

**CORPORACIÓN MEXICANA DE INVESTIGACIÓN EN
MATERIALES**

DIVISIÓN DE ESTUDIOS DE POSGRADO



**Simulación en planeación táctica de la producción tipo taller mediante un
Algoritmo genético.**

TESIS

Que para obtener el grado académico de

Maestra en Ciencia y Tecnología en Ingeniería Industrial y Manufactura

POR

Dulce María Robles de León

SALTILLO, COAHUILA.

21 de Junio de 2018

**Simulación en planeación táctica de la producción tipo taller
mediante un Algoritmo genético.**

Por

Dulce María Robles de León

TESIS

Presentada al programa Interinstitucional en Ciencia y Tecnología

Sede

Corporación Mexicana de Investigación en Materiales S.A. de C.V.

Como requisito parcial para obtener el grado académico de

Maestra en Ciencia y Tecnología en Ingeniería Industrial y
Manufactura.

Programa Interinstitucional en Ciencia y Tecnología

COMIMSA

Saltillo Coahuila, 21 de Junio de 2018

**Simulación en planeación táctica de la producción tipo taller mediante un
Algoritmo genético.**

Por

Dulce María Robles de León

Comité Revisor

Director de Tesis / Tutor Académico:

Dr. Elias Gabriel Carrum Siller.

Profesor Investigador en Posgrado Interinstitucional en Ciencia y
Tecnología en Corporación Mexicana de Investigación en
Materiales S.A. de C.V.

Asesor:

Dr. Pedro Pérez Villanueva.

Coordinador General de Posgrado en Corporación Mexicana de
Investigación en Materiales S.A. de C.V.

Tutor de Planta:



Corporación Mexicana de Investigación en Materiales, S.A de C.V
Gerencia de Desarrollo Humano
División de Estudios de Posgrado

Los abajo firmantes, miembros del Comité Tutorial recomendamos que la Tesis
**“SIMULACIÓN EN PLANEACIÓN TÁCTICA DE LA PRODUCCIÓN TIPO
TALLER MEDIANTE UN ALGORITMO GENÉTICO.”**, realizada por el alumno
DULCE MARÍA ROBLES DE LEÓN matrícula **No. 1606IM1125** sea aceptada
para su defensa como Doctor(a)/Maestro(a) en Ciencia y Tecnología en
INGENIERÍA INDUSTRIAL Y DE MANUFACTURA.

El Comité Tutorial

Tutor Académico

Director de Tesis

Dr. Elias Gabriel Carrum Siller

Tutor en Planta

Ing. Héctor Gutiérrez Becerra

Asesor

Dr. Pedro Pérez Villanueva

Vo. Bo

Dr. Pedro Pérez Villanueva

Coordinación General de Estudios de Posgrado

COMIMSA

Corporación Mexicana de Investigación en Materiales

Gerencia de Desarrollo Humano

División de Estudios de Posgrado

Los abajo firmantes, miembros del Jurado del Examen de Grado del alumno, **DULCE MARÍA ROBLES DE LEÓN** una vez leída y revisada la tesis titulada **“SIMULACIÓN EN PLANEACIÓN TÁCTICA DE LA PRODUCCIÓN TIPO TALLER MEDIANTE UN ALGORITMO GENÉTICO”**, aceptamos que la referida tesis revisada y corregida sea presentada por el alumno para aspirar al grado de Maestra en Ciencia y Tecnología en **INGENIERÍA INDUSTRIAL Y MANUFACTURA** durante el Examen de Grado correspondiente.

Y para que así conste firmamos la presente a los 21 días del mes de Junio del año 2018.

Dr. Darwin Young
Presidente

Dr. Rolando Praga
Secretario

Dr. Elias Carrum
Vocal

Dedicatoria

Esta Tesis va dedicada primeramente a Dios por darme fuerza y salud para salir adelante en esta etapa de mi vida.

A mi familia, en especial a mi mami, sin su apoyo no hubiera podido perseverar en todos mis estudios, nunca me dejo sola y siempre estuvo ahí con sus consejos y amor como la madre que es; A mi pequeña Yuli quien fue mi fuente de inspiración para realizarme como mamá y profesionalista, porque sacrifique tiempo con ella y algunos de sus añitos, tiempo que valió la pena para poderle ofrecer lo mejor de mí y poner ese ejemplo a seguir; A mis hermanos, Lalito y Yaneth que también me apoyaron mucho.

A mi tutor, el Dr. Elias Carrum quien me brindo apoyo incondicional y siempre me supo guiar para no perderme.

Agradecimientos

Quiero agradecer a Dios por no dejarme vencer en el camino, a mi Tutor Dr. Elias Carrum que me tuvo paciencia y nunca me dejó rendirme, al apoyo de mi familia, ya que ellos se sacrificaron conmigo, a mis Maestros, Colaboradores y Doctores que me apoyaron en la tesis y materias, Dr. David González, Dr. Pedro Pérez, Dr. Rolando Praga, Dr. Melvin Álvarez, Mtra. Pilar Granillo y Lic. María Vega; Al Dr. Darwin Young por la orientación brindada al proyecto y excelentes consejos; A mis amigos y compañeros de COMIMSA que siempre estaban ahí para apoyarme incondicionalmente; A [REDACTED] por permitirme la toma de datos para poder llevar a cabo mi experimentación, a CONACYT por brindarme la oportunidad de realizar la maestría y especialmente a COMIMSA por ser un excelente Centro de Investigación, el cual cuenta con excelentes profesionistas y un personal competente, los cuales siempre fueron muy amables y me brindaron su apoyo.

Contenido

SINTESIS	1
ANTECEDENTES DE LA EMPRESA	2
INTRODUCCIÓN	4
CAPÍTULO I	6
1.1. PLANTEAMIENTO DEL PROBLEMA	6
1.2. OBJETIVO GENERAL	8
1.2.1 OBJETIVOS ESPECÍFICOS	8
1.3. PREGUNTAS DE INVESTIGACIÓN	9
1.4. HIPÓTESIS GENERAL	9
1.4.1 HIPÓTESIS ESPECÍFICAS	10
1.5 JUSTIFICACIÓN.	10
CAPÍTULO II	12
2.1. MARCO TEÓRICO	12
2.1.1 Estudio de tiempos y movimientos	12
2.1.2 Análisis de procesos	12
2.1.3 Técnicas del Job shop	13
2.1.4 Plan de producción	14
2.1.5 El Problema de programación Job Shop y métodos de solución	14
2.1.6 Sistemas de producción	15
2.1.7 Métodos de planificación	15
2.1.8 Tipo de talleres	15
2.1.9 UEPS.	18
2.1.10 Simulación	19
2.1.11 Algoritmo Genético	20

2.2 ESTADO DEL ARTE	25
2.3 MODELO PROPUESTO	46
2.4 ALCANCES Y DELIMITACIONES	48
CAPITULO III	49
3.1 METODOLOGÍA	49
CAPITULO IV	51
4.1 EXPERIMENTACIÓN	51
4.1.1 Toma de datos	51
4.1.2. Descripción del modelo	53
4.1.3. Elección de herramientas para el modelo	56
4.1.4 Simulación	61
4.2 RESULTADOS	67
4.2.1. Simulación Basal	67
4.2.2. Resultados de simulación optimizada.	72
CAPITULO V	80
5.1 CONCLUSIONES	80
5.2 TRABAJO FUTURO	82
5.3 RESULTADOS Y ENTREGABLES	83
5.4 APORTACIÓN CIENTÍFICA	83
5.5 APORTACIÓN TECNOLÓGICA	83
BIBLIOGRAFÍA	83
ANEXOS	93

Contenido de ilustraciones

Ilustración 1. Diagrama Ishikawa. Causa-efecto. (Elaboración propia).....	7
Ilustración 2. Pizarrón con la programación de la producción.....	8

Ilustración 3. Técnicas de optimización del Job Shop Fuente: (Elaboración propia)	13
Ilustración 4. Taller de una sola máquina. Fuente: (Santos García J. , 2007).	16
Ilustración 5. Taller de máquina en paralelo. Fuente: (Santos García J. , 2007).	16
Ilustración 6. Taller flow shop. Fuente: (Santos García J. , 2007).	17
Ilustración 7. Taller Job Shop. (Santos García J. , 2007).	18
Ilustración 8. Grafo del Job shop 4x3. Referencia: Ruíz, 2015	27
Ilustración 9. Metodología (elaboración propia)	49
Ilustración 10. Código de colores para cada caso (elaboración propia).	53
Ilustración 11. Variables de experimentación de caso 1 y caso 2.(elaboración propia).	53
Ilustración 12. Matriz con asignación de tiempo de operación para cada caso.(elaboración propia)	57
Ilustración 13. Matriz de asignación (2) de tiempo de operación para cada caso.(elaboración propia)	57
Ilustración 14. Matriz de variables (elaboración propia)	58
Ilustración 15. Resolviendo por Solver (3 casos).	59
Ilustración 16. Resolviendo por Solver (GRADIENTE).	59
Ilustración 17. Resolviendo por Solver (SIMPLEX)	60
Ilustración 18. Resolviendo por Solver (ALGORITMO GENÉTICO).	60
Ilustración 19. Resolviendo por Solver (ALGORITMO GENÉTICO).	61
Ilustración 20 Layout Conceptual del primer piso.	63
Ilustración 21 Layout conceptual de planta segundo piso	64
Ilustración 22. Tiempos y resultados de simulación de caso 1 y 12.	66
Ilustración 23. Job shop (tipo gantt) de caso 1 y 12.	66
Ilustración 24. Casos de estudio con su cantidad de piezas y tiempos de entrada al proceso.	67
Ilustración 25. Fecha en que comienza la simulación y el tiempo que transcurrió.	68
Ilustración 26. Resultado de simulación de los 13 casos.	69
Ilustración 27. Porcentaje de trabajo, esperas, etc... de los casos en cuanto a producción, transporte y el almacenado.	69
Ilustración 28. Tiempo promedio por caso.	69

Ilustración 29. Resultados de simulación basal con 19 casos.....	70
Ilustración 30. Porcentaje de trabajo, esperas, pausas, etc... de los resultados de 19 casos de estudio.	71
Ilustración 31. Resultados de simulación con 26 casos de estudio.....	72
Ilustración 32. Bosquejo de simulación.....	73
Ilustración 33. Tiempo total transcurrido en la simulación con los 13 casos con el caso 9 loteado en 3 partes iguales.	73
Ilustración 34. Resultados de simulación optimizada con 16 casos de estudio.....	74
Ilustración 35. El mejor individuo (secuencia óptima, en posiciones de acuerdo a cada caso).....	74
Ilustración 36. Gráfica de convergencia.	75
Ilustración 37. Resultados de simulación optimizada de 19 casos de estudio.....	76
Ilustración 38. Secuencia óptima de los 19 casos de estudio loteados.	76
Ilustración 39. Gráfica de convergencia de los 19 casos de estudio.....	77
Ilustración 40. Resultados de simulación optimizada de los 26 casos de estudio. ...	78
Ilustración 41. Secuencia optimizada de los 26 casos loteados.....	79
Ilustración 42. Gráfica de convergencia de los 26 casos de estudio.....	79
Ilustración 43. Matriz de tiempo por operación. Parte 1.	93
Ilustración 44. Matriz de tiempos por operación. Parte 2.....	93
Ilustración 45. Matriz de tiempos por operación. Parte 3.	94
Ilustración 46. Matriz de tiempos por operación. Parte 4.....	94
Ilustración 47. Matriz de tiempos por operación. Parte 5.....	95
Ilustración 48. Job shop real. Parte 1.....	95
Ilustración 49. Job shop real. Parte 2.....	95
Ilustración 50. Job shop real. Parte 3.....	96
Ilustración 51. Job shop mejora. Parte 1.....	96
Ilustración 52. Job shop mejora. Parte 2.....	96
Ilustración 53. Job shop mejora. Parte 3.....	97
Ilustración 54. Experimentación de herramienta MuPad de Matlab.	97

Contenido de tablas

Tabla 1. Tiempos por pieza en minutos por operación según el caso. (Elaboración propia).....	52
Tabla 2. Comparación de tiempos de simulación y porcentaje reducido.	82

SINTESIS

Planear la producción es de suma importancia hoy en día en la industria, por lo que en esta investigación se desarrolla una metodología para obtener una planeación de la producción con un énfasis en la reducción del tiempo de terminación de los pedidos, en donde dentro de la experimentación se utilizan distintos métodos y herramientas como un modelo matemático general para representar el problema, posteriormente éste se intenta resolver con ayuda de la herramienta Solver de Excel, donde no fue posible al utilizar Simplex ya que el problema no cuenta con las condiciones de linealidad necesarias, después se utilizó Gradiente en el cual el resultado obtenido fue de 35100 y por último Algoritmos genéticos, en donde la respuesta mejoró a 29700, por lo que en esta parte se valida el uso de algoritmos genéticos para optimizar el tiempo de terminación.

Se opta por utilizar un modelo de simulación en donde se pretende observar las deficiencias en el proceso, donde es posible identificar tiempos muertos, tiempos de espera, espera por material, cuellos de botella, etc., donde posteriormente en la misma simulación con ayuda de un módulo de Algoritmos genéticos, se logra optimizar la secuencia de entradas, para después simular ya con la respuesta mejorada y así obtener la simulación optimizada. La simulación toma en cuenta todas las variables y la realidad se representa sencillamente. El uso del algoritmo encontró la mejor solución de secuencia de operaciones que logro minimizar el tiempo de procesamiento de 29 días, 7 horas y 42 segundos que tomaba al utilizar PEPS a 14 días, 06 horas, 28 minutos y 26 segundos dividiendo en lotes y optimizando la secuencia de operaciones.

Se comprueba la hipótesis utilizando 13 casos de experimentación, reduciendo más del 50% (15 días de diferencia que es un resultado significativo), donde se aumentó el número de casos de experimentación para mejorar la comprobación de hipótesis.

ANTECEDENTES DE LA EMPRESA

La empresa Uniformes [REDACTED] empresa mexicana 100% textil fundada en 1992 se dedica a fabricar diferentes artículos textiles para Industria general, entre las cuales destacan: los uniformes deportivos (béisbol, futbol y basquetbol), escolares (se surte a 27 escuelas de Saltillo y Ramos Arizpe), uniformes deportivos, diario y gala e industriales (se surte diferentes artículos a 16 empresas de la región, como: batas, mandiles, chamarras, etc.).

Se fabrican distintos productos (gorras, bufandas, chaquetas, sudaderas, playeras con distintos tipos de tela, faldas, sacos, pantalones, pants, etc.) dependiendo del pedido, es posible bordar, tejer, pasar por el área de serigrafía o sublimado, de acuerdo al requerimiento del cliente, el cual, es el que describe como quiere que se realice su producto y en qué cantidad, el diseñador es el encargado de plasmar el pedido en papel (en algunos casos) para después ser procesado, por otra parte, el producto puede ir desde un personalizado (bordado) hasta una gran cantidad de sudaderas o playeras sublimadas.

