

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA**



**PROBLEMA DE PROGRAMACIÓN DE LA
PRODUCCIÓN EN LA INDUSTRIA METAL-MECÁNICA
APLICADA A LA FABRICACIÓN DE ANILLOS**

**Tesis presentada por:
XAVIER CANTÚ RODRÍGUEZ**

Como requisito parcial para obtener el grado de:

**DOCTOR EN INGENIERÍA
CON ORIENTACIÓN EN TECNOLOGÍAS DE LA INFORMACIÓN**

AGOSTO 2021

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



PROBLEMA DE PROGRAMACIÓN DE
LA PRODUCCIÓN EN LA INDUSTRIA
METAL-MECÁNICA APLICADA A LA
FABRICACIÓN DE ANILLOS

POR

XAVIER CANTÚ RODRÍGUEZ

COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE

DOCTOR EN INGENIERÍA

CON ESPECIALIDAD EN TECNOLOGÍAS DE LA INFORMACIÓN

AGOSTO 2021



Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica
Subdirección de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis "PROBLEMA DE PROGRAMACIÓN DE LA PRODUCCIÓN EN LA INDUSTRIA METAL-MECÁNICA APLICADA A LA FABRICACIÓN DE ANILLOS", realizada por el alumno Xavier Cantú Rodríguez, con número de matrícula 0953409, sea aceptada para su defensa como requisito para obtener el grado de Doctorado en Ingeniería con Orientación en Tecnologías de la Información.

El Comité de Tesis

[Signature of Dr. Vincent Andre Lionel Boyer]

Dr. Vincent Andre Lionel Boyer
Director

[Signature of Dra. Pamela Jocelyn Palomo Martínez]

Dra. Pamela Jocelyn Palomo Martínez
Revisor

[Signature of Dr. Luis Martín Torres Treviño]

Dr. Luis Martín Torres Treviño
Revisor

[Signature of Dr. Jobish Vallikavungal Devassia]

Dr. Jobish Vallikavungal Devassia
Revisor

[Signature of Dr. Romeo Sanchez Nigenda]

Dr. Romeo Sanchez Nigenda
Revisor

Vo. Bo.

[Signature of Dr. Simón Martínez-Martínez]

Dr. Simón Martínez-Martínez
Subdirector de Estudios de Posgrado



San Nicolás de los Garza, Nuevo León, Junio de 2021

ÍNDICE GENERAL

1. INTRODUCCIÓN	1
1.1. Motivación	1
1.2. Hipótesis	5
1.3. Objetivos	5
1.4. Estructura de la Tesis	6
1.5. Metodología	6
2. REVISIÓN DE LA LITERATURA	8
2.1. Rolado de anillos	8
2.2. Capacidad de máquina	8
2.3. La distribución tipo taller	11
2.4. Tiempos de retraso	12
2.5. Programación de restricciones	17
2.6. Resumen	20
3. DESCRIPCIÓN DEL PROBLEMA	21

3.1. Descripción general del proceso	25
3.2. Parámetros	28
3.3. Variables de decisión	30
3.3.1. Modelo de programación lineal entera mixta	31
3.3.2. Formulación del modelo de programación de restricciones . . .	33
4. EXPERIMENTACIÓN y RESULTADOS	36
4.1. Descripción de las instancias	36
4.2. Entorno experimental	37
4.3. Experimentos computacionales	38
4.4. Comparativos entre MILP y CP	40
4.5. Experimentos adicionales	47
5. CONCLUSIÓN	50
5.1. Trabajos futuros	51
A. Apéndice	52

ÍNDICE DE FIGURAS

3.1. Forja de anillos	21
3.2. Proceso de forja de anillos	23
3.3. Lingotes para corte de anillo	24
3.4. Laminación de anillo calentado por horno	25
3.5. Intervalos de temperaturas	26
3.6. Activación del peso de la pieza en el modelo matemático	34
4.1. Operaciones vs Tiempo de CPU CP & MILP	39
4.2. Comparativo de Tardiness: MILP & CP	39
4.3. Comparativo de Makespan: MILP & CP	41
4.4. Jobs vs Tardiness comparando ambas funciones objetivo	42
4.5. Makespan comparando ambas funciones objetivo con 1, 5, 10 jobs	43
4.6. Tardiness comparando ambas funciones objetivo con 1, 5, 10 jobs	43
4.7. Operaciones vs Tardiness Comparativo porcentual	44
4.8. Operaciones vs Tardiness CP & MILP	45
4.9. Operaciones vs Tardiness CP & MILP	45

4.10. Operaciones vs Tiempo de CPU CP & MILP	46
4.11. Operaciones vs Makespan CP & MILP	47
4.12. Tiempo segundos vs Makespan 3, 4, 5 HORNOS	48
4.13. Tiempo segundos vs Tardiness 3, 4, 5 HORNOS	48
4.14. Jobs vs Makespan comparando ambas funciones objetivo	49

ÍNDICE DE TABLAS

4.1. Instancias generadas para el SPRRO	37
4.2. Comparativo con diferentes cantidades de hornos teóricos en CP	46
A.1. Resultados de experimentación con MILP	53
A.2. Resultados de experimentación con CP	54

CAPÍTULO 1

INTRODUCCIÓN

1.1 MOTIVACIÓN

Las necesidades de la industria y la creciente tendencia de modificar las formas de producir y tomar decisiones en las empresas han motivado desarrollar el presente trabajo el cual es un caso aplicado de investigación operativa (OR por sus siglas en inglés). Esta rama de estudio está basada en la toma de decisiones para la asignación de recursos mediante modelos matemáticos que representan los factores, mediciones y restricciones que existen en un sistema productivo. Los recursos que las empresas asignan están delimitados por la cantidad de operadores, los recursos económicos así como la cantidad de infraestructura que tienen.

Algunos problemas de asignación de recursos son el ruteo de pedidos, la asignación de cargas de trabajo y la programación de la producción entre otros. En todos estos casos se aplican modelos matemáticos y heurísticas para proveer a los tomadores de decisiones herramientas para resolver problemas de asignación.

La programación de la producción es una rama específica de la investigación operativa que consiste en determinar el orden en el cual se deben realizar un con-

junto de trabajos u operaciones. La medición de una decisión está basada en una función objetivo. Por ejemplo, el tiempo total de manufactura (*makespan*) de toda una secuenciación, el retraso total (*tardiness*) y la precocidad del cumplimiento (*earliness*) son algunas funciones objetivo para este tipo de problemas. Estos criterios de optimización se deciden en base a las necesidades de cada compañía o servicio particular y su complejidad aumenta con el tamaño del problema a resolver. En el presente estudio es la cantidad de trabajos.

El problema de programación flexible del taller de trabajo (FJSSP por sus siglas en inglés) es una generalización del problema clásico de programación de distribución tipo taller que permite procesar operaciones en una máquina de un conjunto de máquinas alternativas. El problema clásico de asignación en distribución tipo taller (JSSP por sus siglas en inglés) ha sido abordado mediante programación lineal entera mixta (MILP por sus siglas en inglés).

Algunas premisas de este problema es que se tiene que asignar trabajos (*jobs*), los cuales están formados a su vez por un conjunto de operaciones. Las operaciones de cada job pueden seguir secuencias de proceso en máquinas diferentes de acuerdo al job que se está procesando. Parte de la programación incluye el definir en qué máquina se procesará una determinada operación de un trabajo. El FJSSP es un problema NP-hard que consta de dos subproblemas, los cuales son la asignación y los problemas de programación según Demir y İşleyen (2013).

Estos modelos incluyen posición de secuencia variable propuestos inicialmente por Wagner (1959), en donde se presenta un modelo que especifica la posición relativa de un trabajo a todos los demás trabajos procesados. Los modelos de precedencia variable, planteados originalmente por Manne (1960) denotan la secuencia de operaciones asignadas a la misma máquina. Los modelos basados en tiempo creados por

Bowman (1959) consideran el horizonte de planificación indexado en tiempo discreto.

Algunas restricciones de los problemas planteados son las precedencias de las operaciones, los tiempos de preparación (*setup*), la capacidad de máquina, los tiempos máximos de retención de piezas, la complejidad computacional de acuerdo al tamaño de las instancias por poner algunos ejemplos. Los métodos exactos aplicados a modelos enteros mixtos no pueden proporcionar una solución óptima en tiempos razonables de procesamiento. Por otro lado, el uso de heurísticas permite encontrar soluciones factibles al FJSSP, no obstante sin garantía de encontrar el óptimo global de la función objetivo.

Un trabajo puede ser procesado en una máquina bajo dos condiciones: a) si está disponible y b) la operación inmediata anterior del trabajo ha sido realizada. Los tiempos de preparación son incluidos entre las restricciones del modelo clásico y este problema se vuelve flexible cuando se tiene disponibilidad de más de una máquina para hacer una determinada operación.

Los problemas de investigación operativa miden su complejidad en función del método más eficiente actualmente desarrollado para resolverlo. El problema del presente trabajo es clasificado como una variante del FJSSP. Este tipo de modelo de distribución de manufactura asume un conjunto de trabajos que serán procesados en un conjunto de máquinas con la premisa de que solo pueden ser operados una sola vez por máquina con diferentes secuenciacines de máquinas por cada trabajo (*job*). Las dependencias de las secuencias están basadas en el job a procesar y la disponibilidad de las máquinas.

Un modelo empleado para los problemas clásicos de FJSSP es el desarrollado por Manne (1960), el cual está basado en precedencias mediante un planteamiento

de MILP. Actualmente los modelos MILP para la solución de problemas de programación de la producción no pueden ser empleados por los usuarios de toma de decisiones en la industria debido al tiempo computacional que generan cuando se aplican métodos exactos.

Por otra parte, algunas meta-heurísticas aplicadas al MILP son los algoritmos genéticos (GA, por sus siglas en inglés) en los trabajos de Li y Gao (2016) y Sukkerd y Wuttiornpun (2016), el Taboo Search en los trabajos de Shahvari y Logendran (2017) y Liu y Kozan (2017), la inteligencia de enjambre de partículas presentados por Singh y Mahapatra (2016) y Nouri *et al.* (2018) con excelentes resultados.

La presente tesis aborda un nuevo problema de programación de la producción con restricciones de capacidad de máquina basadas en el peso que puede contener, así como tiempos máximos de espera en proceso y fuera de proceso. Este caso de estudio pertenece al ramo metal-mecánico de los aceros de super-aleaciones. Las super-aleaciones requieren tratamientos térmicos con intervalos de temperatura específicos por tipo de material, por consiguiente una programación de la producción que no considere estos factores tiene un efecto económico en cantidades millonarias para la empresa.

La programación de restricciones (CP, por sus siglas en inglés) se ha vuelto un paradigma de la computación que es empleado para problemas de optimización combinatoria particularmente complejos, especialmente en las áreas de planificación y programación de tareas. Este campo de estudio se relaciona con la programación lógica y la investigación operativa. La *Association for Computing Machinery* (ACM) lo ha considerado como una dirección pilar de la investigación en la computación y está basada en la aplicación de conjuntos de restricciones, las cuales deben ser satisfechas para encontrar una solución. La reducción del universo del discurso o dominio

de las variables reduce el área de búsqueda en algoritmos y permite encontrar soluciones con una mayor velocidad.

El presente trabajo propone un nuevo problema de programación de la producción con dos tipos de modelo matemático para una variante del FJSSP en el proceso de forja de anillos para el mercado aeroespacial y de energía eólica. Los modelos propuestos para la solución son MILP y CP.

1.2 HIPÓTESIS

La secuenciación de las operaciones de producción para la industria de los anillos forjados en el mercado aeroespacial y de energía eólica puede ser resuelto de manera mas eficiente en términos de tiempo computacional mediante un enfoque de CP que mediante las técnicas clásicas de programación lineal.

1.3 OBJETIVOS

El objetivo general de esta investigación es proponer un modelo matemático para una variante del problema de FJSSP con restricciones de capacidad y tiempos de espera de proceso, mismo que está conformado por los siguientes objetivos específicos:

1. Determinar las variables de problema de estudio.
2. Plantear un modelo matemático de CP para el caso de estudio.

3. Hacer un comparativo con los problemas clásicos de MILP.
4. Generar instancias del problema de estudio y hacer las comparaciones pertinentes.
5. Determinar la calidad de las soluciones.
6. Validar la modelación matemática con diferentes parámetros.

1.4 ESTRUCTURA DE LA TESIS

El presente trabajo de investigación tiene la siguiente estructura:

- 1 Una introducción que describe las motivaciones del problema de investigación.
- 2 Una revisión de la literatura que incluye una compilación de las principales palabras clave para poder exponer la contribución de este trabajo.
- 3 La descripción del problema en las cuales se define el proceso de forja, la definición de variables así como el modelo matemático del presente problema.
- 4 La metodología empleada en este proyecto, la cual incluye el desarrollo de modelo clásico, desarrollo del modelo de CP.
- 5 Los resultados de la experimentación con ambos modelos.
- 6 La conclusión y trabajos futuros de esta investigación.

1.5 METODOLOGÍA

Este trabajo de investigación al proponer un modelo matemático para una variante del problema de FJSSP, requiere comparar el modelo contra el modelo clásico

de solución. El desarrollo del modelo propuesto implicó la investigación de datos de la industria para generar las temperaturas teóricas de los productos y de esta manera tener una representación aproximada a la realidad de la empresa.

Revisión de la literatura: Se realizó una revisión de la literatura basado en las características del problema en campo para comparar los desarrollos de anteriores estudios y verificar cuál es la línea que se ha desarrollado, así como los métodos de solución que se han empleado para resolver problemas similares.

Desarrollo de modelo de programación lineal entero mixto: Desarrollar un modelo clásico de programación lineal entera mixta basado en el modelo de Manne.

Desarrollo de modelo de programación de restricciones: Se desarrollará un modelo de CP para el problema de programación de la producción propuesto con el fin de proporcionar un comparativo entre los modelos clásicos de programación lineal entera mixta.

Comparación de los modelos: Los modelos serán comparados mediante el solver en la aplicación CPLEX.

Comparativo de funciones objetivo: Las funciones objetivo que se aplicarán para cada uno de los modelos serán el tiempo total de manufactura denominado makespan y la tardanza total de los trabajos denominada tardiness. Estas funciones objetivo son comunes en la literatura para los problemas flexibles de programación de la producción y de relevancia para la toma de decisiones en la industria.

