://doi.org/10.1109/ACCESS.2020.3037145>.

Por último, se presentan los resultados obtenidos al calcular la trayectoria óptima que deben seguir los dos brazos robóticos para trasladar las piezas al molde, según la posición de las mismas en la bandeja (en revisión). El Capítulo 5 compara los resultados obtenidos preprogramando las tres secuencias de montaje y calculando la trayectoria óptima para cada bandeja. En el siguiente Capítulo se recogen las principales conclusiones obtenidas con nuestra investigación. En el Capítulo 8 se analizan las líneas futuras de investigación que se plantean a partir de los resultados obtenidos.

Capítulo 2

Objetivos

El objetivo general de este proyecto es aplicar tecnologías en robótica colaborativa y automatización industrial para optimizar el proceso de fabricación de calzado, reduciendo el tiempo de montaje. Como método de automatización para la fabricación del calzado se considera el sistema 3DBonding desarrollado por la empresa colaboradora en este Doctorado Industrial, SimplicityWorks, que permite la unión de piezas de diversos materiales textiles mediante la inyección de poliuretano en canales dentro del molde.

Para contextualizar el problema que se aborda en esta tesis, y sobre todo, para que se entiendan mejor los objetivos de la misma, en la Figura 2.1. se presentan las diferentes etapas y niveles sobre el proceso automatizado de pick and place. En primer lugar, las piezas llegan al espacio de trabajo del robot y son localizadas por visión artificial. En esta etapa se controla si están todas las piezas del montaje y se ubican dentro del espacio de trabajo. A continuación, se pasar a la planificación de las tareas a realizar por el robot, con los datos que han sido recopilados por visión artificial (posición y ángulo de las piezas), esta capa de control es el eje principal de investigación de la presente tesis.

El siguiente paso es la planificación de trayectorias para evitar colisiones entre los brazos y con el entorno del robot. Esta línea de investigación es una de las líneas futuras que se plantea como continuación de esta tesis, tal y como se comenta con más detalle en el capítulo 8.

A continuación, detallamos los tres escenarios de trabajo de esta tesis que son los que marcan los objetivos de la misma:

Resultados previos. Los primeros pasos llevaron a realizar una revisión bibliográfica del problema, que se aprovechó para analizar la influencia del diseño de una planta

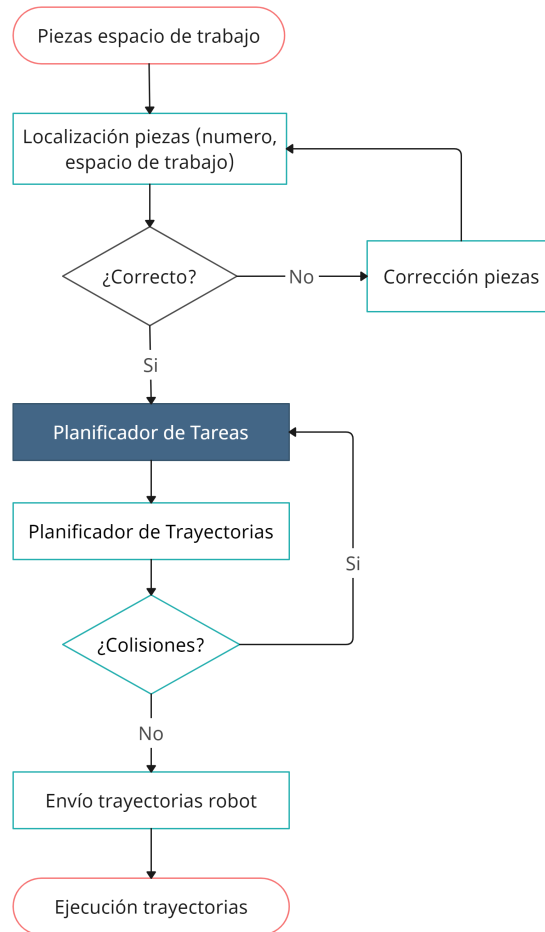


Figura 2.1: Flujograma del sistema de automatización. Resaltada la etapa en la que se va a centrar la tesis.

de calzado y las ventajas de la técnica de inyección directa frente a la técnica de montaje y pegado, sobre la que, en un principio se iba a centrar la resolución del problema.

- Objetivo 1. La primera aproximación al problema fue analizar todas las posibles combinaciones de movimientos de los dos brazos para establecer la o las secuencias de montaje que definieran una trayectoria fija de los dos brazos para realizar el pick and place. Este trabajo ha sido publicado también en una revista de impacto.

- Objetivo 2. Esa solución podía ser válida para montajes de pocas piezas, pero el número de combinaciones de movimientos de los dos brazos crecía de manera exponencial cuando aumentaba el número de piezas del calzado. Esto llevó a una segunda aproximación al problema a través de la modelización del mismo mediante técnicas de programación lineal entera y binaria para obtener la trayectoria óptima de montaje para cada bandeja. Este trabajo ha sido aceptado para su publicación en una revista de impacto.

Capítulo 3

Materiales y Métodos

3.1. Descripción del sistema

El uso de un robot de doble brazo es una solución popular en muchas aplicaciones industriales debido a su capacidad para realizar tareas complejas de manera precisa y eficiente. Sin embargo, el control de un robot de doble brazo presenta un desafío importante debido a la necesidad de evitar colisiones entre los brazos. Para lograr un control preciso, los movimientos de los brazos deben estar sincronizados en tiempo real para evitar choques, lo que puede resultar en daños en el robot y una posible interrupción en la producción.

Para evitar estos problemas se emplean técnicas de monitorización continua de la posición de los brazos para recalcular la trayectoria si es necesario. Además, se utilizan puntos de referencia a lo largo de la trayectoria para sincronizar ambos brazos. Estos puntos de referencia ayudan a garantizar que los movimientos de los brazos se realicen en sincronía y que no haya interferencias entre ellos.

Otro factor importante a considerar en el control de un robot de doble brazo es la distancia entre los brazos, que debe ajustarse para garantizar el correcto transporte de una pieza que requiera a ambos. La configuración adecuada de la distancia entre los brazos es fundamental para garantizar que el robot pueda realizar la tarea asignada sin problemas.

En las operaciones de pick and place, la calibración de la cámara es un procedimiento crítico e imprescindible. La precisión y fiabilidad de la localización de los objetos son cruciales para garantizar el éxito esta tarea, que puede ser compleja y delicada en la que los objetos deben ser recogidos y transportados a un lugar determinado con una gran precisión.

La calibración se refiere al proceso de ajuste de los parámetros de la cámara para garantizar que las imágenes capturadas sean precisas y consistentes. Esto incluye ajustes en la orientación, la posición y el enfoque de la cámara, así como la corrección de distorsiones y la compensación de las variaciones en la iluminación y en las condiciones del entorno.

La mala localización de los objetos puede tener consecuencias graves en la ejecución de la tarea de pick and place, y puede llevar a una ejecución errónea o incluso a la falla del sistema. En la mayoría de los casos, los sistemas de pick and place están diseñados para trabajar en entornos no estructurados, donde los objetos se presentan en diferentes posiciones y orientaciones. Por lo tanto, una calibración precisa de la cámara es fundamental para garantizar una ubicación eficiente y consistente de los objetos.

Es importante destacar que la calibración debe realizarse de manera regular, ya que las condiciones del entorno y los componentes del sistema pueden cambiar con el tiempo, lo que puede afectar la precisión de la misma. Por lo tanto, es esencial realizar una calibración periódica para asegurar que el sistema de pick and place esté funcionando con la precisión y fiabilidad necesarias. En [52] los autores proponen y desarrollan un método de calibración mano-ojo sin marcadores orientado a la tarea utilizando optimización iterativa no lineal.

El sistema de visión utilizado en esta tesis consiste en una cámara industrial ABB conectada al robot que proporciona los datos necesarios para su posterior procesamiento. El programa de visión artificial desarrollado mediante el software de Robotstudio, es capaz de reconocer las diferentes piezas y su ubicación en el plano XY y su orientación en el eje z. Una vez que las piezas han sido identificadas y situadas en el plano, los datos de posición y orientación son enviados al robot para que inicie la tarea de pick and place.

El robot utilizado tiene también una cámara incorporada en la muñeca, una cámara ojo-en-mano. Esto permite una mayor flexibilidad en las tareas de visualización. Una vez finalizada la tarea de pick and place, el robot inicia el control de calidad para comprobar que todas las piezas se han colocado satisfactoriamente. Esta tarea la realiza la cámara de muñeca. La Figura 3.1 muestra el sistema industrial (robot de doble brazo, cámara, bandeja y molde).

Los resultados experimentales se basan en el montaje de dos modelos de zapato que pertenecen a diferentes fabricantes, identificados como Fabricante 1 y Fabricante 2. El modelo del Fabricante 1 es un calzado de seguridad, mientras que el modelo del Fabricante 2 es un modelo “casual”. El tiempo requerido para la tarea de pick and place se calcula mediante simulación en RobotStudio. Ambos modelos constan

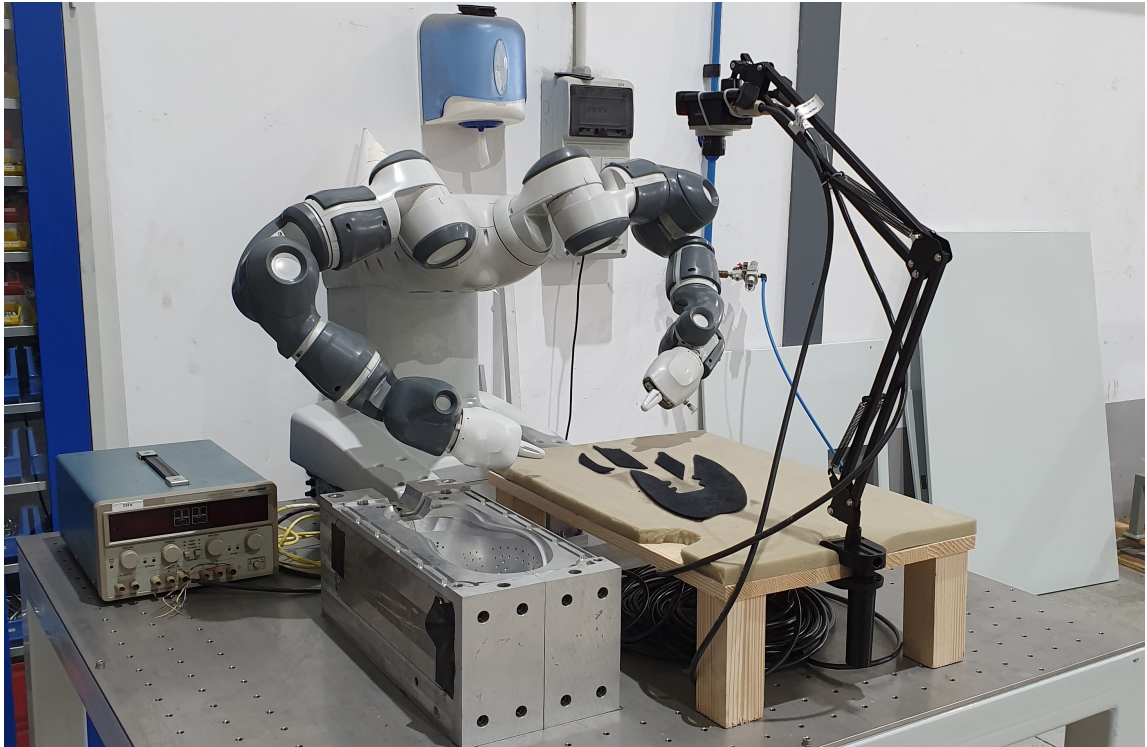


Figura 3.1: Sistema prototipo real: Robot ABB YuMi, cámara, molde de zapato y bandeja.

de cuatro piezas, tres de ellas se agarran con un brazo (las dos piezas laterales y la pieza del talón) y una de ellas (la pieza frontal) se agarra con dos brazos. La diferencia entre ellos radica en el tamaño de las piezas y su posición final. Las imágenes de cada bandeja se toman en posiciones aleatorias de las piezas. En la Figura 3.2 se presentan los modelos (a y b, respectivamente) y las imágenes tomadas de 4 bandejas elegidas al azar para su montaje en el molde (c).

El proceso de pick and place industrial implica ciertas restricciones que se han incorporado a través de la experiencia. Por ejemplo, la pieza 4 requiere el transporte con ambos brazos, lo que significa que su tiempo se suma al tiempo total de la tarea. Se ha decidido que la pieza de doble brazo siempre se recoja, lo cual no afecta al montaje final del modelo.

Por otro lado, en el caso de piezas de un solo brazo lo habitual es que la pieza que esté fuera del alcance de un brazo se recoja con el otro brazo. De esta manera, se puede optimizar el tiempo y la eficiencia de la tarea de pick and place.

Todo el proceso de pick and place industrial comienza cuando las piezas llegan al espacio de trabajo del robot. El primer paso es tomar una foto para localizar las piezas. Esta información se envía al procesamiento externo a través de un socket de comunicación entre el robot y Python.

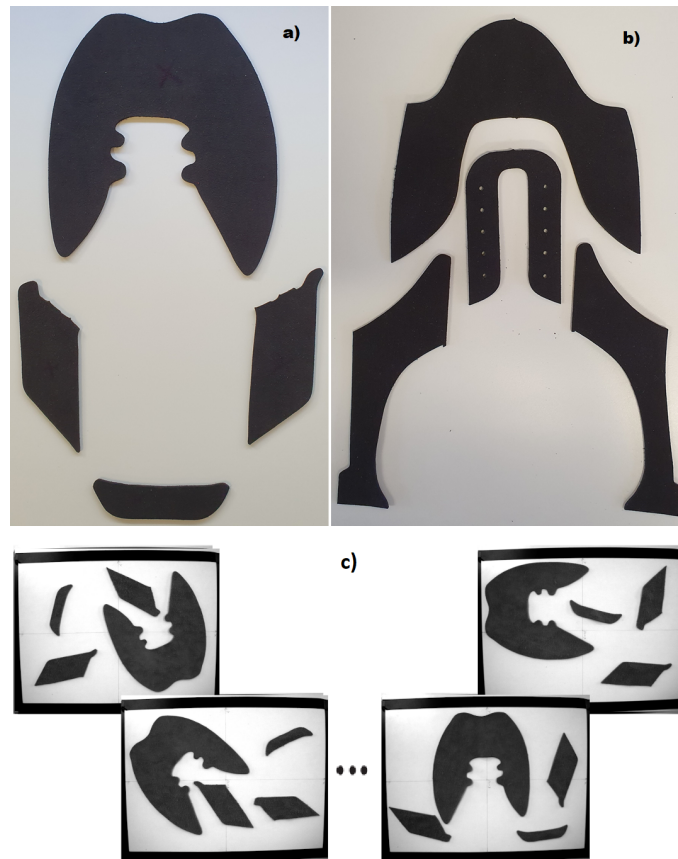


Figura 3.2: Modelos de zapatos: a) Modelo de zapato de cuatro piezas producido por el Fabricante 1, b) Modelo de zapato de cuatro piezas producido por el Fabricante 2. c) Conjunto de bandejas aleatorias.

3.1.1. Características del robot colaborativo y el procesamiento de imágenes

Esta investigación se realiza con un robot ABB YuMi (Figura 3.3 y 3.4). Este robot es un robot colaborativo de doble brazo con 7 grados de libertad (DOF) en cada brazo, lo que lo convierte en un robot con 14 DOF en total. Este tipo de robot ha sido ampliamente estudiado en la literatura. En [53] y [54] se presentan dos métodos diferentes para la solución y el control cinemático. En [55] se realiza un análisis cinemático del ABB YuMi y en [56] se muestra un control más complejo basado en dual quaternion para Yumi.

Su tamaño compacto y facilidad de programación lo hacen fácilmente adaptable a una amplia gama de tareas de pick and place y montaje. Gracias a sus brazos articulados, el robot puede realizar movimientos precisos y suaves que permiten la



Figura 3.3: IRB 14000 YuMi.

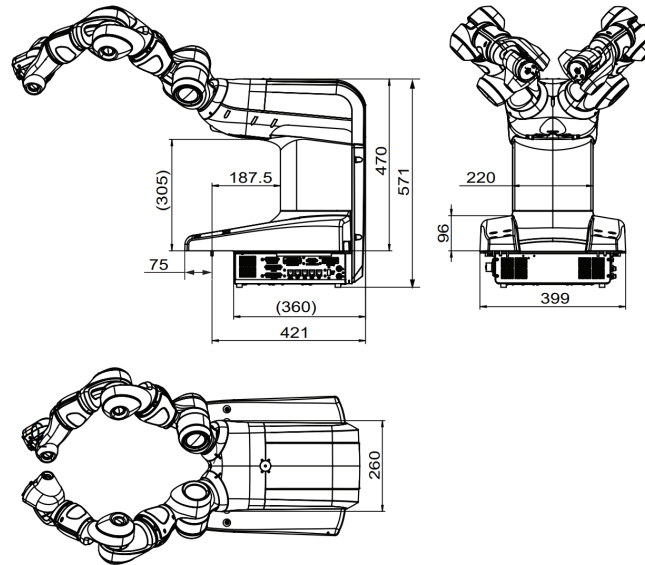


Figura 3.4: Plano de las dimensiones del IRB 14000 YuMi.

manipulación de piezas de formas irregulares o complicadas. Esto hace que el robot sea altamente versátil y adaptable a una variedad de aplicaciones.

La seguridad también es una característica importante del robot YuMi. Este robot ha sido diseñado para trabajar en estrecha colaboración con personas, por lo que ha incorporado una amplia gama de sensores de seguridad para proteger, tanto al robot, como a personas que trabajan a su alrededor.

En cuanto a la programación, YuMi cuenta con una interfaz de usuario intuitiva que facilita la programación y el control del robot. La biblioteca de aplicaciones preprogramadas ayuda a simplificar la programación y acelerar la implementación de proyectos. La programación en línea también es posible, lo que permite la adaptación rápida a las necesidades de producción.

Los sensores y cámaras incorporados permiten una mayor flexibilidad en las tareas de visualización, lo que ayuda a garantizar la precisión y eficiencia en la recogida y colocación de piezas. La cámara de muñeca integrada en un brazo también se utiliza para realizar el control de calidad una vez que se ha completado la tarea de pick and place.