Además, se distribuyen diferentes marcas a nivel nacional como: ropa industrial “Dickies”, camisas y pantalones “Air man”, gorras y sombreros “Dkp’s”, playeras “Yazbek”, playeras “Premium” y “Mayork”, sudaderas y pants “Gildan”.

Dicha empresa al igual que otras maquiladoras se cuenta con múltiples problemas en los cuales destacan tiempo en proceso elevados, las diversas piezas procesadas en una sola máquina, falta material, paros inesperados, utilización de los recursos, tiempos, máquinas, instalación, personas, etc., que tienen una capacidad colocada la cual no se encuentra en provecho de la mejor manera, esto se debe a un mal estudio de capacidad y se desconocen los beneficios tanto productivos como financieros que es posible aportar en el mejoramiento de la empresa.

La especificación de la capacidad dentro de la empresa, es de suma importancia para saber si la demanda será o no satisfactoria a partir de la reserva tecnológica con que cuenta una organización. (Pozo, 2015).

Otro punto importante es al momento de conocer cómo se debe administrar la empresa, donde la clave está en observar que es lo que el cliente busca y así reducir tiempos aumentando la capacidad de producción(Guamanquispe, 2011).

Un problema sobresaliente en las empresas que trabajan bajo pedido sería que al no detallar sus procesos con números, les es difícil comprenderlos, y por ende, no pueden controlarlos. Es necesario realizar un análisis de datos estadístico para comprender al producto o proceso. (Vazquez, 2005)

Una opción para resolver este tipo de problemas es utilizar planeación agregada, donde se puede conocer los aportes de algunas técnicas para optimizar, las cuales permiten obtener información de la forma de trabajar dentro del proceso, por mencionar un ejemplo, un modelo de programación lineal ayuda a bajar costos de producción y aumentar la productividad. La programación lineal es una técnica que constantemente innova y que adjunta cada una de las variables que se involucran dentro del proceso, para después optimizar a fondo cada uno de los medios manejados por la empresa para fabricar un producto y así obtener un plan de producción(Molina, 2014).

INTRODUCCIÓN

En la actualidad la Industria en general ha tenido un aumento significativo en la competitividad de la misma, en especial la industria textil que tenido crecimiento de un 10% desde el 2017, según el Presidente de la cámara de la industria textil.

Con el fin de cumplir con la demanda por la alta competitividad, en los años 80's nace el S&OP (Sales and Operation Planning) o el planeamiento de ventas operaciones, cuyo trabajo es integrar a todas las áreas de una organización desde el marketing hasta la distribución final del producto para obtener el control de la producción.

El S&OP es un proceso para desarrollar los planes tácticos, se compone de la administración de la demanda, Plan Maestro de Producción y los requerimientos para los recursos (tiempo disponible para la producción, maquinaria, almacén, proveedores, etc...).

En este proyecto se pretende llevar a cabo un plan de producción con el fin de reducir tiempo de terminación de los pedidos solicitados.

La planeación de la producción dentro de la industria es una serie de pasos que son utilizados para la organización de las áreas de trabajo con los recursos que cada una tiene, para con ello, obtener el mejor provecho posible para la empresa.

En algunas ocasiones el plan de la producción es incorrecto por considerar mayor capacidad en el proceso a la que realmente es. Por eso, es importante realizar re planificaciones. Es provechoso situar de esta información al recurso que es el que detiene la producción, es decir, el cuello de botella. (Santos García J. , 2007).

Un ejemplo de problema de plan de producción es la programación para problemas tipo taller, el cual es uno de los más mencionados en la literatura de optimización combinatoria, siendo este el más complicado. No se ha publicado el algoritmo de solución a resolver, como dicho problema de optimización en tiempo polinomial. Esto ha llevado a un interés actual en el uso de algoritmos genéticos para abordar el problema.

Por otra parte, Schroeder (2004), menciona que la administración de operaciones con sus 4 cuestiones importantes como capacidad, localización, proceso y distribución, en donde tanto la capacidad como la distribución dan lugar a la configuración productiva del sistema, tienen que ver con la decisión. Algunos autores, como (Domínguez M, 1995), Schroeder (2004), (Alfalla L, 2003), entre otros; han dado a conocer la existencia de 4 clases de

configuraciones genéricas las cuales son: por proyecto, orientada al proceso o funcional (como por ejemplo el Job-shop), la que se enfoca en el resultado, que es el producto (que es el flow-shop) y las híbridas.

Osorio (2008) menciona en su artículo lo que conlleva trabajar con el problema específico tipo taller se lleva a cabo con un grupo de trabajos y otro de máquinas, en donde cada proceso consiste en desarrollar una secuencia de operaciones que se tienen que procesar en alguna máquina durante cierto tiempo. No se puede realizar más de una operación por máquina en un tiempo. Por su parte, la programación se define como la determinación de los tiempos en que comienza el proceso y en el que termina, con sus respectivos recursos; En este caso el problema consiste en hallar una metodología o programación con el fin de satisfacer la terminación de todos los trabajos en el mínimo de tiempo posible. (Osorio Gómez, 2008)

Dentro de la industria de manufactura, el tiempo que se ocupa en terminar todos los trabajos es considerado como la diferencia de tiempo entre el inicio y el fin de una secuencia de trabajos o tareas. Si no se reduce el tiempo de terminación de algún producto, el pedido no será entregado en tiempo acordado. (Skorin-Kapov, 1993).

Al minimizar los tiempos tanto de flujo como finales, retrasos, trabajos sin completar, etc. Se convierten en los criterios más objetivos dentro de la programación de problemas tipo taller, siempre y cuando se cuide al cliente (Domínguez Machuca y otros, 1995). Si a todo lo mencionado no se le da un mínimo valor, se puede encontrar una asignación de un factor para cada criterio (Jain y Meeran, 1998).

CAPÍTULO I

1.1. PLANTEAMIENTO DEL PROBLEMA

Actualmente la empresa Uniformes [REDACTED] cuenta con 11 áreas de producción las cuales son: Diseño (con 2 computadoras para utilizar en diseño, tejido, vinil y bordado), Plotter (impresión de papel, donde en dicha área se cuenta con 3 máquinas: plotter 1, 2 y 3), Corte de tela(cuenta con 2 procesos diferentes: corte manual que se utiliza para pocas piezas y corte automático que se utiliza para un gran número de piezas, Serigrafía (con 2 máquinas llamadas pulpos, uno manual y otro automático que incluye un horno de secado), Sublimado (con una sola máquina: calandra), Bordado(cuenta con 2 máquinas multi-bordado, es decir, cada máquina cuenta con 5 bordadoras), Tejido(con 3 máquinas tejedoras) , Costura (con 5 máquinas, las cuales 4 son utilizadas comúnmente y una más de repuesto), Láser(con una máquina la cual opera la misma persona de Diseño), para el área de Detallado, se utiliza a la persona que realiza el aseo de la empresa para realizar el detallado de las piezas que salen de las máquinas tejedoras o de las piezas que salen de costura y por último el área de Empaque cuya labor la realiza la persona que está encargada del área de ventas.

El área de diseño con 1 sola persona que se encarga de crear los pedidos del cliente la cual aparte de diseñar también se encarga del área de láser, el área de plotter cuenta con 1 persona que se encarga de las impresiones del plotter y corte de papel al igual que el área de corte de tela que cuenta con 1 persona encargada de cortar toda tela que se convierta en pedido y por último, tejido que cuenta con una sola persona que a su vez se encarga del mantenimiento de la empresa; El área de serigrafía con 2 personas las cuales también se dedican a el mantenimiento de la empresa, sublimado con 3 (incluyendo el supervisor de área), bordado con 2 y costura con 4 personas.

Con el fin de mejorar algunas de las operaciones antes mencionadas, se realizó un diagrama Ishikawa o diagrama de causa-efecto para analizar las problemáticas en la empresa.

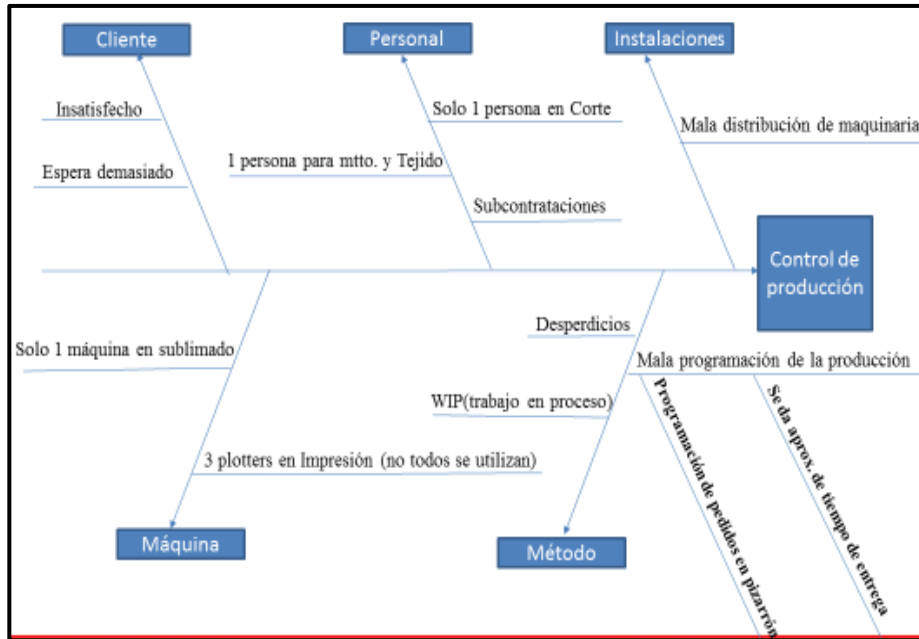


Ilustración 1. Diagrama Ishikawa. Causa-efecto. (Elaboración propia)

El proceso de pedidos en la empresa se realiza de la siguiente manera: llega el cliente a recepción, hace su pedido, el cual es calendarizado en un pizarrón para dar seguimiento y acomodarse lo más pronto posible para ser producido.

Actualmente, la empresa ████ trabaja con una calendarización de acuerdo con un sistema bajo pedido PEPS (Primeras Entradas, Primeras Salidas) donde los primeros pedidos que entran a ser procesados, son los primeros que salen, y los demás se quedan en fila de espera.

De acuerdo con su forma de calendarización, el plan de la producción se realiza en base a los pedidos ya realizados y anotados en el pizarrón que se encuentra en la *Ilustración 2*, donde se van eliminando de éste las órdenes que ya salieron de producción, por lo que la forma de programación de la producción es poco eficiente, ya que lo hacen por medio de la planeación bajo pedido (PEPS).

Paquete	Fecha Cierre
121 ch sin cue	10-Febrero-2016
120 ch 44 Greta	11-Febrero-2016
100M ⁴⁴ ₄₀ ⁴⁴ ₄₀	11-Febrero-2016
100M T.M	13-Febrero-2016
100M T.V	13-Febrero-2016
100M	14-Febrero-2016
100M T.M	15-Febrero-2016
100M ⁴⁴ ₄₀ ⁴⁴ ₄₀	15-Febrero-2016
100M	17-Febrero-2016
100M	
100M	
100M	
100M	
100G T.N	16-Feb.
100G	

del Mes	Calendario	Fecha Termina
1 Frente Tap Mangas	10-02-16 GB	
3 Espaldas 2 par Mangas	11-02-16 GB	
	12-02-16 GB	
	13-02-16 GB	
	13-02-16 GB	
	14-02-16	
	15-02-16	
	15-02-16	

Ilustración 2. Pizarrón con la programación de la producción

Debido a una mala planeación de la producción, las áreas de corte de tela y sublimado son las que tienen mayor cantidad de trabajo ya que estas proveen de material al área de costura, el cual no puede trabajar si no tiene material para coser.

Si se logra planear correctamente la producción, se obtiene como ventaja principal el conocer la fecha de conclusión de los pedidos ya hechos, dando así al cliente, un tiempo preciso de entrega de su pedido, donde si esto no se cumple, la relación entre la empresa y el cliente se estropea y la consecuencia a esto son los costos de envío como multa al no cumplir con la fecha acordada.

1.2. OBJETIVO GENERAL

Desarrollar un plan de producción para encontrar la secuencia óptima que minimice el tiempo de terminación.

1.2.1 OBJETIVOS ESPECÍFICOS

- Simulación del proceso actual.

- Usar el simulador como función de evaluación mediante un algoritmo genético.
- Optimizar la secuencia de entradas.
- Implementación de interfaz

Simular el proceso actual.

Se crea una simulación de la situación actual de la empresa, según la planeación que utilizan, donde actualmente trabajan con un sistema bajo pedido, PEPS (primeras entradas, primeras salidas), es aquí donde se pueden apreciar las deficiencias dentro del proceso.

Simular el proceso actual con un algoritmo genético para optimizar.

Con la simulación de la situación actual de la empresa, ya creada, se implementa un algoritmo genético, ya integrado en el software, para poder optimizar la secuencia de operaciones, eligiendo la mejor secuencia que minimice el tiempo de terminación.

Optimizar la secuencia de entradas.

Realizar el plan de producción por medio de un algoritmo genético va a indicar la secuencia de caso a procesar y así lograr minimizar el tiempo de terminación, reduciendo tiempos de espera, tiempos extra, etc...

1.3. PREGUNTAS DE INVESTIGACIÓN

- 1- ¿Qué técnicas de optimización existen para encontrar la secuencia óptima?
- 2- ¿Qué herramientas se pueden utilizar para encontrar deficiencias en el proceso?
- 3- ¿Mediante simulación, es posible representar el proceso, lo más real posible comparando con el modelo matemático?
- 4- ¿Qué ventaja se logró al utilizar Algoritmos genéticos?

1.4. HIPÓTESIS GENERAL

Mediante la simulación en planeación táctica de la producción tipo taller mediante un algoritmo genético se podrá reducir el tiempo de terminación en al menos un 25%.

1.4.1 HIPÓTESIS ESPECÍFICAS.

- Mediante el uso de simulación se podrá obtener una mejor representación del proceso que usando un modelo matemático.
- El uso de Algoritmos genéticos, planteados en la simulación, es capaz de generar una solución factible para el caso de estudio.

1.5 JUSTIFICACIÓN.

Al obtener el plan de producción, es preciso detallar tiempos asignados a cada operación, las personas encargadas de llevar a cabo cada tarea, las piezas de salida del proceso, los porcentajes de eficiencia y productividad de personal o de maquinaria, etc. Es la que indica:

- ¿Qué producir?
- ¿Cuánto producir?
- ¿Cómo producir?