CAPÍTULO 2

REVISIÓN DE LA LITERATURA

2.1 ROLADO DE ANILLOS

El proceso de rolado de anillos ha basado su investigación principalmente en el desarrollo de materiales; no se tienen muchos antecedentes en la revisión de la literatura de problemas específicos para este tipo de manufactura en cuanto a su programación de la producción. Los investigadores Baykasoğlu y Ozsoydan (2018) se han basado en la programación de los hornos mediante algoritmos constructivos para máquinas en paralelo mediante optimización dinámica. La evolución del diseño de equipos de laminación de anillos ha sido estudiada por Allwood *et al.* (2005); Hua *et al.* (2017). Incluyeron una discusión detallada del diseño y fabricación de pre-formas.

2.2 CAPACIDAD DE MÁQUINA

El presente trabajo define la capacidad de máquina como la cantidad máxima en kilogramos que pueden contener los hornos, ya que el resto de las máquinas no pueden procesar mas de una pieza a la vez. La capacidad máxima se vuelve una

restricción del problema propuesto. El proceso de los hornos es clave en la fabricación de los anillos aleados, ya que permite la elongación del material así como proporcionar las propiedades mecánicas a los materiales. La insuficiencia o exceso de temperatura, al igual que el tiempo de procesamiento influyen en la dureza del material.

Los investigadores Cui *et al.* (2014) han abordado la definición de capacidad de máquina desde diferentes perspectivas. La definición propuesta en su trabajo es la tasa de fallo como una forma de disponibilidad del equipo, en donde los paros por mantenimiento ocasionarían un tiempo de espera antes de estar en operación. La incertidumbre por las fallas ha sido estudiada con el makespan y tardiness ponderado como funciones objetivo por Shahvari y Logendran (2018) mediante un algoritmo de tres fases.

Mokhtari y Dadgar (2015) estudiaron la tasa de fallas y disponibilidad de máquina mediante MILP para el problema de los trabajos tardíos por paros no programados. Se realizaron las simulaciones de descompostura mediante el método Montecarlo. Las fallas y situaciones inesperadas también fueron estudiadas por Lin *et al.* (2016a) como un factor de impacto en la cantidad de piezas que puede producir una máquina con modelos de programación entera mixta y branch and bound.

Por otra parte, Khoukhi *et al.* (2017) estudiaron las restricciones de indisponibilidad de máquina por mantenimiento preventivo. Éstas se abordaron mediante un modelo de programación no lineal entera mixta (MINLP) y la aplicación de algoritmos de colonia de hormigas (ACO). Chen *et al.* (2016) minimizaron la carga de trabajo, el tiempo de espera y la tardanza aplicando algoritmos genéticos multi-propósito (MOGA). Esta disponibilidad puede ser estocástica, Lin *et al.* (2016b) proponen casos donde el número de máquinas disponibles en cada departamento

representa un nivel en la metodología de branch and bound.

Otra definición de capacidad de máquina es planteada por Deep y Singh (2015). Su propuesta está basada en la reconfiguración y flexibilidad de acomodo, basados en el flujo de materiales.

Algunos trabajos han estudiaron la complejidad del problema del flujo en sistemas flexibles mediante dos estrategias: La reconfiguración mediante heurísticas de vibración amortiguada por parte de Mehdizadeh *et al.* (2016) y su aplicación en vehículos para la agricultura en los estudios de Bochtis *et al.* (2014).

La definición de capacidad es estudiada también por Leinberger *et al.* (1999), en la cual se considera la cantidad de producto que puede contener un empaque. Garillot *et al.* (2009) nos presentan una generalización del problema clásico de embalaje de contenedores unidimensionales donde la capacidad del contenedor y los tamaños de los artículos se representan mediante vectores d -dimensionales.

La cantidad de tareas que una maquina puede realizar es la definición que Raduly-Baka y Nevalainen (2015) abordan para la minimización de los tiempos de preparación mediante diseño experimental y políticas de prioridad. En este problema Chen *et al.* (2017) incluyeron las máquinas que tienen múltiple herramental.

El comportamiento dinámico de los sistemas en los problemas reales de planta no pueden ser representados completamente en un modelo matemático por diversos factores como retrasos inesperados, colas, averías. Lee y Kim (2002) resolvieron el problema de disponibilidad de máquinas proponiendo un enfoque híbrido que combina un modelo analítico y una simulación.

El concepto de capacidad de maquina visto como la cantidad de materia prima que puede cargar una maquina no ha sido abordado en la literatura revisada hasta la actualidad por la particularidad del problema, lo cual representa una oportunidad de aporte académico por parte del presente trabajo.

2.3 LA DISTRIBUCIÓN TIPO TALLER

La distribución de taller mejor conocida como *job shop* en la literatura consiste en una agrupación que distribuye las máquinas existentes por proceso. El problema de programación de la producción en este sistema implica que los trabajos a realizar no llevan necesariamente la misma secuencia de operaciones en las mismas máquinas.

Este caso clásico corresponde a un problema de planificación de tareas y se vuelve un problema de optimización combinatoria. Se puede plantear como la planificación de N trabajos (*jobs*) en un conjunto de M unidades físicas o máquinas. Cada trabajo consta de un conjunto de operaciones O que deben ser ejecutadas en forma secuencial, pero que en cada trabajo pueden variar en la asignación de máquinas.

Las restricciones del problema clásico pueden resumirse de la siguiente manera:

1. Una tarea no puede ser procesada por una misma máquina dos veces.
2. No hay restricciones de precedencia entre operaciones de distintas tareas.
3. Las operaciones no pueden ser interrumpidas.
4. Cada máquina tiene la capacidad de realizar una sola tarea.

5. No se especifican fecha de tarea lista a ser procesada (*release times*) como tampoco la fecha de entrega (*due date*).

El problema flexible es denominado job shop scheduling problem (JSSP) con flexibilidad de máquinas por Nouri *et al.* (2015).

González *et al.* (2015) planteó una estrategia de búsqueda de dispersión ha estableciendo nuevos límites superiores para varias instancias conocidas. Una programación entera mixta con dosificación en máquinas de procesamiento y lotes en paralelo es planteada por Ham (2017) y resuelta mediante una estrategia de branch and bound generando cronogramas efectivos para la programación.

Las líneas de flujo flexibles con tiempos de configuración dependientes de la secuencia fue abordada por Kurz y Askin (2003) mediante tres heurísticas (incluyendo Método de Johnson) donde se aplican heurísticas greedy para comparar las reglas de secuenciación. El problema de programación del taller de flujo flexible de dos etapas sin espera (NWTSSFSSP) fue propuesto por Rabiee *et al.* (2014), considerando máquinas paralelas no relacionadas y empleando recocido simulado y búsquedas de vecindario variable (VNS).

2.4 TIEMPOS DE RETRASO

El tiempo de retraso (*lagtime*) es el tiempo entre dos eventos relacionados y otra restricción planteada en este trabajo de investigación. Algunas causas de estos retrasos son la disponibilidad de transporte, que fue estudiado por Afsar *et al.* (2016). Estos planteamientos fueron validados mediante los modelos de Laurence y grafos disyuntivos induciendo un retraso del tiempo entre operaciones y el problema

del enrutamiento. Gschwind (2015) empleo el labeling algorithm para resolver los problemas que también generan retrasos por disponibilidad de vehículos que satisfagan el emparejamiento y la precedencia, las capacidades y las ventanas de tiempo.

Los tiempos de retraso fueron simulados por Francis (2016) para generar retrasos máximos y mínimos en la entrega de pedidos como otro tipo de lag time. Fueron resueltos mediante heurísticas estocásticas como scatter search y tabu search. Las restricciones de ventana de tiempo fueron modeladas con MIP usando CPLEX 12.5 por Zhang *et al.* (2015a) en problemas de programación de taller con retrasos, considerando los horarios factibles e infactibles en la ejecución con procedimientos de vecindarios. Zhang *et al.* (2015b) propone como definición de ventana de tiempo el tiempo de transferencia de un trabajo de una máquina a otra.

Un problema de taller de flujo con permutaciones y rezagos de tiempo entre operaciones fue estudiado por Fondrevelle *et al.* (2006) para proporcionar evidencia de la complejidad del problema para dos máquinas. Kreter *et al.* (2016) proponen un problema de flujo de taller para minimizar los tiempos de retraso mediante algoritmos greedy.

Algunas aplicaciones en la asignación de recursos médicos han sido estudiadas por Lee y Yih (2014), quienes proponen un problema aplicado para la programación de cirugías mediante algoritmos genéticos en la primera etapa y una decisión heurística en la segunda etapa. Los investigadores Vanhoucke y Coelho (2016) y Min y Yih (2010) presentan un caso de estudio donde las múltiples cirugías generan restricciones de recursos. Algunos pacientes tienen que esperar para poder tener un quirófano aunado a la incertidumbre en los tiempos de servicio.

Hong *et al.* (2019) estudiaron los problemas con tiempos de retraso mediante

relajación lagrangiana modelados con MILP para tiempos de espera que no pueden ser excedidos en un intervalo dado empleando programación dinámica. Detienne *et al.* (2016) propusieron otras estrategias con lagrangianos para el problema de intervalos de tiempo que no pueden ser, mismos que han sido resueltos mediante branch and bound en casos de flow shop con dos máquinas simples.

Kreter *et al.* (2017) realizan la propuesta del problema de resource-constrained project scheduling problem (RCPSP) donde se aplican los tiempos de retraso a proyectos. La literatura revisada hasta el momento no menciona problemas de tiempos de retraso en donde la integridad del producto se vea comprometida en caso de excedentes, lo cual aporta al problema de investigación un caso de estudio interesante.

Amrouche y Boudhar (2016) y Amrouche *et al.* (2017) demostraron que cuando hay retrasos de tiempos los problemas con dos maquinas secuenciadas se vuelven NP-Hard y posteriormente que en casos especiales de reingreso y retraso de tiempo exacto el problema también se vuelve también NP-Hard.

Artigues *et al.* (2011) aplicaron un algoritmo de Branch and Bound, donde se encontraron mejores soluciones que las instancias tomadas en los problemas que involucran retrasos de tiempo ajustados para el problema de job shop clásico. Para otras configuraciones dependientes de la secuencia Defersha y Chen (2012) aplicaron algoritmos genéticos con restricciones de fechas de inicio de operación de la máquina.

Botta-Genoulaz (2000) presentaron seis nuevas heurísticas para resolver la programación de O operaciones con M máquinas paralelas idénticas cuando los trabajos están sujetos a restricciones de precedencia, varios retrasos y fechas de vencimiento. Una estrategia para resolver este problema son los algoritmos meméticos empleados por Caumond *et al.* (2008) con problemas del tipo $(JS|t_{i,s_i})$ de acuerdo con la no-

tación de Graham *et al.* (1979) la cual clasifica los problemas de scheduling en una triplete de parámetros definidos a continuación:

α Describe el ambiente de máquina con una única entrada.

β Describe características del proceso y restricciones.

γ Describe la función objetivo a resolver

Las configuraciones dependientes de la secuencia y tiempo de inicio de operación de la máquina fueron estudiadas por Defersha y Chen (2012) para el problema del job shop en donde se aplicaron algoritmos genéticos. Se abordó el problema de programación de Job-Shop y Flow-Shop con dos máquinas, no más de dos operaciones por trabajo, y tiempos de retraso mediante Tabu Search por Dell'Amico (1996).

Dhouib *et al.* (2013) realizan una formulación de programación matemática de enteros mixtos para resolver el problema de la programación del taller de flujo de permutación con retardos de tiempo mínimos y máximos, también aplicándolo a problemas de flow shop.

Un algoritmo genético (GA), y tres heurísticas (Johnson (1954), Palmer (1965) y SPTCH) son empleadas por Farahmand-Mehr *et al.* (2014) para resolver el problema de programación del taller de flujo híbrido generando soluciones casi óptimas en un tiempo computacional corto para problemas en dos diferentes tamaños denominados pequeños y mediano-grandes .

Fondrevelle *et al.* (2009) abordó el problema de programación del job shop clásico con permutación de la máquina m , donde se definen los retrasos de tiempo exactos entre las operaciones consecutivas de cada trabajo mediante tres esquemas

de ramificación (método de Ignall y Schrage (1965) y dos variaciones del método de Potts Potts (1980)) con el objetivo de minimizar la tardanza.

La planificación de los recursos humanos y la programación de la producción son problemas que han sido estudiados por Frihat *et al.* (2014) y resueltos mediante programación lineal mixta y programación de restricciones, demostrando por experimentación numérica la ventaja de la descomposición y generación de cortes en términos de tiempo de resolución.

Hamdi y Loukil (2015a) explora el problema de programación del taller de flujo de permutación con demoras de tiempo mínimas y máximas aplicando relajación lagrangiana y un algoritmo de subgradiente . Posteriormente se abordó el problema mediante el algoritmo de Moore a descomposición de Benders Hamdi y Loukil (2015b) y la propuesta de nuevas acotaciones para problemas de tamaño mayor Hamdi y Loukil (2017).

Los investigadores Harbaoui *et al.* (2016) estudiaron una configuración dependiente de la secuencia con retrasos de tiempo aplicado a una compañía de producción de pasta y resuelto mediante un algoritmo genético. Se demostró con este estudio que el algoritmo propuesto puede producir resultados que están muy cerca de los horarios óptimos si se conocen o muy cerca de los límites inferiores.

Karoui *et al.* (2010) realizaron un comparativo entre el job shop clásico y el flow shop con retrasos de tiempo mínimos y máximos resuelto mediante adaptaciones de la heurística climbing discrepancy search (CDS) para encontrar las principales discrepancias en la complejidad de estos problemas .

El problema de programación de taller de flujo con retrasos de tiempo fue pro-

puesto por los investigadores Wang *et al.* (2014) y Ning *et al.* (2017) con el objetivo de minimizar el intervalo de espera y satisfacer las limitaciones de tiempo es resuelto. Proponen un algoritmo greedy encontrando las mismas soluciones que los algoritmos genéticos en menos del 1 % del tiempo computacional.

Un problema de programación de una sola máquina con retrasos de tiempo mínimos y máximos es propuesto por Sheen y Liao (2007). Se minimizó el tiempo de espera mediante un algoritmo de branch and bound y resuelto en un entorno de fabricación en el que el siguiente trabajo debe realizarse dentro de un intervalo de tiempo específico después de la finalización del trabajo inmediatamente anterior.

Ye *et al.* (2017) validaron cinco heurísticas de búsqueda de vecindario eficientes, aplicadas al problema de la programación de taller de flujo utilizando la heurística del problema de flujo de permutación (PFSP).

La minimización de las variaciones de tiempo en taller de flujo son resueltas mediante algoritmos de aproximación por Yu *et al.* (2017) en ambientes de dos máquinas con restricciones de tiempo de espera y omisión de trabajos.