El procesamiento de imágenes se realiza mediante un programa desarrollado en RobotStudio que se comunica con una cámara ABB externa conectada a un PC. La visión por ordenador permite reconocer los diferentes tipos de piezas. Cuando

se detecta la pieza, el punto de agarre (x,y) es preprogramado y la orientación en el eje z se guarda para ser utilizada por el programa de simulación. Cada pieza se clasifica según su posición y orientación para buscar una relación cuando las piezas se encuentran en una zona cercana y reducir el número de trayectorias diferentes que deben programarse. El espacio de trabajo de visión se divide en cuatro cuadrantes (I,II,III,IV) y la orientación se divide en cuatro sectores ($[0, \frac{\pi}{2}]$; $[\frac{\pi}{2}, \pi]$; $[\pi, \frac{3\pi}{2}]$; $[\frac{3\pi}{2}, 0]$)

3.1.2. Generación de secuencias

El estudio de las secuencias de movimientos del robot para el ensamblaje de zapatos es un tema de gran interés en la industria manufacturera debido a su potencial para mejorar la eficiencia y la calidad del proceso de producción.

Abordamos inicialmente el problema fijando 30 secuencias de movimiento de los brazos del robot en función de que brazo coge cada una de las piezas y en que orden. En la Tabla 3.1 se muestran las combinaciones entre los brazos en la recogida. El número representa el orden en la recogida y la letra es el brazo: R = Brazo derecho, L = Brazo izquierdo y B = Ambos brazos. Estas secuencias dan como resultado el tiempo total en la tarea de pick and place.

La pieza 4, que se recoge con dos brazos, se coloca siempre en la última posición de recogida y supone un adición en el tiempo independientemente de la posición en la que se encuentre, como ya hemos comentado anteriormente. En los casos en que

Tabla 3.1: Secuencia de combinación entre los brazos en el pick para el primer y segundo modelo.

Secuencia	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Pieza 1	1L	1R	2R	2L	3L	3L	1L	1R	3L	3R	2R	2L	1L	2R	2R
Pieza 2	2R	2L	1L	1R	2R	2L	3L	3R	1L	1R	3L	3R	2R	1L	3R
Pieza 3	3L	3R	3L	3R	1L	1R	2R	2L	2R	2L	1L	1R	3R	3R	1L
Pieza 4	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B
Secuencia	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Pieza 1	1L	3R	3R	1L	2L	1L	2L	3R	3R	1L	1L	2R	1R	1R	2R
Pieza 2	3R	1L	2R	2L	1L	3R	3R	1L	2L	1R	2R	1L	1L	2R	1R
Pieza 3	2R	2R	1L	3R	3R	2L	1L	2L	1L	2R	1R	1R	2R	1L	1L
Pieza 4	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B	4B

los dos brazos recogen piezas al mismo tiempo, la tercera pieza restante la recoge siempre el brazo derecho porque es el que está más cerca de la zona de recogida, evitándose así posibles colisiones.

De esta forma la secuencia 1, la cual denotaremos por S_1 , implica que el brazo izquierdo coge primero la pieza 1, el brazo derecho coge la pieza 2, el izquierdo irá a por la pieza 3 y finalmente los dos brazos cogerán la pieza 4.

3.2. Análisis Estadístico de las secuencias de montaje

Se ha determinado que los datos más importantes para el desarrollo del modelo son el tiempo de secuencia medio, mínimo y la ejecución correcta de las trayectorias de cada brazo. Se ha considerado como conjunto de entrenamiento 243 bandejas aleatorias asociadas al modelo de zapato del Fabricante 1. Como conjunto de validación se han considerado 222 bandejas aleatorias asociadas al Fabricante 2.

Los tiempos de montaje se han resumido mediante medias, desviaciones típicas, mínimos, máximos y rangos. Las clasificaciones se han resumido mediante recuentos y porcentajes. Para obtener la secuencia de montaje óptima se han utilizado árboles de decisión condicionales (CTREE - Conditional Tree).

En los árboles de inferencia condicional, la selección de las variables se realiza en dos fases. En primer lugar, se formula una hipótesis global de independencia en términos de hipótesis parciales. Es decir, se evalúa si existe dependencia entre la variable de respuesta y cada una de las variables explicativas.

Si no se puede rechazar la hipótesis nula de independencia se interrumpe el proceso recursivo. Por el contrario, si se rechaza la hipótesis global de independencia, el siguiente paso es medir el nivel de asociación entre la variable dependiente y cada una de las variables explicativas, lo que permite generar nuevas divisiones del árbol de forma secuencial [57].

Algunas de las ventajas de esta herramienta son: la fácil interpretación de los modelos debido a la estructura del árbol resultante [58], y su versatilidad en caso de relaciones no lineales y en el manejo de variables numéricas y categóricas simultáneamente.

3.3. Problema de programación lineal entera y binaria (BILP)

3.3.1. Modelo de 4 piezas

Para resolver el problema se considera un modelo de zapato para cuyo montaje se utilizan tres piezas de brazo simple y una pieza de brazo doble. Dado que la pieza final es la recogida por ambos brazos, sólo se considera el problema con tres nodos de recogida y tres nodos de colocación, asociados a las piezas de recogida mediante un único brazo. A estos nodos se añade un nodo de inicio por cada brazo. Los ocho nodos se colocan en un único plano cartesiano para resolver el problema. Un resumen de la notación utilizada en el modelo BILP se presenta en la Tabla 3.2.

Durante la tarea de pick and place hay que tener en cuenta el espacio de trabajo del robot, cuando la rotación de la pieza está entre 90 y 270 grados es necesaria una rotación previa, porque la recogida de las piezas se realiza en un ángulo de aproximadamente 30° con respecto al plano XY, por lo que si el punto de recogida de la pieza está orientado fuera del espacio de trabajo del robot es posible que la muñeca del robot esté fuera del mismo. Esto es especialmente relevante en el caso de piezas de dos brazos, ya que una rotación fuera del espacio podría provocar colisiones entre los brazos.

Tabla 3.2: Notación del modelo BILP.

T	Se hace referencia al conjunto de nodos de bandeja (piezas). Todos ellos tienen una posición asignada y conocida. Cada nodo estará representado por la letra i : 1,2,3
M	Se hace referencia al conjunto de nodos (piezas) del molde. Todos ellos tienen una posición asignada y conocida. Cada nodo estará representado por la letra j : 1,2,3.
K	Se hace referencia al conjunto de brazos. Habrá tantos como brazos. Cada brazo estará representado por la letra k : 1,2
S	Se refiere al conjunto de brazos disponibles en cada caso. Cada posición inicial se representará con la letra s : 1,2
X_{kij}	Es una variable binaria. Su valor en la solución determina qué recorrido desarrolla cada brazo. Si esta variable es 1, es decir, está activada, indica que el brazo k recorrerá el movimiento descrito por los nodos consecutivos i - j , donde i es el nodo inicial y j es el nodo final. Por el contrario, si el valor es 0, esta trayectoria no tendrá lugar en la tarea.
D_{ij}	Se refiere a la distancia euclidiana entre el nodo i y el nodo j . Esta distancia se calcula a partir de la posición de la pieza en el plano cartesiano del problema considerado. Para cada movimiento i - j existe otro movimiento j - i tal que $D_{ij} = D_{ji}$. Todas las distancias deben ser positivas y no se tienen en cuenta las distancias entre nodos incompatibles. Este vector es de orden $3(s+p)+2(s-p)$, donde p es el número de piezas de un solo brazo a transportar y s es el número de brazos robóticos.

La Figura 3.5 muestra todos los posibles itinerarios del problema. En color negro se representan las trayectorias que son obligatorias después de visitar el nodo de recogida. En azul se representan las trayectorias que se pueden realizar. Este problema se resuelve como un (BILP) y se modeliza de la siguiente forma:

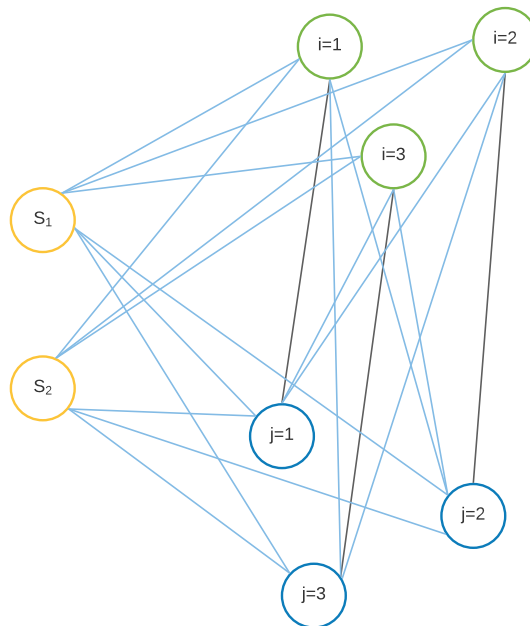


Figura 3.5: Diagrama con todas las posibilidades. S son las posiciones de inicio de cada brazo. Los nodos i son las posiciones de recogida y los nodos j son las posiciones de colocación.

$$\min \sum_{k \in K} \sum_{i,j} D_{ij} X_{kij} \quad (1)$$

$$\text{s.t.} \quad \sum_{i=1}^p X_{ksi} = 1 \quad \forall k = s, k \in K, s \in S \quad (2)$$

$$\sum_{k=1}^2 X_{kij} = 1 \quad \forall i = j, i \in T, j \in M \quad (3)$$

$$\sum_{j=1}^p X_{kjs} = 1 \quad \forall k = s, k \in K, s \in S \quad (4)$$

$$\sum_{k=s=1}^2 X_{ksi} + \sum_{k=1}^2 \sum_{j=i}^2 X_{kji} = 1 \quad \forall i \in T \quad (5)$$

$$\sum_{i=j=1}^p X_{kij} \leq p - 1 \quad \forall k \in K \quad (6)$$

$$\sum_{k=s=1}^2 \sum_{i \neq j} X_{kji} + \sum_{k=s=1}^2 X_{kjs} = 1 \quad \forall j \in M \quad (7)$$

$$\sum_{i \neq j} X_{kji} + X_{kjs} - X_{khj} = 0 \quad \forall j \in M, k = s, h = j, h \in T \quad (8)$$

$$\sum_{j=1}^p X_{kji} + X_{ksi} - X_{kih} = 0 \quad \forall i \in T, k = s, h = i, h \in M \quad (9)$$

- **Función objetivo (1):** Viene definida por la suma de todas las distancias recorridas por ambos brazos para colocar las piezas sueltas desde la bandeja hasta el molde. Por lo tanto, el problema de optimización debe minimizar el valor de la misma.

- **Tipología de las Restricciones**

Restricciones iniciales(2): El brazo debe comenzar en el nodo inicial. Por lo tanto, la trayectoria desde el nodo inicial hasta el nodo asociado a la pieza $i \in T$ solo puede darse una vez. Habrá tantas restricciones como n^0 de brazos (s).

Restricciones de colocación (3): Cada pieza solo puede ir a un lugar del molde por lo que este movimiento tiene que ser realizado obligatoriamente por un brazo. Todos los movimientos solo pueden ser realizados como máximo por un brazo, incluso puede haber movimientos que no sean cruzados por ningún brazo. Habrá p restricciones.

Regreso del brazo a la posición inicial (4): Cada brazo debe volver a su posición inicial (nodo inicial s) una vez que ha terminado la tarea, es decir, cada brazo solo puede ir una vez desde un nodo j al nodo inicial s. Habrá tantas restricciones de este tipo como brazos (s).

Los nodos de bandeja solo se visitan una vez (5): Cada nodo de bandeja $i \in T$ solo se visita una vez. A estos nodos se puede llegar desde el nodo inicial $s \in S$, si es la primera vez, o desde un nodo $j \in M$ si está después del lugar de otra pieza. Habrá p restricciones.

Un brazo no puede recoger todas las piezas (6): Se da como condición que un brazo no pueda transportar todas las piezas. Parece lógico pensar que si las piezas se transportan entre ambos brazos, se minimizara el tiempo total de la tarea. Habrá tantas restricciones de este tipo como brazos (s).

De cada nodo del molde solo sale un brazo una vez (7): El nodo $j \in M$ es el nodo de colocación de cada pieza, por lo tanto, solo se puede colocar una pieza por nodo, por lo que solo se puede visitar una vez considerando todos los brazos. Si un brazo ha cogido o colocado una pieza en un nodo, el mismo brazo o el otro no podrán volver a hacerlo. Habrá p restricciones.

Entrada y salida de un nodo del molde (8): Si un brazo llega a cualquier nodo de colocación $j \in M$, es obligatorio que el brazo salga de él para ir al siguiente nodo de recogida $i \in T$ o nodo inicial $s \in S$ si ha terminado su tarea. Habrá p restricciones.

Entrada y salida de un nodo de la bandeja (9): Si un brazo llega a cualquier nodo de recogida $i \in T$ es obligatorio que el brazo salga de él para ir al nodo $j \in M$ de colocación. Puede alcanzarse desde el nodo inicial del brazo o desde un nodo de recogida anterior. Habrá p restricciones.

Se utilizará el software RobotStudio, Phyton y R para desarrollar y probar el algoritmo. La modelización del problema de minimización se ha realizado con la función $lp()$ del paquete *lpsolve* de R. En el cual mediante la introducción de las restricciones en forma matricial binaria se ha minimizado la distancia para obtener el valor objetivo.

La primera fase del proceso será la obtención de las coordenadas de las piezas en la bandeja (posiciones de pick) mediante RobotStudio, con estos datos y sabiendo las posiciones fijas de place se ejecuta el modelo BILP para la optimización de la secuencia, devolviendo éste la secuencia a ejecutar a RobotStudio.

3.3.2. Modelo de n piezas

Tras el desarrollo del modelo matemático para la optimización de la secuencia de pick and place se observó que al aumentar el numero de piezas para obtener la secuencia óptima se creaban bucles entre dos piezas y el brazo implicado no acababa en la posición final, tal y como se puede observar en la Figura 3.6. Por lo tanto, se incorporó la siguiente restricción para evitar estos bucles.

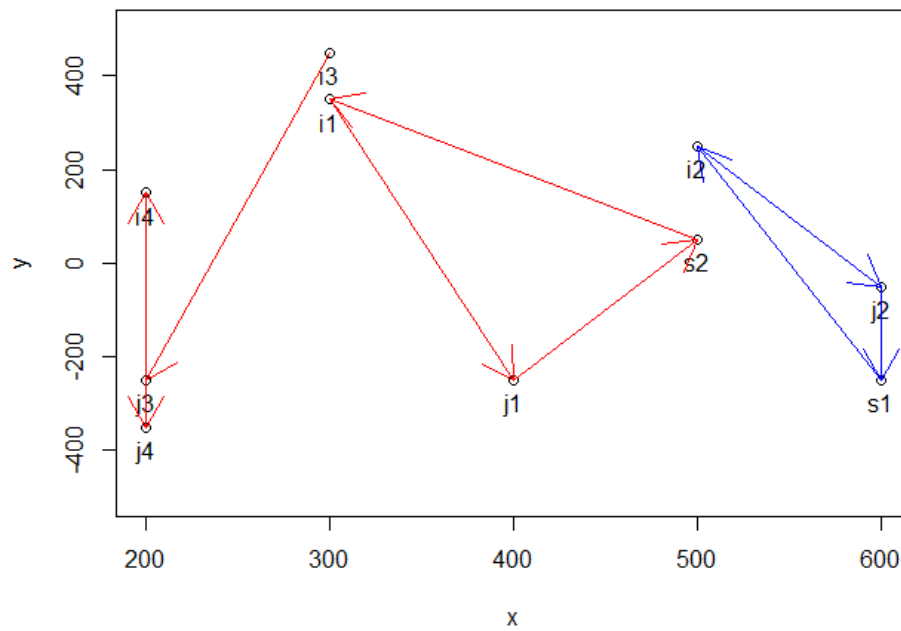


Figura 3.6: Secuencia optimizada mediante el BILP para un modelo de zapato con 4 piezas individuales donde se detecta un bucle en la trayectoria roja.

Bucles (10). No se puede generar un camino en el cual se genere un bucle cerrado entre dos piezas sin terminar ni empezar en el punto de inicio. Habrá $3(p-2)$ restricciones.

$$\sum_{j=1} X_{kij} + \sum_{j \neq 1} X_{kij} < p \quad \forall i \in T, k = s, j \in M \quad (10)$$

Para la resolución del modelo de n piezas se ha trasladado el desarrollo a Python el cual utilizará las mismas funciones que R mediante la función $r()$ del paquete *rpy2*. La metodología que sigue el programa es la misma que la presentada anteriormente pero en este caso conectando RobotStudio con Python.

Capítulo 4

Resultados

4.1. Diseño de una fábrica de calzado de inyección de polímero.

Se aborda, en primer lugar, el problema del diseño de una fábrica de calzado de inyección de polímero, para poder contextualizar el entorno de trabajo en el que se sitúa el problema que se resuelve con esta tesis.

La automatización de los procesos de producción pasa en primer lugar por una reestructuración de la propia fábrica. Por eso, se plantea como se podría diseñar una planta de producción de calzado que utiliza un sistema de montaje de inyección directa, para minimizar los costes de producción y los tiempos asociados a los procesos.

Se desarrolla un sistema experto respaldado por una simulación de eventos discretos. Inicialmente, se recopila conocimiento experto fundamental para crear una base de datos. A continuación, se emplea la simulación de la planta de producción para validar la solución propuesta. El simulador proporciona una amplia y precisa información sobre la solución, incluyendo tiempos de tarea y número de trabajadores, entre otros aspectos relevantes. Para este diseño se ha utilizado la edición gratuita del software de simulación AnyLogic.

La base de conocimientos utilizada en este proyecto se basa en un conjunto predeterminado de reglas establecidas por expertos, fundamentadas en el proceso de producción. Estas reglas se obtienen tanto de fuentes bibliográficas, como del conocimiento aportado por los expertos involucrados. Existen varias técnicas para establecer reglas, como se describe en [59]. En esta aplicación específica, el conocimiento fue proporcionado por los coautores del trabajo, también se utilizaron herramientas de

modelado de procesos industriales y se cuenta con la colaboración de expertos externos, quienes aportan información sobre el proceso de producción, sus características y restricciones.