Al determinar la planeación de la producción, es posible reducir tiempos muertos.

A continuación, se muestra una tabla con un análisis a 16 días (días que fueron requeridos por la empresa para concluir la producción de los pedidos mencionados en la experimentación).

El análisis real de la empresa con una planificación tipo FIFO (primeras entradas, primeras salidas), la cual tiene como restricción el no producir el siguiente pedido sin haber concluido el anterior, muestra las horas trabajadas por área en los 16 días, donde con ayuda de los datos, se puede observar que el cuello de botella se encuentra en el área de sublimado y corte, ya que dichas áreas cuentan con los tiempos de terminación más elevados, afectando así sus áreas precedentes con tiempos de espera prolongados.

Tabla 1. Tiempo trabajado por área a un estudio de 16 días, según el resultado del diagrama de Gantt

Personas	Máquina	Hrs	Días	Días Trabajados	Hora extra pagada
1	PC1	2,5	1	16	0
	Láser	70	9		
1	Corte Automático	121,5	15	16	14
	Corte Manual	0,5	1		
1	Plotter 1	62,5	7	16	0
	Plotter 2	56,5	7		
	Plotter 3	3,5	1		
2	Sublimado	115,5	14	16	10,5
1	PC2	2,5	1	16	0
	Tejedora 1	78,5	9		
	Tejedora 2	10	2		
	Tejedora 3	11,5	2		
1	Detallado	7	2	16	0
1	Costura 1	89	13	16	9
1	Costura 2	84	11	16	0
1	Costura 3	82	11	16	0
1	Costura 4	75,5	11	16	11,5
1	Bordado 1	70	10	16	0
1	Bordado 2	71	10	16	0
1	Empaque	8	10	16	0
				Piezas Ela	
Pieza	Maquila 1	72	9	446	
Pieza	Maquila 2	53	7	296	
Pieza	Maquila 3	50	6	196	
Pieza	Maquila 4	49	6	196	
Pieza	Maquila 5	49	6	196	

CAPÍTULO II

2.1. MARCO TEÓRICO

En el marco teórico se abarcan definiciones básicas, complementarios, específicos e importantes que se utilizaron en la metodología, los cuales ayudarán a comprender cada parte del proyecto y analizar su desarrollo, donde como primer paso se tiene la comprensión del estudio de los tiempos y movimientos y las diferentes técnicas de Job shop que se encontraron en la literatura, el problema en específico y sus métodos de solución; La simulación con sus ventajas y desventajas; La última parte del proyecto que es la optimización mediante Algoritmos genéticos, se detallan conceptos y su pseudocódigo.

2.1.1 ESTUDIO DE TIEMPOS Y MOVIMIENTOS

Para Molina (2014) el estudio de tiempos y movimientos es una técnica que beneficia a través de la precisión a un tiempo estándar donde se toma en cuenta la fatiga, demoras personales y retrasos inevitables.

Las técnicas y habilidades disponibles junto con el estudio de tiempos y movimientos deben de ir adjuntas para lograr eficiencia en la relación hombre-máquina. Ya determinado el método el trabajo realizado tiene como cargo, el determinar el tiempo que requiera cada tarea, verificar que las normas o estándares se cumplan y que los trabajos se recompensen.

Los objetivos principales del estudio de tiempos y movimientos son:

- Minimizar el tiempo necesario en los trabajos.
- Ahorrar los recursos y reducir costos.
- Reducir los movimientos innecesarios y acelerar los eficientes.
- El método a experimentar debe haberse estandarizado anteriormente.

2.1.2 ANÁLISIS DE PROCESOS

En los últimos años se ha investigado la pregunta ¿Qué es análisis de procesos? En cuanto a la investigación de operaciones, de la mano de los conceptos de: modelo, sistema y variable.

Estos conceptos se engloban en el análisis de proceso, que está referido a la aplicación de ciertos métodos y al puntualizar problemas como parte del progreso de procedimientos para dar solución. (Bischoff, 1997)

2.1.3 TÉCNICAS DEL JOB SHOP

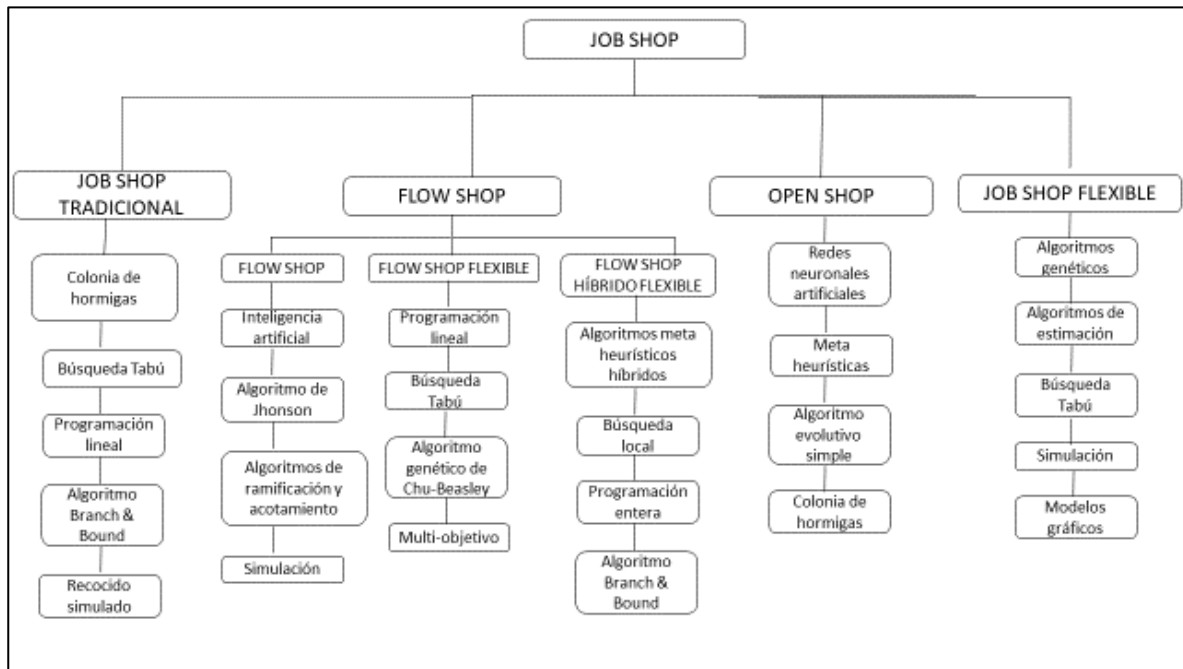


Ilustración 3. Técnicas de optimización del Job Shop Fuente: (Elaboración propia)

La *ilustración 3* muestra la comparación de 3 tipos de Job Shop que son Job shop tradicional, Flow shop, Open shop y Job shop flexible. Del Flow shop se dividen 3 tipos: Flow shop normal, Flow shop flexible y Flow shop híbrido flexible, los cuales cuentan con distintas técnicas de optimización para su solución.

Según las características de cada clasificación del Job shop, la más apropiada para utilizar en este trabajo es la del Job shop tradicional, ya que sus tareas no pasan obligatoriamente por todos sus procesos en el mismo orden, en cambio, en el Flow shop las operaciones pasan por todos los procesos o maquinaria para llevarse a cabo en un mismo orden, el Open shop clasifica por sistemas con 2 tipos de máquinas, no trabaja en serie y no hay restricciones de categorización respecto a las operaciones y Job shop flexible, existe

flexibilidad dentro del proceso donde no es relevante el número de tarea u operación a ser procesado.

2.1.4 PLAN DE PRODUCCIÓN

Para González (2006) el plan de producción tiene como fin determinar el costo de la producción en conjunto con costos de ventas; El almacén es el encargado de poseer la diferencia entre la producción cuando ésta supera a la demanda y, al contrario, que provea producción cuando la demanda baje. Es de vital importancia que se cuente con los recursos necesarios como materia prima, mano de obra, maquinaria, etc... enfocadas en planificación de recursos que se subdividen en: aumento de capacidad futura o redistribución de exceso de carga.

2.1.4.1 PCP (PLANEACIÓN Y CONTROL DE LA PRODUCCIÓN)

González (2006) menciona que los sistemas que requieren de información, es necesario darles seguimiento para verificar desvíos u otros manejos de información, determinándose esto: Gestión de la producción asistida por ordenador (GPAO) la cual es de gran importancia en empresas de producción, esto refiere a los sistemas ERP (Enterprise Resource Planning). En cualquier sistema siempre se tendrán entradas, algún tipo de proceso y salidas.

2.1.5 EL PROBLEMA DE PROGRAMACIÓN DE PROBLEMAS TIPO TALLER Y MÉTODOS DE SOLUCIÓN

Escamilla Fuster (2016) describe que un problema de programación de problemas tipo taller se compone de n trabajos, donde cada uno consta de una secuencia ordenada de operaciones. Por restricción se tiene que cada operación se debe realizar utilizando una de las M máquinas existentes en el problema. El proceso de la operación j en la máquina i es una operación (i, j) y su duración es p_{ij} . Para obtener un plan rentable se debe determinar un tiempo inicial a todas las operaciones, tomando en cuenta las siguientes restricciones:

- Las operaciones de cada trabajo deben estar realizadas según su secuencia.

- Cada máquina puede hacer una operación al mismo tiempo.
- No hay prioridad en las operaciones.

Otra parte importante dentro del problema es el objetivo en la programación, el cual lleva consigo el tiempo de procesamiento de los trabajos. Uno de los objetivos más utilizados es minimizar el tiempo de terminación, con el fin de completar todos los trabajos en el menor tiempo posible (tiempo de terminación C_{\max}).

2.1.6 SISTEMAS DE PRODUCCIÓN

Estos sistemas actúan como procesos de conversión por lo que los inputs o entradas son convertidos en productos o servicios (outputs o salidas) que sirven para clientes o usuarios.

2.1.7 MÉTODOS DE PLANIFICACIÓN

Con el objetivo de mejorar el uso de recursos para evitar caer en un mal plan de la producción y a su vez reducir gastos se pueden utilizar diversos métodos.

Por una parte, Elsayed y Boucher (1985) dicen que para obtener un buen plan de la producción se puede utilizar la planificación agregada lineal siempre y cuando la función es descrita de la mejor manera y se incluyen todas las limitaciones, donde la respuesta obtenida es mejorada.

Velázquez (2008), por otro lado, menciona que para el plan de la producción es importante definir los términos de carga y capacidad:

Se define como carga a la adición de tiempos tanto el de proceso como el de permutación entre trabajos que son establecidos a cada máquina. La capacidad es un tiempo del cual se sitúa algún medio para llevar a cabo una operación. También es posible reconocerla fácilmente, ya que será una máquina con suficiente inventario que espera para ser procesado.

2.1.8 TIPO DE TALLERES

Santos García (2007) redacta algunos conceptos de talleres con cierto número de máquinas, como el taller de una máquina, el cual consta de una sola tarea realizada en una sola máquina, por esta razón es posible obtener el mejor secuenciamiento de tareas. En algunas

empresas no se da esta opción, por lo que también es posible congregarse varias máquinas en estaciones de trabajo (celdas).

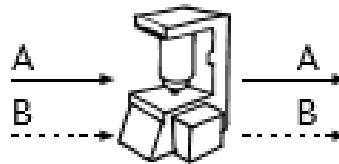


Ilustración 4. Taller de una sola máquina. Fuente: (Santos García J. , 2007).

Otro tipo de taller es el de máquinas en paralelo, el cual consta de m máquinas iguales que procesan n trabajos que estén disponibles en el momento. A veces es asignado algún pedido a máquinas distintas por el tiempo de procesamiento o costos distintos.

Para resolver problemas que se presentan en el control de los distintos tipos de talleres, actualmente existen muchas técnicas de optimización como algoritmos, por ejemplo, el algoritmo de asignación o de Khun, el cual también se puede utilizar con algún método de transporte.

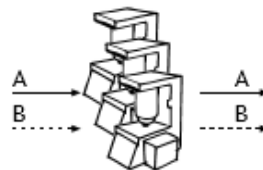


Ilustración 5. Taller de máquina en paralelo. Fuente: (Santos García J. , 2007).

Las reglas de despacho son otro método de solución a los problemas de talleres, ya que conceden conceptualizar las preferencias entre las tareas que pueden hallarse en el taller. Las reglas de despacho son simples si refieren del producto, donde por ejemplo se pueden encontrar algunos tiempos como el de proceso o el de la entrega de pedido; también es posible obtenerlos a través de ecuaciones entre distintas variables (como la holgura). Las más comunes reglas de despacho que se manipulan en piso son:

- FIFO (First In First Out) o PEPA (Primero en Entrar, Primero en Atender): Se emplea muy seguido y, normalmente, en piezas transitorias, donde se designa FEFO (First Expiration First Out).

- LIFO (Last In First Out) o UEPA (Último en Entrar, Primero en Atender): No popular, más sin embargo, cuando la materia prima se encuentra ocupando gran territorio con producción elevada (maquinaria de amplio tamaño) suele ser muy utilizado.
- SPT (Sort Process Time): Clasifica las tareas desde el mínimo tiempo de proceso al mayor. De las mayormente utilizadas.
- LPT (Longest Process Time): Lo contrario al SPT, va del mayor al menor.
- EDD (Earliest Due Date): Clasifica los trabajos de acuerdo a la fecha en que se entrega el pedido, de forma gradual, es decir, el primer pedido de la lista es el primero en entregarse.
- Holgura mínima: Considera el tiempo sobrante total hasta que se termine el trabajo. De esta forma los trabajos son programados con anterioridad, y así con mayores posibilidades de retrasarse.

2.1.8.1 EL FLOW SHOP

La condición del Flow shop es que las máquinas estén de acuerdo con que el flujo de las piezas tenga un solo sentido.

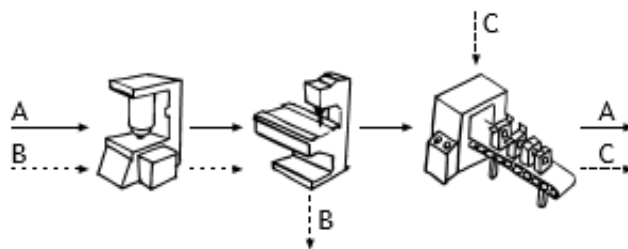


Ilustración 6. Taller flow shop. Fuente: (Santos García J. , 2007).

Este tipo de problema es de problema N-P duro, ya que su representación es combinatoria.