Zhang y van de Velde (2015) introdujeron un problema de programación de taller de flujo por intervalos que incluye tiempos de inicio y finalización fijos para cada trabajo y un tiempo de transporte determinado.

2.5 PROGRAMACIÓN DE RESTRICCIONES

La programación de restricciones o constraint programming (CP) es un paradigma computacional que es empleado para resolver problemas combinatorios y se

apoya en técnicas de inteligencia artificial, ciencias de la computación e investigación operativa. En este paradigma se definen las restricciones en las cuales se generan soluciones factibles al problema a resolver. Las restricciones difieren de la programación imperativa en que no especifican un paso o secuencia de pasos a ejecutar, sino más bien las propiedades de una solución a encontrar; aunado a esto los usuarios también deben especificar un método para resolver estas restricciones.

Los modelos de *constraint programming* han sido empleados recientemente en la manufactura como un nuevo paradigma para la solución de problemas de optimización combinatoria. En años recientes los investigadores han empleado el CP para máquinas que producen lotes en paralelo y algunas restricciones tales como el mantenimiento preventivo y tiempos de preparación dependientes de la secuencia de procesamiento. A estos modelos se han aplicado algoritmos bio-inspirados como colonias de hormigas (ACO), algoritmos del lobo gris (*gray wolf algorithm*), entre otros; y se han validado con los modelos tradicionales de MILP. Los principales enfoques han sido para mantenimientos preventivos y también se han aplicado modelos de taller de flujo flexible con restricciones de ventanas de tiempo y restricciones de capacidad.

El CP se ha aplicado ampliamente para resolver problemas complejos de optimización. En la última década, el CP se ha utilizado de manera eficiente en combinación con algunos algoritmos de optimización, se ha demostrado que es más rápido, eficiente y fácil de usar para resolver problemas de programación de la producción. Uno de las principales ventajas de utilizar la programación de restricciones es que, uno puede hacer uso de lógica, restricciones y expresiones en lugar de restricciones de programación complejas.

El problema clásico de FJSSP ha sido modelado mediante programación lineal entera mixta. Los investigadores Ham y Cakici (2016) recientemente han probado

modelos de CP contra los de MILP para resolver problemas con máquinas paralelas en lotes. Otros investigadores como Sahraeian y Namakshenas (2015) aplican grafos disyuntivos en ILog Cplex con parámetros como la eficiencia del tiempo.

Ku y Beck (2016) realizaron comparativos entre los diferentes solvers como CPLEX, GUROBI y SCIP; y fueron validados con cuatro enfoques de MILP y comparados contra el CP en el problema clásico de JSSP.

Dos modelos de CP fueron planteados por Samarghandi y Behroozi (2017) y Nagata y Ono (2018) para el problema clásico con enfoque de minimización de makespan, aplicando una política sin tiempos de espera y combinando meta-heurísticas bio-inspiradas como la inteligencia de partículas en el trabajo de Gregor *et al.* (2018).

Ribeiro *et al.* (2015) realizaron una revisión exhaustiva mediante 17 instancias del modelo clásico del JSSP y dos modelos de CP contra uno de MILP buscando mejorar el sistema en términos de rendimiento de optimización, tiempo de procesamiento computacional, mayor utilización de recursos e inventario mínimo de trabajo en proceso. Los resultados indican que con el modelo MILP se resolvieron el 60 % de las instancias mientras que el modelo aplicado de CP-1 55 % y CP-2 88 % de las instancias

El CP ha sido aplicado en la programación de proyectos en los trabajos de Kreter *et al.* (2017) y Schnabel *et al.* (2018) y a la agenda de horarios laborales en los trabajos de Chapados *et al.* (2014) y Latorre-Núñez *et al.* (2016). En los modelos de CP recopilados en la revisión de la literatura no se ha encontrado alguno que combine restricciones de tiempo de espera y la capacidad de máquina como parte de las restricciones del sistema planteado.

2.6 RESUMEN

El problema del JSSP ha sido resuelto mediante CP abordando la tardanza ponderada por Sahraeian y Namakshenas (2015) y Ribeiro *et al.* (2015). El CP fue validado mediante varios métodos de solución por Gregor *et al.* (2018). La capacidad de máquinas está definida como la capacidad de carga de vehículos en el trabajo de Afsar *et al.* (2016). También se aplicó un método de scatter search introduciendo tiempos de espera y tiempos de preparación pero con un modelo diferente en el trabajo de González *et al.* (2015).

Shen *et al.* (2018) proponen un FJSSP mediante los criterios de preparación dependientes de secuencias. La fabricación de lotes transmitidos por Novas (2019), y lotes paralelos por Ham y Cakici (2016); algunas aplicaciones tales como impresoras online han sido estudiadas por Lunardi *et al.* (2021).

La disponibilidad de máquinas debido a mantenimientos programados en ambientes de FJSSP han sido propuestos por Khoukhi *et al.* (2017) y Mokhtari y Dadgar (2015) y la disponibilidad de horarios para operación de las máquinas por Zhang *et al.* (2015a) son aportes a la definición de capacidad de máquinas en FJSSP. Otra aplicación del problema flexible a la asignación de recursos de hospital fue propuesto por Lee y Yih (2014) considerando tiempos de retraso y conjuntos difusos, empleando un algoritmo genético para resolver el problema.

De nuestro mejor conocimiento, el aporte académico de este trabajo de investigación es un nuevo problema basado en una variante del FJSSP aplicado en la industria metal-mecánica y el desarrollo de un modelo de CP y MILP para su solución.

CAPÍTULO 3

DESCRIPCIÓN DEL PROBLEMA

El presente estudio está basado en la programación de la producción en la industria de forja de anillos de super-aleaciones para el ramo aeroespacial y de energía eólica. El proceso de forja de anillos mediante el proceso de rolado tiene diversas etapas que incluyen corte, prensado, forja y calentamiento por horno, así como los respectivos traslados del anillo. Cada proceso es realizado por una máquina que tiene cierta disponibilidad de tiempo y capacidad en kilogramos para realizar el proceso. Los anillos de forja metálica se fabrican mediante un proceso que incluye transformaciones mecánicas del material, dichas operaciones incluyen pre-formar un anillo y someterlo a procesos de prensado, horneado, y forjado mediante tensión como lo muestra la figura 3.1.



FIGURA 3.1: Forja de anillos

Los anillos de forja son empleados en la industria aeroespacial, de construcción y minería, de maquinaria en general, de petróleo y gas, de generación de energía y energía eólica. Este proceso de manufactura supera en calidad a otros tipos de forjado de metales como el acero vaciado, ya que el rolado de materiales en caliente evita la generación porosidades en el material y por consiguiente proporciona alta resistencia y ductilidad.

El problema de programación de la producción es definir qué se va a producir y en qué secuencia, cuándo y con qué recursos. La programación de la producción es una actividad indispensable en las plantas de manufactura que, aplicada en forma adecuada disminuye tiempos de ocio y aumenta el cumplimiento de entrega de los pedidos o trabajos.

Basado en la clasificación de los problemas de programación de la producción, este problema es una variante de la distribución de tipo taller flexible (flexible job shop scheduling problem, FJSSP) con variantes específicas presentado como programación de la producción para anillos de forja. Cada paso del proceso de laminado de anillos puede verse como una operación de un trabajo (pieza de trabajo) en el contexto de los problemas de programación flexible del taller de trabajo.

Para el caso de la manufactura de anillos aleados cada una de las piezas a procesar es considerada un trabajo o *job*. De esta forma, cada uno de los trabajos tiene diversas operaciones en máquinas como la cortadora, prensa, horno y roladora. Cada job está compuesto por al menos cinco operaciones (una operación por cada paso del proceso de laminación del anillo). Como cada anillo tiene requisitos físicos específicos y un rango de temperatura a manipular, el número de veces que las dos últimas máquinas realizar una operación en la pieza de trabajo depende de los re-

quisitos específicos del producto final.

Por lo tanto, dado un conjunto de trabajos (anillos requeridos por el cliente) el problema de optimización consiste en definir una secuencia de producción que incremente la utilización/capacidad de los hornos mediante la minimización del makespan como función objetivo, considerando también validar el modelo mediante el tardiness para ver su desempeño. El proceso completo de forja de anillos se muestra en la figura 3.2

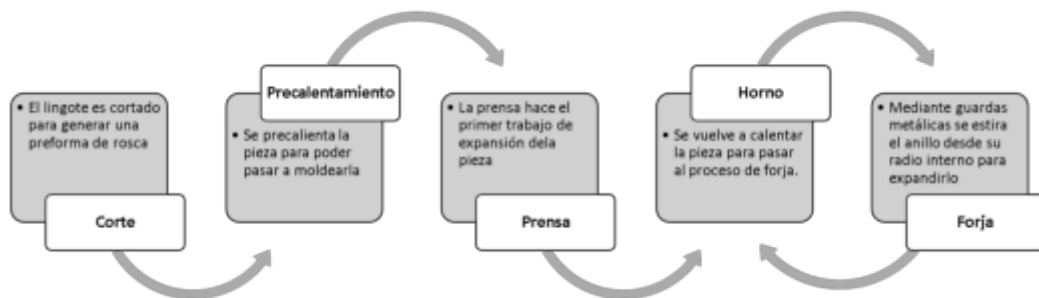


FIGURA 3.2: Proceso de forja de anillos

La cantidad de veces que un anillo tiene que re-iterar el proceso de horno-forja no muestran una distribución probabilística normal, lo cual complica el proceso de secuenciación ya que no se sabe con certeza cuantas veces requerirá entrar al horno una determinada pieza, no obstante; se contemplan los históricos de las piezas para poder tener una estimación al respecto.

El horno puede calentar varias piezas de trabajo simultáneamente, mientras que las máquinas laminadoras pueden trabajar con una pieza a la vez. Como se puede ver, el proceso se compone de cinco pasos principales, los cuales son corte, precalentado, prensado, calentamiento y laminado. Los últimos dos pasos del proceso se aplican repetidamente hasta cumplir con los requisitos del producto.

El primer paso de este proceso es prensar una pieza cilíndrica de metal como la mostrada en la figura 3.3 proporcionada por Sidenor (22 de Agosto de 2013) donde la pieza es prensada y perforada hasta obtener la pre-forma de una rosca, esto forma la base del anillo que será elongado. Una vez que el anillo tiene forma de rosca, la pieza de trabajo se envía al horno si y solo si no se excede su capacidad en kilogramos, de lo contrario, espera afuera. La pieza de trabajo se pre-calienta (alrededor de 2 horas o más, dependiendo del tipo de metal) en un horno que mantenga la temperatura adecuada para evitar el choque térmico y afectar las propiedades mecánicas de la pieza.



FIGURA 3.3: Lingotes para corte de anillo

Una vez que la pieza se encuentra en el rango de temperatura requerido para su procesamiento se envía a la máquina de laminación. Este proceso consiste en estirar la pieza calentada mediante esfuerzo mecánico como lo muestra la figura 3.4 encontrada en la página Euskali (22 de Agosto de 2013).



FIGURA 3.4: Laminación de anillo calentado por horno

El material, una vez que ha adquirido la forma de rosca, es calentado para ser deformado mecánicamente, pero una vez que está demasiado frío para poder ser estirado tiene que ser devuelto al horno. Este último proceso puede incluir herramientas de pre-forma que le darán a la parte interna, externa o ambas del anillo algún troquel específico para el mercado en que se aplicará. Los tipos de materiales son para mercados de alta especificidad, de manera que un defecto a lo largo del proceso es costoso. Las herramientas de pre-forma sirven para el mercado específico.

3.1 DESCRIPCIÓN GENERAL DEL PROCESO

Algunas variables como las características del material, los tiempos de calentamiento y espera en el proceso de horneado para las piezas forjadas y las piezas pre-moldeadas afectan las decisiones de programación; ya que el material puede perder sus propiedades por exceso de tiempo en horneado. Los problemas de programación de la producción son representados mediante modelos matemáticos que pueden ser resueltos a través de métodos exactos o heurísticas. Estos modelos pueden representar total o parcialmente la realidad del proceso para buscar la optimización de la función objetivo que se pretende modelar.

Cada pedido posee diferentes tipos de materiales y cada tipo de material una cantidad diversa de anillos a producir. La cantidad de operaciones del horno están

basadas en los tipos de material, el tiempo de horneado y las temperaturas a manejar en cada material como lo muestra la figura 3.5.

Los intervalos de temperatura para algunos de los principales materiales de la empresa se pueden ver en la figura 3.5. Los materiales de forja son calentados en temperaturas desde 955 a 1170 grados centígrados. Existen algunos que empatan en su temperatura de procesamiento, pero otros definitivamente no pueden ser procesados a la par generando demoras en el proceso. Estos hornos registran una usabilidad de menos del 20% en la práctica ya que las combinaciones de pedidos que son programados no permiten que las piezas regresen oportunamente a los hornos por no tener el intervalo de temperatura adecuado para el material debido a otro proceso. El proceso de horneado tiene el principal conflicto en la programación de la producción, ya que no tiene un aprovechamiento adecuado de la utilización.

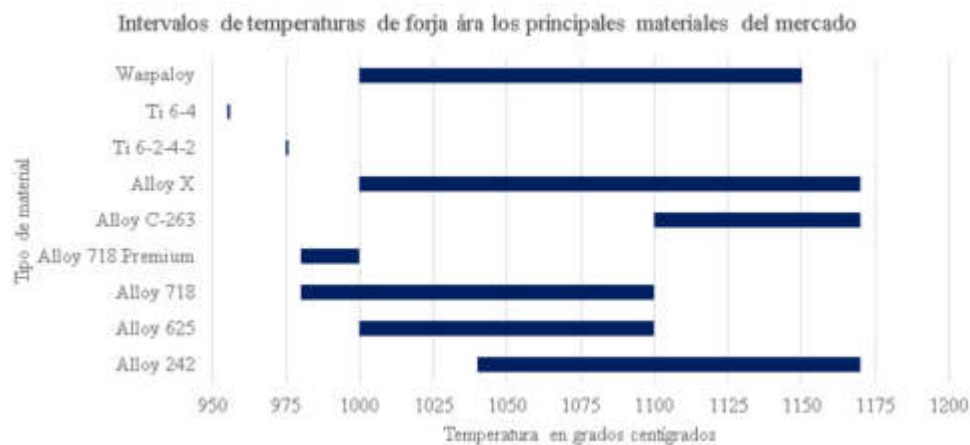


FIGURA 3.5: Intervalos de temperaturas

El horno tiene la restricción de la cantidad de piezas que se pueden calentar y es el cuello de botella del proceso. Las piezas tienen un tiempo de preparación dependiente debido a los cambios que tiene que realizar el horno en su temperatura para calentar las piezas. Se tienen tres hornos para elegir en dónde procesar una determinada pieza. Las capacidades de las máquinas son de una sola pieza a excepción

del horno el cual puede procesar mas de una pieza. Esto depende del tamaño, peso y temperatura requerida para el proceso, así como de la cantidad a fabricar. En este mercado se agrupan los principales productos en 8 tipos diferentes, pero la mezcla de productos es de mas de 50 diferentes y por consiguiente la secuenciación tiene restricciones particulares a considerar.