Se utilizan herramientas de modelado de procesos debido a que no todo el conocimiento puede ser expresado únicamente en forma de reglas por los expertos [60]. Para representar el sistema de manera adecuada, parte de la información necesaria se introduce en una herramienta de simulación, lo que permite definir correctamente entidades, tareas, variables y sus interrelaciones. Asimismo, se organizaron los datos de diseño de manera ordenada para facilitar su accesibilidad y comprensión por parte del usuario.

En esta investigación se consideraron principalmente el conocimiento experto sobre el proceso, las dependencias entre los distintos procesos y el tiempo promedio de las tareas. El resultado principal se enfoca en la mejor distribución de los procesos en la planta.

El sistema de base de conocimientos recibe el conocimiento experto codificado mediante hechos, reglas, heurísticas y procedimientos. El sistema experto consta de tres elementos: base de datos, base de conocimientos y el motor de inferencia. La arquitectura general de la base de datos de conocimiento y el sistema experto se presenta en [61]. La base posee conocimiento del dominio expresado en términos de lógica matemática.

Existen varias formas de representar las relaciones entre clases, atributos y procedimientos de clase. En este trabajo se utiliza el Lenguaje Unificado de Modelado (UML) debido a su sencillez, uso intuitivo y amplia documentación [62]. La Figura 4.1 representa el diagrama de clases del marco conceptual de referencia, donde el área de procesos del diagrama se describió en seis categorías principales: Corte de piezas superiores, Fabricación, Prueba de calidad, Corte rápido, Limpieza/pulido y Embalaje. Dicha clasificación se detalló siguiendo la formalización propuesta por [63],[64]. Este estudio se centra en el proceso de fabricación, los demás procesos, al no formar parte del mismo, no forman parte de este diseño.

Los subprocesos considerados son los siguientes: cosido, pulido del cuero, preformado de piezas, pegado, durado, fresado, taladrado y rectificado. Cabe destacar que el procedimiento es válido para el diseño de cualquier tipo de fábrica de producción de calzado, independientemente del proceso de producción utilizado: técnica de inyección directa o técnica de montaje y pegado.

La Tabla 4.1 muestra los tiempos por producto para las dos técnicas de montaje citadas, teniendo en cuenta que algunos procesos son idénticos en ambas. Es fácil observar que el proceso de inyección directa tiene menos operaciones, ahorrando

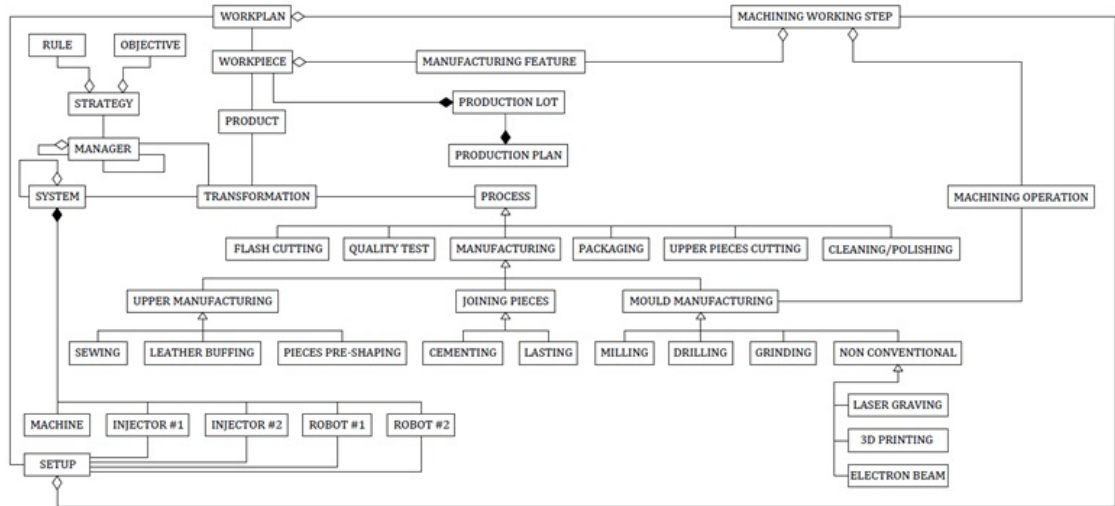


Figura 4.1: Diagrama de clases del marco conceptual de referencia.

tiempo en el proceso global. Para la estimación de los tiempos de operación de los procesos hemos contado con la ayuda de la empresa Simplicity Works Europe SL (Elche, España) [65]. Su actividad empresarial requiere un profundo conocimiento de los procesos de fabricación industrial, que permite, en el caso de la producción de calzado, conocer el conjunto de procesos, los tiempos parciales de producción y los flujos asociados.

Existen varios tipos de máquinas de inyección rotativa con diferente número de estaciones e inyectores. En este caso, las características de la máquina rotativa considerada serían las siguientes:

- Número de estaciones: 24
- Ciclo del tiempo: 16 s
- Tiempo de desplazamiento entre estaciones: 2 s
- Número de inyectores: 2

Una vez establecidos los agentes y los parámetros se puede modelizar el sistema teniendo en cuenta las siguientes fases:

- Entrega, almacenamiento y envío de materiales y productos: En primer lugar, se modela un sistema de entrega y almacenamiento de materias primas.

Tabla 4.1: Características de la secuencia óptima para el Fabricante 1 y Tiempos de fabricación por producto. Esta información está asociada a la inyección directa y al proceso de montaje y pegado.

Tarea	Inyección directa	Montado y Pegado
	Tiempo por zapato (s)	
Colocación del material y ajuste de las proyecciones		6.5
Corte		39
Marcado superior		30
Marcado		9
Fresado		30
Uniones temporales		45
Costura de punteras		37
Costura de cuartos		67
Costura del talón		37
Encolado entre la lengüeta, la parte superior y el forro		52
Encolado entre la parte superior y el forro		47
Pegado de espuma en la lengua		22
Moldeador superior		52
Moldeador de talones		52
Ojales		36
Montaje superior	-	17
Fijación del talón y de la caña	-	17
Marcado y cepillado	-	26
Suela y empeine encolados	-	40
Secado y activación de la cola	-	20
Suela de unión prensada	-	38
Corte rápido	10	-
Última eliminación		6
Limpieza		13
Plantilla y cordones		51
Embalaje		14
Corte	39	-

La capacidad de almacenamiento o el intervalo de entrega pueden controlarse mediante los parámetros previamente definidos para evitar la escasez de existencias.

- Factor limitante: En segundo lugar, se configura el agente “Rotary” que representa la máquina de inyección rotativa. Simultáneamente, se define la relación entre la máquina rotativa y el resto del sistema. La Figura 4.2 muestra los bloques lógicos utilizados para modelar el movimiento de la máquina de inyección rotativa.
- Operaciones previas al factor limitante: Tras haber definido correctamente el factor limitante, hay que detallar las operaciones previas y el número de recursos necesarios para mantener la máquina rotativa a pleno rendimiento.
- Factor postlimitante: Por último, se modelan las operaciones de acabado y envasado. En este momento, es necesario establecer el número de recursos necesarios para extraer cada producto de la inyectora rotativa sin obtener nuevas colas.

Para conseguir un modelo lo más simple posible, las tareas que son similares entre sí se agrupan en un único parámetro temporal. En la Figura 4.3 se muestran

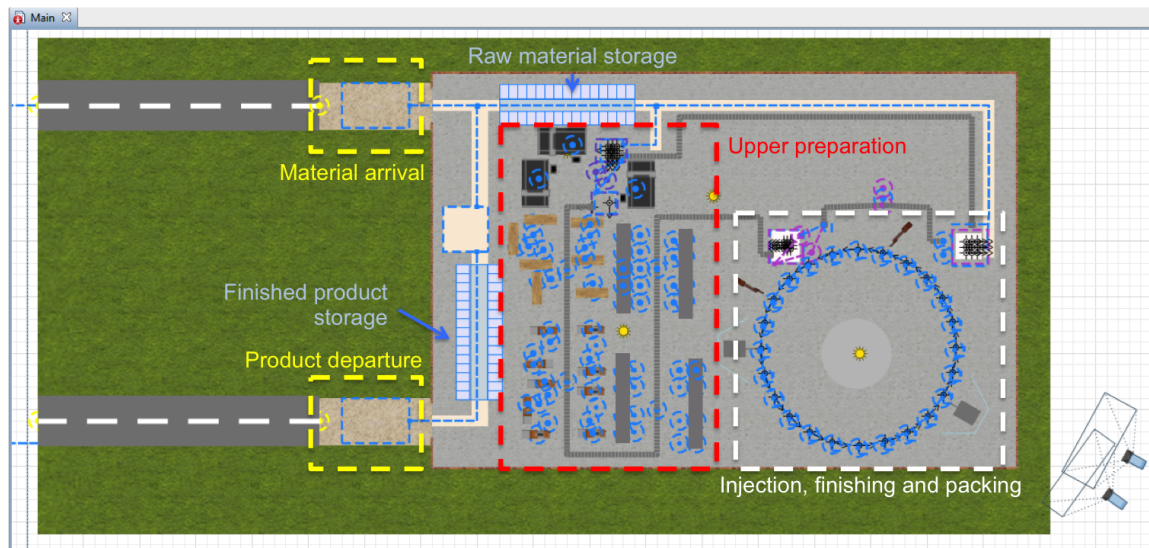


Figura 4.2: Bloques lógicos en el agente “Rotary” y configuración de la transición entre las dos fases. Estos bloques están asociados al proceso de inyección directa.

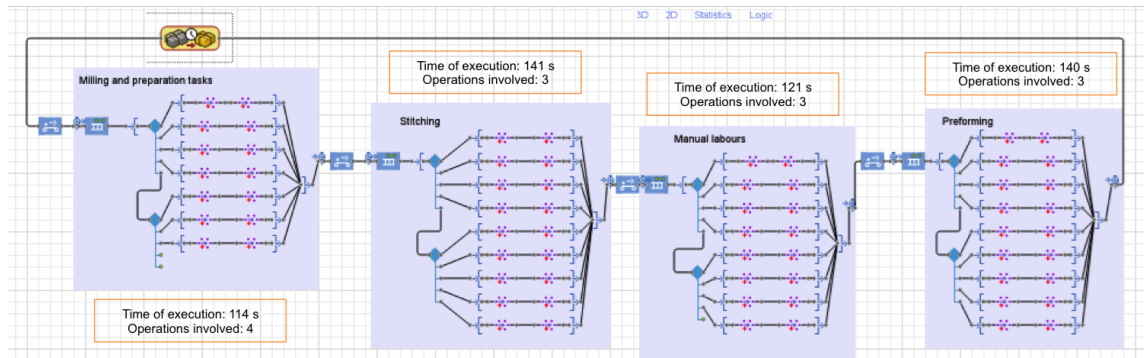


Figura 4.3: Diagrama de bloques del proceso de inyección directa.

los bloques lógicos que se ejecutarán durante la simulación. Además, la Figura 4.4 muestra la representación espacial del modelo final en 2D. Es importante destacar algunos de los resultados proporcionados por las simulaciones una vez finalizado y optimizado el modelo con la inyectora rotativa trabajando a pleno rendimiento.

Con los recursos empleados podemos asegurar que el único cuello de botella del sistema es el factor limitante y el producto invierte 45.925 min. de media desde que entra en el sistema hasta el final del sistema. Después de simular 8 horas de trabajo, se han producido 787 pares de zapatos y sólo quedan 11 productos a la espera de ser colocados en la máquina de inyección rotativa que tardará sólo 13.017 min. en completar el montaje.

El enfoque propuesto permite obtener resultados relativamente precisos de una forma menos lenta y costosa que probando alternativas en una planta de producción real. Una información importante que puede extraerse de este trabajo es el número de zapatos producidos por trabajador durante una jornada laboral (se consideró un turno y 8 horas de trabajo). La tabla 4.2 muestra que la productividad de un trabajador en una fábrica de inyección directa es de 36.60 zapatos/día/turno, mientras que la productividad de un trabajador en una fábrica de montaje y pegado es de 28.08 zapatos/día/turno. Esto significa que una fábrica de inyección directa es un 30.34 % más productiva que una de montaje y pegado.

El estudio de la tesis se centra en la obtención de la mejor secuencia en la tarea de pick and place para el sistema de producción de Simplicity Works Europe, llamado 3DBounding. Este sistema tiene como procedimiento principal de fabricación el posicionado de las piezas del corte dentro de un molde autocerrado que posteriormente se inyecta. El poliuretano fluye por unos canales internos que generan la unión entre las piezas del corte conformando el calzado. Este método de fabricación se puede

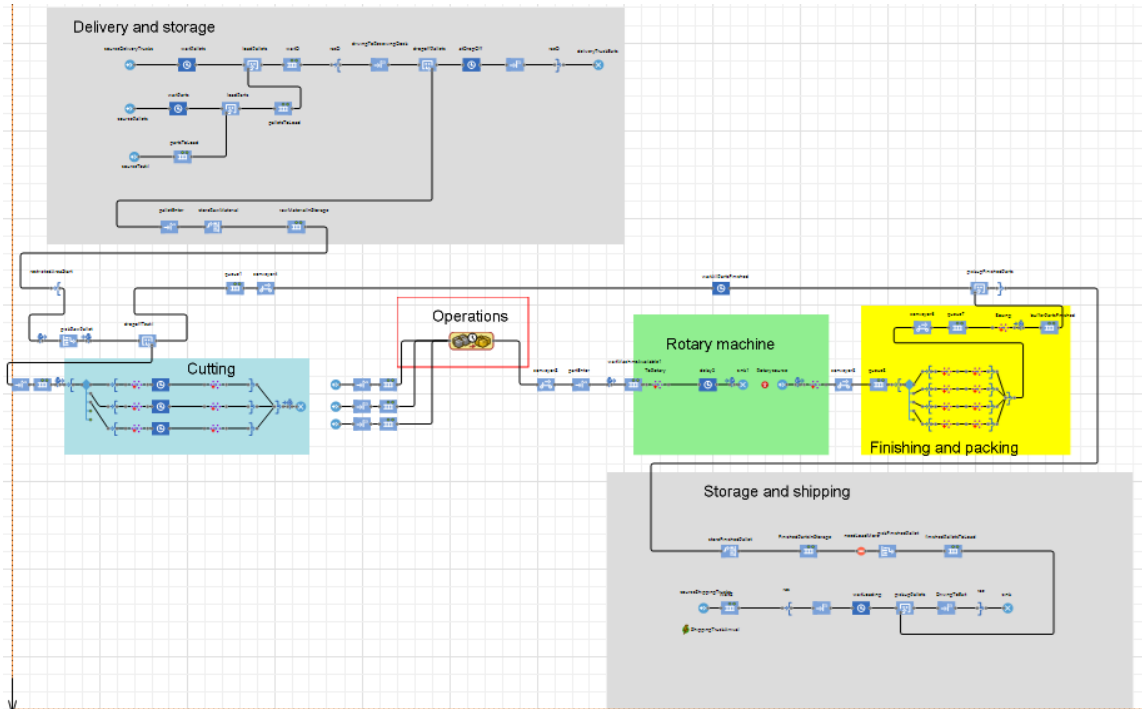


Figura 4.4: Representación espacial del sistema de inyección directa.

Tabla 4.2: Características de la secuencia óptima para el Fabricante 1 y Tiempos de fabricación por producto. Esta información está asociada a la inyección directa y al proceso de montaje y pegado.

	Jornada laboral (8h)	Nº de trabajadores	Ratio de productividad de calzado por trabajador y día
Inyección directa	1574 pares de zapatos	37	36.60
Montado y Pegado	1320 pares de zapatos	43	28.08

considerar una evolución de la inyección directa de poliuretano.

4.2. Análisis de las secuencias de montaje

El sistema de simulación se ha desarrollado con el software RobotStudio y el programa de lenguaje RAPID. El robot con el que se ha trabajado es el manipulador de doble brazo ABB, YuMi. El programa recibe la información sobre la posición y orientación de las piezas. Con esta información, los puntos de recogida que se

establezcan cambiarán de una secuencia a otra, ya que los puntos de colocación serán los mismos para cada pieza en todas las secuencias. La Figura 4.5 muestra el sistema en RobotStudio.

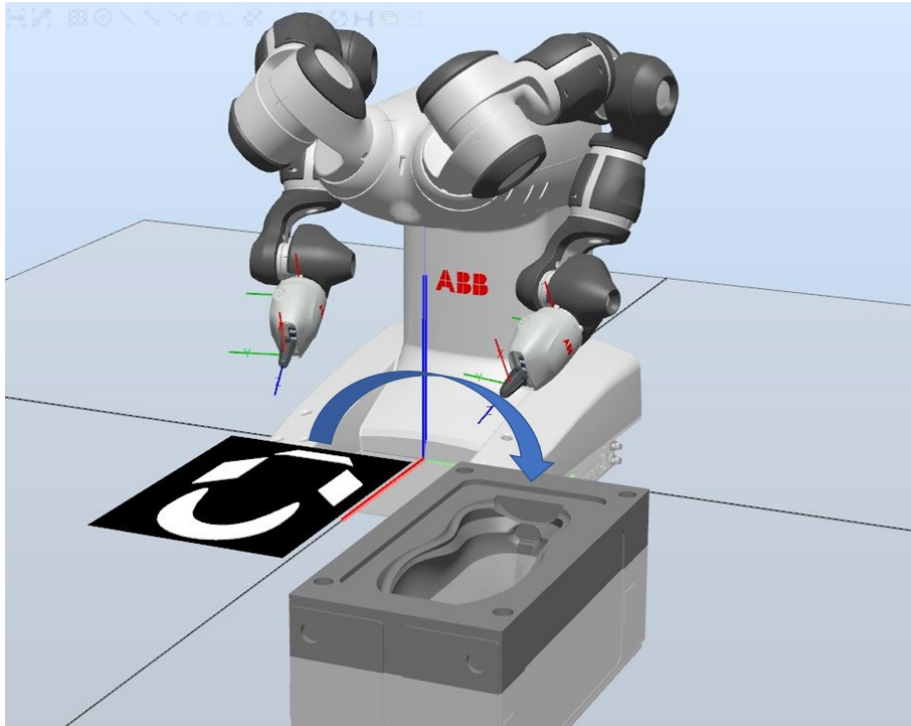


Figura 4.5: Sistema simulado en Robotstudio.