Las posibles combinaciones para la programación del mismo son: $m * (n!)$ si los n trabajos son procesados en las m máquinas, es decir, si se tiene un plan de 10 trabajos en 5 máquinas se dispone más de 2.000.000 alternativas distintas. Actualmente el flow shop solo se ha resuelto utilizando 2 máquinas y el algoritmo que ayuda a minimizar el tiempo de terminación es el Algoritmo de Johnson.

2.1.8.2 EL JOB SHOP

Esta filosofía se caracteriza por tener m máquinas y cada trabajo n que se programe, fluye por una ruta distinta, o también es posible que un trabajo se opere por más de una vez en una misma máquina en 2 etapas diferentes de un proceso. (Santos García J. , 2007).

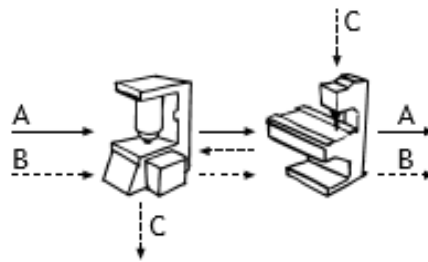


Ilustración 7. Taller Job Shop. (Santos García J. , 2007).

2.1.9 UEPS

Últimas entradas, primeras salidas.

Admite que el inventario que se proporcionó al final, sea el primero en salir o ser vendido, es decir, que este último inventario es valorado a los precios más viejos y el precio del artículo ya vendido, con los precios actuales.

PEPS es mejor que UEPS en cuanto a costos más bajos y mayor utilidad, el promedio ponderado se encuentra entre PEPS y UEPS. En economía de precios incrementados, las empresas pagarán menos impuestos que si utilizarán UEPS. (Jiménez & Espinoza, 2007).

2.1.10 SIMULACIÓN

Simulación es una técnica cuyo objetivo es llevar a cabo experimentos muestrales con el fin de modelar un sistema, donde el conjunto de variables viene siendo el modelo, que a la par de ecuaciones se relacionan, y cuentan con restricciones.

La experimentación dentro de la simulación involucra costos altos, lentitud y pruebas destructivas, a continuación se muestran algunas ventajas y desventajas del uso de simulación (Cao Abad, 2002):

Ventajas:

- En caso de que la solución analítica no sea efectiva.
- Si se desea experimentar antes de llevar a cabo algún proyecto.
- Es útil en sistemas que se desarrollan lánguidamente en el tiempo.

Desventajas:

- La construcción es hacendosa.
- El modelo omite variables importantes.

La simulación es una herramienta que mediante un análisis de sensibilidad valida un modelo, realiza experimentos con el fin de conocer el proceso y las estrategias para controlarlo o tener intervención con el mundo real.

En otra opinión, Ruíz y Oregui (2002) las ventajas de la simulación son:

- Hacer posible un análisis de un sistema bajo situaciones experimentales que no son factibles, en costo.
- Hacer más fácil el estudio a largo plazo.
- Permite anexión de variables de incertidumbre.
- Trabaja en conjunto con un alto número de variables.
- Genera un número significativo de resultados y datos en poco tiempo.

Conceptos:

- Variable: Medida de un atributo, está en una función matemática, se clasifica en exógena (independiente del sistema) y endógena (generada por el sistema).
- Parámetro: Cantidad de variables que se hacen constantes.

Clasificación de modelos de simulación:

- Discretos o continuos: Es referido al tipo de variable que se usa en el modelo, caracterizadas por tomar cualquier número real, en cambio la discreta es restringida a valores específicos o tiene ausencia de atributos.
- Determinísticos o estocásticos: Modelo a partir de un grupo de variables terminantes, y modelo estocástico es el que involucra elementos con incertidumbre y aleatoriedad en el sistema.
- Estocásticos o dinámicos: Dinámicos, si los valores de sus variables están cambiando constantemente en el tiempo y los estocásticos no cambian a través del tiempo.
- Con o sin retroalimentación: Sin retroalimentación es cuando no se tiene supuesto el valor de salida del modelo que no vuelve como valor de entrada, modificando la salida.

Para el mismo fin se definen las etapas que definen el modelo de simulación: definición de objetivos, análisis de sistema, síntesis de sistema, verificación, validación, experimentación y documentación.(Parra, Rodriguez, & Beltrán, 2005).

2.1.11 ALGORITMO GENÉTICO

El paradigma neodarwinista de la evolución dice que el proceso de evolución en las especies contiene 4 pasos importantes: reproducción, mutación, competitividad y la selección, todos sustentados en estas palabras: “supervivencia del más capaz”.

Dicho paradigma ha recurrido a computadoras digitales, en donde se ha tratado de simular la evolución de ciertas especies, reproduciendo en poco tiempo para la máquina lo que a la naturaleza ha tomado tanto tiempo desarrollar.

Es por esto que es importante resaltar que los primeros empleos de simulación dentro de sistemas genéticos fueron los realizados por el australiano A. S. Fraser en el campo de la biología. Fraser realizó un trabajo sobre la epítasis y los sistemas naturales, pero supuestamente no fue considerada la posibilidad de aplicar su metodología a sistemas artificiales, Friedberg no probó que el sistema funcionará simulando los principios de la

evolución natural pero es de suponerse que fue aceptado. Por otro lado, sus coautores Dunham y North indicaron que la aplicación para la solución de problemas de la evolución natural resulto ser una herramienta más poderosa de lo que se pensaba, llamándose así “Computación evolutiva”.

La computación evolutiva surge a partir de los trabajos de Bremermann, Fraser, Friedberg, y otros (Cziko, 1995) en los sesentas. La ciencia en los equipos de cómputo no permitió que los algoritmos se esparcieran para fines científicos sino hasta casi treinta años después. En 1970 los trabajos separados de Holland, Rechenberg, Schewefel y Fogel permitieron que se desarrollara dicha área de tal manera que ahora contamos con varios algoritmos evolutivos tales como los Algoritmos genéticos, las estrategias evolutivas y la programación genética. Durante los ochentas el desarrollo en la computación permitió que los algoritmos evolutivos fueran utilizados para resolver problemas del mundo real significativamente complejos, sobre todo los que tienen que ver con optimización. Las soluciones ofrecidas fueron ampliamente divulgadas en parte por la ciencia de las mismas sobre otras técnicas convencionales y por otro lado por el bajo costo computacional que involucraban.

Lo que comprende a la computación evolutiva es el proceso de la evolución natural, es decir, el proceso continuo que da lugar al surgimiento de estructuras muy complejas y adaptadas a su medio (Darwin, 1997). El algoritmo evolutivo es una similitud con la evolución natural y se ve como un proceso de búsqueda. Los procesos evolutivos perturban a poblaciones de organismos, por lo que los algoritmos evolutivos laboran con poblaciones de soluciones, todas a la vez. Se han reconocido en el proceso evolutivo tres formas generales que son aplicados a todos los organismos:

- Evaluación
- Selección
- Variación

El proceso de evaluación establece la medida de la capacidad de reproducción que tiene un individuo (o solución, respetando la analogía) y el cual se representa por medio de un grado o rango.

Como segunda parte, el proceso de selección ordinariamente es no determinista, donde solamente sobreviven los individuos más rápidos y fuertes, tienen más posibilidad de heredar sus atributos y dejar descendencia. En la naturaleza la selección natural se aplica por el entorno en que se desarrolla el individuo y por la constante lucha que tiene que hacer con otros individuos por los recursos del entorno.

Y por último, el proceso de variación está incluido en la reproducción entre los individuos de la población en donde hay un intercambio de características por lo que el producto (o sus hijos) comparten atributos similares de cada uno de sus padres. (Torres, 2006).

Unos de los principales autores en aplicar la computación evolutiva es Hans Bremmerman, porque su investigación y desarrollo estuvo enfocado a utilizar cada uno de los principios de la evolución natural como los medios de optimizar, donde el autor no solo utilizo los términos de “aptitud”, “selección”, “mutación”, “población”, y “genotipo”, que era raro que se llevaran a cabo para resolver algún problema, confirmó que la evolución biológica es parte del proceso de optimización. Si la evolución natural es considerada como tal, porque ha llegado a optimizar a los organismos volviéndolos fuertes para sobrevivir, entonces, es posible realizar un ajuste en la evolución para encontrar una aún mejor respuesta y llegar a la mejor de las mejores soluciones, es decir, la solución óptima.

Parte importante dentro de la solución de problemas en la computación evolutiva, son los Algoritmos genéticos, tales que hoy en día son muy comunes alrededor del mundo, quizá por su generalidad y sencillez conceptual, que les proporciona una gran disposición de uso y se aplican a un gran número de problemas.

Las partes que componen los Algoritmos genéticos comienzan por la población (P), la cual es una parte esencial dentro de un algoritmo de la computación evolutiva, que es un grupo de buenas respuestas, donde la medida de la población cambia o varía cada que pasan algunas generaciones, pero normalmente permanece sin cambios. Las partes que componen a la población son denominadas individuos, de los cuales, su estructura es según se domine al problema, lo cual quiere decir que dependiendo del problema es cómo se ideara una representación. Este problema es mayormente visto en los Algoritmos genéticos que en las Estrategias Evolutivas debido a que en las estrategias no es necesaria una codificación y en los Algoritmos si la es. Las Estrategias Evolutivas y los Algoritmos genéticos son métodos

que podemos detallar como débiles en la terminología de la inteligencia artificial. Esto es, la forma de trabajar de estos algoritmos no es dependiente del dominio del problema. Todo problema debe tener como característica una medida que compare soluciones que compiten. La aptitud es un mecanismo que se deriva de la acción de dominar el problema, la cual le da un valor equitativo a cada organismo de la población para que se represente la solución de calidad. Una mejor solución para un problema es un individuo con mayor aptitud, dependiendo de las circunstancias, este puede dar un resultado correcto u hasta óptimo. Un individuo con carencia en aptitud se caracteriza por una peor solución. Sería ideal lograr que el valor de aptitud y la solución representada por cada individuo se relacionen altamente, esto es posible con una correcta codificación para representación del problema y a través de una acertada función de aptitud.

Después de la población, la segunda parte del proceso de evolución en un Algoritmo genético es la selección, la cual se encarga de elegir por preferencia a las entidades con más grado de aptitud dentro de la población. El nivel en el que se seleccionan los más aptos individuos se define como presión de la selección. Luego se presenta la recombinación, que se basa en intercambiar información entre los organismos de una población, mientras que la mutación provoca variantes aleatorias en los individuos. Los organismos vivos pueden ser percibidos como una dicotomía entre su información genética, llamada genotipo. En comparación con las Estrategias, los Algoritmos genéticos evolucionan a los individuos a través de sus genotipos, que por el contrario las Estrategias evolutivas hacen lo propio a través de sus fenotipos. Por esta razón es que en los Algoritmos es común encontrar que los individuos de una población sean codificaciones, casi siempre binarias, de la solución de un problema, en cambio, en las Estrategias evolutivas los individuos no son codificaciones sino fenotipos, es decir, un organismo está formado a partir de las variables que se busca optimizar. Ejemplo de ello son la recombinación y la mutación. La recombinación es de vital importancia para los Algoritmos y la mutación para las Estrategias. En los Algoritmos, la búsqueda prospera por medio de la recombinación del material genético de los individuos más aptos mientras que en las Estrategias la búsqueda progresa por medio de su mutación. Otra diferencia significativa es que en los Algoritmos la selección es aleatoria pero en las Estrategias es determinística. (Hernandez Aguirre & Buckles P. Bill, 2017)

Para la resolución de problemas, se utiliza la programación de Algoritmos genéticos, la cual es posible adaptar al problema específico que se presente, a continuación se presenta el pseudocódigo de programación de un Algoritmo genético simple.

Algoritmo Genético. Pseudocódigo:

P ← Creates (NTI, NTB)

For i=1: NTGen

Fe ← Evaluation (P)

PS ← Selection (Fe)

P ← Variation (PS)

end

El algoritmo genético se compone de diferentes partes que a continuación se explican:

En la primera parte del pseudocódigo: P ← Creates (NTI, NTB), se crea la población de acuerdo al número total de individuos y la representación fenotípica o el número total de bytes.

Para comenzar el ciclo, se coloca for, donde “i” representa cada generación.

En la parte de evaluación, se califica a la población con el fin de obtener la función de evaluación, en este caso en particular el tiempo de terminación, hasta obtener todas las posibles combinaciones de secuencia.

En la parte de selección, se elige a la mitad de la población (por ejemplo si son 100 individuos solo selecciono 50, que serían los mejores), si el objetivo de la función es maximizar escojo los mayores tiempos, si es minimizar, que es el caso en particular, escojo los menores tiempos.

Posteriormente se elige al azar 2 números de NTI, se observa cual es el de menor tiempo de terminación y según el seleccionado, es la secuencia que se elegirá.

En la parte de variación, se seleccionan 2 NTI de los mejores, escogidos anteriormente en el proceso de selección, y se aplica el cruce correspondiente y mutación para variar la población, el ciclo continuo hasta formar una nueva población con el mismo número de NTI y NTB.

Al finalizar, se realiza una recopilación de información de las mejores, las promedio y las peores secuencias.

2.2 ESTADO DEL ARTE

Schuh, Potente y Thomas (2013) en su artículo mencionan uno de los problemas en cuanto a una mala planeación de la producción, el cual es provocado por el incumplimiento de los tiempos de entrega, también existe falta de transparencia y funcionamiento del control, selección de decisiones.

La configuración para la mejora en el plan de producción determina la forma de liberación de pedidos, ordenes de priorización y de la planificación de capacidades. Los empleados no saben el impacto que tiene la secuenciación y cumplimiento dentro del proceso, las responsabilidades de un mal control en la industria son el departamento de máquinas y herramientas o Ingeniería de producción. (Wiendahl H., 2006).

Una metodología utilizada en primer estancia fue Primeras entradas, primeras salidas, la cual funcionó para lograr mejores resultados.

Las restricciones utilizadas dentro de la metodología fueron la estructura organizativa y necesidad de una alta calidad. (Schuh, Potente, Fuchs, & Thomas, 2011).

Un recurso vital para planear correctamente la producción son los empleados de piso, ellos prefieren crear sus propios planes de producción porque creen que sus intervenciones aún pueden salvar las fechas de vencimiento de sus órdenes.

Los empleados tratan de mantener alta la capacidad de la maquinaria, kan-ban fracasó por sus hábitos y métodos de pago, la organización entre empleados y las decisiones influyen en el control de producción y es bueno que los empleados aporten ideas. (Wiendahl, Ein sozio-technischer Ansatz für das industrielle Auftragsmanagement, 2008).

Para un buen plan de la producción se tienen que seguir 4 tareas que consisten en: Orden de generación, liberación de la orden, secuencia y control de la capacidad.

Al generar orden, se determina el tamaño de lote, trabajo de proceso (WIP), esparcimiento de tiempos de producción; Se eligen a partir de las órdenes del cliente, pronósticos o nivel de inventario, determina también pedidos en espera de ser procesados. (Lödding, 2012).