Cada máquina está dedicada a un tipo de operación de trabajo, excepto los hornos, los cuales son utilizados para el precalentamiento y calentamiento de las piezas. Estas dos acciones son operaciones diferentes.

Se considera que que la capacidad de los hornos depende de tres factores:

- Peso máximo en kilogramos
- Disponibilidad de espacio
- Rango de temperatura compatible

Cuando la pieza en el proceso de laminación ha perdido temperatura ya no puede ser elongada y tiene que regresar al proceso de calentamiento en el horno. Las restricciones de capacidad mencionadas con anterioridad generan tiempos de espera para el ingreso de las piezas.

En la empresa se ha contemplado la obtención de tecnología. La justificación de esta decisión está basada en la utilización de las máquinas, pero siendo su programación realizada de manera empírica (hecha por los programadores de la producción) no se tiene un método de programación que genere mejores soluciones que las que pueda tomar un ser humano desde su experiencia en el proceso. La empresa no puede justificar la compra de más hornos para su proceso ya que la utilización del horno

no justifica económicamente la compra de nueva tecnología.

El problema resultante se denomina problema de programación de la producción de anillos de forja (SPRRO, por sus siglas en inglés). En el SPRRO, la demanda de anillos se modela como un conjunto de trabajos que deben ser programado dentro de un horizonte de planificación. Un comparativo de modelos matemáticos para el problema específico de rolado de anillos es presentado comparando los modelos tradicionales de MILP contra un modelo de CP.

El modelo base de MILP es tomado del planteamiento de Özgüven *et al.* (2010). La función objetivo propuesta para el modelo de constraint programming fue la minimización de la tardanza, siendo definida esta como el la máxima diferencia entre el tiempo de entrega menos el tiempo de terminación de cada una de las jobs. Los modelos matemáticos fueron desarrollados para esta investigación.

3.2 PARÁMETROS

Note que la capacidad de los hornos se da en términos del peso máximo que pueden soportar y además, el conjunto de piezas de trabajo que se procesan, simultáneamente, debe ser compatible en su rango de temperatura. Los tiempos de carga y descarga de las piezas de trabajo se consideran un retraso. (δ_m) entre operaciones consecutivas en la máquina m .

Relaciones de precedencia entre las operaciones de la misma pieza de trabajo se consideran, así como un tiempo de preparación a_{ijm} entre operaciones consecutivas i y j asignadas a la misma máquina m ; el tiempo total que una pieza de trabajo puede estar en un horno m es un tiempo variable entre el tiempo de procesamiento p_{im} y

el tiempo de procesamiento más un tiempo máximo de retención. Además, todas las operaciones tienen una fecha de vencimiento deseada en la que deben realizarse. Para una mejor explicación de la formulación propuesta, introducimos la siguiente notación.

Las relaciones de precedencia consideran las operaciones de la misma pieza de trabajo, así como un tiempo de configuración a_{ijm} entre operaciones consecutivas i y j asignadas a la misma máquina m ; el tiempo total que una pieza de trabajo puede estar en un horno m es un tiempo variable entre tiempo de procesamiento p_{im} y el tiempo de procesamiento más un tiempo máximo de mantenimiento.

Los tiempos de carga y descarga de las piezas de trabajo se consideran un retraso (δ_m) entre operaciones consecutivas en la máquina m . Además, todas las operaciones tienen una fecha de vencimiento deseada d en la que deben realizarse. La notación matemática completa del modelo es descrita a continuación:

M Conjunto m de maquinas

O Conjunto de operaciones

C_m Capacidad de la maquina m (en términos del peso o número de unidades)

$\underline{l}_{i,j}$ Tiempo de espera mínimo (lag) entre el tiempo de inicio de la operación j y el tiempo final de la operación i ($\underline{l}_{i,j} = -\infty$ si no existe relación de precedencia entre las operaciones i y j)

$\bar{l}_{i,j}$ Tiempo de espera máximo (lag) entre el tiempo de inicio de la operación j y el tiempo final de la operación i ($\bar{l}_{i,j} = \infty$ si no existe relación de precedencia entre las operaciones i y j)

- p_{im} Tiempo de procesamiento entre la operación i en la maquina m ($p_{im} = +\infty$ si la maquina m no puede procesar la operación i)
- d Fecha de vencimiento deseada para terminar todas las operaciones ($d=\infty$ si la fecha de vencimiento no es especificada)
- h_{im} Máximo tiempo de espera de la operación i en la maquina m ($h_{im}=0$ si no es requerido tiempo de sostenimiento en la maquina m para la operación i)
- w_{im} Peso de la operación i en la maquina m (peso en número de unidades)
- δ_m Tiempo de retraso de entrada y salida entre dos consecutivas operaciones en la maquina m
- a_{ijm} Tiempo de preparación en la maquina m cuando se procesa la operación i después de la operación j ($a_{ijm} = 0$ si no hay tiempos de preparación)
- \mathcal{M} Un numero grande

3.3 VARIABLES DE DECISIÓN

- s_i Tiempo de inicio de la operación i
- c_i Tiempo de completamiento de la operación i
- e_i Tiempo extra requerido para completar la operación i (tiempo de sostenimiento)
- $x_{ij} = 1$ si la operación i es procesada antes de la operación j . 0 en cualquier otro caso.
- $y_{im} = 1$ si la operación i es procesada por la maquina m . 0 en cualquier otro caso.
- $z_{ijm} = 1$ si la operación i es procesada y sobrepuesta antes de la operación j . 0 en cualquier otro caso

3.3.1 MODELO DE PROGRAMACIÓN LINEAL ENTERA MIXTA

$$\text{Min} \sum_{i \in \mathcal{O}} e_i \quad (3.1)$$

restricto a:

$$c_i \leq d + e_i \quad \forall i \in \mathcal{O} \quad (3.2)$$

$$c_i \geq s_i + \sum_{m \in \mathbf{M}} p_{im} y_{im} \quad \forall i \in \mathcal{O} \quad (3.3)$$

$$c_i \leq s_i + \sum_{m \in \mathbf{M}} (p_{im} + h_{im}) y_{im} \quad \forall i \in \mathcal{O} \quad (3.4)$$

$$\sum_{m \in \mathbf{M}} y_{im} = 1 \quad \forall i \in \mathcal{O} \quad (3.5)$$

$$s_j \geq c_i + l_{ij} \quad \forall i, j \in \mathcal{O} \quad (3.6)$$

$$s_j \leq c_i + \bar{l}_{ij} \quad \forall i, j \in \mathcal{O} \quad (3.7)$$

$$s_j \geq c_i + a_{ijm} - \mathcal{M}(3 - x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.8)$$

$$s_i \geq c_j + a_{jim} - \mathcal{M}(2 + x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.9)$$

$$s_j \geq s_i + \delta_m - \mathcal{M}(3 - x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.10)$$

$$s_i \geq s_j + \delta_m - \mathcal{M}(2 + x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.11)$$

$$c_j \geq c_i + \delta_m - \mathcal{M}(3 - x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.12)$$

$$c_i \geq c_j + \delta_m - \mathcal{M}(2 + x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.13)$$

$$s_j \geq c_i - \mathcal{M}(3 + z_{ijm} - x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.14)$$

$$s_i \geq c_j - \mathcal{M}(2 - z_{jim} - x_{ij} - y_{im} - y_{jm}) \quad \forall i < j \in \mathcal{O}, \forall m \in M \quad (3.15)$$

$$\sum_{j \neq i} w_{jm} z_{ijm} \leq (C_m - w_{im}) y_{im} \quad \forall i \in \mathcal{O}, \forall m \in M \quad (3.16)$$

El objetivo (3.1) minimiza la tardanza de las operaciones. La restricción (3.2) calcula el tiempo extra (después del límite de tiempo deseado) requerido para terminar cada operación. Las restricciones (3.3) y (3.4) establecen que el tiempo de finalización de cada operación debe estar entre el tiempo de procesamiento y el

tiempo de procesamiento más el tiempo permitido. Las restricciones (3.5) aseguran que cada operación está asignado a una máquina. Las restricciones (3.6) y (3.7) establecen el mínimo y retraso máximo de tiempo entre operaciones con relación de precedencia, respectivamente. Las restricciones (3.8) y (3.9) garantizan los tiempos de configuración dependientes de la secuencia. Las restricciones (3.10) - (3.13) modelan la cola mediante primeras entradas y primeras salidas en las máquinas. Las restricciones (3.14) y (3.15) se utilizan para detectar si dos operaciones se procesan al mismo tiempo en la misma máquina. Las restricciones (3.16) evitan exceder la capacidad de las máquinas. El valor de \mathcal{M} se puede estimar de la siguiente manera:

$$p_i = \max_{m \in M} \{p_{im} + h_{im} \mid p_{im} < \infty\} \quad \forall i \in \mathcal{O} \quad (3.17)$$

$$l_i = \max_{j \in \mathcal{O}} \{l_{ij} \mid l_{ij} \geq 0\} \quad \forall i \in \mathcal{O} \quad (3.18)$$

$$a_i = \max_{m \in M, i \in \mathcal{O}} \{a_{ijm} \mid p_{im} < \infty\} \quad \forall i \in \mathcal{O} \quad (3.19)$$

$$\mathcal{M} = \sum_{i \in \mathcal{O}} p_i + \max\{l_i, a_i\} \quad (3.20)$$

Si se quiere minimizar el tiempo total de manufactura la formulación matemática sería la siguiente:

$$\text{Min } C_{max}$$

sujeto a: 3.2-3.16

$$C_{max} \geq c_j \quad \forall j \in \mathcal{O}$$

3.3.2 FORMULACIÓN DEL MODELO DE PROGRAMACIÓN DE RESTRICCIONES

$$\text{Min } \sum_{i \in \mathcal{O}} e_i \quad (3.21)$$

$$e_i = \max \{0, d - c_i\} \quad \forall i \in \mathcal{O} \quad (3.22)$$

$$c_i \geq s_i + \sum_{m \in M} p_{im} y_{im} \quad \forall i \in \mathcal{O} \quad (3.23)$$

$$c_i \leq s_i + \sum_{m \in M} (p_{im} + h_{im}) y_{im} \quad \forall i \in \mathcal{O} \quad (3.24)$$

$$\sum_{m \in M} y_{im} = 1 \quad \forall i \in \mathcal{O} \quad (3.25)$$

$$s_j \geq c_i + l_{ij} \quad \forall i, j \in \mathcal{O} \quad (3.26)$$

$$s_j \leq c_i + \bar{l}_{ij} \quad \forall i, j \in \mathcal{O} \quad (3.27)$$

$$(y_{im} = 1 \wedge y_{jm} = 1) \implies \{s_j \geq c_i + a_{mij} \vee s_i \geq c_j + a_{mij}\} \quad \forall i, j \in \mathcal{O} \quad (3.28)$$

$$(y_{im} = 1 \wedge y_{jm} = 1) \implies$$

$$\{s_j \geq s_i + \max(c_i - s_i) - (c_j - s_j) + \delta_m \vee s_i \geq s_j +$$

$$\max(c_j - s_j) - (c_i - s_i) + \delta_m$$

$$\forall i, j \in \mathcal{O}, \forall m \in M \quad (3.29)$$

$$u_{im}(t) = \begin{cases} w_{im} & \text{si } s_i \leq t \leq c_i \\ 0 & \text{Otro caso} \end{cases} \quad \forall i \in \mathcal{O}, \forall m \in M \quad (3.30)$$

$$\max \sum_{i \in \mathcal{O}} y_{im} u_{im} \leq C_m \quad \forall m \in M \quad (3.31)$$

El objetivo (3.21) minimiza la tardanza de las operaciones. Las restricciones (3.22) calcula el tiempo extra (después del límite de tiempo deseado) requerido para terminar cada operación. Las restricciones (3.23) (3.24) establecen que el tiempo de finalización de cada operación debe estar entre el tiempo de procesamiento y

el tiempo de procesamiento más el tiempo máximo de mantenimiento permitido. Las restricciones (3.25) aseguran que cada operación se asigne a una máquina. Las restricciones (3.25) asegura que cada operación está asignada a una máquina. Las ecuaciones (3.26) y (3.27) establecen los tiempos mínimos y máximos de espera entre las operaciones. La restricción (3.28) garantiza la dependencia de los tiempos de preparación. La restricción (3.29) modelan la priorización mediante primeras entradas y primeras salidas (*FIFO*) en las máquinas.

Expresiones (3.30) y (3.31) evitar exceder la capacidad de las máquinas. Tenga en cuenta que usamos un función clásica para calcular la capacidad utilizada por la operación i en la máquina m en tiempo t dentro del horizonte de planificación, una representación gráfica de esta función es vista en la figura 3.6.

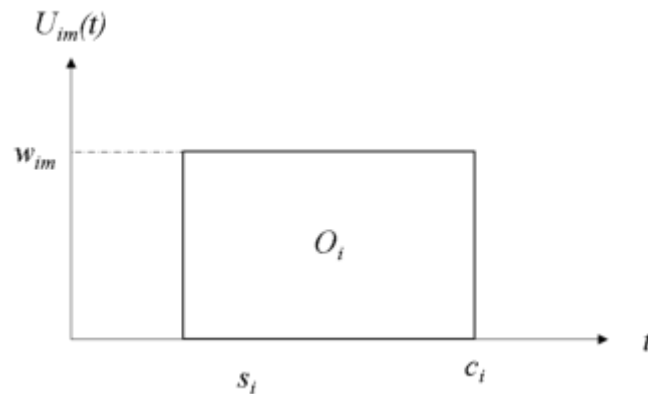


FIGURA 3.6: Activación del peso de la pieza en el modelo matemático

Si se quiere minimizar el tiempo total de manufactura es mediante la función objetivo 3.32 con las restricciones de tiempo de la ecuación 3.33

$$\text{Min } C_{max} \quad (3.32)$$

$$Cmax \geq c_j \forall j \in \mathcal{O} \quad (3.33)$$

EXPERIMENTACIÓN Y RESULTADOS

4.1 DESCRIPCIÓN DE LAS INSTANCIAS

Las instancias se muestran en la tabla 4.1 y se generaron basándonos en las posibles situaciones que se producen en planta así como posibles casos teóricos. En la primer columna se muestran los nombres de las instancias que describen como primer coeficiente el numero de trabajos, posteriormente el numero de productos, el número de hornos permitidos y finalmente la cantidad de tiempo en segundos permitido respectivamente. Se generaron instancias con las condiciones actuales de la empresa (3 hornos). En la tercer columna se muestra la cantidad de operaciones promedio. Cada una de estas fueron generadas en 10 experimentos con diferentes semillas.