4.2.1. Valores ausentes, colisiones y asociaciones

Las 243 bandejas aleatorias de piezas asociadas al modelo del Fabricante 1 han sido acopladas en el molde 30 veces, una por cada una de las secuencias de montaje definidas (Tabla 3.1), lo que implica la resolución del montaje 7,290 veces. En el caso del Fabricante 2 estaríamos hablando de resolver el montaje del zapato $222 \times 30 = 6,660$ veces.

Toda la información recogida nos permite identificar patrones de comportamiento que se reproducen con mayor o menor asiduidad. En primer lugar, observamos la aparición de valores ausentes asociados a que no se ha completado el proceso de montaje, en la mayoría de los casos por colisiones entre los brazos 4.6. Esto ocurre,

sobre todo, en las secuencias S25 a S30, probablemente porque desarrollan la tarea de pick and place moviendo ambos brazos demasiado cerca. El número de colisiones en las seis últimas secuencias es similar para ambos fabricantes, lo que denota un comportamiento común entre las secuencias, independientemente del modelo fabricado. Este problema es debido a que en estas seis secuencias algunas piezas se transportan simultáneamente, mientras que las veinticuatro primeras secuencias son secuenciales. El elevado riesgo de colisión nos lleva a descartar estas seis secuencias del análisis

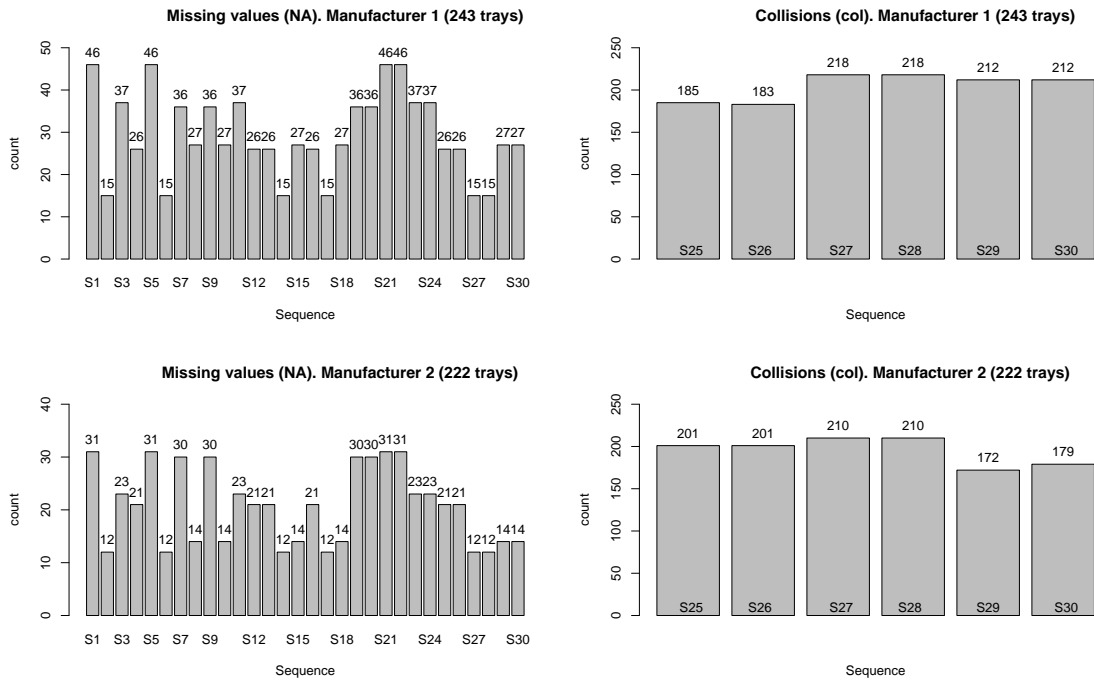


Figura 4.6: Valores ausentes y número de colisiones para los modelos de zapato Fabricante 1 y Fabricante 2.

Se observan varios grupos de secuencias con una correlación alta en cuanto a los tiempos de montaje de cada bandeja. La Figura 4.7 nos muestra la asociación existente entre las secuencias S1, S5, S21 y S22. Todas ellas, además, coinciden en las bandejas que no resuelven, coincidiendo la razón del fallo de montaje. No ocurre lo mismo al analizar los resultados de las secuencias S1 y S2, las cuales presentan un menor nivel de asociación.

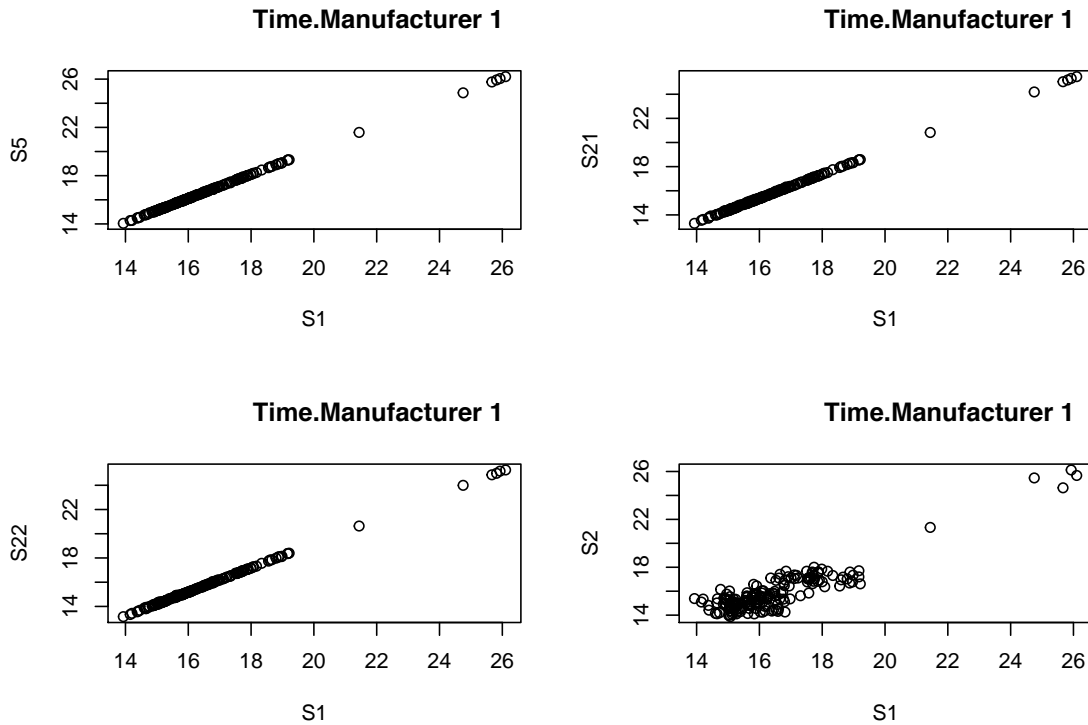


Figura 4.7: Asociación entre cada grupo de secuencias.

Estos patrones se cumplen en los dos modelos analizados, lo que nos lleva, teniendo en cuenta el análisis de los valores perdidos y las colisiones a agrupar las secuencias en 6 grupos por similitud de comportamiento en el montaje:

- **Grupo 1-** S1, S5, S21 y S22
- **Grupo 2-** S2, S6, S14 y S17
- **Grupo 3-** S3, S11, S23 y S24
- **Grupo 4-** S4, S12, S13 y S16
- **Grupo 5-** S7, S9, S19 y S20
- **Grupo 6-** S8, S10, S15 y S18

4.2.2. Resultados de cada secuencia

Los tiempos medios de montaje por bandeja en el modelo Fabricante 1 van desde 15.18 sg. (S18) a 17.03 sg. (S9), Figura 4.8a, sin que ninguna secuencia complete el número total de bandejas (197 a 228). La variabilidad observada con cada secuencia es muy parecida, oscilando entre 1.69 y 1.91. La S15 es capaz de montar una de las bandejas en el menor tiempo posible, 12.71 sg., mientras que S3 y S11 son las que tardan más en montar una de ellas, 26.64sg. (Tabla 4.3)

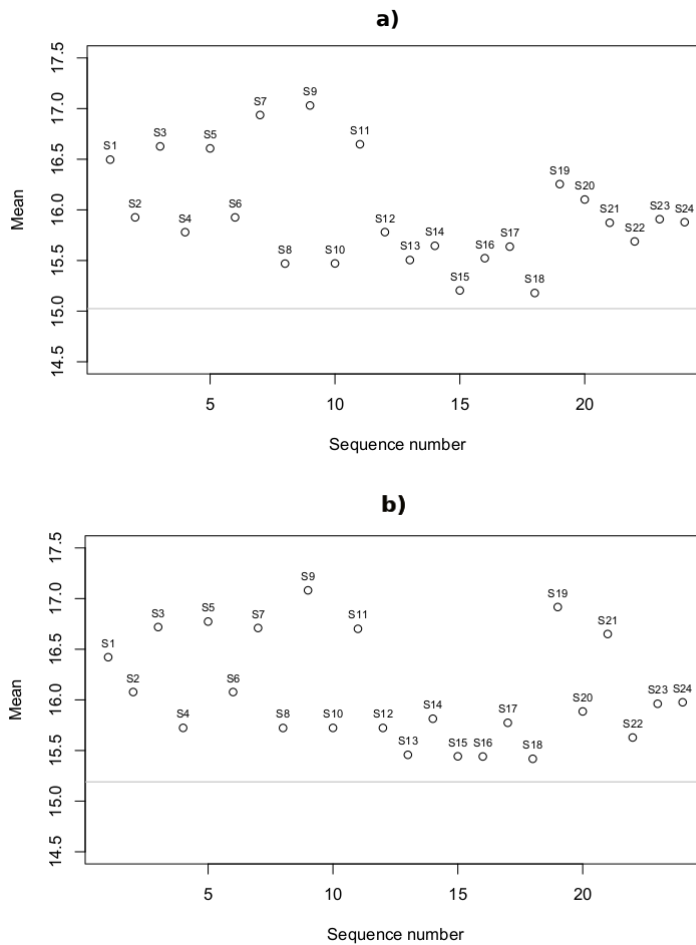


Figura 4.8: Distribución del tiempo medio para cada secuencia para a) el modelo Fabricante 1 y b) el modelo Fabricante 2

Tabla 4.3: Principales parámetros para el tiempo de montaje con cada secuencia del modelo del Fabricante 1.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
n	197	228	206	217	197	228	207	216	207	216	206	217
media (s)	16.50	15.93	16.63	15.78	16.61	15.93	16.94	15.47	17.03	15.47	16.65	18.78
sd	1.91	1.70	1.77	1.82	1.91	1.70	1.86	1.74	1.86	1.74	1.77	1.82
min (s)	13.93	13.87	14.34	13.55	14.05	13.87	14.43	12.89	14.54	12.90	14.34	13.52
max (s)	26.10	26.13	26.64	25.63	26.20	26.16	26.55	25.58	26.62	25.59	26.64	25.62
	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24
n	217	228	216	217	228	216	207	207	197	197	206	206
media (s)	15.50	16.64	15.20	15.52	15.64	15.18	16.25	16.10	15.87	15.69	15.91	15.88
sd	1.81	1.69	1.73	1.81	1.70	1.74	1.85	1.86	1.91	1.91	1.77	1.76
min (s)	13.29	13.61	12.71	13.29	13.53	12.84	13.75	13.66	13.30	13.16	13.64	13.57
max (s)	25.38	25.92	25.26	25.38	26.01	25.47	25.82	25.72	25.46	25.26	25.90	25.88

Si se hubiera podido seleccionar para cada bandeja la secuencia que lo resuelve más rápidamente, se habría obtenido un tiempo medio por bandeja de 15.03 sg.

En el modelo del Fabricante 2 se observa que los tiempos medios por bandeja necesarios para realizar el montaje oscilan entre 15.42 sg. (S18) y 17.08 sg. (S9), Figura 4.8b. Las bandejas completadas para cada secuencia varían entre 192 y 210 de las 222 completadas con el Fabricante 2. La variabilidad observada oscila entre 1.23 y 1.30. La bandeja más rápida en montar necesita 13.26 sg. (S22), mientras que la más lenta se va a 26.56 sg. (S9). (Tabla 4.4)

Si se hubiera podido elegir para cada bandeja la secuencia que la resuelve más rápidamente, se habría obtenido un tiempo medio por bandeja de 15.19 sg. Los resultados son, por tanto, muy parecidos con ambos fabricantes.

Se puede apreciar en las Figura 4.9a y 4.9b que las secuencias que más frecuentemente aparecen como las mejores son S13, S14, S15, S16, S17 y S18, siendo la importancia de las secuencias S20, S22 y S24 residual. El resto de secuencias no consiguen mejores resultados en el montaje de ninguna bandeja. La secuencia más eficiente, en ambos fabricantes, es la S18, teniendo el resto un comportamiento parecido, aunque la S16 es más frecuente con el Fabricante 2.

Si sólo consideramos estas 6 secuencias más frecuentes, Figura 4.9c y Figura 4.9d, vemos que las secuencias que absorben las eliminadas son S13, S15 y S18. Se observa también que estas secuencias están recogidas en tres de los seis grupos comentados anteriormente. La S14 y S17 forman parte del grupo 2, la S13 y S16 del grupo 4, y

Tabla 4.4: Principales parámetros para el tiempo de montaje con cada secuencia del modelo del Fabricante 2.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
n	191	210	199	201	191	210	192	208	192	208	199	201
media (s)	16.42	16.08	16.72	15.72	16.77	16.08	16.71	15.72	17.08	15.72	16.70	15.72
sd	1.27	1.30	1.30	1.24	1.27	1.30	1.28	1.24	1.28	1.24	1.30	1.24
min (s)	13.97	13.91	14.58	13.63	14.36	13.92	14.18	13.72	14.56	13.72	14.60	13.62
max (s)	25.38	25.55	25.62	25.31	25.74	25.55	26.16	24.70	26.56	25.70	25.60	25.30
	S13	S14	S15	S16	S17	S18	S19	S20	S21	S22	S23	S24
n	201	210	208	201	210	208	192	192	191	191	199	199
media (s)	15.46	15.81	15.44	15.44	15.77	15.42	16.92	15.89	16.65	15.63	15.96	15.98
sd	1.23	1.30	1.24	1.23	1.30	1.24	1.27	1.27	1.27	1.26	1.29	1.29
min (s)	13.48	13.80	13.44	13.34	13.61	13.40	14.43	13.44	14.27	13.26	13.90	13.95
max (s)	24.95	25.20	24.34	24.92	25.32	24.50	26.33	25.30	25.65	24.57	24.90	24.85

la S15 y S18 del grupo 6.

Si elegimos de cada par la más frecuente estaríamos reduciendo a 3 las secuencias necesarias para trasladar las piezas de cada bandeja al molde, Figura 4.9e y Figura 4.9f, obteniendo la misma distribución de frecuencias en ambos fabricantes.

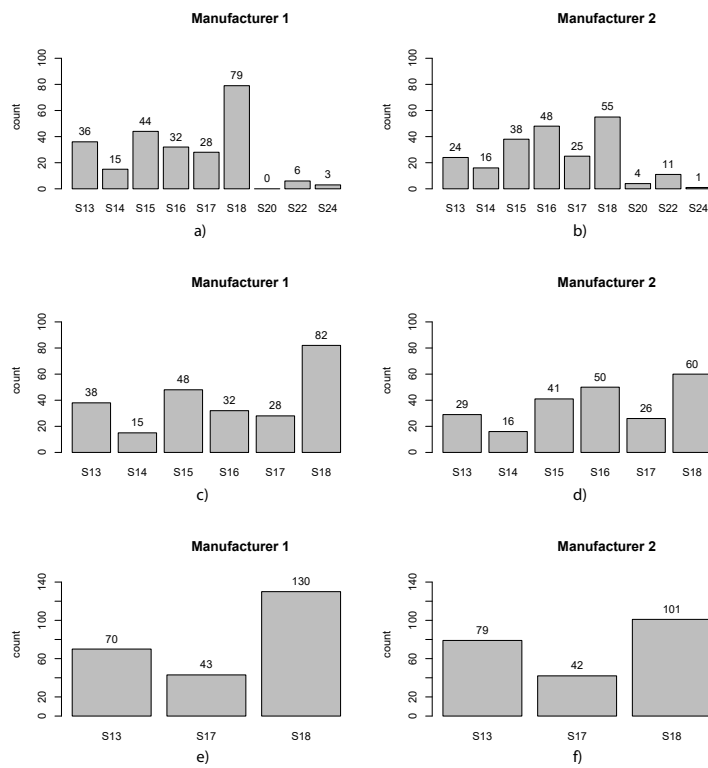


Figura 4.9: Distribución mejores secuencias. Izquierda: modelo del Fabricante 1. Derecha: Modelo del Fabricante 2.

4.2.3. Selección de la secuencia adecuada

Si se utiliza S18 con todas las bandejas estaremos cogiendo la mejor solución posible en un 53.50 % de las ocasiones en el caso del Fabricante 1, y un 45.50 % en el caso del Fabricante 2. Para intentar mejorar este porcentaje de acierto se calcula la distancia euclídea entre el punto donde se encuentra cada pieza y lo que se considera la esquina de la bandeja P(160, -160). También consideramos la rotación de la pieza (Rotación 1 ($\pi, \frac{3\pi}{2}$], Rotación 2 ($\frac{3\pi}{2}, 0$], Rotación 3 ($0, \frac{\pi}{2}$], Rotación 4 ($\frac{\pi}{2}, \pi$]) y el cuadrante en el que está situada la pieza (I, II, III, IV). Utilizamos el carácter ‘a’ para la rotación y el carácter ‘c’ para identificar el cuadrante en el que se encuentra la pieza. Además, la pieza 1, talón, se asocia a una “_t”, la pieza 2, izquierda, a una “_i” y la pieza 3, derecha, a una “_d”.

Con estas variables definidas a partir de la posición de las piezas en cada bandeja se plantea la identificación de la mejor secuencia de montaje, centrándonos en las seis secuencias más frecuentes. Para ello, como ya se ha comentado se utilizará un árbol condicional de clasificación.

La Figura 4.10 representa el árbol de clasificación teniendo en cuenta las seis mejores secuencias para las bandejas del Fabricante 1 (muestra de ajuste). En este árbol las variables que se tendrían en cuenta son el cuadrante donde se ha colocado cada pieza (“_c_d”, “_c_i” y “_c_t”), la distancia euclídea de la pieza derecha (“_dis_d”) y sobre todo el ángulo de la pieza “talón” (“_a_t”). La variable que más discrimina es el cuadrante en el que está situada la pieza derecha, “_c_d”, separando el cuadrante I de los cuadrantes II, III y IV.