El control depende de la capacidad de la ocupación de las máquinas y asignación de empleados y mide la productividad y costo de producción y su objetivo es ampliar los cuellos de botella, la separación de máquinas y asignación de operadores permite mayor flexibilidad dentro de la producción, donde sí es limitado el WIP los tiempos de producción se pueden controlar. (Münzberg, 2009) (Lödding, 2012).

Schuh, Franzkoch, Potente y Fuchs (2010) desarrollaron un modelo de cadena de valor enfocada al plan de la producción de 3 etapas: Corriente de valor, plan de producción, datos maestros de la administración. (Baumann, 2011).

En este artículo, la simulación es usada para la configuración para un plan de la producción, donde gracias a esta se puede observar lo que pasa dentro del proceso, y con esto hacer una correcta configuración y control dentro del mismo. (Schuh, Franzkoch, Potente, & Fuchs, 2010).

Schuh, Potente y Thomas (2013) mencionan otro factor importante dentro del plan para controlar la producción, cuando ésta inicia, se tiene que supervisar el progreso, el que controla la producción es el encargado de integrar las órdenes de prioridad al plan de producción, capaz de optimizar el rendimiento de producción debido a cambios.

Se tiene que saber: ¿Con qué frecuencia, cuánto tiempo y dónde se hace? La experiencia del trabajador puede determinar la decisión, por lo que los autores recomiendan un modelo organizado que incluya: datos con certeza, personas con necesidad de comunicarse, interactuar y decidir temas de programación definida.

Aparte del modelo, se simboliza un sistema socio-económico con 4 elementos: personal, información, tiempo y tema.

Los autores utilizan el sistema ERP: (Enterprise Resource Planning) Planificación de recursos de la empresa que utiliza producción suave y tiempos cortos como parte del plan de control en la producción.

La liberación de la orden es de gran importancia en el control, ya que determina la cantidad de pedidos en el sistema de producción; La intensidad de pedidos urgentes afecta el cumplimiento de normas de secuenciación. En caso de que un operador se ausente, debe de haber un sustituto para relevarlo y que esto no afecte a la programación.

Con el fin de representar el problema que anteriormente se menciona, se utiliza una metodología basada en principios de un Grupo de Tecnología (GT), donde las máquinas se representan en renglones y las columnas contienen el proceso, también se utilizó una red neuronal para crear el GT, utilizando un Algoritmo genético multi-objetivo para crear soluciones óptimas, donde son analizadas y comparadas.

Ruíz, Catrillón y Sarache (2015) en su artículo menciona en que consiste la problemática con la programación de tareas, la cual corresponde en la definición en la secuencia que mayormente beneficie a un lote de productos con variación en flujos y el variado tamaño del mismo, donde la característica está en la diferencia en el secuenciamiento que conlleva cada pedido que va entrando al sistema (Chase, Aquilano, & Jacobs, 2006).

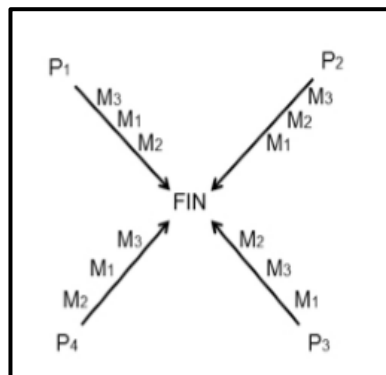


Ilustración 8. Grafo del Job shop 4x3. Referencia: Ruíz, 2015

Para mencionar un ejemplo de lo anterior, en la *ilustración 8* se muestran 4 pedidos P en 3 máquinas M, el objetivo consiste en encontrar el menor tiempo posible para el conjunto de pedidos (tiempo de terminación), cuando se exige agregar más pedidos, es vital una solución multi objetivo para resolver dicha problemática.

Precisamente para la solución multi objetivo, Frutos y Tohmé (2012) mencionan en su investigación cómo entre los métodos multi objetivo se encuentran los algoritmos genéticos entre los cuales destacan el NSGA II (Non-dominated Sorting Genetic Algorhythm II), SPEA II (Strength Pareto Evolutionary Algorhythm II), NSGA II con mayor eficiencia, el cual inicia a partir de una población y después se encarga de seleccionar los individuos que están en o cerca de una frontera de Pareto, en donde por parte individual se satisface el conjunto de objetivos y por último los individuos se mezclan en un proceso iterativo.

Dentro de la investigación, se incorporó una metodología con el algoritmo de selección, esta metodología fue aplicada a un caso de estudio para compararla con PEPS y

NSGA II, donde la metodología superó a PEPS en variables de tiempo de terminación, tiempos muertos y costos de energía.

En cuanto al tiempo de terminación y tiempos ociosos no se obtuvo respuesta alta comparada con NSGA II, pero si se vio en cuanto a costos en la energía y eficiencia elevada en el nivel de iteraciones requeridas para alcanzar al mejor tiempo de terminación.

Los objetivos de la metodología a utilizar son:

- Reducir tiempo de terminación
- Reducir el costo de recursos utilizados.

La metodología se basa de 7 intentos:

Planteamiento del problema, creación de parentales, generaciones, creación de filiales, computo de funciones, selección de mejores sucesores, selección de mejores individuos.

Para dar validez a la resolución del problema con los Algoritmos anteriormente mencionados, se utilizó MATLAB para comparar NSGA II y PEPS con 1.000 iteraciones, 10 individuos parentales y 1 mutación de 30%.

Como conclusión, la metodología de subpoblaciones propuesta tiene mejores resultados que NSGA II, y se requiere menos iteraciones en cuanto a tiempo computacional para el SP que para el NSGA II. (Frutos Alazard & Tohmé Hauptmann, 2012)

Por otra parte con otras propuestas de un algoritmo genético mejorado, Meisel y Prado (2010) en su artículo mencionan el objetivo de este, el cual trata de establecer un orden en la muestra y tiempo en un grupo de tareas determinar el orden de emisión y tiempo de un conjunto de trabajos sobre máquinas.

El problema que los autores presentan es de tipo N-P Hard porque el algoritmo incrementa de manera exponencial su tiempo computacional cada que crece el número de variables y restricciones(Garey & Johnson, 1976).

Se ha propuesto meta heurística para darle solución al problema, redes de Hopfield, algoritmos de Ramificación y acotación (branch & bound), etc... dentro de la investigación es propuesto un algoritmo genético mejorado (AGM) y enfriamiento simulado propuesto (ESP) donde primero se hace una formulación matemática del JSP y se presentan 2 algoritmos para solución del problema.

Dentro del AGM se propone la hibridación donde al mutar se hace una estrategia de enfriamiento simulado y se realizó un método de codificación en donde el orden en la cantidad de tareas se ve en el cromosoma, según el valor más apto encontrado en cada generación es aplicada la búsqueda local, solamente si este valor es mejor que el valor más apto encontrado en las generaciones corridas.

Para la validación del AGM se realiza un diseño de experimentos con 30 repeticiones y para validar el ESP los parámetros como temperatura final e inicial se analizan.

Para comparar los métodos anteriormente mencionados se probó una hipótesis con discrepancia en las medias, y de acuerdo al análisis de experimentos para encontrar los parámetros óptimos se analizó que para el AGM no es necesario elevar el tamaño en la población.

Para una mejor validación del AGM y ESP cada parte del problema se corrió 30 veces, donde se observó un desviación relativamente baja con respecto al tiempo de terminación promedio, donde se concluye que los algoritmos utilizados tienen una confiabilidad alta para así obtener mejores respuestas.

Como conclusión, en cuanto al AGM, éste cuenta con poca eficacia computacional para problemas NP-hard (15 o más trabajos con 10 máquinas), ya que los tiempos computacionales son relativamente altos, por lo que se recomienda utilizar menor número de máquinas para obtener mejor respuesta.

El ESP, por otra parte, es un algoritmo con alta eficiencia computacional para problemas con elevado número de trabajos y máquinas, ya que logra converger en menos de 4 minutos.

Ya aplicado el ESP, efectivamente converge más rápido a comparación con el AGM y el tiempo de terminación logro acercarse al mejor valor real.

Tupia y Mauricio (2004) en su artículo en el que plantean un problema de planeación de producción por medio de un modelo matemático, en donde se dan m máquinas (procesadores) y n trabajos con T_{ij} elementos de tiempo para cada trabajo i –ésima que se lleva a cabo en la máquina j –ésima y lo que se quiere es acomodar las n tareas en las m máquinas con su respectiva secuencia de ejecución para satisfacer la optimalidad que requieren para resolver. El problema de programación es de complejidad ya que algunos

trabajos dependen de otros que le anteceden o preceden y las máquinas no son iguales, por lo que a cada trabajo le corresponden otras tareas y tienen que esperar.

Al modelo se le agrega la heterogeneidad de las máquinas, cada tarea tarda diversos tiempos en ejecución, el objetivo sería minimizar el tiempo rezagado en ejecución en máquinas.

No existen métodos exactos que resuelvan el problema N-P Hard, según los autores, y plantean un modelo matemático de optimización combinatoria que se muestra a continuación.

$$\text{Min } X_0 \tag{1}$$

s.a.

$$X_0 \geq \sum_{i=1}^n T_{ij} * X_{ij} \quad \forall j \in 1 \cdot M \tag{2}$$

$$X_0 \geq \sum_{j=1}^m X_{ij} = 1 \quad \forall i \in 1 \cdot N \tag{3}$$

Dónde:

X_0 = tiempo de terminación

X_{ij} = 0, si la máquina j – ésima no ejecuta la tarea i – ésima

1, c.c.

Los métodos para resolución del problema pueden ser exactos o aproximados, los exactos se encargan de examinar el orden de las tareas o procesos en producción y el análisis de búsqueda para todas las posibles permutaciones y solo son útiles para algunas medidas, en cambio los aproximados si se enfocan en solucionar las variantes más complejas y obtienen soluciones muy buenas para justificar, como métodos heurísticos para resolver Job shop con una restricción donde las máquinas solo son capaz de operar una operación a la vez, sin paros, el objetivo sería determinar un programa que minimice el tiempo de terminación.

Como métodos de resolución de problemas de tipo N-P hard se pueden mencionar algunos, por ejemplo, Bard y Feo (1989) presentan el método de búsqueda en grafos en el cual se encarga de recorrer un árbol cuyos nodos son las secuencias de operaciones y no consideran costos extra por atrasar pedidos ni brincos de tareas de máquina a máquina.

Como otro método de solución se encuentran los algoritmos genéticos, donde según Goncalves (2011) y Davis (1985) utilizan representación cromosómica q se basan en tasas de elitismo aleatoria, asimilan un ambiente laboral cambiante en línea de producción flexible y se dividieron los trabajos en cortos cromosomas para que las poblaciones generadas sean lo más variadas posible, parecido al loteo. (Davis, 1985)

La búsqueda tabú es otra opción de solución, en donde Taillard (1994) menciona que dicho método cuenta con listas que son matrices formadas por renglones (operaciones) y columnas (máquinas) al movimiento de una aspirante a operación i –ésima (tarea) para acomodar de una máquina a otra.

Las máquinas y tareas se subdividen en: máquinas iguales y operaciones que no dependen de otras, máquinas iguales y operaciones que si dependen de otras, máquinas distintas y operaciones no dependientes, máquinas diferentes y operaciones dependientes (modelo más complejo). (Taillard, 1994)

Para resolver la última se pueden utilizar algoritmos voraces, donde el algoritmo es voraz porque se basa principalmente en buscar a la mejor alternativa para formar el resultado, al aplicarse, la función que se va a mejorar arroja óptimos locales, en la estructura consideran que dentro de la muestra existe mínimo una operación que no cuenta con predecesores, siendo esta la operación inicial. para máquinas idénticas, programación discreta para trabajos que no dependen de otros, esto es utilizado para máquinas diferentes (Tupia & Mauricio, 2004) , se adaptó el trabajo de Campello (1992) considerando los diferentes tiempos de ejecución para cada máquina.

Ya ejecutado el algoritmo se tienen 16 combinaciones (máquina-tarea) y se enfrentó el algoritmo con una meta heurística GRASP (Eds. M. Solar, 2004.).

Para determinar la calidad de la solución se aplicará un modelo matemático (programación lineal usando LINDO (L, 1997) para obtener una solución exacta, se configuraron las instancias $n \times m$ para experimentos exactos.

Los autores concluyen que el algoritmo voraz encuentra una solución muy cerca a la solución exacta, el tiempo del computador ejecutando el algoritmo es insignificante (por debajo de los 2 seg.)

Chang, Damodaran y Melouk (2004) por otra parte en su artículo mencionan que cuando el tiempo de terminación es el mismo se reduce el problema de loteo, el problema de máquinas en paralelo también es N-P hard y proponen un algoritmo de recocido simulado (SA).

Chandru (1993) propone heurísticas para minimizar el tiempo de terminación en máquinas paralelas idénticas del procesamiento de lotes.

Brucker (1998) en cambio, desarrolla un algoritmo de programación dinámica usando una nueva técnica de corrida dinámica.

Con las ideas anteriormente mencionadas, Wang (2002) aplica la misma técnica para determinar la solución factible para tiempos de entrega, ellos combinan el algoritmo con un esquema clave aleatoria para desarrollar un algoritmo genético. La SA fue aplicada para solucionar la problemática del agente viajero.

Se plantea a continuación, un modelo para las máquinas en paralelo por lotes:

$$\begin{aligned} \text{Min } C_{\max} & \quad (4) \\ \text{s.t.} & \end{aligned}$$

$$\sum_{b \in B} \sum_{m \in M} X_{jbm} = 1 \quad \forall j \in J \quad (5)$$

$$\sum_{j \in J} \sum_{m \in M} S_j X_{jbm} \leq S \quad \forall b \in B \quad (6)$$

$$P_{bm} \geq P_j X_{jbm} \quad \forall m \in M, b \in B, j \in J \quad (7)$$

$$C_{\max} \geq \sum_{b \in B} P_{bm} \quad \forall m \in M \quad (8)$$

$$X_{jbm} \text{ binario} \quad (9)$$

$$C_{\max}, P_{bm} \geq 0 \quad (10)$$

Sets:

J = Set de trabajos ($j \in J$),

M =Set de máquinas ($m \in M$),

B =Set de batches ($b \in B$)

Parámetros:

P_j =Tiempo de procesamiento de trabajo j

s_j =Tamaño de trabajo j

S = Capacidad de la máquina

P_{bm} =Tiempo de proceso de batch b procesado por la máquina m .