INSTANCIAS	JOBS	OPERACIONES
sprro 1-1-3-200	1.000	7.800
sprro 5-1-3-200	5.000	39.000
sprro 5-5-3-200	5.000	38.800
sprro 10-1-3-200	10.000	78.000
sprro 10-5-3-200	10.000	81.400
sprro 10-10-3-200	10.000	79.000
sprro 15-1-3-200	15.000	117.000
sprro 15-5-3-200	15.000	122.200
sprro 15-10-3-200	15.000	119.200
sprro 20-1-3-200	20.000	156.000
sprro 20-5-3-200	20.000	165.000
sprro 20-10-3-200	20.000	158.000
sprro 25-1-3-200	25.000	195.000
sprro 25-5-3-200	25.000	202.600
sprro 25-10-3-200	25.000	195.600
sprro 30-1-3-200	30.000	234.000
sprro 30-5-3-200	30.000	242.200

TABLA 4.1: Instancias generadas para el SPRRO

4.2 ENTORNO EXPERIMENTAL

Los modelos matemáticos propuestos de MILP y CP fueron modelados en C++ y se resuelven con el CPLEX 12.6.3. Todas las pruebas se llevaron a cabo en una computadora con un procesador Intel Xeon E5 de cuatro núcleos a 3,7 GHz con 12 GB de memoria RAM. Se ajustó un rango de límites de tiempo (200 a 3600 segundos) para verificar la eficiencia del MILP y CP. Se realizó el comparativo de los resultados de los modelos matemáticos empleando como función objetivo el makespan.

El tiempo de convergencia en segundos mostró en ambos casos comportamientos de tipo exponencial, pero el comportamiento del CP muestra una reducción de hasta un 80 % en los tiempos de convergencia de la solución.

Los resultados de las instancias generadas para la experimentación considerando el número de máquinas en el sistema se presenta en la tabla 5.1. Solo se consideraron una cortadora, una prensa para estas instancias.

4.3 EXPERIMENTOS COMPUTACIONALES

Una de las funciones objetivo empleadas para esta investigación fue la tardanza, definida como la sumatorio de los máximos entre 0 y la diferencia entre el tiempo de entrega y el tiempo de terminación de la actividad. Se muestra el tiempo computacional en segundos empleados por ambos modelos matemáticos en la figura 4.1 así como el tiempo en segundos. La figura 4.2 muestra cómo se incrementa la diferencia de los resultados de ambos modelos matemáticos en segundos. En el caso de la figura 4.1 se puede apreciar la asíntota superior generada por el tiempo máximo en que se dejó correr el algoritmo.

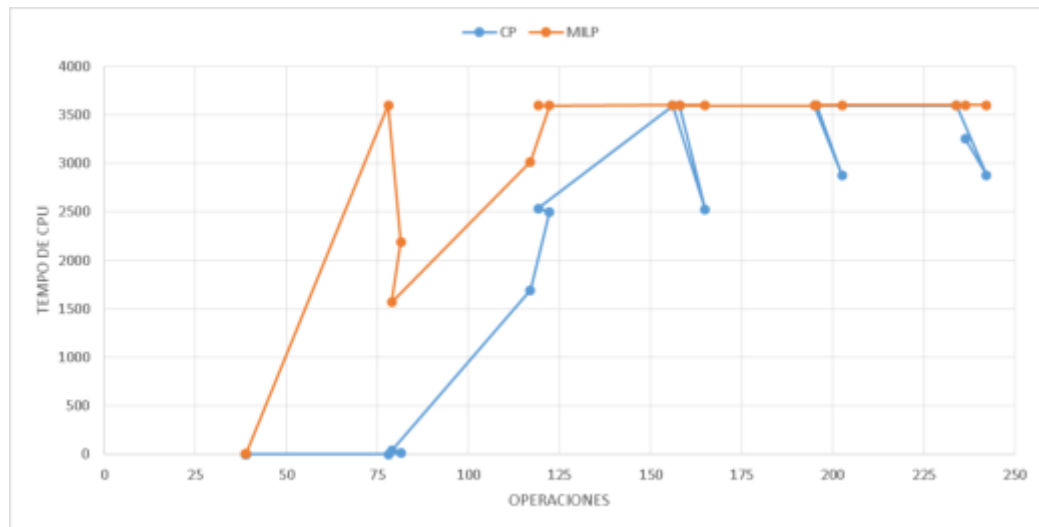


FIGURA 4.1: Operaciones vs Tiempo de CPU CP & MILP

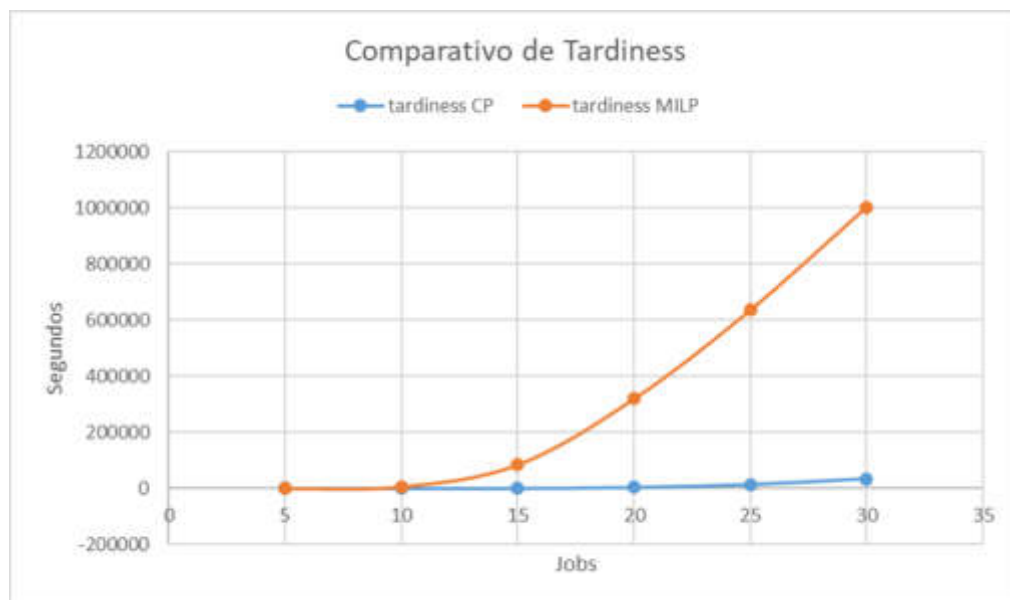


FIGURA 4.2: Comparativo de Tardiness: MILP & CP

Se encuentra que el modelo de CP proporciona soluciones con una mayor velocidad al resolver el modelo matemático de MILP, dando una ventaja competitiva en la toma de decisiones a un usuario del modelo como lo muestra la figura 4.2. La asíntota generada en los resultados de MILP es debido a la restricción de tiempo computacional para correr el procedimiento. El tiempo en segundos que generan los

modelos matemáticos aplicando el tardiness como función objetivo es mostrado en la figura 4.2.

4.4 COMPARATIVOS ENTRE MILP Y CP

El comparativo de los modelos matemáticos de MILP y CP se encuentran en el apéndice A. Estos resultados incluyen factores tales como:

- Tardiness
- Makespan
- Tiempo de CPU
- GAP contra resultados de CPLEX

La cantidad de segundos de makespan en segundos con respecto a la cantidad de trabajos son mostrados en la figura 4.3 en donde se hace el comparativo de los modelos de MILP y CP.

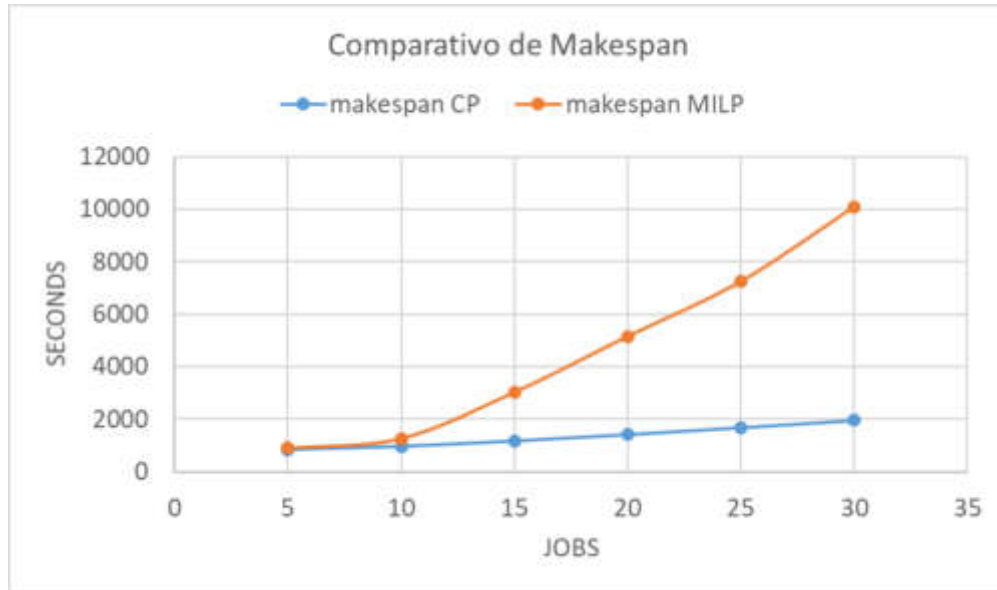


FIGURA 4.3: Comparativo de Makespan: MILP & CP

La diferencia porcentual indica que mientras más grande es el problema en términos de cantidad de trabajos mejores son los resultados que ofrece un modelo matemático de CP para el problema de asignación. Una disminución de hasta el 80 % se aprecia en más de 230 operaciones de acuerdo a los resultados del promedio de las instancias. Los resultados se comportaron exponencialmente realizando la experimentación hasta con 30 trabajos diferentes con las mismas instancias para poder realizar un comparativo adecuado. Los resultados son mostrados en la figura 4.3.

Se graficó el resultado del tardiness aplicando ambas funciones objetivo (makespan y tardiness) para realizar el comparativo el cual se muestra en la figura 4.5. Otra diferencia en la experimentación se presenta en la variable objetivo del tardiness, el modelo de CP no presentó trabajos tardíos sino hasta instancias de más de 11 operaciones.

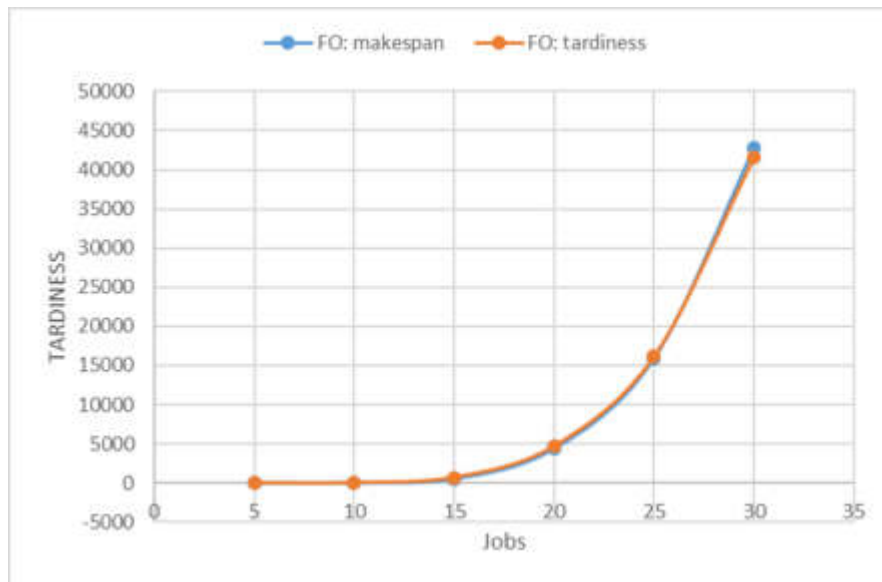


FIGURA 4.4: Jobs vs Tardiness comparando ambas funciones objetivo

Se registran los resultados del makespan en la figura 4.6 y del tardines en la figura 4.7 en instancias específicas de 1, 5 y 10 jobs aplicando el comparativo de las funciones objetivo de makespan y tardiness. Se aplicó como función objetivo el makespan en instancias con 1, 5 y 10 trabajos. Se graficó el resultado del makespan y el tardiness con el makespan como función objetivo obteniendo los resultados mostrados en la figura 4.6.