El nodo terminal 3 identifica las bandejas en las que la pieza derecha se encuentra en el cuadrante I en una posición con menos de -134.743 grados de ángulo de rotación. En este caso tendríamos 9 bandejas de las cuales el 80 % se resolverían con S13 y el otro 20 % con S18, por lo que cuando aparezca una bandeja con estas posiciones deberíamos de utilizar S13.

El nodo 9 identifica 14 bandejas en las que la pieza derecha se encuentra en el cuadrante I en una posición con más de 4.815 grados de ángulo de rotación y la distancia de la pieza derecha a la esquina de la bandeja es inferior a 199.168 mm. Para estas bandejas se debería utilizar S14 por ser la secuencia que tiene asociada la probabilidad más alta.

En la Tabla 4.5 las columnas representan qué secuencia ha sido la más rápida en el montaje y las filas, la secuencia predicha por el modelo ajustado mediante al árbol. El 58.44 % de las bandejas utilizadas como muestra de ajuste (Fabricante 1) están correctamente clasificadas, mientras que en la muestra de validación (Fabricante 2) ese porcentaje se reduce a un 28.38 %. El tiempo medio de montaje de las bandejas

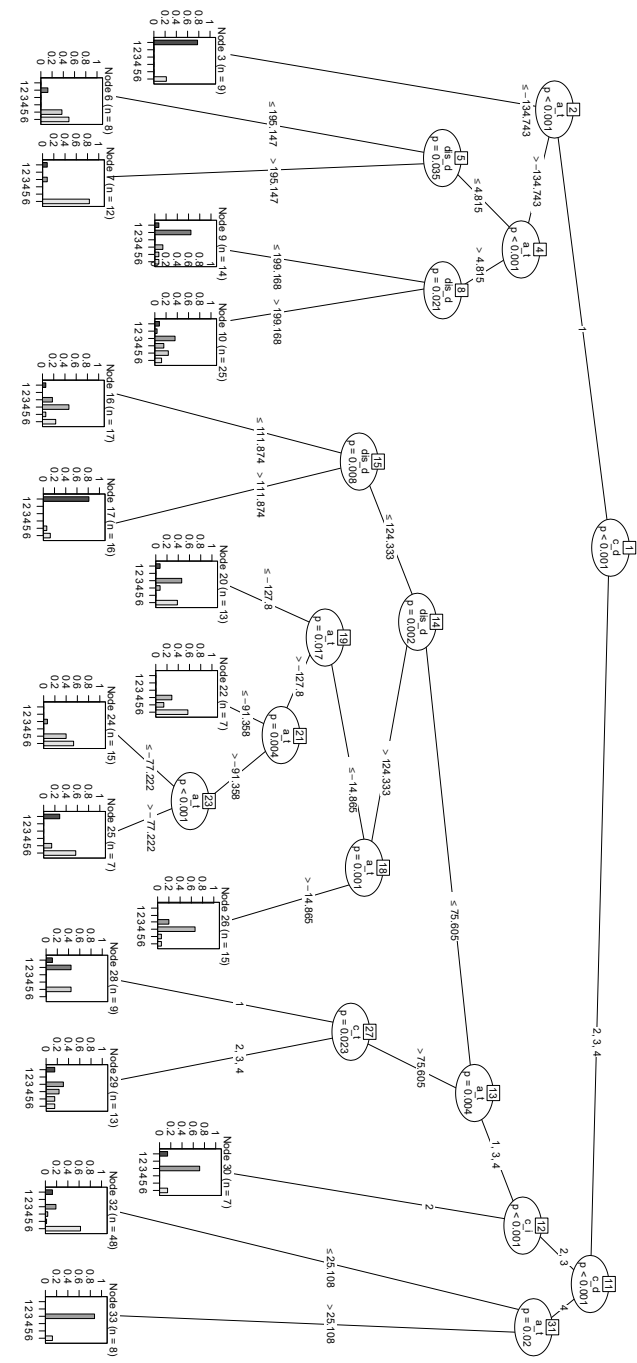


Figura 4.10: Árbol de decisión condicional con el modelo de las seis mejores secuencias.

Tabla 4.5: Ajuste (modelo de zapato Fabricante 1) y validación (modelo de zapato Fabricante 2) en árbol de seis secuencias.

Ajuste	S13	S14	S15	S16	S17	S18
S13	20	0	0	0	1	4
S14	2	13	0	2	5	1
S15	6	1	31	8	8	12
S16	1	0	6	18	2	5
S17	0	0	0	0	0	0
S18	9	1	11	4	12	60
Validación	S13	S14	S15	S16	S17	S18
S13	0	0	7	2	0	4
S14	2	1	2	5	2	4
S15	9	12	19	9	14	19
S16	5	1	4	15	0	5
S17	0	0	0	0	0	0
S18	13	2	9	19	10	28

del primer Fabricante sería de 15.14 sg. y del segundo Fabricante 15.51 sg.

Estos resultados mejoran los obtenidos con cada una de las secuencias por separado. En el caso del Fabricante 1 hemos reducido el tiempo medio por bandeja en 0.04 sg respecto de la mejor (S18), aunque seguimos lejos del mejor tiempo posible, 15.03 sg. En cuanto al Fabricante 2, se observa un incremento del tiempo medio en 0.10 sg respecto a la mejor secuencia individual, 15.42 (S18).

Para intentar mejorar estos resultados reducimos el árbol a tres posibles resultados: S13, S17 y S18. En este árbol, Figura 5.7, los parámetros a tener en cuenta son el cuadrante donde se ha colocado la pieza derecha (“*c.d*”), la distancia euclídea de las piezas derecha e izquierda (“*dis.d*” y “*dis.i*”) y el ángulo de las piezas derecha y talón (“*a.d*” y “*a.t*”). Cuando se colocan las piezas, la primera característica que se tiene en cuenta es el cuadrante en el que se encuentra la pieza derecha, distinguiendo entre los cuadrantes I y II, y el III y IV.

Considerando sólo estas tres secuencias clasificaríamos correctamente el 69.55 % de las bandejas del primer fabricante, con un tiempo medio por bandeja de 15.14 sg., mientras que las bandejas del Fabricante 2 correctamente clasificadas fueron el 42.34 % con un tiempo medio por bandeja de 15.48 sg.

Otro de los aspectos a tener en cuenta es el número de bandejas cuyo montaje se ha podido completar. En la Tabla 4.6 se observa que S18 completaría 216 de las

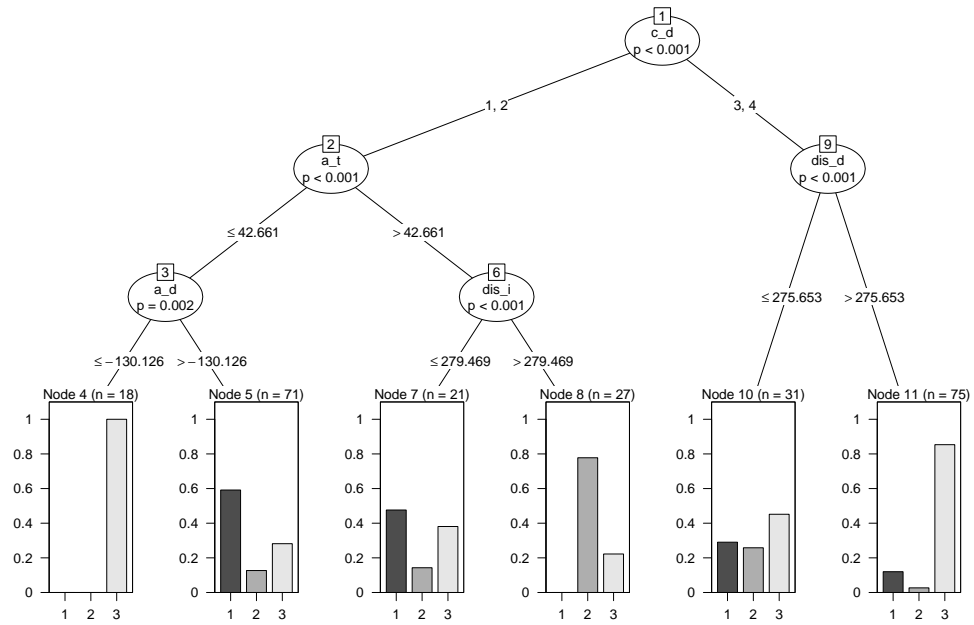


Figura 4.11: Árbol de decisión condicional con el modelo de las tres mejores secuencias.

Tabla 4.6: Ajuste (modelo de zapato Fabricante 1) y validación (modelo de zapato Fabricante 2) en árbol de tres secuencias.

Ajuste	S13	S17	S18
S13	52	12	28
S17	0	21	6
S18	18	10	96
Validación	S13	S17	S18
S13	42	6	40
S17	1	1	10
S18	36	35	51

Tabla 4.7: Resumen resultados.

	Fabricante 1			Fabricante 2		
	n	%	Tiempo(sg)	n	%	Tiempo(sg)
S18	216	88.89	15.18±1.74	208	93.96	15.42±1.24
Árbol (6 sec.)	237	97.53	15.14±1.70	216	97.30	15.51±1.30
Árbol (3 sec.)	236	97.12	15.14±1.70	217	97.75	15.48±1.30

243 bandejas del Fabricante 1, es decir, un 88.89% de las mismas, mientras que esta misma secuencia con el modelo del Fabricante 2 completaría 208 de 222 (93.69%). Por tanto, la utilización del árbol permite mejorar el número de modelos montados sin afectar sustancialmente al tiempo medio de montaje por bandeja.

4.3. Optimización de la trayectoria de montaje

El apartado anterior nos llevaría a programar 3 secuencias de montaje para, en función de los nodos obtenidos con el árbol, montar el modelo con las piezas de cada bandeja. Aún así no se consigue el montaje del 100% de las bandejas.

Las pruebas para comprobar la viabilidad del modelo se han desarrollado con 250 bandejas del modelo Fabricante 1 y otras 250 bandejas del modelo Fabricante 2. En total se han evaluado 500 casos para comprobar la robustez y viabilidad del modelo. Cada bandeja tiene posiciones aleatorias para las tres piezas siempre dentro del espacio de trabajo del robot. El BILP se ha programado en R utilizando la librería `lpsolve`, considerando 3 piezas individuales y un robot con 2 brazos, lo que nos lleva a un problema con 27 restricciones. La trayectoria de cada brazo se analiza para cada bandeja. La Figura 4.12 (fila superior) muestra la trayectoria de los brazos para tres bandejas seleccionadas al azar del modelo Fabricante 1. El cuerpo del robot se sitúa en la posición (250,0) y sus efectores finales (última articulación del robot donde se une la herramienta) en S1 y S2. Los nodos i1, i2 e i3 representan las posiciones de recogida y los nodos j1, j2 y j3 representan las posiciones de colocación. Las líneas rojas representan las trayectorias del brazo derecho y las líneas azules las del brazo izquierdo.

La Figura 4.12 (fila inferior) representa la solución de tres bandejas seleccionadas al azar del modelo del Fabricante 2. La mayor diferencia, respecto al modelo del Fabricante 1, radica en la posición de la primera pieza que es el nodo j1. Algunas trayectorias presentadas podrían presentar conflictos y colisiones durante la

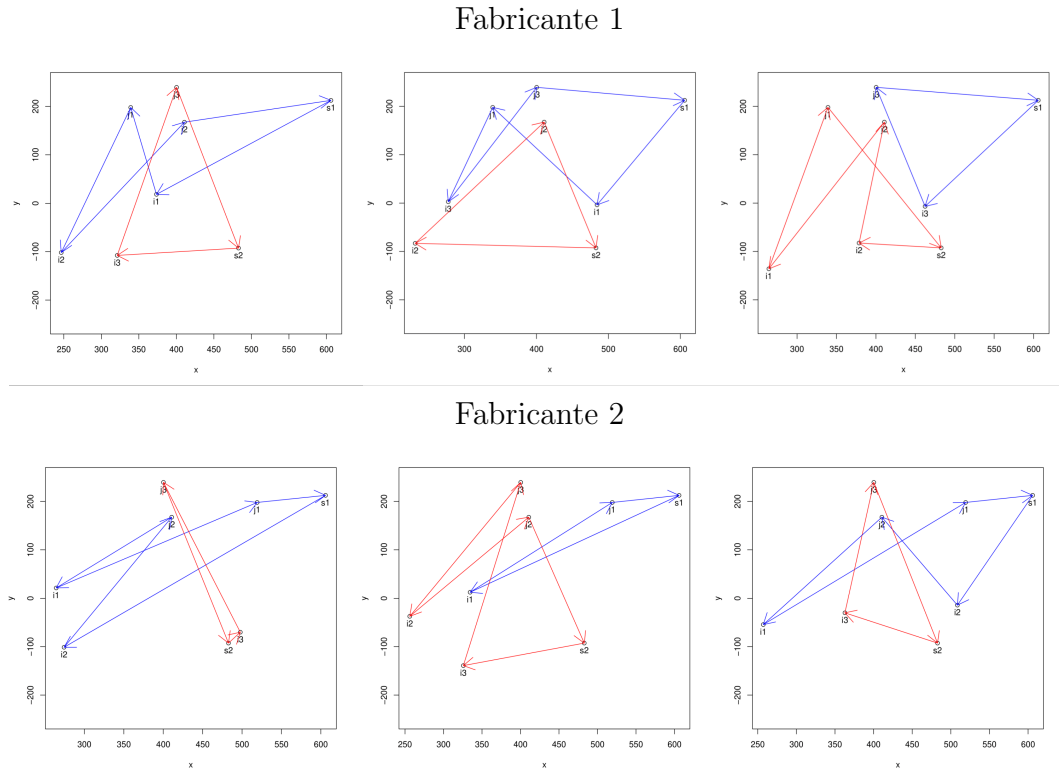


Figura 4.12: Soluciones óptimas para diferentes posiciones de montaje. Fila superior para el modelo de zapato del Fabricante 1. Fila inferior para el modelo de zapato del Fabricante 2.

ejecución de la trayectoria, para evitar estos problemas en el programa del robot se llevan a cabo dos estrategias. La primera medida adoptada es introducir puntos de sincronización, es decir, ambos brazos deben permanecer en estos puntos al mismo tiempo para asegurar que no haya colisiones, y la segunda medida adoptada es dar mayor velocidad al brazo que realiza más operaciones para obligar a que ambos brazos empiecen y terminen al mismo tiempo.

La Tabla 4.8 muestra un resumen de los principales resultados obtenidos al realizar el montaje óptimo de cada modelo. La distancia media por bandeja montada recorrida por los dos brazos es de $1,971.9 \pm 159.9$ mm. para el modelo del Fabricante 1. Se aprecia un valor ligeramente inferior en el caso del Fabricante 2 debido a la posición de la pieza 1 en la tarea de colocación. Los intervalos de confianza al 95 % son bastante estrechos, 39.6 mm y 35.1 mm, respectivamente, lo que confirma la

solidez del modelo.

Tabla 4.8: Datos de la secuencia óptima para el Fabricante 1 y el Fabricante 2.

	Media	Sd	95%CI	Mediana	Min	Max
Fabricante 1	1971.9	159.9	1952.1-1991.7	1973.2	1634.5	2356.4
Fabricante 2	1948.4	141.4	1930.8-1965.9	1971.9	1563.5	2220.8

Se observa que la distribución de distancias recorridas es prácticamente simétrica para el modelo del Fabricante 1, y asimétrica a la izquierda para el modelo del Fabricante 2, no observándose valores atípicos en ninguno de los dos casos, Figura 4.13. La caja representa el 50 % de los datos centrales y la línea negra representa la mediana. La gráfica de densidad nos muestra lo parecidas que son ambas distribuciones, aunque, como ya se ha comentado, se esperan distancias recorridas mayores en el modelo del Fabricante 1 por la posición final de la pieza 1 en el molde de montaje.

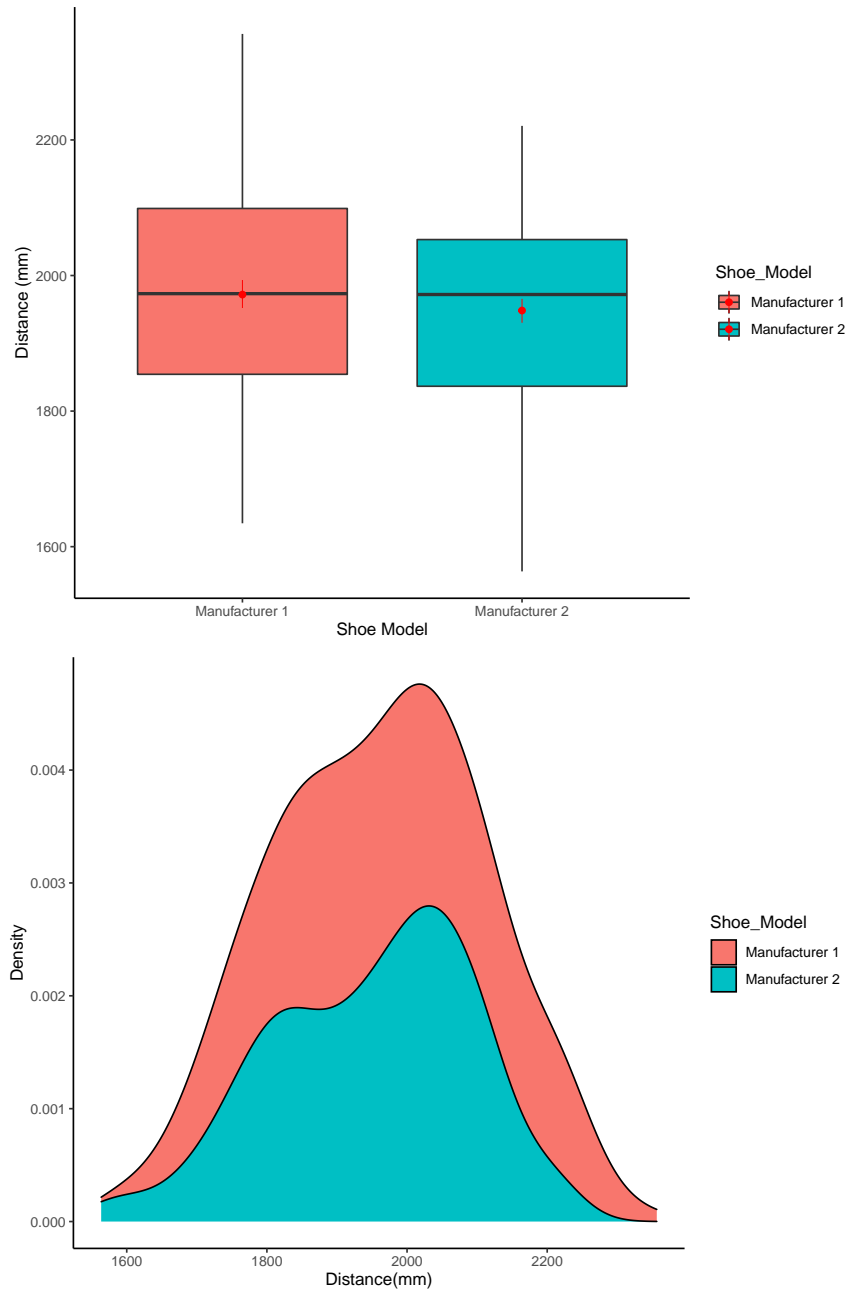


Figura 4.13: Diagrama de cajas y gráfico de densidad para la distribución de distancias recorridas con cada fabricante.