C_{\max} = Tiempo de terminación

En la ecuación (4) se muestra la función objetivo, en (5) un trabajo se asigna a un grupo de piezas y es procesado en una máquina, en (6) tamaño total de todos los trabajos asignados al lote asegurándose de no exceder la capacidad de la máquina, en (7) el tiempo de proceso del lote es el tiempo de procesamiento máximo de todos los trabajos en el lote, (8) determina el tiempo de terminación, (9) decisión binaria, (10) variables no negativas.

Ya realizada la representación del problema con el modelo matemático, se procede a la experimentación, donde el tiempo de corrida de SA es comparado con CPLEX (herramienta para resolver problemas de programación lineal, entera, entera mixta y cuadrática), el experimento se basa en 2 y 4 máquinas, y 10,25 y 50 trabajos.

Ya con la comparación, se analizan los resultados, los cuales demuestran que SA requiere tiempos de corridas cortas cuando se compara con CPLEX, pero SA garantiza mejores resultados cortos que CPLEX.

Los autores concluyen que SA fue desarrollada para minimizar el tiempo de terminación para la capacidad de máquina en procesamiento de lote que procesa trabajos con tamaños no idénticos y SA encuentra mejores resultados en tiempos de corridas cortos, CPLEX con 25 y 50 trabajos.

Como representación de un problema tipo taller, Osorio, Castrillón y Orejuela (2008) en su artículo mencionan que en el Job shop flexible (FJSSP) las distintas operaciones pueden procesarse en alguna de las máquinas (que son iguales) dentro del taller, el problema serían los paros en las tareas y los tiempos de preparación que van ligados a la secuencia programada,

donde se busca reducir el tiempo de flujo, tiempo de terminación, retrasos, tardanza y número de trabajos retrasados.

De una manera global, el problema de Job shop se compone por $n \times m = (n!)^m$ posibles soluciones y esto es lo que es precisamente el N-P Hard.

El FJSSP pertenece a un ramo de Job shop, pero FJSSP es más complejo que el Job shop tradicional por asignar múltiples operaciones a las máquinas.

Como primer paso en la experimentación, se tiene que asignar los trabajos a los recursos existentes y luego darle secuencia según su asignación, se busca utilizar modelos monolíticos (problemas de asignación y secuencia en un solo paso) y el problema utiliza Job shop donde se necesita definir recursos (máquinas y personas). Los autores plantean 2 niveles de solución, que se mencionan a continuación.

- Definen un Plan agregado de la producción (asignación de máquinas y personas a las tareas).
- Con lo que resulte del plan agregado va al nivel más bajo y con un modelo de simulación resuelven el problema de programación.

El problema de nivel superior consiste en asignar los procesos a los centros de trabajo, minimizando los tiempos de ejecución y que los centros estén balanceados, ya que si se recarga al centro todo el tiempo de terminación se incrementa.

Posteriormente se representa el problema de FJSSP por medio de un modelo matemático, haciendo algunas configuraciones.

M Máquinas ($I = 1, 2 \dots M$)

N Operaciones ($J = 1, 2 \dots N$)

Nivel superior= L centros de trabajos ($R = 1, 2 \dots L$) donde a cada centro lo conforma m máquinas ($i = 1, 2 \dots m$) $m \in M$ por centro L se asigna n operaciones ($j = 1, 2 \dots n$) $n \in N$ cada operación j implica operaciones ($o = 1, 2 \dots k$) cada operación o trabajo j contiene un tiempo en proceso en máquina i , P_{oij} .

Se tomaron en cuenta algunas consideraciones importantes:

- Los trabajos están a disposición para ser procesados en $t = 0$ (fecha de liberación).
- Las máquinas están a disposición para ser operadas en $t = 0$ (disponibilidad).
- La preparación, en tiempo, va dependiendo del secuenciamiento.
- Es posible que se presenten paros en las operaciones, ya en las máquinas sin que haya concluido la operación en proceso para pasar a otra, al volver a ejecutar la operación solo es necesario el tiempo que falte para la determinación de la tarea (migración).
- Recirculación en Job shop (1 trabajo pasa varias veces por alguna máquina), tanto que el trabajo que contiene todas sus operaciones se procese en una sola máquina.
- La máquina no cuenta con la capacidad de que en el mismo tiempo procese más de una operación.
- Todas las máquinas tienen la capacidad para realizar la misma operación (flexibilidad).
- Las máquinas que realizan el trabajo son iguales (paralelas).

Para dar ser más específicos, por medio de la programación de la producción con todas las operaciones, se da detalle del lugar donde son procesadas (máquinas), existencia de paros (la frecuencia con la que la máquina procesa la operación) y en qué momento se realizan (inicio y fin de los procesos o secuencias de trabajo).

El modelo presenta 2 tipos de agregación: las operaciones en trabajos y el de máquinas en centros de trabajo.

Modelo de programación lineal:

$$Z = \text{Max} \sum_{J=1}^N T_{R,J} X_{R,J} \quad \forall R \in (1, \dots, L) \quad (11)$$

$T_{R,J}$ se refiere al tiempo que tarda en procesar el trabajo J en el centro R y $X_{R,J}$ es una variable con valor binario que clasifica si el trabajo J se está procesando o no en el centro R y el objetivo consiste en encontrar el valor más bajo de Z .

$$\text{Min } Z \tag{12}$$

s.a.

$$Z \geq \sum_{J=1}^N T_{R,J} X_{R,J} \quad \forall R, R = 1, \dots, L \tag{13}$$

$$X_{RJ} = 1, \text{ si el trabajo } J \text{ se trabaja en el centro } R \\ 0, \text{ c.c.} \tag{14}$$

La ecuación (11) define el tiempo de terminación, en (12) función objetivo donde busca el valor mínimo, en restricción (13) garantiza balance en los centros de trabajo.

L =Número de centros de trabajo

N =Número de trabajos a ser programados

En la ecuación (14) variables de decisión.

El problema de asignación en cada centro se basa en establecer la máquina en la cual serán operados los lotes que dividen las operaciones de un trabajo.

Para la resolución del problema, se toma la misma metodología que Tahar (2006) donde se busca reducir el problema de cada centro que tiene 1 recurso y se transforma en un problema del agente viajero, la secuencia se divide entre la cantidad en máquinas que constituyen al centro:

Se aplicó un modelo mediante el uso de un software de programación Legin con máquinas idénticas, el problema se basa en 6 tareas, las cuales deben ser programadas en 6 máquinas iguales.

El problema fue resuelto, asignando 3 de los trabajos a cada centro, con el fin de que la efectividad del modelo sea lo mejor posible, se subdividen los problemas, ya que según su tamaño se facilita su resolución a través de métodos con flexibilidad en cuanto a tiempos. El modelo se caracteriza por el loteo.

La metodología que se utilizó, cuenta con algunas limitantes, entre las cuales se encuentra la garantía a una solución solamente para problemas que utilicen hasta 7 operaciones y 3 máquinas, la sinergia obtenida da lugar al modelo para que se replique a varios centros, siempre y cuando la limitante se respete, permitiendo así utilizar este modelo para problemas grandes.

Para concluir, La metodología utilizada es la mejor opción que ofrece una gran ventaja al otorgar un excelente balance entre la conformación de máquinas y la carga de los centros de trabajo, adaptado el modelo adecuadamente y así mejorar la efectividad en el proceso.

Wen Yang y Beck (2016) dicen que el Job shop es un problema $n \times m$ donde J es el conjunto de n trabajos y M el conjunto de m máquinas, donde se da una lista de las máquinas $(\sigma_i^j, \dots, \sigma_h^j, \dots)$ representando el orden de procesamiento de j en máquina m .

- σ_h^j es la h – ésima operación del trabajo j .
- σ_m^j es la última operación del trabajo j .
- P_{ij} variable no negativa la cual representa el tiempo de procesamiento de j en i .
- Cada máquina no puede procesar más de un trabajo a la vez.
- No hay interrupciones.

El principal objetivo consiste en hallar la programación de J en M que minimice el tiempo de terminación. La minimización de tiempo de terminación para JSP es NP-Hard para $n \geq 3$ y $m \geq 2$.

Para programación entera mixta se utilizan 3 formulaciones:

- Índice-tiempo
- Rango-basado
- Disyuntiva

Siendo la última la más efectiva.

Liao C (1992) propone una modificación a la formulación disyuntiva y muestra empíricamente que ese modelo es mejor al original.

Se presentan 4 modelos MIP (programación entera mixta):

- Modelo disyuntivo
- Índice de tiempo
- Rango basado
- Modelo propuesto por Liao que es el modelo disyuntivo de Liao.

La diferencia entre el modelo disyuntivo y el propuesto por Liao es que adhiere continuidad, variables de decisión que reducen las restricciones lineales y límites superiores, él dice que los límites en variables son más fáciles de manejar que las restricciones.

En el modelo de índice-tiempo propuesto por Bowman (1959), la variable de decisión X_{ijt} es 1 si el trabajo j empieza en el tiempo t en máquina i .

Modelo de índice-tiempo:

$$\min C_{max} \quad (15)$$

s.t.

$$\sum_{i \in H} X_{ijt} = 1, \quad \forall j \in J, i \in M \quad (16)$$

$$\sum_{i \in H} (t + P_{ij}) \cdot X_{ijt} \leq C_{max}, \quad \forall j \in J, i \in M \quad (17)$$

$$\sum_{j \in J} \sum_{t \in T_{jt}} X_{ijt} \leq 1, \quad \forall j \in M, t \in H, \text{ donde} \quad (18)$$

$$T_{ijt} = \langle t - P_{ij} + 1, \dots, t \rangle$$

$$\sum_{i \in H} (t + P_{\sigma_{h-1}^j, j}) \cdot x_{\sigma_{h-1}^j, jt} \leq \sum_{i \in H} t \cdot x_{\sigma_h^j, jt}, \quad \forall j \in J, h = 2, \dots, m \quad (19)$$

$$X_{ijt} \in \{0, 1\}, \quad \forall j \in J, i \in M, t \in H \quad (20)$$

Dónde:

σ_h^j Es la h – ésima operación del trabajo j .

σ_m^j Es la última operación del trabajo j .

P_{ij} Variable no negativa la cual representa el tiempo de procesamiento de j en i .

Variable de decisión $X_{ijt} \begin{cases} 1, & \text{si el trabajo } j \text{ empieza en el tiempo } t \text{ en máquina } i \\ 0, & \text{c.c.} \end{cases}$

Ecuación (15) función objetivo, restricción (16) asegura que cada trabajo puede empezar exactamente una vez en cada máquina, restricción (17) asegura que el tiempo de terminación es al menos el tiempo más largo de todas las operaciones, restricción (18) asegura

que las máquinas no sobrepasen la capacidad, (19) es la restricción de precedencia, donde todas las operaciones de un trabajo son ejecutadas en el orden dado.

El rango basado es parecido al modelo índice-tiempo, pero va más enfocado a la posición.

Se hizo una comparación de modelos utilizando CPLEX, GUROBI y SCI.

En CPLEX donde todos los modelos menos el propuesto por Liao, son parcialmente consistentes, y donde indican que ambos modelos disyuntivos se realizan mejor que rango-basado e índice-tiempo, realizando mejor en problemas de 10 x 10.

Para problemas de 15 x 15 se muestra que el modelo disyuntivo original encuentra mejores soluciones factibles más rápido que el modelo propuesto por Liao.

Como resultados se obtuvo que GUROBI es probado óptimamente más rápido con el modelo propuesto por Liao y es mejor realizado con rango-basado y con índice-tiempo para problemas menores de 5 x 5, por otra parte también se obtiene que SCIP es menos eficiente que CPLEX y GUROBI.

Como parte de los resultados se compara MIP con programación de restricciones (CP), donde para MIP los resultados se comparan usando CPLEX y con modelo CP se usando un hilo, ambos se resuelven rápidamente con problemas de 10 x 10.

CP encuentra mejores soluciones factibles más rápido que MIP.

CP domina a MIP en el límite de tiempo computacional.

Se concluye que el modelo disyuntivo es el más eficiente para los problemas de programación lineal para problemas Job shop para todos los tamaños donde el modelo disyuntivo original es más eficiente para CPLEX y GUROBI.

Shen, Burnham y Smalov (2015) mencionan que el objetivo en su artículo es optimizar la duración de fermentación la cual depende de 3 productos cerveceros para ser producidos en 3 diferentes capacidades de vasijas.

El método del Algoritmo genético es implementado en Matlab para optimizar un sistema de producción y es formulado un modelo matemático.

Para el modelo matemático se utilizaron distintas variables, donde hay n órdenes de varios productos de cerveza que son arribados continuamente para formar una cola de espera, para producir en una capacidad limitada en condiciones de vasija y fermentación.

Cada orden se acumula en un lote de producción, cada lote producido puede ser procesado una vez en cada vasija, cada vasija tiene que ser limpiada antes de cada operación.

Se asume que 3 productos de cerveza son producidos en 3 vasijas paralelas de diferente capacidad cada una y se obtiene un tiempo mínimo de producción.

El cambio ocurre cuando el siguiente lote es transformado en diferentes vasijas.

Dónde:

P_i = Número de los tipos diferentes de producto i , $i = \{1,2,3\}$

v_j = Número de vasijas j , $j = \{1,2,3\}$

$X_{p_i v_j}$ = Número de ocurrencias de p_i hecha en vasija v_j

$m_{p_i v_j}$ = Variable de decisión: coeficiente 1, trabajando en la misma vasija, si es coeficiente 0, trabajando en diferentes vasijas

t_{p_i} = La fermentación de p_i

n_{p_i} = Número de ordenes n a ser programadas por p_i , $n = \{1,2,3, \dots, n\}$

d_{p_i} = Tiempo de entrega de p_i

S_{p_i} = Tiempo de preparación para p_i

$C_{p_i v_j}$ = Tiempo de cambio para p_i y v_j

C_{v_j} = Tiempo de limpieza para v_j

V_{cap} = Capacidad de v_j

Función objetivo:

Se tienen 3 tipos diferentes de cerveza producidos en 3 vasijas separadamente.

$$T = t_{p_i} \sum_{i=1}^n \sum_{j=1}^n x_{p_i v_j} + \sum_{j=1}^{m_1} \left(\sum_{i=1}^{n_1} S_{p_i} + \sum_{i=1}^{n_1} C_{v_j} \right) + \sum_{j=1}^{m_2} \left(S_{p_i} \sum_{i=1}^{n_2} C_{v_j} + \sum_{i=1}^{n_2} C_{p_i v_j} \right) \quad (21)$$

Restricciones:

Cada vasija puede producir solamente un lote de órdenes durante un proceso de producción:

$$\sum_{i=1}^n m_{p_i v_j} = 1 \quad (22)$$

$$\sum_{j=1}^n m_{p_i v_j} = 1 \quad (23)$$

Dónde:

$$m_{p_i v_j} \in \{0,1\}, \forall_i \in \{1,2,3,\dots,m\}, \forall_j \in \{1,2,3,\dots,n\}$$

La capacidad de las vasijas no puede ser excedida cuando las órdenes del mismo tipo de cerveza se acumulan en el proceso.

$$0 \leq n_{p_i} \leq V_{cap} \quad (24)$$

Los tiempos de entrega también se consideran y la producción debe terminar antes o durante el tiempo de entrega.

$$t_{p_i} + \left(\sum_{i=1}^{n_1} S_{p_i} + \sum_{i=1}^{n_1} C_{v_j} \right) \leq d_{p_i} \quad (25)$$

$$t_{p_i} + \left(\sum_{i=1}^{n_2} S_{p_i} + \sum_{i=1}^{n_2} C_{v_j} + \sum_{i=1}^{n_2} C_{p_i v_j} \right) \leq d_{p_i} \quad (26)$$

Para solución al modelo se utilizó un Algoritmo genético, el cual consideró las restricciones con límite de capacidad, 3 productos diferentes en 3 máquinas paralelas, cada vasija puede procesar en un lote solamente y no hay interrupciones en el procesamiento, se requiere capacidad en vasijas y el tiempo de producción no debe exceder el tiempo de entrega.