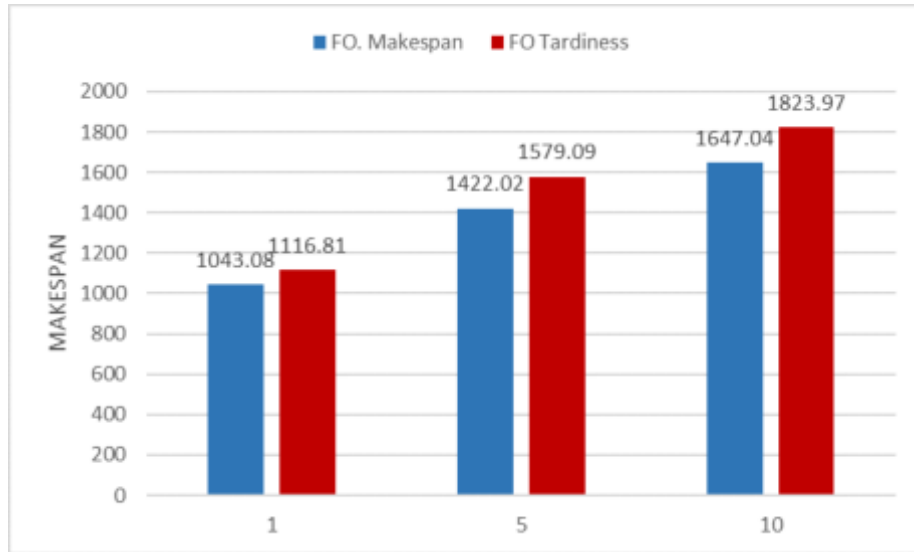


FIGURA 4.5: Makespan comparando ambas funciones objetivo con 1, 5, 10 jobs

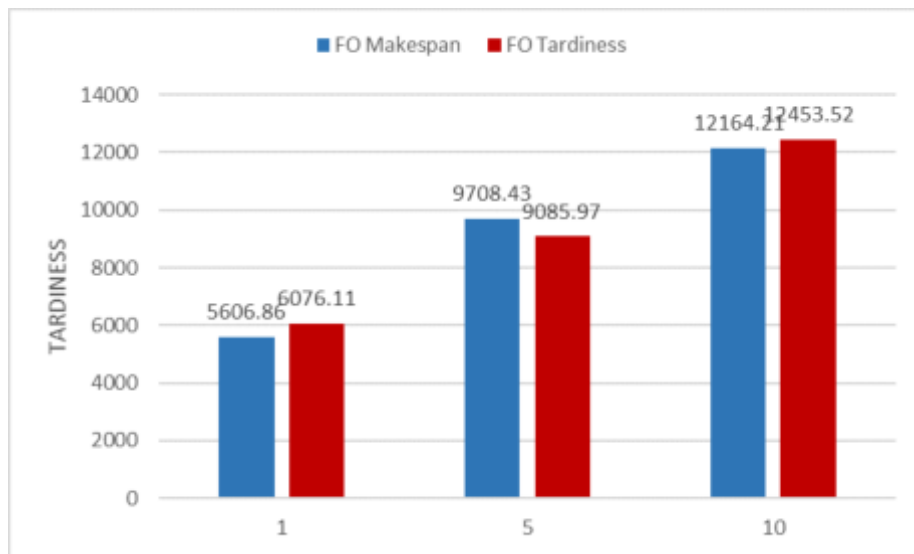


FIGURA 4.6: Tardiness comparando ambas funciones objetivo con 1, 5, 10 jobs

Se aprecia un cambio de crecimiento del 9.62% en promedio de tardanza cuando cambiamos la función objetivo. Cuando la medición se realiza sobre el tardiness en ambas funciones objetivo podemos verificar que en 1 y 10 trabajos el tardiness es inferior cuando la función objetivo sigue siendo el makespan. Estos resultados implican que es mejor utilizar una función objetivo basada en el makespan ya que esto

garantiza mejores tiempos de cumplimiento en tardiness para casi todos los escenarios. Una ampliación en la segmentación de la cantidad de trabajos proporcionará una mayor comprensión de los efectos de utilizar las opciones de las funciones objetivo. La política del número de operaciones a realizar en vez del número de trabajos muestra que no hay diferencia significativa en la calidad de la solución cuando la función objetivo es el makespan medido en segundos.

El crecimiento porcentual del tardiness muestra un patrón disperso en el aumento de operaciones cuando aplicamos el modelo de MILP como lo muestra la figura 4.8. Esto demuestra que no hay una correlación directa entre el número de operaciones y el tardiness a diferencia del makspan.

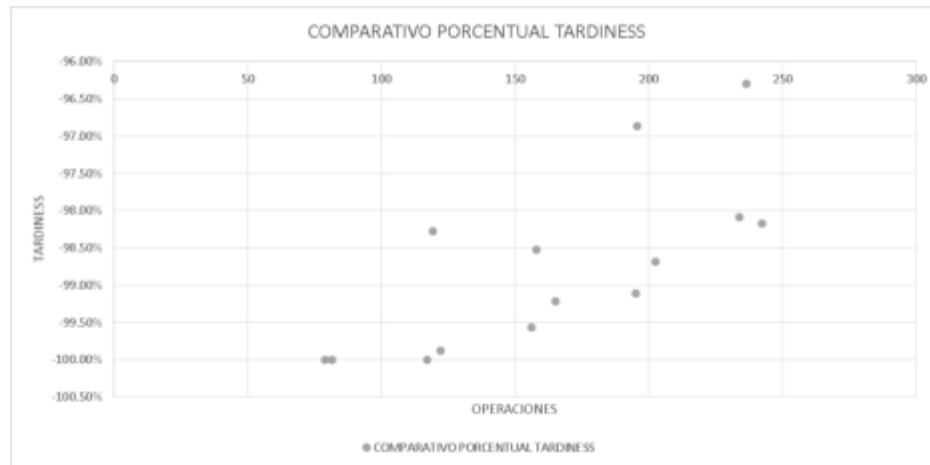


FIGURA 4.7: Operaciones vs Tardiness Comparativo porcentual

Se fijaron tiempos máximos para realizar el comparativo entre CP y MILP en segundos para el solucionador CPLEX en donde el modelo de CP demuestra tener una mayor velocidad de convergencia y encontrar soluciones factibles para cantidades de operaciones superiores a las 100 se genera una diferencia en el resultado del tardiness. Las figuras 4.9 y 4.10 muestran el comportamiento del tardiness promedio con cada tipo de modelo matemático basado en el número de operaciones:

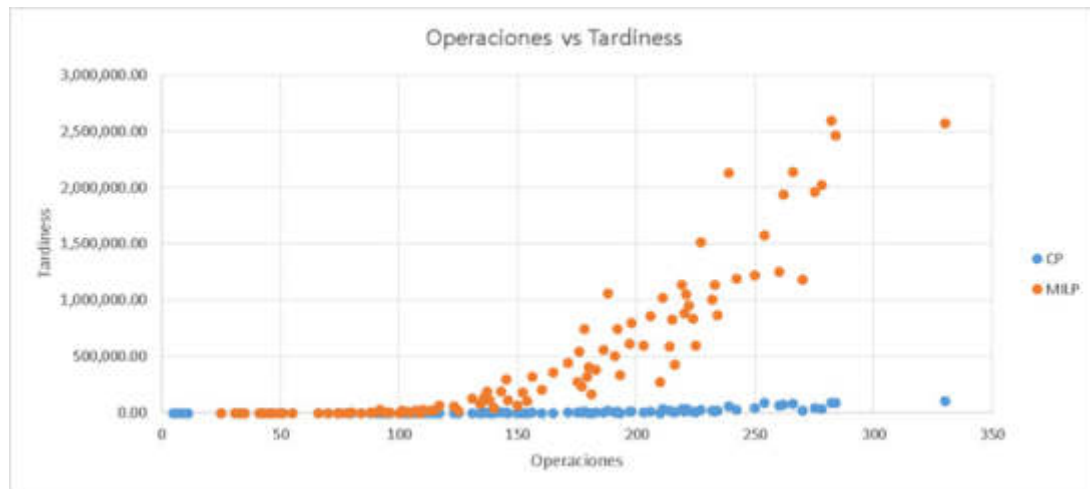


FIGURA 4.8: Operaciones vs Tardiness CP & MILP

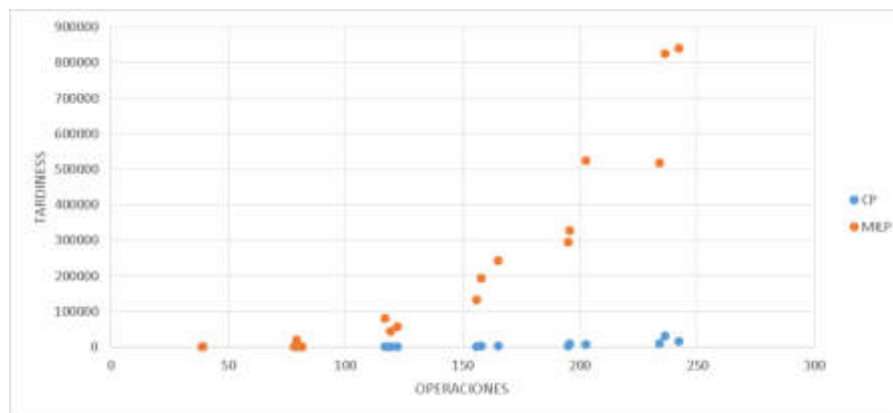


FIGURA 4.9: Operaciones vs Tardiness CP & MILP

Se muestra que en instancias pequeñas no existe una diferencia significativa entre el CP y el MILP ya que ambos resuelven el problema en casi los mismos tiempos, no obstante, para problemas con un mayor número de trabajos a procesar (como comúnmente se dan en la industria) el MILP se vuelve impráctico como lo muestran la figura 4.11.

Los resultados experimentales dan evidencia de que el modelo de CP genera

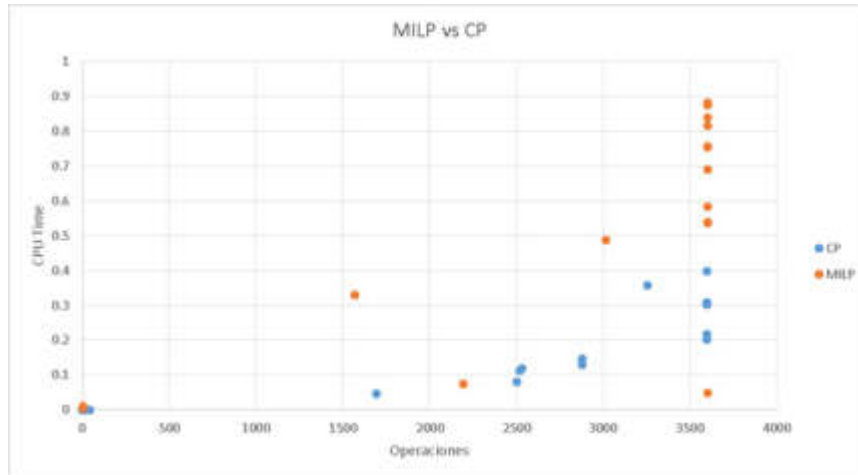


FIGURA 4.10: Operaciones vs Tiempo de CPU CP & MILP

soluciones en tiempos menores que el MILP, lo que significa una mayor rentabilidad del uso de estos modelos en campo real. La información sobre la calidad de las soluciones fue abordada para conocer el comportamiento del CP con las funciones objetivo de makespan y tardiness. La tabla 4.2 muestra el comparativo de los tiempos de makespan y tardiness con diferentes escenarios de cantidad de hornos. Se corrieron iteraciones para solución del problema en 200, 500, 1000 y 3600 segundos para poder comparar la mejora en las soluciones generadas con respecto a ambas variables.

Tpo de computo	3 hornos		4 hornos		5 hornos	
	M	T	M	T	M	T
200	1410.12	12552.42	1273.14	8114.89	1238.98	6577.77
500	1346.36	9122.86	1220.53	4853.66	1192.99	3821.37
1000	1310.58	7494.44	1200.58	3717.67	1168.24	2822.75
3600	1281.57	6233.87	1173.46	2601.29	1149.71	1852.57

TABLA 4.2: Comparativo con diferentes cantidades de hornos teóricos en CP

El comparativo de las soluciones planteadas por el makespan y el tardiness cambiando la función objetivo son mostradas en las figuras 4.12 y 4.6.

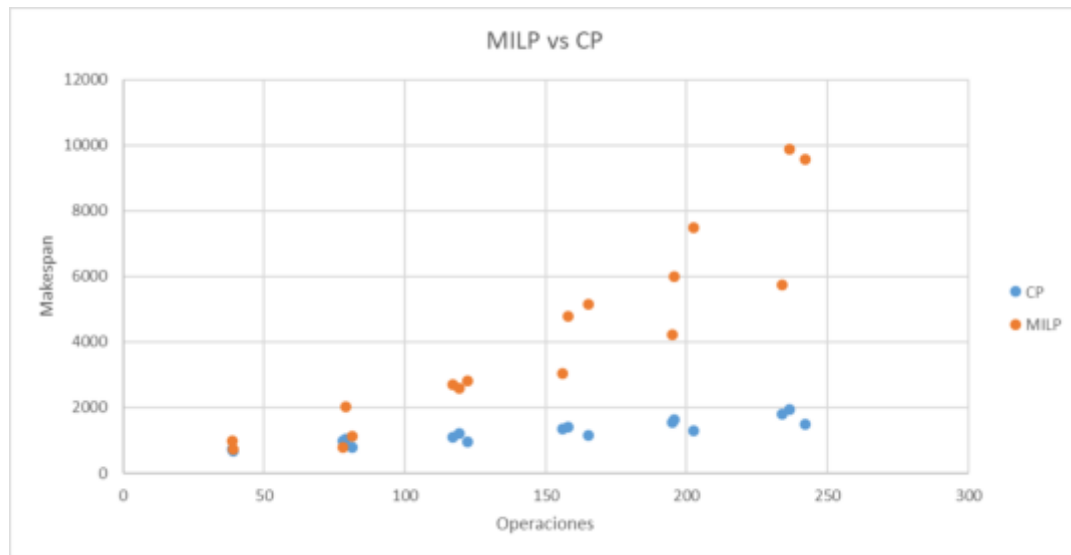


FIGURA 4.11: Operaciones vs Makespan CP & MILP

La figura 5.16 muestra el tiempo de makespan con respecto al numero de operaciones demostrando que en el modelo de MILP los tiempos se disparan de forma exponencial. El modelo de CP también muestra un modelo exponencial, sin embargo con una curva de valores inferiores debido a la restricción del tiempo limite del solver.

Teniendo las instancias del modelo de CP se realizaron pruebas con el algoritmo dando maquinas opcionales, es decir; hornos extra. Las instancias que fueron validadas son para 200, 500, 1000 y 3600 segundos de operación en funciones objetivo de makespan M y tardiness T para verificar la eficiencia del algoritmo.

4.5 EXPERIMENTOS ADICIONALES

Podemos apreciar que hay una reducción del 9.15% de cambiar de un horno a 4 en el makespan lo cual resulta benéfico, el cambio en el makespan es de 2.54% de incrementar los hornos hasta 5 de manera que no es factible incrementar la cantidad

de hornos en más de una unidad. Así mismo el cambio en la disminución del tardiness es de 44.18 % de 3 a 4 máquinas, sin embargo de 4 a 5 solo es del 21.43 %. Las figuras 4.13 y 4.14 muestran el comportamiento de estos resultados en el promedio de las instancias.