4.3.1. Coste computacional

El tiempo empleado en calcular la mejor secuencia consideramos que es una característica esencial. Si el tiempo en encontrar la secuencia más rápida es superior a la ganancia que se obtiene con la misma frente a una secuencia aleatoria o ya prefijada, el modelo no es útil.

El tiempo medio para encontrar la trayectoria óptima es de 0.0201 ± 0.0102 sg. en las bandejas del modelo del Fabricante 1, mientras que para el Fabricante 2 sería de 0.0181 ± 0.0075 sg., Tabla 4.9. Se aprecia una menor dispersión en la distribución de tiempos de cálculo del modelo del Fabricante 2 ($cv=0.507$ vs $cv=0.414$).

Tabla 4.9: Resumen del coste computacional en segundos.

	Media	Sd	Mediana	Min	Max
Fabricante 1	0.0201	0.0102	0.0200	<0.0001	0.0500
Fabricante 2	0.0181	0.0075	0.0200	0.0100	0.0400

La distribución de tiempos de cálculo es ligeramente asimétrica a la derecha en el caso del Fabricante 1. La del Fabricante 2 presenta un valor atípico, se necesitan 0.04 sg. para obtener la trayectoria óptima de una bandeja, mientras que el resto está entre 0.01 y 0.03. En ambos casos el 50% de los tiempos está por debajo de los 0.02 sg, aunque haya más variabilidad en el modelo del Fabricante 1. Estos tiempos confirma la rapidez para resolver el problema de optimización (véase Figura 4.14))

En la Figura 4.15 se presenta un análisis del coste computacional global a medida que aumenta el número de bandejas analizadas. El comportamiento de ambos modelos es lineal, teniendo valores de tiempo similares para calcular la mejor trayectoria.

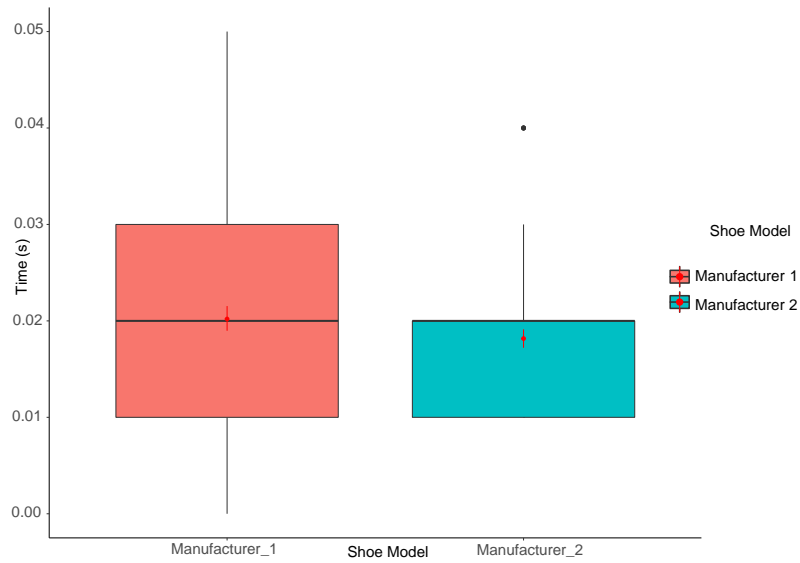


Figura 4.14: Diagrama de cajas del coste computacional en segundos.

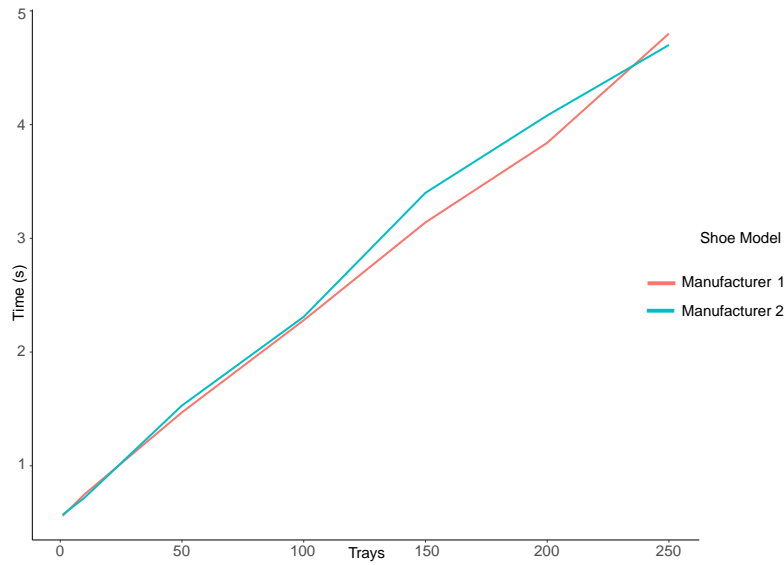


Figura 4.15: Coste computacional en función del número de bandejas para ambos modelos.

4.3.2. Generalización para n piezas

Se realizaron pruebas para comprobar la robustez del algoritmo para 4, 5, 6 y 7 piezas. En cada caso, se produjeron 250 bandejas y se obtuvo la solución óptima. En la Figura 4.16 se muestra el gráfico de la secuencia de una bandeja para cada uno de los casos. El brazo derecho se representa con líneas rojas y el izquierdo con líneas azules.

En el análisis realizado para los diferentes números de piezas, se ha obtenido la distancia óptima media y el coste computacional, como se puede observar en la Figura 4.17. En el caso de la distancia óptima, ésta sigue una función proporcional que aumenta con el número de piezas, mientras que el coste computacional aumenta exponencialmente, debido a que el número de restricciones aumenta exponencialmente con el número de piezas.

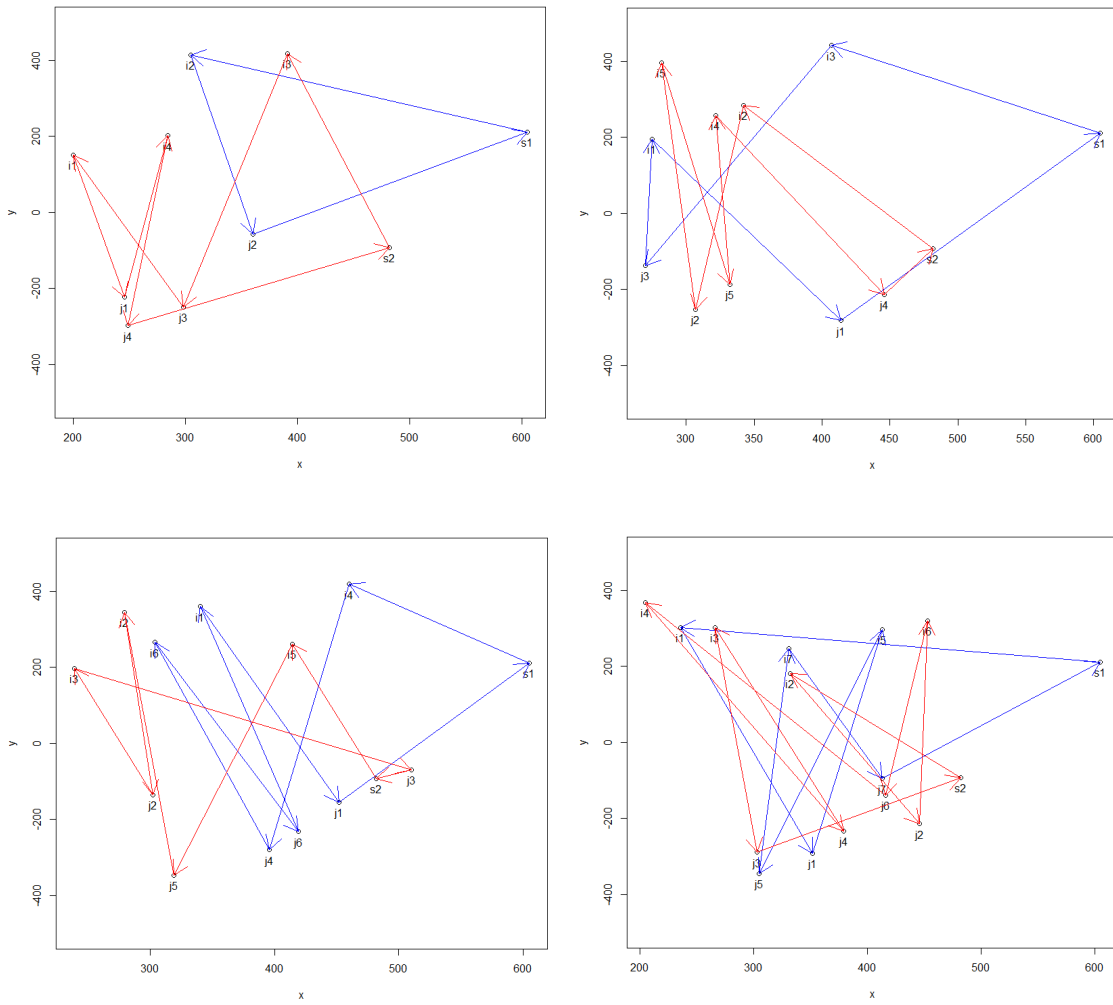


Figura 4.16: Soluciones óptimas para distintos números de piezas. La línea azul corresponde al brazo rojo y la línea azul al brazo izquierdo.

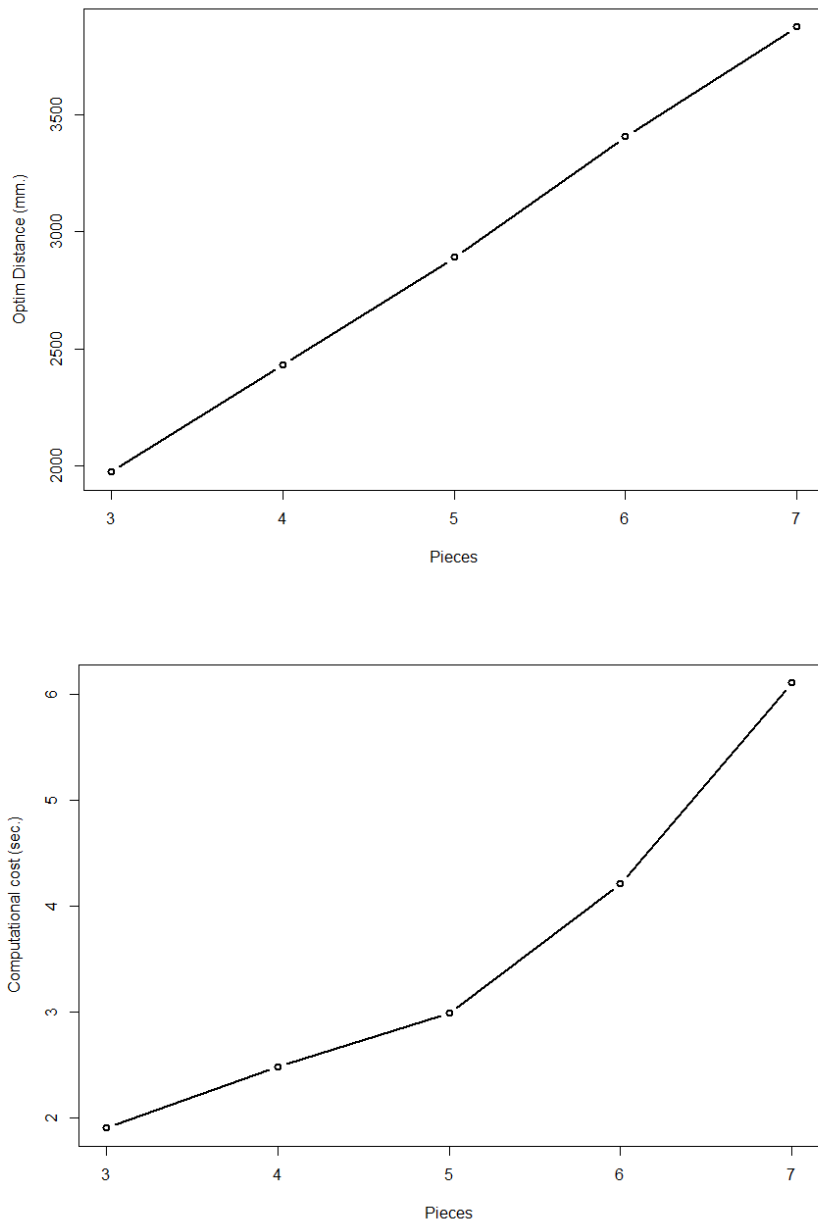


Figura 4.17: Distancia óptima media y coste computacional para distinto número de piezas.

Capítulo 5

Discusión

La investigación realizada en [66] presenta un Modelo de Árbol de Decisión para optimizar el tiempo de la tarea, y reduce todas las secuencias posibles para tener un número reducido de ramas en el árbol, minimizando el coste computacional. Para comparar ambos modelos, han sido seleccionadas las secuencias óptimas de [66] que son la Simulación 13 (pieza 1 recogida por el brazo izquierdo y piezas 2 y 3 recogidas por el brazo derecho), la Simulación 17 (pieza 3 recogida por el brazo derecho, pieza recogida 1 por brazo izquierdo y pieza 2 recogida por brazo derecho) y la Simulación 18 (pieza 3 recogida por el brazo derecho, pieza recogida 2 por brazo derecho y pieza 1 recogida por brazo izquierdo) del modelo de árbol de decisión, comparando la distancia que recorren los brazos en estas secuencias con el modelo BILP presentado también en esta tesis.

La información obtenida se presenta en la Figura 8, donde se representa la distancia para cada una de las simulaciones y fabricantes frente a la distancia obtenida por el modelo BILP. La línea roja muestra los puntos medios entre ambos modelos, cuando la distancia recorrida es la misma. La distancia media recorrida en la Simulación 13 para el modelo del Fabricante 1 es de 2111.1 mm y su desviación estándar es de 163.8. Para la Simulación 17, la distancia media recorrida es de 2148.3 mm y su desviación estándar de 156.6 para el modelo del Fabricante 1 mientras que para la simulación 18 la distancia media recorrida es de 2148.3 mm y su desviación estándar de 156.6. La distancia media recorrida en la simulación 13 para el modelo del Fabricante 2 es de 2128.4 mm y su desviación estándar es de 147.86. En la simulación 17, la distancia media recorrida es de 2173.5 mm y la desviación típica de 171.4 mm para el modelo del Fabricante 2. Para la simulación 18, la distancia media recorrida es de 2173.5 mm y la desviación típica de 171.4 mm para el modelo del Fabricante

2. Como se puede apreciar a la hora de minimizar la distancia la simulación 17 y la 18 resultan con los mismos valores, ya que las parejas pieza-brazo son las mismas solo cambia el orden, por lo que hemos escogido la simulación 18 para seguir con el estudio comparativo, debido a que es la simulación que tiene más frecuencia. Por lo tanto, la mejora de utilizar el modelo BILP resulta en una disminución media por bandeja: 9.0 % para el modelo Fabricante 1 y 11.6 % para el modelo Fabricante 2.

Por un lado, estos resultados refuerzan la validez de la propuesta mediante árboles de decisión, mientras que por otro, demuestran que la nueva propuesta permite reducir los tiempos de ejecución de las tareas con un coste computacional asumible. Se ha conseguido reducir las trayectorias que requiere un robot de doble brazo para colocar todas las piezas dentro del molde. Teniendo en cuenta que este proceso de pick and place supone el 83 % del tiempo de montaje de un zapato, este enfoque logra una reducción de aproximadamente el 7.1 % y el 9.0 % del tiempo total con respecto al modelo del árbol de decisión. Los tiempos medios de los modelos Fabricante 1 y Fabricante 2 en el modelo de árbol de decisión son de 15.14 y 15.51 segundos respectivamente. Al proceso de pick and place hay que añadir 1.5 seg. de tiempo de inicio y 5 seg. de tiempo final. Esto suma un tiempo total de tarea de 21.64 seg. para el modelo Fabricante 1 y de 22.01 seg. para el modelo Fabricante 2. Si consideramos que un robot de doble brazo puede procesar 1308 moldes durante una jornada laboral de 8 horas, y teniendo en cuenta que con el árbol de simulación se tarda, de media, 21.64 seg. en procesar un molde, se reduciría el tiempo total de montaje de todas las bandejas en 42.46 min. para el modelo Fabricante 1 y 55.66 min. para el modelo Fabricante 2, lo que implica que en el mismo tiempo se montan 129 moldes más para el modelo Fabricante 1 y 171 moldes más para el modelo Fabricante 2 con el modelo BILP propuesto. Como puede verse, los puntos mostrados en la Figura 8 están siempre por encima de la línea roja, como máximo con el modelo de árbol de decisión se obtiene la misma distancia recorrida. Así que el modelo BILP es siempre igual o mejor que el modelo de árbol de decisión en la minimización de la distancia.