El algoritmo genético simula y optimiza el sistema de producción. Un cromosoma representa la lista de órdenes (3 productos), cada individuo o población tiene un límite de

procesamiento ya establecido y un tiempo de preparación y cambio. La evolución es representada como las insatisfacciones.

El tamaño de la población en el algoritmo es de 100 y 400 generaciones, la solución óptima es obtenida después de las 95 generaciones, el algoritmo genético logra mejorar el tiempo de terminación promedio= 850.

Fowler (2004) sugirió una simulación de fábrica en tiempo real para evaluar un proceso de producción, la cual corre simultáneamente la operación real e inmediatamente proporciona resultados para las decisiones a corto plazo (Fowler 2004). Hoy en día esta idea es abordada, por ejemplo, en contexto con la iniciativa Industry 4.0 que tiene como fin la creación de fábricas inteligentes mediante el uso de sistemas cibernéticos (CPS) que viene siendo elementos virtuales y físicos, así como tecnologías digitales como la realidad aumentada y el Internet.

Pask (2017) propone la utilización de Fuzzy y simulación de Montecarlo para ayudar significativamente a los que se encargan de tomar la decisión. (Pask, Lake, Yang, Tokos, & Sadhukhan, 2017)

Kagermann (2013) del mismo modo, busca la simulación como el enfoque de los duplicados digitales pretende un sistema del mundo real con un duplicado virtual. Este enfoque permite comprender cómo un producto es transformado durante la producción y también para trazar las características de este a lo largo de su ciclo de vida Grieves (2014). Sin embargo, hasta ahora la mayoría de las aplicaciones de simulación se centran en niveles estructurales individuales, tales como un sistema de producción completo, una celda de producción, máquinas y equipos, o procesos Landherr (2013). Para simular una fábrica entera, los nuevos enfoques de simulación tienen que utilizar simulación multiescala de sistemas de producción con sus estructuras y procesos. (Schönemann, Multiscale Simulation approach for battery production system, 2017)

Müller (2015) presentaron WorkDesigner (aplicación de software de simulación económica), este utiliza un modelo de simulación fundamentado en la edad y estrés para evaluación de trabajo de empleados en la industria manufacturera donde los parámetros principales son los trabajadores y las estaciones de trabajo, para los empleados la capacidad y habilidad física y para la estación de trabajo, iluminación, clima, ruido, postura de trabajo e

intensidad de trabajo; Basado en las fórmulas proporcionadas por Feller y Müller , el estrés general se calcula para cada proceso de trabajo, lo que conduce a la determinar la utilización por el empleado de su capacidad de trabajo al final. En general, el objetivo de WorkDesigner es responder a la pregunta: ¿cómo se ve afectado el personal por la digitalización de la producción? (Gust, Müller, Feller, & Schiffmann, 2017)

Ham (2017) presenta junto a los demás autores una metodología de prueba basada en simulación para la inspección electrónica de la unidad de control de motores (ECU), se requiere la verificación efectiva de ECU en su fase de fabricación por lo que ellos utilizan un modelo de sistema de motor virtual, el cual especifica las operaciones de cada función durante la simulación, esto ha sido implementado y aplicado exitosamente en una compañía coreana automotriz.

Con el fin de verificar si ECU es defectuoso o no, el sistema de inspección asegura que todos los actuadores reciban señal de ECU durante la simulación y de esa manera se comprobó un diagnóstico. (Ham, Ko, & Park, 2017)

Santana Robles (2017) propone junto a los demás autores las redes de Petri coloreadas jerárquicas para un modelo que se basa en los procesos de negocio de una cadena de suministro textil para obtener una mejor comprensión de su comportamiento, utilizaron el software CPN Tools, el cual permitió construir, editar, verificar sintaxis y simular el modelo discreto, la jerarquía representa la cadena de suministro de la industria textil como sistema complejo, mostrando en los resultados la utilización de recursos y el conocimiento de cómo se modela la cadena textil en fabricación para inventario (make to stock). Se observó cómo la empresa fue afectada por incumplimiento de proveedores, así como el inventario en la orden del cliente. Se obtuvo el promedio de ocupación de los equipos de producción. (Santana Robles, Granillo, Isidro, & Isaías, 2017).

Con un enfoque más centrado en la metodología utilizada en este artículo, Supsomboon, S. & Vajasovimon, A (2016) proponen un modelo de simulación para producciones tipo Job shop, el cual ayudo a mejorar las operaciones, resolver problemas y tomar las mejores decisiones. Se creó un layout de las líneas de producción en el software de Plant simulation, validando el modelo con la prueba estadística t-student, utilizando en la experimentación 3 estrategias: A: Crear el layout del proceso por tecnología de grupo y

ampliación de trabajos, B: Activación de máquinas ociosas, ampliación de trabajos y trabajadores adicionales, y por último la C: Trabajadores adicionales y 2 turnos. Donde como conclusión muestran que la estrategia B es mejor en cuanto a utilización de trabajadores y la C en costos de operación y salida de piezas por día (Supsomboon & Vajasovimon, 2016).

Castrillón (2016), junto a sus aliados, mencionan en su artículo en qué consiste el problema de producción Job shop, el cual se encarga de establecer el más factible secuenciamiento en los pedidos para que el tiempo de terminación se reduzca significativamente.

Como parte de la metodología de Job shop, ellos plasman principalmente el objetivo que es una red y ésta se define como una mezcla de tiempos de proceso en cada nodo que compone dicha red, por las funciones de transferencia, como producto con los pesos (que se crean por algoritmos de programación de la red).

La metodología consiste en:

Paso 1. Identificar problemas de programación, donde se realizó una matriz de $n \times m$ en donde $n = \text{pedidos a programar}$, $m = \text{número de máquinas}$ y

$$T_{ij} = \text{Tiempo que la tarea } i \text{ necesita en la máquina } j.$$

Mencionan que el uso de algoritmos genéticos es la mejor técnica inteligente para dar solución al problema de programación Job shop, dentro del artículo, se presenta un algoritmo evolutivo híbrido con una red neuronal, con el objetivo de reducir el tiempo de fin dentro de la problemática del Job shop flexible, resultado de esto, un algoritmo robusto y estable que hace que la zona de búsqueda se amplié y así, evitar tropezar en los óptimos locales.

Como parte del modelo que representa el problema, se utilizan restricciones como: Cantidad de máquinas o proceso, precedencia de pedidos, etc...

Paso 2. Algoritmo, en donde se programó la red neuronal y un diagrama de Gantt, la red combino algunas entradas que son producto del cálculo del peso, la mezcla de entradas se realiza a través de una sumatoria, que es aplicada a una función de transferencia que genera una salida satisfactoria.

Cada salida construirá la programación de cada máquina.

Existen varias combinaciones de pesos w , las que generan distintas salidas validas al sistema, por lo que se requiere ejecutar un diagrama de Gantt con el objetivo de la obtención del secuenciamiento con mejor desempeño.

Paso 3. Validación, con la red y programada, se tomó un problema el cual represente todas las de la red que se propone y se valida la salida formada, si esta es invalida, se ajustan los pesos.

Paso 4. Se hizo la comparación de los resultados obtenidos con FIFO, LIFO, LPT Y SPT.

Paso 5. Análisis de estabilidad, fue hecho un análisis de varianza con un intervalo de confianza de un 99.5%.

Para concluir, se encontró la secuencia óptima y la solución fue encontrada por la red neuronal con un tiempo de terminación de 99 minutos. Este problema tiene 514 posibles soluciones (Omar D. Castrillón, 2016).

Zhang (2017), junto a los demás autores, toman en cuenta el tiempo de procesamiento más corto y usan balanceo de líneas en las máquinas, tomando en cuenta el Algoritmo genético multi-poblacional basado en programación multi-objetivo de Job shop flexible con el fin de minimizar el lapso de trabajo más largo de las piezas de trabajo, la carga de cada máquina y la carga total de las máquinas con programación del Job shop flexible.

Se realizó un Gantt de 3x4 del a programación Job shop flexible, solo se tomó en cuenta las máquinas con el menor tiempo de tiempo de terminación. El autor expande la escala basada en variación operacional con una estancia de 10x10 y otra de 15x10.

El método de peso Random es utilizado para encontrar la carga total de todas las máquinas en el Job shop y la varianza de carga de máquinas.

Como resultado, el algoritmo propuesto es eficiente en búsqueda óptima, de asignación de máquinas, comparando con otros algoritmos de planificación Job shop flexible es mejor en calidad de población y en la convergencia, a pesar de que esta es prematura, la carga total de máquinas y la varianza disminuyen significativamente y comenzó a converger a partir de la generación 50.

El algoritmo comienza con n poblaciones, los ordena de manera descendiente de fitness, extrae a los mejores organismos de una población, para formar a la mejor población, procesa sinérgicamente toda la población y selecciona la solución óptima, de acuerdo al valor fitness adquirido.

Las 2 estancias sirvieron para validar la eficacia del algoritmo multi-objetivo y también fue utilizado un algoritmo de Kasem.

Resultado final. Para las 2 estancias se probaron 3 algoritmos (Random, Kasem y el propuesto por los autores), con 3 funciones objetivo (T_{max} , ω_T y ω_M). Donde el algoritmo propuesto fue mejor que los demás en calidad de convergencia, búsqueda óptima y asignación de máquinas.

Para concluir, el proceso de programación Job shop logro minimizar el tiempo de tiempo de terminación más largo de cada pieza de trabajo, carga de trabajo y el total de carga de trabajo de las máquinas. Al comparar el algoritmo propuesto con el convencional algoritmo de programación Job shop flexible, el propuesto tiene mejor convergencia, donde esta se presenta después de la 50 generación. (Zhang, Wen, & Zhu, 2017).

2.3 MODELO PROPUESTO

Se propone un modelo donde se toman en cuenta algunas partes de la literatura como:

$$\sum_{i \in H} X_{ij} = 1, \quad \forall j \in J, i \in M \quad \text{tomada de (Wen Yang \& Beck, 2016)} \quad (27)$$

$$\left. \begin{array}{l} \sum_{i=1}^n m_{p_i v_j} = 1 \\ \sum_{j=1}^n m_{p_i v_j} = 1 \end{array} \right\} \text{tomadas de (Shen, Burnham, \& Smalov, 2015)} \quad (28)$$

Haciendo algunas modificaciones según la configuración que se tiene en la empresa el modelo que se propone se muestra a continuación:

$$\text{Min } C_{max} \quad C_{max} = ps_{ij} \quad \forall_{i,j} = J_i \quad (29)$$

s.t.

ASIGNACIÓN

$$\text{Máquinas} \quad \sum_{k=1}^m X_{ijkl} \leq a_k \quad k = 1, 2, \dots, m \quad (30)$$

$$\text{Operaciones} \quad \sum_{j \in J_i} X_{ijkl} = 1 \quad j \in J_i \quad (31)$$

PRECEDENCIA

$$\sum ps_{ij} \leq t_{j+1} \quad (32)$$

BINARIO

$$\sum_{k=1}^m \sum_{j \in J_i} X_{ijkl} \begin{cases} 1, & \text{si } O_{ij} \text{ es asignado en la máquina } k \\ 0, & \text{c.c.} \end{cases} \quad (33)$$

NO NEGATIVIDAD Y ENTERO

$$ps_{ij} \geq 0 \quad \forall_{i,j} \quad (34)$$

VARIABLES

i = Casos (1, ..., n)

j = Operaciones (1, ..., J_i)

k = Máquinas (1, ..., m)

l = Secuencia de operaciones asignadas en máquina k (1, ..., d_k)

J_i = Número total de operaciones de trabajo

X_{ijkl} = Variable de asignación

O_{ij} = Operación

ps_{ij} = Tiempo de procesamiento de operación O_{ij}

a_k = Número de máquinas iguales.

La ecuación (29) función objetivo, el cual es minimizar el tiempo de terminación (tiempo máximo), en (30) restricción de asignación a máquina, donde X_{ijkl} debe ser asignado a una de las máquinas, en las cuales dado el caso hay varias máquinas idénticas, en (31) restricción de asignación de operación en donde se debe asignar una operación , (32) restricción de precedencia en donde la suma de la operación anterior más la actual debe de ser mayor a la operación anterior, en (33) restricción binaria donde es 1 si la operación es asignada a la máquina k , en (34) la restricción de no negatividad en donde el valor de la variable X_{ijkl} debe ser mayor a 0 y con valor entero.

2.4 ALCANCES Y DELIMITACIONES

ALCANCES

- Obtener un plan de producción.
- Realizar una interfaz gráfica mediante el software de Plant Simulation.

DELIMITACIONES

- En la empresa no se cuenta con un sistema de inventarios, por lo cual es necesario mencionar que inicialmente se tendrá como supuesto que siempre se cuenta con materia prima.
- No se mencionará relación con inventarios y los que deriven de estos, proyecciones de nuevos pedidos ni demanda.