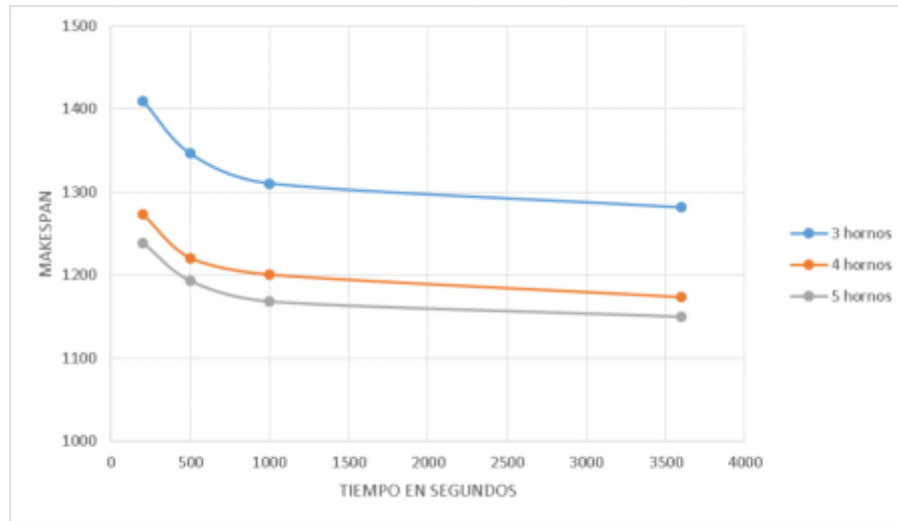


FIGURA 4.12: Tiempo segundos vs Makespan 3, 4, 5 HORNOS

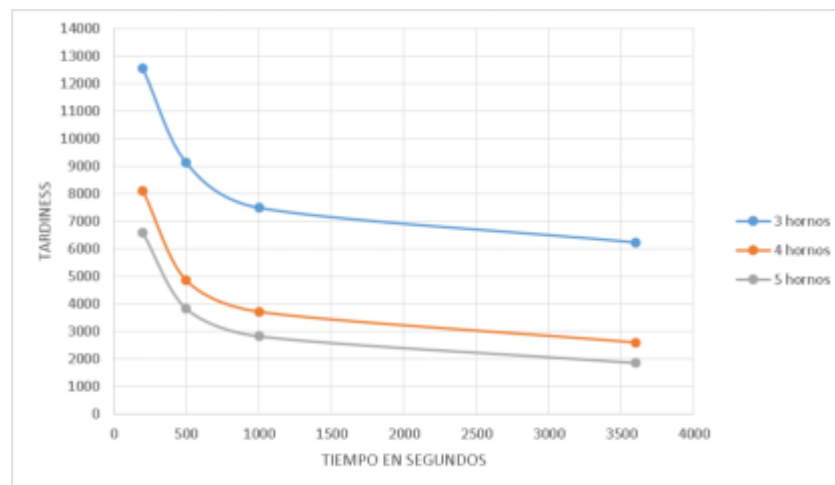


FIGURA 4.13: Tiempo segundos vs Tardiness 3, 4, 5 HORNOS

La figura 4.15 muestra el comportamiento del makespan con respecto a la cantidad de trabajos empleados en el modelo y la regresión de este comportamiento es

exponencial con una determinación del 99.79 %.

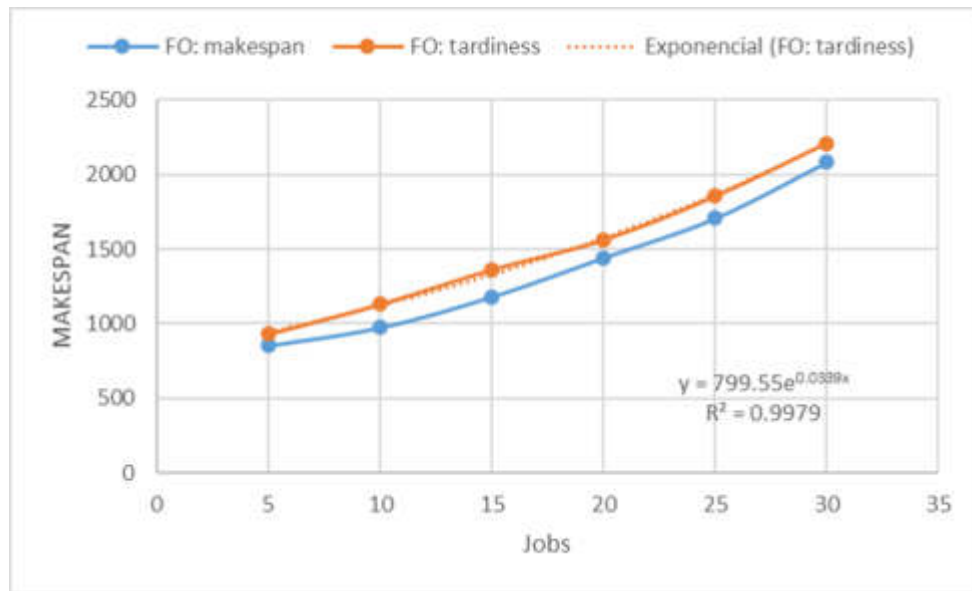


FIGURA 4.14: Jobs vs Makespan comparando ambas funciones objetivo

CAPÍTULO 5

CONCLUSIÓN

Como conclusión, el presente trabajo estudió las ventajas de los modelos de CP aplicados a problemas de FJSSP con variantes en la cantidad de máquinas para mejorar la utilización del proceso. Se demostró que los modelos de CP tienen la ventaja de reducir el tiempo de búsqueda en este tipo de problemas aplicando métodos exactos por medio de CPLEX.

De esta manera también se experimentó con cambios en los parámetros de los modelos tales como la cantidad de máquinas implantadas en un sistema real con valores aproximados a un modelo tomado de la industria ya que los porcentajes de utilización el tiempo total de manufactura y la tardanza no tuvieron un cambio relativo después de las cuatro máquinas. La experimentación se realizó con 3, 4 y 5 máquinas para validar el desempeño del sistema general.

Este ramo de la industria particular tiene problemáticas fuertes debido a malas decisiones de planeación con horizontes de tiempo muy bajos. La capacidad de reacción que se tiene para los tomadores de decisiones es crucial para su seguimiento de este tipo de negocios y las conclusiones que fueron obtenidas de este documento son fácilmente exportables a otra rama de la industria que podrían beneficiarse de

modelos de programación de la producción con restricciones y metaheurísticas.

5.1 TRABAJOS FUTUROS

- Validar el modelo con otras heurísticas
- Comparar la complejidad computacional de problema

Como trabajos futuros propuestos para el nuevo problema de programación de la producción se buscará la aplicación de diferentes heurísticas al modelo de CP y el comparativo de la complejidad computacional contra otros problemas similares.

APÉNDICE A

APÉNDICE

En las siguientes tablas se presenta los resultados de diferentes instancias del problema. Las columnas representan los siguientes conceptos:

1. Instancia con No. de trabajos y Cantidad de tiempo del algoritmo.
2. Numero de operaciones promedio
3. Tardanza (tardiness)
4. Tiempo total de procesamiento (makespan)
5. Tiempo de CPU
6. Diferencia contra el óptimo (Gap)

INSTANCIA	OP. PROMEDIO	TARDINESS	MAKESPAN	TIEMPO CPU	GAP
sprro 1-1-3-200	39.000	0.000	717.800	3.417	0.010
sprro 5-1-3-200	38.800	0.000	991.100	5.710	0.003
sprro 5-5-3-200	78.000	0.000	777.900	3599.458	0.048
sprro 10-1-3-200	81.400	289.500	1138.100	2192.504	0.073
sprro 10-5-3-200	79.000	21199.600	2017.800	1568.086	0.331
sprro 10-10-3-200	117.000	81494.000	2690.200	3016.480	0.488
sprro 15-1-3-200	122.200	57637.800	2800.300	3599.199	0.540
sprro 15-5-3-200	119.200	43285.100	2589.000	3599.239	0.537
sprro 15-10-3-200	156.000	133378.300	3029.600	3599.310	0.583
sprro 20-1-3-200	165.000	243580.200	5135.500	3599.329	0.757
sprro 20-5-3-200	158.000	192979.600	4771.100	3599.230	0.754
sprro 20-10-3-200	195.000	294190.900	4219.000	3599.238	0.691
sprro 25-1-3-200	202.600	525531.300	7498.000	3599.285	0.839
sprro 25-5-3-200	195.600	327303.600	6004.200	3599.383	0.816
sprro 25-10-3-200	234.000	517660.000	5736.200	3599.316	0.756
sprro 30-1-3-200	242.200	840058.500	9578.300	3599.393	0.877
sprro 30-5-3-200	236.400	824764.700	9867.400	3599.903	0.882

TABLA A.1: Resultados de experimentación con MILP

INSTANCIA	OP.PROMEDIO	TARDINESS	MAKESPAN	TIEMPO CPU	GAP
sprro 1-1-3-200	39.000	0.000	677.800	0.005	0.000
sprro 5-1-3-200	38.800	0.000	717.800	0.202	0.000
sprro 5-5-3-200	78.000	0.000	989.000	0.377	0.000
sprro 10-1-3-200	81.400	0.000	777.900	12.026	0.000
sprro 10-5-3-200	79.000	0.000	1034.400	40.341	0.000
sprro 10-10-3-200	117.000	0.000	1100.700	1694.751	0.046
sprro 15-1-3-200	122.200	70.000	963.200	2501.563	0.080
sprro 15-5-3-200	119.200	747.200	1210.200	2533.891	0.118
sprro 15-10-3-200	156.000	577.300	1341.300	3597.207	0.203
sprro 20-1-3-200	165.000	1916.700	1147.800	2521.088	0.113
sprro 20-5-3-200	158.000	2841.100	1402.400	3597.204	0.217
sprro 20-10-3-200	195.000	2614.100	1539.100	3597.333	0.309
sprro 25-1-3-200	202.600	6928.700	1305.700	2878.482	0.130
sprro 25-5-3-200	195.600	10247.000	1644.400	3598.049	0.302
sprro 25-10-3-200	234.000	9919.400	1790.800	3598.004	0.398
sprro 30-1-3-200	242.200	15377.300	1477.700	2879.830	0.147
sprro 30-5-3-200	236.400	30583.700	1931.700	3256.157	0.358

TABLA A.2: Resultados de experimentación con CP

BIBLIOGRAFÍA

- AFSAR, H., P. LACOMME, L. REN, C. PRODHON y D. VIGO (2016), «Resolution of a Job-Shop problem with transportation constraints: a master/slave approach», *IFAC-PapersOnLine*, **49**(12), págs. 898 – 903, 8th {IFAC} Conference on Manufacturing Modelling, Management and Control {MIM} 2016Troyes, France, 28—30 June 2016.
- ALLWOOD, J. M., A. E. TEKKAYA y T. F. STANISTREET (2005), «The development of ring rolling technology», *Steel research international*, **76**(2-3), págs. 111–120.
- AMROUCHE, K. y M. BOUDHAR (2016), «Two machines flow shop with reentrance and exact time lag», *RAIRO-Operations Research*, **50**(2), págs. 223–232.
- AMROUCHE, K., M. BOUDHAR, M. BENDRAOUCHE y F. YALAOUI (2017), «Chain-reentrant shop with an exact time lag: new results», *International Journal of Production Research*, **55**(1), págs. 285–295.
- ARTIGUES, C., M.-J. HUGUET y P. LOPEZ (2011), «Generalized disjunctive constraint propagation for solving the job shop problem with time lags», *Engineering Applications of Artificial Intelligence*, **24**(2), págs. 220–231.
- BAYKASOĞLU, A. y F. B. OZSOYDAN (2018), «Dynamic scheduling of parallel heat treatment furnaces: A case study at a manufacturing system», *Journal of manufacturing systems*, **46**, págs. 152–162.

- BOCHTIS, D. D., C. G. SØRENSEN y P. BUSATO (2014), «Advances in agricultural machinery management: A review», *Biosystems Engineering*, **126**, págs. 69 – 81.
- BOTTA-GENOULAZ, V. (2000), «Hybrid flow shop scheduling with precedence constraints and time lags to minimize maximum lateness», *International journal of production economics*, **64**(1-3), págs. 101–111.
- BOWMAN, E. H. (1959), «The schedule-sequencing problem», *Operations research*, **7**(5), págs. 621–624.
- CAUMOND, A., P. LACOMME y N. TCHERNEV (2008), «A memetic algorithm for the job-shop with time-lags», *Computers & Operations Research*, **35**(7), págs. 2331–2356.
- CHAPADOS, N., M. JOLIVEAU, P. L'ECUYER y L.-M. ROUSSEAU (2014), «Retail store scheduling for profit», *European Journal of Operational Research*, **239**(3), págs. 609 – 624.
- CHEN, J. C., T.-L. CHEN y H. HARIANTO (2017), «Capacity planning for packaging industry», *Journal of Manufacturing Systems*, **42**, págs. 153 – 169.
- CHEN, J. C., T.-L. CHEN, B. R. PRATAMA y Q.-F. TU (2016), «Capacity planning in thin film transistor – Liquid crystal display cell process», *Journal of Manufacturing Systems*, **39**, págs. 63 – 78.
- CUI, W.-W., Z. LU y E. PAN (2014), «Integrated production scheduling and maintenance policy for robustness in a single machine», *Computers & Operations Research*, **47**, págs. 81 – 91.
- DEEP, K. y P. K. SINGH (2015), «Design of robust cellular manufacturing system for dynamic part population considering multiple processing routes using genetic algorithm», *Journal of Manufacturing Systems*, **35**, págs. 155 – 163.
- DEFERSHA, F. M. y M. CHEN (2012), «Jobshop lot streaming with routing flexibility, sequence-dependent setups, machine release dates and lag time», *International Journal of Production Research*, **50**(8), págs. 2331–2352.

- DELL'AMICO, M. (1996), «Shop problems with two machines and time lags», *Operations Research*, **44**(5), págs. 777–787.
- DEMIR, Y. y S. K. İŞLEYEN (2013), «Evaluation of mathematical models for flexible job-shop scheduling problems», *Applied Mathematical Modelling*, **37**(3), págs. 977–988.
- DETIENNE, B., R. SADYKOV y S. TANAKA (2016), «The two-machine flowshop total completion time problem: Branch-and-bound algorithms based on network-flow formulation», *European Journal of Operational Research*, **252**(3), págs. 750 – 760.
- DHOUB, E., J. TEGHEM y T. LOUKIL (2013), «Lexicographic optimization of a permutation flow shop scheduling problem with time lag constraints», *International Transactions in Operational Research*, **20**(2), págs. 213–232.
- EUSKALI (22 de Agosto de 2013), «<https://euskalforging.com/proceso-productivo/>», .
- FARAHMAND-MEHR, M., P. FATTAHI, M. KAZEMI, H. ZAREI y A. PIRI (2014), «An efficient genetic algorithm for a hybrid flow shop scheduling problem with time lags and sequence-dependent setup time», *Manufacturing Review*, **1**, pág. 21.
- FONDREVELLE, J., A. OULAMARA y M.-C. PORTMANN (2006), «Permutation flowshop scheduling problems with maximal and minimal time lags», *Computers & Operations Research*, **33**(6), págs. 1540–1556.
- FONDREVELLE, J., A. OULAMARA, M.-C. PORTMANN y A. ALLAHVERDI (2009), «Permutation flow shops with exact time lags to minimise maximum lateness», *International journal of production research*, **47**(23), págs. 6759–6775.
- FRANCIS, A. (2016), «Comparing Time and Quantity Scales for Relationship and Float Calculations», *Procedia Engineering*, **164**, págs. 49 – 56, selected papers from Creative Construction Conference 2016.