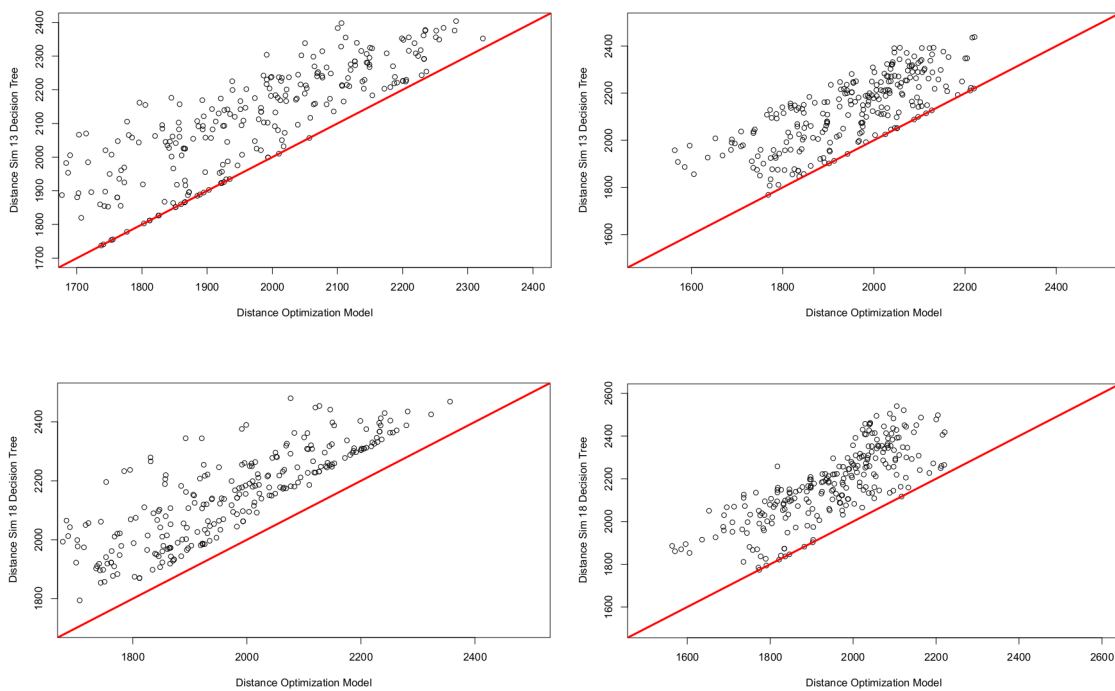


Figura 5.1: Comparativa entre el modelo de árbol de decisión y el modelo BILP. a) Comparativa entre la simulación 13 y el enfoque BILP para el modelo del Fabricante 1, b) Comparativa entre la simulación 18 y el enfoque BILP para el modelo del Fabricante 1, c) Comparativa entre la simulación 13 y el enfoque BILP para el modelo del Fabricante 2, d) Comparativa entre la simulación 18 y el enfoque BILP para el modelo del Fabricante 2.

Capítulo 6

Conclusiones

Hemos desarrollado en esta tesis doctoral un algoritmo BILP para pick and place en la producción de calzado con un robot colaborativo de doble brazo. Para llegar al mismo se ha presentado en primer lugar un diseño de planta de producción que nos permite valorar las ventajas de la fabricación mediante técnicas de inyección directa frente a la técnica clásica de montaje y pegado. Como técnica de inyección directa hemos considerado el 3D Bonding que se caracteriza por inyectar polímero en un molde para fabricar el zapato.

Nos hemos centrado en la resolución del pick and place mediante la fijación de 30 combinaciones de movimientos (secuencias) de los brazos robóticos. Esto nos ha permitido valorar, sobre todo, los problemas para montar las piezas de una bandeja en el molde. Hemos observado que un porcentaje pequeño, pero no despreciable, de las casi 500 bandejas consideradas no han podido ser completadas.

Hemos podido plantear estas secuencias porque trabajamos con un problema pequeño, donde sólo tenemos que trasladar tres piezas de forma individual y una cuarta para la que necesitamos los dos brazos. El trabajo se centra en la colocación de las piezas que se cogen con un solo brazo. Para validar las propuestas se han considerado dos modelos diferentes de calzado.

La utilización de árboles de decisión nos ha permitido reducir a tres las secuencias de colocación, de forma que una vez analizada, mediante la captura de la imagen, la posición y orientación de las piezas en la bandeja, el robot elige la secuencia más probable de estas tres para el traslado de las piezas al molde.

La utilización de esta metodología permite un análisis a priori que no interfiere en el montaje de la bandeja, es decir, los cálculos son realizados previamente. Se programan las tres secuencias en el robot y una vez asignada la bandeja al nodo

correspondiente, en base, como hemos indicado, a la posición y orientación de las piezas, el robot ejecuta automáticamente la secuencia más probable asignada a ese nodo.

En una segunda parte se ha buscado la optimización de la trayectoria de montaje de cada bandeja, controlando las posibles colisiones. Esto nos permite reducir a cero las bandejas que no son montadas y que, por tanto, pueden afectar a la cadena de producción. Se han resuelto más de 500 casos y en todos ellos se alcanza una solución válida, obteniendo la trayectoria óptima.

Los resultados obtenidos demuestran la robustez del modelo. No hay datos fuera del espacio de trabajo, todas las trayectorias se calculan de forma óptima, dando resultados consistentes. Otro punto a tener en cuenta es el tiempo necesario para calcular esta secuencia óptima. Los resultados dan una media de 0.02 sg. para calcular cada bandeja. Este tiempo aumenta linealmente cuando se incrementa el número de bandejas.

En comparación con la propuesta anterior, observamos que la utilización de árboles de decisión conlleva un estudio previo para fijar las secuencias, mientras que en el BILP no es necesario, y el tiempo de computación necesario para encontrar la trayectoria óptima es mínimo. El BILP, frente a los árboles, permite completar el montaje de todas las bandejas.

Capítulo 7

Conclusions

In this thesis, we have developed a BILP algorithm for pick and place in the production of shoes with a dual-arm collaborative robot. In order to achieve this, we have firstly presented a manufacturing plant design that allows us to assess the advantages of manufacturing using direct injection techniques as opposed to the classic assembly and gluing techniques. As a direct injection technique, we considered 3D Bonding, which is characterised by the injection of polymer into a mould to produce the shoe.

We focused on the resolution of the pick and place process by defining 30 combinations of movements (sequences) of the robot arms. This allowed us to assess, in particular, the problems associated with the assembly of the parts of a shell in the mould. We found that a small but not negligible percentage of the nearly 500 trays considered could not be completed.

We were able to set up these sequences because we were working with a small problem where we only had to move three pieces individually and a fourth for which we needed both arms. The work focuses on the placement of the pieces that are picked up with only one arm. To validate the proposals, two different shoe models were considered.

The use of decision trees has allowed us to reduce the number of placement sequences to three, so that once the position and orientation of the parts in the tray have been analysed by image acquisition, the robot selects the most likely of these three sequences to transfer the parts to the mould.

The use of this method allows an a priori analysis that does not interfere with the assembly of the tray, i.e. the calculations are made beforehand, the three sequences are programmed in the robot and once the tray has been assigned to the correspon-

ding node, based, as we have said, on the position and orientation of the parts, the robot automatically executes the most probable sequence assigned to that node.

In a second part, we have tried to optimise the assembly path of each tray, checking for possible collisions, which allows us to reduce to zero the number of trays that are not assembled and can therefore affect the production line. More than 500 cases have been solved and in all of them a valid solution is reached, obtaining the optimal trajectory.

The results obtained demonstrate the robustness of the model. There is no data outside the workspace, all trajectories are calculated in an optimal way, giving consistent results. Another point to consider is the time taken to calculate this optimal sequence. The results show an average of 0.02 seconds to calculate each tray, and this time increases linearly as the number of trays increases.

In comparison with the previous proposal, we can see that the use of decision trees requires a previous study to determine the sequences, whereas in the BILP this is not necessary and the computation time required to find the optimal trajectory is minimal. The BILP, unlike trees, allows the assembly of all the trays to be completed.

Capítulo 8

Líneas futuras

Una vez avanzado en la problemática del planificador de tareas, tal y como se comentó en los objetivos, una línea de investigación que se plantea es el desarrollo de algoritmos de planificación de trayectorias en tiempo real, en la Figura 2.1. se resaltan las capas de control con las que se continuaría esta investigación. Estos algoritmos permiten planificar y generar trayectorias de forma eficiente, teniendo en cuenta las restricciones de movimiento del robot y los objetos que se deben recoger y transportar. Este tipo de algoritmos es especialmente importante en entornos no estructurados, donde los objetos pueden estar en diferentes posiciones y orientaciones, y es necesario planificar la ruta óptima en tiempo real.

En concreto, los pasos a seguir a partir de este momento serían los siguientes:

1. Incorporación al modelo de los datos de aceleración/desaceleración de los brazos teniendo en cuenta el robot utilizado. Esto nos llevará a minimizar el tiempo de montaje, y muy probablemente a un problema no lineal.
2. Incorporación de restricciones de alcanzabilidad al problema de optimización.
3. Evaluación de la solución óptima en el planificador de trayectorias. Si dicha solución no fuese válida se plantean dos escenarios de trabajo:
 - Incorporar automáticamente al problema una restricción que no permita alcanzar la solución óptima obtenida y volver a resolverlo.
 - La generación de un conjunto factible de soluciones al problema evaluando para cada una de ellas su viabilidad en el planificador de trayectorias, parando cuando nos encontremos a una distancia mínima de la solución óptima desechada.

Por otro lado, la utilización de técnicas de visión por computadora avanzadas nos puede ayudar a mejorar la eficiencia y precisión del proceso de detección y localización de objetos, lo que a su vez puede reducir el tiempo necesario para ejecutar la tarea de pick and place.

Se plantea también la integración de sensores de proximidad y fuerza en el sistema. Este tipo de sensores pueden ayudar a detectar y evitar colisiones entre el robot y los objetos circundantes, lo que puede reducir el tiempo necesario para realizar ajustes y correcciones durante la ejecución de la tarea de pick and place. Además, pueden ayudar a mejorar la precisión en el agarre y transporte de los objetos.

Bibliografía

- [1] Schmél, F. “Structure of Production Costs in Footwear Manufacture,” United Nations Industrial Development Organization: Vienna, Austria, 2003.
- [2] Boër, C.R.; Dulio, S. “The EUROShoE project. In Mass Customization and Footwear: Myth, Salvation or Reality?” Springer: London, UK, 2007.
- [3] Jeng, Y.-R.; Liu, D.-S.; Yau, H.-T. “Fast numerical algorithm for optimization mold shape of direct injection molding process,” *Mater. Manuf. Process.* 2013, 28, 689–694.
- [4] Denkena, B.; Scherger, S. “A concept for shoe last manufacturing in mass customisation,” *CIRP Ann.* 2005, 54, 341–344.
- [5] Jatta, F.; Zanoni, L.; Fassi, I.; Negri, S. “A roughing/cementing robotic cell for custom made shoe manufacture,” *Int. J. Comput. Integr. Manuf.* 2004, 17, 645–652.
- [6] Lee, C.-Y.; Kao, T.-L.; Wang, K.-S. “Implementation of a robotic arm with 3d vision for shoes glue spraying system,” *Proceedings of the 2nd International Conference on Computer Science and Artificial Intelligence*, Shenzhen, China, 8–10 December 2018; pp. 562–565.
- [7] Mendez, J.B.; Perez-Vidal, C.; Heras, J.V.S.; Perez-Hernandez, J.J. “Robotic pick-and-place time optimization: Application to footwear production,” *IEEE Access* 2020, 8, 209428–209440.
- [8] Serweta W., Olejniczak Z. and Matusiak M., “Improve of Footwear Comfort Sensation with Material Packages and Knitted Fabrics,” *Fibres and Textiles in Eastern Europe* 2019, 27 (3) 85-90.

-
- [9] Kim S., Seong H., Her Y. and Chun J., “A study of the development and improvement of fashion products using a FDM type 3D printer,” *Fashion and Textiles* 2019, 6 (9).
- [10] SimplicityWorks. (2020). 3D Bonding. SimplicityWorks Europe S.L. <http://www.simplicity.works/bond>
- [11] Y. Gao, J. Lin, Z. Chen, M. Fang, X. Li and Y. -H. Liu, “Deep-Learning Based Robotic Manipulation of Flexible PCBs,” 2020 IEEE International Conference on Real-time Computing and Robotics (RCAR), Asahikawa, Japan, 2020, pp. 164-170, doi: 10.1109/RCAR49640.2020.9303294.
- [12] Z. Shiller and S. Dubowsky, “On computing the global timeoptimal motions of robotic manipulators in the presence of obstacles,” *IEEE Trans. Robot. Automat.*, vol. 7, no. 6, pp. 785-797, Dec. 1991.
- [13] T. Fletcher, “The undersea mobot,” *Nucl. Electron. Lab. Hughes Aircr. Company, Culver City, CA, USA, Tech. Rep.*, 1960.
- [14] R. O. Ambrose, H. Aldridge, R. S. Askew, R. R. Burrige, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark, “Robonaut: NASA’s space humanoid,” *IEEE Intell. Syst.*, vol. 15, no. 4, pp. 57-63, Jul. 2000.
- [15] S. Stavridis and Z. Doulgeri, “Bimanual assembly of two parts with relative motion generation and task related optimization,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 7131-7136.
- [16] C. Smitha, Y. Karayiannidisa, L. Nalpantidisa, X. Gratala, P. Qia, D. V. Dimarogonasa, and D. Kragica, “Dual arm manipulation- A survey,” *Robot. Auton. Syst.*, vol. 60, no. 10, pp. 1340-1353, 2012.
- [17] Beuke F., Alartartsev S. and Jessen Verl A. “Responsive and Reactive Dual-Arm Robot Coordination,” *IEEE International Conference on Robotics and Automation ICRA* 316-322, 2018.
- [18] Gafur, N., Kanagalingam, G.; Ruskowski, M., “Dynamic collision avoidance for multiple robotic manipulators based on a non-cooperative multi-agent game,” <http://arxiv.org/abs/2103.00583>, 2021.

- [19] Y. Jiang, Y. Wang, Z. Miao, J. Na, Z. Zhao and C. Yang, “Composite-Learning-Based Adaptive Neural Control for Dual-Arm Robots With Relative Motion,” in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 3, pp. 1010-1021, March 2022, doi: 10.1109/TNNLS.2020.3037795.
- [20] P. Yu and D. V. Dimarogonas, “Distributed Motion Coordination for Multirobot Systems Under LTL Specifications,” in *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 1047-1062, April 2022, doi: 10.1109/TRO.2021.3088764.
- [21] Ahmadi R. H. and Mame J. W., “Routing heuristics for automated pick-and-place machines.” *European Journal of Operation*, 117(3), 533–552, 1999.
- [22] Ayob M. and Kendall G., “A triple objective function with a Chebychev dynamic pick-and-place point specification approach to optimise the surface mount placement machine,” *European Journal of Operational Research*, 164(3), 609–626, 2005.
- [23] Gao, J., Zu, X., Liu, A.B., Meng, Q.Y. and Zhang, R.T. “An Iterated Hybrid Local Search Algorithm for Pick-and-Place Sequence optimisation,” *European Journal of Operational Research*, Volume 10, 11, 633, 2018.
- [24] Boudjedir C.E. , Bouri M. and Boukhetala D., “Model-Free Iterative Learning Control With Nonrepetitive Trajectories for Second-Order MIMO Nonlinear Systems-Application to a Delta Robot,” *IEEE Transactions on Industrial Electronics*, vol. 68, 8, pp. 7433-7443, 2021.
- [25] Tahriri F., Mousavi M., Yap H.J., Siti Zawiah, M.D. and Taha, Z. “Optimizing the robot arm movement time using virtual reality robotic teaching system” *International Journal of Simulation Modelling* 14(1) 28-38,2015.
- [26] Wong C., Shackleford S., Potter D., Richardson J. -P., McDermott L. and Nolan J., “Robotic Task Sequencing and Motion Coordination for Multiarm Systems,” *IEEE/ASME Transactions on Mechatronics*,, 2022. doi: 10.1109/T-MECH.2022.3178019.
- [27] Wang Z., Gan Y. and Dai X., “Assembly-Oriented Task Sequence Planning for a Dual-Arm Robot,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8455-8462, July 2022, doi: 10.1109/LRA.2022.3183786.

-
- [28] Shahria, M.T.; Sunny, M.S.H.; Zarif, M.I.I.; Ghommam, J.; Ahamed, S.I.; Rahman, M.H. “A Comprehensive Review of Vision-Based Robotic Applications: Current State, Components, Approaches, Barriers, and Potential Solutions.” *Robotics* 2022, 11, 139. <https://doi.org/10.3390/robotics11060139>
- [29] Ozawa, R.; Tahara, K.; “Grasp and dexterous manipulation of multi-fingered robotic hands: a review from a control view point.” *Advanced Robotics* 2017, 1030-1050. <https://doi.org/10.1080/01691864.2017.1365011>
- [30] Kong X., Gao Y, Wang T., Liu J. and Xu W. “Multi-robot Task Allocation Strategy based on particle Swarm optimisation and Greedy Algorithm,” *IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China*, pp. 1643-1646, 2019.
- [31] Chernikov S.Y., Blagodarov D.A., Safonov Y.M., Gosteva L.O. and Fedortsov N.N. “Reference Trajectory optimisation for an Industrial Robot,” *10th International Conference on Electrical Power Drive Systems (ICEPDS), 2018*.
- [32] Goertz R.C. “Fundamentals of general-purpose remote manipulators,” *Nucleonics*, 10 (11): 36-45, 1952.
- [33] Aminzadeh V., Wurdemann H.A., . Dai J.S., Reed J. and Purnell G. “A new Algorithm for Pick-and-Place Operations,” *Industrial Robot: An International Journal*, 2010.
- [34] Bobrow J.E., Dubowsky S. and Gibson J.S. “Time-Optimal Control of Robotic Manipulators Along Specified Paths,” *International Journal Robot Res.* 4 (3) 3–17, 1985.
- [35] Zhang K., Gao X.S., Li H.B. and Yuan C.M. “A Greedy Algorithm for Feed-rate Planning of CNC Machines along Curved Tool Paths with Confined Jerk for Each Axis,” *Robot Comput. Integr. Manuf.* 28:472–83, 2012.
- [36] Zhang Q. and Li S.R. “Efficient Computation of Smooth Minimum Time Trajectory for CNC Machining,” *International Journal Adv. Manuf. Technol.*, 68 (1-4) 683–92, 2013.
- [37] Zhang Q., Li S.R. and Gao X.S. “Practical smooth minimum time trajectory planning for path following Robotic manipulators,” *Am Control Conf, USA*, 17–9, 2013.

-
- [38] Y. Liu, C. Guo and Y. Weng, “Online Time-Optimal Trajectory Planning for Robotic Manipulators Using Adaptive Elite Genetic Algorithm With Singularity Avoidance,” *IEEE Access*, vol. 7, pp. 146301-146308, 2019. doi: 10.1109/ACCESS.2019.2945824.
- [39] Cheng K. P. , Mohan R. E., Khanh Nhan N. H. and Le A. V. “Multi-Objective Genetic Algorithm-Based Autonomous Path Planning for Hinged-Tetro Reconfigurable Tiling Robot,” *IEEE Access*, vol. 8, pp. 121267-121284, 2020.
- [40] Goldberg D. E. “Genetic algorithms in search optimisation and machine learning,” *MA: Addison-Wesley, Boston.* , 1989.
- [41] Holland J. H. “Optimal trajectories of robot arms minimizing constrained actuators and travelling time,” *MI: University of Michigan Press*, 1975.
- [42] Jung E., and Kim I. “Hybrid Imitation Learning Framework for Robotic Manipulation Tasks,” *Sensors*, 21 (10), 2021.
- [43] Alazzam AR. “Using BUA Algorithm to Solve a Sequential pick-and-place Problem,” *International Conference on Information and Computer Technologies (ICICT), 2018*.
- [44] Zanchettin, A.M., Messeri, C., Cristantielli, Rocco P. “Trajectory optimisation in collaborative robotics based on simulations and genetic algorithms,” *Int J Intell Robot Appl*, 2022. <https://doi.org/10.1007/s41315-022-00240-4>
- [45] Bessonnet G. and Lallemand J.P.. “Adaptation in natural and artificial systems,” Ann Arbor, Proceedings., *IEEE International Conference on Robotics and Automation, Cincinnati, OH, USA*, pp. 112-117 vol.1, 1990.
- [46] Gecks T. and Henrich D. “Human-robot cooperation: Safe pick-and-place operations,” *In Proceedings of the IEEE International workshop on robots and human interactive communication*, (pp. 549–554). Nashville, USA, 2005.
- [47] Daoud S. , Chehade H., Yalaoui F. and Amodeo L. “Efficient metaheuristics for pick-and-place robotic systems optimisation,” *J. Intell. Manuf. vol.*, 25, no. 1, pp. 27–41, 2014.
- [48] G. A. Calangiu, I. Sarkany, M. Stoica, and F. Sisak, “Modeling a knowledge base for generating new trajectories for a robot arm located in a cell of a flexible

- fabrication line,” in Proc. 12th Int. Conf. Optim. Electr. Electron. Equip., May 2010, pp. 813-818.
- [49] Y. Xie and J. Li, “Path planning based on robot posture control in spraying,” in Proc. 35th Chin. Control Conf. (CCC), Jul. 2016, pp. 6098-6102.
- [50] G. Wang, X. Shang, Y. Yan, J. K. Allen, and F. Mistree, “A tree-based decision method for the configuration design of reconfigurable machine tools,” *J. Manuf. Syst.*, vol. 49, pp. 143-162, Oct. 2018.
- [51] D. Liu, Z. Wang, B. Lu, M. Cong, H. Yu, and Q. Zou, “A reinforcement learning-based framework for robot manipulation skill acquisition,” *IEEE Access*, vol. 8, pp. 108429-108437, 2020, doi: 10.1109/ACCESS.2020.3001130.
- [52] Y. Zhou et al., “Robust Task-Oriented Markerless Extrinsic Calibration for Robotic Pick-and-Place Scenarios”, in *IEEE Access*, vol. 7, pp. 127932-127942, 2019, doi: 10.1109/ACCESS.2019.2913421.
- [53] Huang L. and Jiang R. “A new method of inverse kinematics solution for industrial 7DOF robot”, *Proceedings of the 32nd Chinese Control Conference, Xian, China*, pp. 6063-6065, 2013.
- [54] Li L., Gruver W. A., Zhang Q. and Yang Z. “Kinematic control of redundant robots and the motion optimizability measure”, *IEEE Transactions on Systems, Man, and Cybernetics, piece B (Cybernetics)*, vol. 31, no. 1, pp. 155-160, 2001.
- [55] Zaplana I., Claret J.A., Basanez L. “Análisis Cinemático De Robots Manipuladores Redundantes: Aplicación a los Robots Kuka LWR 4+ y ABB YuMi” *RIAI: Revista Iberoamericana de Automática e Informática Industrial*, 15, 192-202, 2018.
- [56] Liang J., Zhang G., Wang W., Hou Z., Li J., Wang X. and Han C., “Dual quaternion based kinematic control for Yumi dual-arm robot”, 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, Korea (South), pp. 114-118, 2017.
- [57] Hothorn T., Hornik K., Zeileis A. “Unbiased recursive partitioning: A conditional inference framework” *Journal of Computational and Graphical Statistics* 15. (3): 651-674, 2016.

- [58] Molnar C. “Recursive partitioning by conditional inference” *Seminar paper. Department of Statistics University of Munich*, 2013.
- [59] Matelli, J. A., Bazzo, E., & da Silva, J. C. “An expert system prototype for designing natural gas cogeneration plants.” *Expert Systems with Applications* 36, 8375-8384, 2009.
- [60] Dym, C. L., & Levitt, R. E. “Knowledge-based systems in engineering.” New York: McGraw-Hill, 1991.
- [61] Kumar, S.L. “Knowledge-based expert system in manufacturing planning: State-of-the-art review.” *Int. J. Prod. Res.*, 57, pp. 4766-4790, 2018.
- [62] Gabbar, H. A., Suzuki, K., & Shimada, Y. . “Plant object-orientated model formalization: A case study: HDS plant design.” *Design Studies*, 24, 101-108, 2003.
- [63] Altिंग, L. “Manufacturing Engineering Processes. Marcel Dekker Inc. 1994.
- [64] Colledani, M., Terkaj, W., Tolio, T. & Tomasella, M. “Development of a Conceptual Reference Framework to Manage Manufacturing Knowledge Related to Products”, *Processes and Production Systems* (pp.259-284). 10.1007/978-3-540-78431-9-15, 2008.
- [65] SimplicityWorks, 2023, 3D Bonding. SimplicityWorks Europe S.L. <http://www.simplicity.works/bond>
- [66] Borrell Méndez J., Perez-Vidal C., Segura Heras J. V. and Pérez-Hernández J. J. “Robotic Pick-and-Place Time optimisation: Application to Footwear Production”, *IEEE Access*, vol. 8, pp. 209428-209440, 2020.

Glosario

PCB	Printed Circuit Board	6, 8
ILC	Iterative Learning Control	9
MI-MO	Multi input multi output	9
RTSP	Robotic Task Sequencing Problems	9
TSP	Travelling Salesman Problem	11
GA	Genetic Algorithm	11
HIL	Hybrid Imitation Learning	11
BC	Behaviour Cloning	11
SC	State Cloning	11
BUA	Best Uniform Algorithm	11
CTREE	Conditional Tree	21
BILP	Binary Integer Linear Problem	22, 23, 24, 26, 27, 55, 56, 57, 59, 76

Anexo 1 - Código en Python del BILP

Se adjunta el código de Python del modelo matemático BILP para n piezas y 2 brazos robots. El primer código es el programa principal en el cual se llama a una función que genera la matriz de números enteros binarios.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu May 19 10:58:48 2022
4
5 @author: Jorge
6 """
7 from BILP_Matrix import *
8
9
10
11 print("Introduce el numero de piezas: ")
12 num_piezas = input(); num_piezas=int(num_piezas)
13 print("introduce el numero de robots: ")
14 num_robots = input(); num_robots=int(num_robots)
15 print("El problema tiene ", num_piezas, " piezas y ", num_robots, "
    robots" , sep="")
16
17
18 """
19 num_piezas=6
20 num_robots=2
21 """
22
23
24 A=[]
25 b=[]
26 dir_ =[]
```

```
27
28
29
30 #Posiciones de Pick i1,i2,i3
31 i1=(305,-99)
32 i2=(255,-115)
33 i3=(270,15)
34
35 #Posiciones de place j1,j2,j3
36 j1=(339,197)
37 j2=(410,167)
38 j3=(400,239)
39
40 #Posiciones de brazos s1 y s2
41
42 s1=(605,212)
43 s2=(482, -92)
44
45
46
47 A = GeneradorMatrizA(A,num_robots,num_piezas)
48
49 b = Generador_b(num_robots, num_piezas)
50
51 dir_ = Generador_dir(num_robots, num_piezas)
```


Código que genera la matriz de restricciones de números enteros binarios.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu May 19 10:53:13 2022
4
5 @author: Jorge
6 """
7
8 from numpy import *
9 from itertools import *
10
11
12
13
14
15
16
17
18 A=[]
19 b=[]
20 dir_ =[]
21
22
23 vec_aux_i = [0, 0, 1, 1]
24 vec_aux_j = [0, 1, 1, 0]
25
26 def Variables(num_piezas ,num_robots):
27     num_variables = num_robots*(num_robots*num_piezas)+ num_robots*(
28     num_piezas*num_piezas)
29     return num_variables
30
31 def restricciones(num_piezas ,num_robots):
32     num_restricciones= int(3*(num_piezas+num_robots)+2*(num_piezas*
33     num_robots)+(num_piezas-1)*((num_piezas*num_robots)/2))
34     return num_restricciones
35
36 def GeneradorListVar(num_piezas ,num_robots):
37
38     a = zeros((num_piezas),int)
39     for i in range(num_piezas):
40         a[i]=i+1
41     temp = combinations(a, 2)
42     combinaciones = list(temp)
```

```

42
43
44     lista = ''
45     for robot in range(num_robots):
46         for combinacion in combinaciones:
47             for i in range(2):
48                 lista += 'X'+str(robot+1)+'i'+str(combinacion[
vec_aux_i[i*2]])+'j'+str(combinacion[vec_aux_j[i*2]])
49                 lista += ' '
50                 lista += 'X'+str(robot+1)+'j'+str(combinacion[
vec_aux_i[2*i+1]])+'i'+str(combinacion[vec_aux_j[2*i+1]])
51                 lista += ' '
52     return lista
53
54
55 def GeneradorVar(num_piezas, num_robots):
56
57
58     variables = ''
59     for robot in range(num_robots):
60         for i in range(num_piezas):
61             variables += 'X'+str(robot+1)+'s'+str(robot+1)+'i'+str(i
+1)
62             variables += ' '
63
64         for robot in range(num_robots):
65             for i in range(num_piezas):
66                 variables += 'X'+str(robot+1)+'i'+str(i+1)+'j'+str(i+1)
67                 variables += ' '
68
69         for robot in range(num_robots):
70             for i in range(num_piezas):
71                 for j in range(num_piezas):
72                     if i == j:
73                         pass
74                     else:
75                         variables += 'X'+str(robot+1)+'j'+str(i+1)+'i'+
str(j+1)
76                         variables += ' '
77
78                 variables += 'X'+str(robot+1)+'j'+str(i+1)+'s'+str(robot
+1)
79                 variables += ' '
80     return variables
81

```

```
82 def GeneradorMatrizA(A,num_robots,num_piezas):
83
84     num_variables = Variables(num_piezas,num_robots)
85
86
87     #Restricciones
88
89     # 1. Inicio:Solo puedo ir a una pieza de la bandeja desde la
90     # posicion de cada brazo
91     a1=zeros((num_robots,num_variables),int)
92
93     for i in range(num_robots):
94         for j in range(num_piezas):
95             a1[i,j+i*num_piezas]=1
96
97     A=a1
98
99
100    # 2. Colocacion: solo un brazo puede colocar una pieza
101    a2=zeros((num_piezas,num_variables),int)
102
103    for i in range(num_piezas):
104        #for j in range(2):
105            a2[i,num_robots*num_piezas+i]=1
106            a2[i,num_robots*num_piezas+i+(num_piezas)]=1
107
108    A=vstack([A,a2])
109
110
111    # 3. Retorno brazo 1 y brazo 2 a su posicion inicial
112
113    a3=zeros((num_robots,num_variables),int)
114
115    for i in range(num_robots):
116        for j in range(num_piezas):
117            a3[i,i*(num_piezas**2)+((2*(num_piezas*num_robots)+(
118            num_piezas)+(num_piezas)*j)-1)]=1
119
120    A=vstack([A,a3])
121
122
123    # 4. Cada nodo debe ser visitado solo una vez
124
125    a4=zeros((num_piezas,num_variables),int)
```

```

125     aux=array(list(range(1,num_piezas+1)))
126     for i in range(num_piezas):
127         a4[i,i]=1
128         a4[i,i+num_piezas]=1
129     for k in range(num_robots):
130         for i in range(num_piezas):
131             cont=0
132             for j in delete(aux,i):
133                 a4[j-1,num_piezas*(i + num_piezas*k)+2*(num_piezas*
num_robots)+cont]=1
134                 cont=cont+1
135     A=vstack([A,a4])
136
137
138
139     # 5. Un brazo no puede llevar todas las piezas
140
141     a5=zeros((num_robots,num_variables),int)
142
143     for i in range(num_robots):
144         for j in range(num_piezas):
145             a5[i,j+(i*num_piezas)+num_robots*num_piezas]=1
146
147
148     A=vstack([A,a5])
149
150
151     # 6. De cada nodo del molde solo puede salir un brazo
152
153     a6=zeros((num_piezas,num_variables),int)
154
155     for i in range(num_piezas):
156         for j in range(num_piezas):
157             a6[i,(i*num_piezas)+(2*(num_piezas*num_robots))+j]=1
158             a6[i,(i*num_piezas)+(2*(num_piezas*num_robots)+(num_piezas
**2))+j]=1
159
160     A=vstack([A,a6])
161
162
163     # 7. Si un brazo llega a un nodo del molde debe poder salir de
el
164
165     a7=zeros((num_piezas*num_robots,num_variables),int)
166

```

```

167     for i in range(num_piezas*num_robots):
168         for j in range(num_piezas):
169             a7[i,(2*(num_robots*num_piezas)+(i*num_piezas))+j]=1
170             a7[i,(num_piezas*num_robots)+i]=-1
171
172     A=vstack([A,a7])
173
174
175
176     # 8. Si un brazo llega a un nodo de la bandeja debe poder salir
177     de el
178
179     a8=zeros((num_piezas*num_robots,num_variables),int)
180     aux=array(list(range(1,num_piezas+1)))
181     for i in range(num_piezas*num_robots):
182         #for j in range(num_piezas):
183             a8[i,i]=1
184             a8[i,i+num_piezas*num_robots]=-1
185     for k in range(num_robots):
186         for i in range(num_piezas):
187             cont=0
188             for j in delete(aux,i):
189                 a8[(k*num_piezas)+(j-1),num_piezas*(i + num_piezas*k
190 )+2*(num_piezas*num_robots)+cont]=1
191                 cont=cont+1
192
193     A=vstack([A,a8])
194
195     # 9. Condiciones de bucle
196
197     lista = GeneradorListVar(num_piezas,num_robots)
198     variables = GeneradorVar(num_piezas,num_robots)
199
200     a9=zeros((int(num_robots*((num_piezas-1)*((num_piezas-1)+1)/2)),
201 num_variables),int)
202     vec_aux = zeros(int(len(lista)/7),int)
203
204     for i in range(0,len(lista),7):
205         for j in range(0,len(variables),7):
206             if lista[i:i+7] == variables[j:j+7]:
207                 int(j/7)
208                 vec_aux[int(i/7)] = int(j/7)

```

```
209     con = 0
210     row = 0
211     for i in range(len(vec_aux)):
212
213         a9[row][vec_aux[i]] = 1
214         if con == 3:
215             con = 0
216             row += 1
217         else:
218             con += 1
219
220     A=vstack([A,a9])
221
222     return A
223
224 def Generador_b(num_robots,num_piezas):
225
226
227     b1=zeros((num_robots),int)
228     b3=zeros((num_robots),int)
229     b5=zeros((num_robots),int)
230     for i in range(num_robots):
231         b1[i]=1
232         b3[i]=1
233         b5[i]=num_piezas-1
234
235
236
237     b2=zeros((num_piezas),int)
238     b4=zeros((num_piezas),int)
239     b6=zeros((num_piezas),int)
240     for i in range(num_piezas):
241         b2[i]=1
242         b4[i]=1
243         b6[i]=1
244
245
246     b7=zeros((num_robots*num_piezas),int)
247     b8=zeros((num_robots*num_piezas),int)
248     for i in range(num_robots*num_piezas):
249         b7[i]=0
250         b8[i]=0
251
252
253     b9=zeros((int(num_robots*((num_piezas-1)*((num_piezas-1)+1)/2)))
```

```
,int)
254     for i in range(int(num_robots*((num_piezas-1)*((num_piezas-1)+1)
255                   /2))):
256         b9[i]=num_piezas-1
257
258     b = concatenate((b1,b2,b3,b4,b5,b6,b7,b8,b9), axis = None)
259
260     return b
261
262 def Generador_dir(num_robots,num_piezas):
263
264     #dir_1=[zeros((num_robots),int)]
265     dir_1=[]
266     dir_3=[]
267     dir_5=[]
268     for i in range(num_robots):
269         dir_1.append('=')
270         dir_3.append('=')
271         dir_5.append('<=')
272
273
274     dir_2=[]
275     dir_4=[]
276     dir_6=[]
277     for i in range(num_piezas):
278         dir_2.append('=')
279         dir_4.append('=')
280         dir_6.append('=')
281
282     dir_7=[]
283     dir_8=[]
284     for i in range(num_robots*num_piezas):
285         dir_7.append('=')
286         dir_8.append('=')
287
288     dir_9=[]
289     for i in range(int(num_robots*((num_piezas-1)*((num_piezas-1)+1)
290                   /2))):
291         dir_9.append('<=')
292
293     dir_ = dir_1 + dir_2 + dir_3 + dir_4 + dir_5 + dir_6 + dir_7 +
294     dir_8 + dir_9
```

295

```
return dir_
```