- FRIHAT, M., C. SADFI y A. B. HADJ-ALOUANE (2014), «Optimization of integrated employee timetabling and Hybrid Job Shop Scheduling under time lag constraints», en IEEE (editor), *2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, IEEE, págs. 282–287.
- GARILLOT, F., G. GONTHIER, A. MAHBOUBI y L. RIDEAU (2009), «Packaging mathematical structures», en *International Conference on Theorem Proving in Higher Order Logics*, Springer, págs. 327–342.
- GONZÁLEZ, M. A., A. ODDI, R. RASCONI y R. VARELA (2015), «Scatter search with path relinking for the job shop with time lags and setup times», *Computers & Operations Research*, **60**, págs. 37–54.
- GRAHAM, R. L., E. L. LAWLER, J. K. LENSTRA y A. R. KAN (1979), «Optimization and approximation in deterministic sequencing and scheduling: a survey», en *Annals of discrete mathematics*, tomo 5, Elsevier, págs. 287–326.
- GREGOR, M., M. HRUBOŠ y D. NEMEC (2018), «A comparative analysis of constraint programming and metaheuristics for job-shop scheduling», en *2018 Cybernetics & Informatics (K&I)*, IEEE, págs. 1–6.
- GSCHWIND, T. (2015), «A comparison of column-generation approaches to the Synchronized Pickup and Delivery Problem», *European Journal of Operational Research*, **247**(1), págs. 60 – 71.
- HAM, A. (2017), «Flexible job shop scheduling problem for parallel batch processing machine with compatible job families», *Applied Mathematical Modelling*, **45**, págs. 551–562.
- HAM, A. M. y E. ÇAKICI (2016), «Flexible job shop scheduling problem with parallel batch processing machines: {MIP} and {CP} approaches», *Computers & Industrial Engineering*, **102**, págs. 160 – 165.
- HAMDI, I. y T. LOUKIL (2015a), «Minimizing total tardiness in the permutation

- flowshop scheduling problem with minimal and maximal time lags», *Operational Research*, **15**(1), págs. 95–114.
- HAMDI, I. y T. LOUKIL (2015b), «Upper and lower bounds for the permutation flowshop scheduling problem with minimal time lags», *Optimization Letters*, **9**(3), págs. 465–482.
- HAMDI, I. y T. LOUKIL (2017), «The permutation flowshop scheduling problem with exact time lags to minimise the total earliness and tardiness», *International Journal of Operational Research*, **28**(1), págs. 70–86.
- HARBAOUI, H., O. BELLENGUEZ-MORINEAU y S. KHALFALLAH (2016), «Scheduling a two-stage hybrid flow shop with dedicated machines, time lags and sequence-dependent family setup times», en *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, págs. 002 990–002 995.
- HONG, I.-H., C.-C. CHOU y P.-K. LEE (2019), «Admission control in queue-time loop production-mixed integer programming with Lagrangian relaxation (MILP-LAR)», *Computers & Industrial Engineering*, **129**, págs. 417 – 425.
- HUA, L., J. DENG y D. QIAN (2017), «Recent development of ring rolling theory and technique», *International Journal of Materials and Product Technology*, **54**(1-3), págs. 65–87.
- IGNALL, E. y L. SCHRAGE (1965), «Application of the branch and bound technique to some flow-shop scheduling problems», *Operations research*, **13**(3), págs. 400–412.
- JOHNSON, S. M. (1954), «Optimal two-and three-stage production schedules with setup times included», *Naval research logistics quarterly*, **1**(1), págs. 61–68.
- KAROUI, W., M.-J. HUGUET, P. LOPEZ y M. HAOUARI (2010), «Climbing discrepancy search for flowshop and jobshop scheduling with time lags», *Electronic Notes in Discrete Mathematics*, **36**, págs. 821–828.

- KHOUKHI, F. E., J. BOUKACHOUR y A. E. H. ALAOUI (2017), «The “Dual-Ants Colony”: A novel hybrid approach for the flexible job shop scheduling problem with preventive maintenance», *Computers & Industrial Engineering*, **106**, págs. 236 – 255.
- KRETER, S., J. RIECK y J. ZIMMERMANN (2016), «Models and solution procedures for the resource-constrained project scheduling problem with general temporal constraints and calendars», *European Journal of Operational Research*, **251**(2), págs. 387 – 403.
- KRETER, S., A. SCHUTT, P. J. STUCKEY y J. ZIMMERMANN (2017), «Mixed-integer linear programming and constraint programming formulations for solving resource availability cost problems», *European Journal of Operational Research*, págs. –.
- KU, W.-Y. y J. C. BECK (2016), «Mixed Integer Programming models for job shop scheduling: A computational analysis», *Computers & Operations Research*, **73**, págs. 165 – 173.
- KURZ, M. E. y R. G. ASKIN (2003), «Comparing scheduling rules for flexible flow lines», *International Journal of Production Economics*, **85**(3), págs. 371–388.
- LATORRE-NÚÑEZ, G., A. LÜER-VILLAGRA, V. MARIANOV, C. OBREQUE, F. RAMIS y L. NERIZ (2016), «Scheduling operating rooms with consideration of all resources, post anesthesia beds and emergency surgeries», *Computers & Industrial Engineering*, **97**, págs. 248–257.
- LEE, S. y Y. YIH (2014), «Reducing patient-flow delays in surgical suites through determining start-times of surgical cases», *European Journal of Operational Research*, **238**(2), págs. 620 – 629.
- LEE, Y. H. y S. H. KIM (2002), «Production–distribution planning in supply chain considering capacity constraints», *Computers & industrial engineering*, **43**(1-2), págs. 169–190.

- LEINBERGER, W., G. KARYPIS y V. KUMAR (1999), «Multi-capacity bin packing algorithms with applications to job scheduling under multiple constraints», en *Proceedings of the 1999 International Conference on Parallel Processing*, IEEE, págs. 404–412.
- LI, X. y L. GAO (2016), «An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem», *International Journal of Production Economics*, **174**, págs. 93–110.
- LIN, Y.-K., D.-H. HUANG y C.-F. HUANG (2016a), «Estimated network reliability evaluation for a stochastic flexible flow shop network with different types of jobs», *Computers & Industrial Engineering*, **98**, págs. 401 – 412.
- LIN, Y.-K., D.-H. HUANG y J.-S. LIN (2016b), «Reliability evaluation of a multistate flexible flow shop with stochastic capacity for multiple types of jobs», *Journal of Manufacturing Systems*, **41**, págs. 287 – 298.
- LIU, S. Q. y E. KOZAN (2017), «A hybrid metaheuristic algorithm to optimise a real-world robotic cell», *Computers & Operations Research*, **84**, págs. 188 – 194.
- LUNARDI, W. T., E. G. BIRGIN, D. P. RONCONI y H. VOOS (2021), «Metaheuristics for the online printing shop scheduling problem», *European Journal of Operational Research*, **293**(2), págs. 419–441.
- MANNE, A. S. (1960), «On the job-shop scheduling problem», *Operations research*, **8**(2), págs. 219–223.
- MEHDIZADEH, E., S. V. D. NIAKI y V. RAHIMI (2016), «A vibration damping optimization algorithm for solving a new multi-objective dynamic cell formation problem with workers training», *Computers & Industrial Engineering*, **101**, págs. 35 – 52.
- MIN, D. y Y. YIH (2010), «Scheduling elective surgery under uncertainty and downstream capacity constraints», *European Journal of Operational Research*, **206**(3), págs. 642–652.

- MOKHTARI, H. y M. DADGAR (2015), «Scheduling optimization of a stochastic flexible job-shop system with time-varying machine failure rate», *Computers & Operations Research*, **61**, págs. 31 – 45.
- NAGATA, Y. y I. ONO (2018), «A guided local search with iterative ejections of bottleneck operations for the job shop scheduling problem», *Computers & Operations Research*, **90**, págs. 60 – 71.
- NING, Z., Y. SONG, L. KAIDIAN y C. SIYU (2017), «Effective iterated greedy algorithm for flow-shop scheduling problems with time lags», *Chinese Journal of Mechanical Engineering*, **30**(3), págs. 652–662.
- NOURI, M., A. BEKRAR, A. JEMAI, S. NIAR y A. C. AMMARI (2018), «An effective and distributed particle swarm optimization algorithm for flexible job-shop scheduling problem», *Journal of Intelligent Manufacturing*, **29**(3), págs. 603–615.
- NOURI, H. E., O. B. DRISS y K. GHÉDIRA (2015), «Hybrid metaheuristics within a holonic multiagent model for the flexible job shop problem», *Procedia Computer Science*, **60**, págs. 83–92.
- NOVAS, J. M. (2019), «Production scheduling and lot streaming at flexible job-shops environments using constraint programming», *Computers & Industrial Engineering*, **136**, págs. 252 – 264.
- ÖZGÜVEN, C., L. ÖZBAKIR y Y. YAVUZ (2010), «Mathematical models for job-shop scheduling problems with routing and process plan flexibility», *Applied Mathematical Modelling*, **34**(6), págs. 1539–1548.
- PALMER, D. (1965), «Sequencing jobs through a multi-stage process in the minimum total time—a quick method of obtaining a near optimum», *Journal of the Operational Research Society*, **16**(1), págs. 101–107.
- POTTS, C. (1980), «An adaptive branching rule for the permutation flow-shop problem», *European Journal of Operational Research*, **5**(1), págs. 19–25.

- RABIEE, M., R. S. RAD, M. MAZINANI y R. SHAFAEI (2014), «An intelligent hybrid meta-heuristic for solving a case of no-wait two-stage flexible flow shop scheduling problem with unrelated parallel machines», *The International Journal of Advanced Manufacturing Technology*, **71**(5), págs. 1229–1245.
- RADULY-BAKA, C. y O. S. NEVALAINEN (2015), «The modular tool switching problem», *European Journal of Operational Research*, **242**(1), págs. 100 – 106.
- RIBEIRO, M. S. D. C. *et al.* (2015), «Comparing mixed & integer programming vs. constraint programming by solving job-shop scheduling problems», *Independent Journal of Management & Production*, **6**(1), págs. 211–238.
- SAHRAEIAN, R. y M. NAMAKSHENAS (2015), «On the optimal modeling and evaluation of job shops with a total weighted tardiness objective: Constraint programming vs. mixed integer programming», *Applied Mathematical Modelling*, **39**(2), págs. 955 – 964.
- SAMARGHANDI, H. y M. BEHROOZI (2017), «On the exact solution of the no-wait flow shop problem with due date constraints», *Computers & Operations Research*, **81**, págs. 141 – 159.
- SCHNABEL, A., C. KELLENBRINK y S. HELBER (2018), «Profit-oriented scheduling of resource-constrained projects with flexible capacity constraints», *Business Research*, **11**(2), págs. 329–356.
- SHAHVARI, O. y R. LOGENDRAN (2017), «An Enhanced tabu search algorithm to minimize a bi-criteria objective in batching and scheduling problems on unrelated-parallel machines with desired lower bounds on batch sizes», *Computers & Operations Research*, **77**, págs. 154 – 176.
- SHAHVARI, O. y R. LOGENDRAN (2018), «A comparison of two stage-based hybrid algorithms for a batch scheduling problem in hybrid flow shop with learning effect», *International Journal of Production Economics*, **195**, págs. 227 – 248.

- SHEEN, G.-J. y L.-W. LIAO (2007), «A branch and bound algorithm for the one-machine scheduling problem with minimum and maximum time lags», *European journal of operational research*, **181**(1), págs. 102–116.
- SHEN, L., S. DAUZÈRE-PÉRÈS y J. S. NEUFELD (2018), «Solving the flexible job shop scheduling problem with sequence-dependent setup times», *European Journal of Operational Research*, **265**(2), págs. 503–516.
- SIDENOR (22 de Agosto de 2013), «<https://www.sidenor.com/es/>», .
- SINGH, M. R. y S. MAHAPATRA (2016), «A quantum behaved particle swarm optimization for flexible job shop scheduling», *Computers & Industrial Engineering*, **93**, págs. 36 – 44.
- SUKKERD, W. y T. WUTTIPORNPUN (2016), «Hybrid genetic algorithm and tabu search for finite capacity material requirement planning system in flexible flow shop with assembly operations», *Computers & Industrial Engineering*, **97**, págs. 157 – 169.
- VANHOUCKE, M. y J. COELHO (2016), «An approach using {SAT} solvers for the {RCPSP} with logical constraints», *European Journal of Operational Research*, **249**(2), págs. 577 – 591.
- WAGNER, H. M. (1959), «An integer linear-programming model for machine scheduling», *Naval research logistics quarterly*, **6**(2), págs. 131–140.
- WANG, C., J. LI, F. YE y Y. YANG (2014), «Recharging schedules for wireless sensor networks with vehicle movement costs and capacity constraints», en *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, IEEE, págs. 468–476.
- YE, S., N. ZHAO, K. LI y C. LEI (2017), «Efficient heuristic for solving non-permutation flow-shop scheduling problems with maximal and minimal time lags», *Computers & Industrial Engineering*, **113**, págs. 160–184.

- YU, T.-S., H.-J. KIM y T.-E. LEE (2017), «Minimization of waiting time variation in a generalized two-machine flowshop with waiting time constraints and skipping jobs», *IEEE Transactions on Semiconductor Manufacturing*, **30**(2), págs. 155–165.
- ZHANG, H., S. COLLART-DUTILLEUL y K. MESGHOUNI (2015a), «Cyclic Scheduling of Flexible Job-shop with Time Window Constraints and Resource Capacity Constraints», *IFAC-PapersOnLine*, **48**(3), págs. 816 – 821, 15th IFAC Symposium on Information Control Problems in Manufacturing.
- ZHANG, W., J. LIU, C. LIU, Q. ZHENG y W. ZHANG (2015b), «Workload modeling for virtual machine-hosted application», *Expert Systems with Applications*, **42**(4), págs. 1835 – 1844.
- ZHANG, X. y S. VAN DE VELDE (2015), «Two-machine interval shop scheduling with time lags», *Journal of Scheduling*, **18**(4), págs. 359–368.