CAPITULO III

3.1 METODOLOGÍA

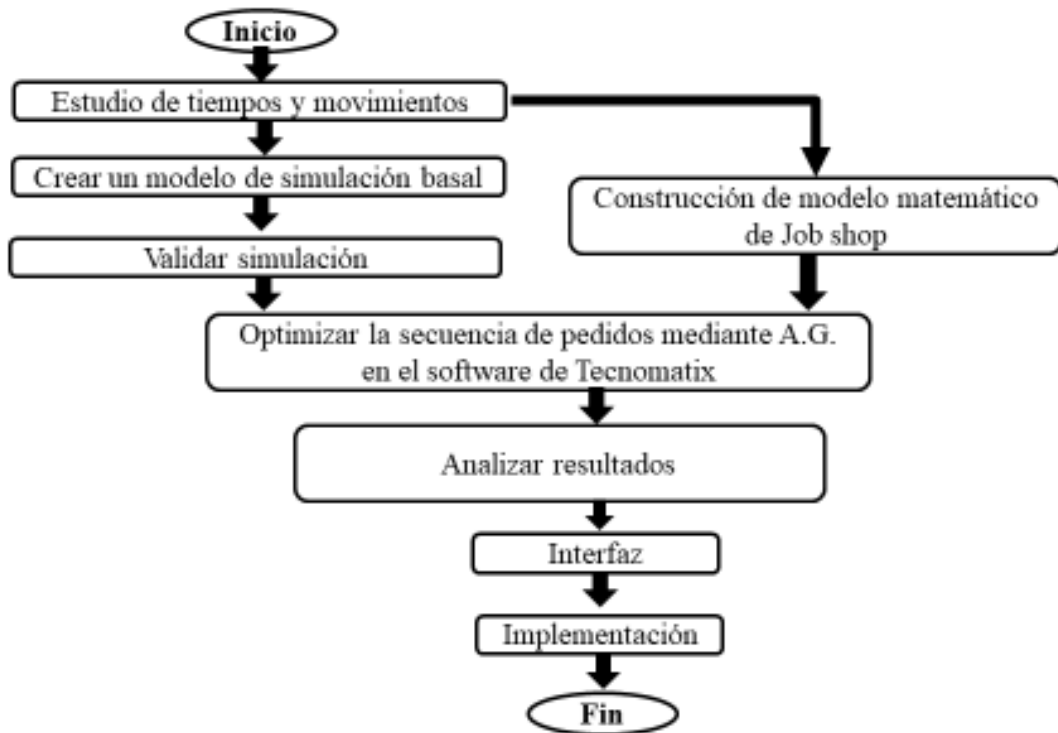


Ilustración 9. Metodología (elaboración propia)

- **Estudio de tiempos y movimientos**

Se tomaron tiempos actuales del proceso o tiempos takt en segundos para introducir estos tiempos en las operaciones de la simulación.

- **Se crea modelo de simulación como actualmente labora la empresa**

Se realizó una simulación en el software Plant Simulation del proceso actual para compararlo después con el modelo de mejorado del proceso; Dentro de la simulación se tomaron en cuenta los tiempos takt y se realizó el experimento con 13 casos de estudio, los cuales pasan por diferentes áreas de producción.

- **Se valida la simulación con la realidad**

La simulación se valida con la situación real en base al tiempo, tomando en cuenta el tiempo de terminación que arrojo el diagrama de Gantt y la simulación, para ya con esto

compararlos con lo que realmente se cronometró en planta según el tiempo de terminación de los 13 casos de experimentación.

- **Modelo matemático de Job shop**

Se realizó un modelo matemático para facilitar la comprensión del problema de Job shop, basándose en otros modelos para que se ajuste mejor al problema.

- **Se simula la respuesta de optimización**

El modelo matemático y el algoritmo genético ayudaron a encontrar una secuencia de operaciones óptima la cual se simulará y se compara con la simulación basal.

- **Análisis de resultados**

Se descartaron algunas herramientas y métodos utilizados dentro de la fase de experimentación, que no ayudaron a encontrar la respuesta óptima; Utilizando el módulo de Algoritmos genéticos dentro del simulador, se logró obtener la secuencia mejorada, la cual al comparar el tiempo de terminación de la simulación basal con la simulación de secuencia optimizada tuvo una mejora significativa.

- **Interfaz gráfica**

Se realizó una interfaz, como entregable, para facilitar la manipulación de datos en el simulador y después se procede a su implementación.

CAPITULO IV

4.1 EXPERIMENTACIÓN

Para iniciar la experimentación primero se obtienen los datos necesarios de la planta, con el fin de realizar un estudio de tiempos y movimientos, habiendo realizado este, se crea un modelo general usando los datos (tiempos en segundos por operación, personal por área y maquinaria utilizada) previamente adquiridos, con el fin de crear una representación matemática del estándar de trabajo.

Habiendo creado la representación del problema (modelo matemático general), se comparan diferentes herramientas con el fin de obtener una mejor respuesta, como:

- Diagrama de Gantt
- Solver de Excel.
- Plant Simulation (software de simulación).

Ya que se haya decidido la mejor opción para de acuerdo al planteamiento del problema, se creó el modelo de simulación de la planta, ya con este terminado se validó mediante la comparación del real contra el simulado.

4.1.1 Toma de datos

En esta primera etapa se tomaron los tiempos en minutos por operación de cada área de la empresa como avance del estudio de tiempos y movimientos en donde se recolectó información como tiempos, personal por área, flujo de producción y maquinaria por área, tomando en cuenta 13 casos de experimentación, como se muestra a continuación:

Tabla 2. Tiempos por pieza en minutos por operación según el caso. (Elaboración propia).

Caso	Operaciones	Tiempo x pza. (min)
Caso 1	Diseño	30
	Corte	3,75
	Plotter	7,5
	Sublimado	7,5
	Costura	7,5
	Empaque	3,75
Caso 2	Corte	0,75
	Costura	2,4
	Bordado	0,9
	Empaque	0,3
Caso 5	Diseño	30
	Tejido	20,25
	Detallado	0,75
	Corte	7,5
	Costura	7,5
	Empaque	0,75
Caso 6	Diseño	30
	Tejido	20
	Detallado	1
	Corte	13
	Costura	14
	Empaque	2
Caso 7	Diseño	30
	Tejido	2,3
	Detallado	1,7
	Costura	1,7
Caso 9	Diseño	30
	Tejido	39
	Corte	15
	Costura	36
	Empaque	1,5
Caso 10	Corte	4,2
	Costura	27
	Empaque	0,75
Caso 12	Costura	3,6
Caso 13	Diseño	30
	Plotter	8,4
Caso 14	Costura	1,8
Caso 15	Diseño	30
	Láser	10
	Bordado	8,8
Caso 16	Diseño	30
	Láser	12
	Bordado	16,8

También se realizó una matriz con tiempos de pieza elaborada por minuto. (Anexos).

Con la información recopilada se construyó un diagrama de Gantt con tiempos reales y otro con una mejora de reducción del tiempo de terminación con solamente acomodados

visuales, el diagrama ayudó a encontrar espacios en blanco, que son tiempos muertos y estos se traducen a menor efectividad.

Para la experimentación se tomaron en cuenta 13 distintos tipos de pedidos que se trabajan en [REDACTED], los cuales son representados con colores diferentes para cada caso, como se muestra a continuación:

CASO 1	[Red]	8 Uniformes
CASO 2	[Blue]	200 Chaquetines
CASO 5	[Purple]	Uniformes (40 chalecos)
CASO 6	[Orange]	Uniformes (30 pants)
CASO 7	[Yellow]	UNIFORMES 35 Cuellos, 70 puños y 35 resortes
CASO 8	[Dark Blue]	UNIFORMES 60 Tiras laterales de pants
CASO 9	[Brown]	100 sweter
CASO 10	[Light Green]	200 Faldas
CASO 12	[Black]	UNIFORMES 70 Cuellos, 140 puños y 40 resortes
CASO 13	[Light Purple]	50 Uniformes basket
CASO 14	[Pink]	400 Resortes
CASO 15	[Magenta]	300 Playeras
CASO 16	[Green]	100 gorras

Ilustración 10. Código de colores para cada caso (elaboración propia).

4.1.2. Descripción del modelo

El modelo matemático con los 13 casos está compuesto por 1220 variables, lo que convierte a este un problema NP-Hard, se experimentó con solo 2 casos de los 13 para poder optimizar con distintas herramientas, donde se obtuvieron las siguientes variables:

TR	OP	Pc1	Cort Automatico	Corte Manual	Plotter 1	Plotter 2	Plotter 3	Calandra	Costura 1	Costura 2	Costura 3	Costura 4	Bordado 1	Bordado 2	Empaque
		M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14
Caso1	1	X1	X11	X21	X31	X41	X51	X61	X71	X81	X91	X101	X111	X121	X131
	2	X2	X12	X22	X32	X42	X52	X62	X72	X82	X92	X102	X112	X122	X132
	3	X3	X13	X23	X33	X43	X53	X63	X73	X83	X93	X103	X113	X123	X133
	4	X4	X14	X24	X34	X44	X54	X64	X74	X84	X94	X104	X114	X124	X134
	5	X5	X15	X25	X35	X45	X55	X65	X75	X85	X95	X105	X115	X125	X135
	6	X6	X16	X26	X36	X46	X56	X66	X76	X86	X96	X106	X116	X126	X136
Caso2	1	X7	X17	X27	X37	X47	X57	X67	X77	X87	X97	X107	X117	X127	X137
	2	X8	X18	X28	X38	X48	X58	X68	X78	X88	X98	X108	X118	X128	X138
	3	X9	X19	X29	X39	X49	X59	X69	X79	X89	X99	X109	X119	X129	X139
	4	X10	X20	X30	X40	X50	X60	X70	X80	X90	X100	X110	X120	X130	X140

Ilustración 11. Variables de experimentación de caso 1 y caso 2.(elaboración propia).

Se adquirieron 21 variables con las cuales se conformó el modelo a experimentar, quedando de la siguiente forma:

Caso 1y 2

Función objetivo:

$$\begin{aligned} \min z = & 14400x_1 + 3600x_{186} + 2250x_{190} + 1800x_{247} + 4500x_{251} + 3600x_{490} \\ & + 3600x_{551} + 3600x_{612} + 3600x_{736} + 3600x_{798} + 7200x_{801} + 3600x_{859} \\ & + 7200x_{862} + 3600x_{920} + 7200x_{923} + 3600x_{981} + 7200x_{984} \\ & + 2700x_{1046} + 2700x_{1107} + 1800x_{1165} + 900x_{1169} \end{aligned}$$

En la función objetivo se coloca min Z como representación del objetivo, que es minimizar el tiempo máximo de las operaciones.

Restricción de asignación:

$$\begin{aligned} 14400x_1 & \geq 1 \\ 3600x_{490} + 3600x_{551} + 3600x_{612} & \geq 1 \\ 3600x_{186} + 1800x_{247} & \geq 1 \\ 3600x_{736} & \geq 1 \\ 3600x_{798} + 3600x_{859} + 3600x_{920} + 3600x_{981} & \geq 1 \\ 1800x_{1165} & \geq 1 \\ 2250x_{190} + 4500x_{251} & \geq 1 \\ 7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} & \geq 2 \\ 2700x_{1046} + 2700x_{1107} & \geq 2 \\ 900x_{1169} & \geq 1 \end{aligned}$$

Las operaciones deben ser asignadas a una sola máquina, en caso de que las máquinas sean iguales (por ejemplo costura o bordado) se asigna a la más disponible.

Restricción de precedencia.

$$\begin{aligned} 14400x_1 \\ 14400x_1 + 3600x_{490} + 3600x_{551} + 3600x_{612} & \geq 1400x_1 \\ 3600x_{186} + 1800x_{247} \end{aligned}$$

$$3600x_{736} + 14400x_1 + 3600x_{490} + 3600x_{551} + 3600x_{612} + 3600x_{186} + 1800x_{247} \\ \geq 14400x_1 + 3600x_{490} + 3600x_{551} + 3600x_{612} + 3600x_{186} + 1800x_{247}$$

$$3600x_{798} + 3600x_{859} + 3600x_{920} + 3600x_{981} + 3600x_{736} + 14400x_1 + 3600x_{490} \\ + 3600x_{551} + 3600x_{612} + 3600x_{186} + 1800x_{247} \\ \geq 3600x_{736} + 14400x_1 + 3600x_{490} + 3600x_{551} + 3600x_{612} + 3600x_{186} \\ + 1800x_{247}$$

$$2250x_{190} + 4500x_{251}$$

$$7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} + 2250x_{190} + 4500x_{251} \\ \geq 2250x_{190} + 4500x_{251}$$

$$2700x_{1046} + 2700x_{1107} + 7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} + 2250x_{190} \\ + 4500x_{251} \\ \geq 7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} + 2250x_{190} + 4500x_{251} \\ \geq 2250x_{190} + 4500x_{251}$$

$$900x_{1169} + 2700x_{1046} + 2700x_{1107} + 7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} \\ + 2250x_{190} + 4500x_{251} \\ \geq 2700x_{1046} + 2700x_{1107} + 7200x_{801} + 7200x_{862} + 7200x_{923} \\ + 7200x_{984} + 2250x_{190} + 4500x_{251} \\ \geq 7200x_{801} + 7200x_{862} + 7200x_{923} + 7200x_{984} + 2250x_{190} + 4500x_{251} \\ \geq 2250x_{190} + 4500x_{251}$$

Esta restricción representa las operaciones que preceden de otra según su orden y si no tienen precedencia son independientes y pueden empezar a trabajar simultáneamente con otras operaciones, como es el caso del área de diseño y corte, las cuales no dependen de otra operación anterior a estas; En caso contrario se tiene el área de plotter, que precede de diseño y sublimado que depende tanto de corte como de plotter.

Se coloca el símbolo de \geq para representar que la sumatoria de las siguientes operaciones debe ser mayor a la operación de la cual preceden.

4.1.3. Elección de herramientas para el modelo

En esta etapa se describe como se selecciona la herramienta más adecuada para encontrar la solución al problema planteado, comparando las siguientes herramientas:

- Solver
- Algoritmos Genéticos
- Plant Simulation

A continuación se describen las restricciones para cada herramienta.

Solver: Siendo este un complemento de Excel para la solución de problemas de optimización se limita a trabajar con problemas que cuentan con pocas variables y no es posible resolver problemas NP-hard. Solver es utilizado para encontrar el mínimo o máximo valor óptimo dentro de una función, celda objetivo, que se compone de limitantes en los valores en otras celdas de una hoja de cálculo, este cuenta con 3 métodos de solución los cuales son:

Método simplex (LP): usado para resolver únicamente problemas lineales

Gradiente generalizado reducido (GRG): Se utiliza para problemas que son no lineales suavizados.

Algoritmo genético: Algoritmos evolucionarios que se utilizan para resolver problemas no suavizados.

Para la experimentación se optó primeramente por utilizar Solver, donde fue necesaria la creación de la matriz de tiempos y de variables de los 16 casos con todas las máquinas de la empresa.

Se realizó una matriz de asignación para cada uno de los 13 casos con el tiempo en segundos por pieza (dependiendo de cuantas piezas sean por caso) de cada operación (j) en cada máquina (k) en donde el caso (i) se pueda realizar, en la máquina en donde el caso o la operación no se realizan se coloca un cero, pero al utilizar el algoritmo genético programado se utiliza un número muy grande (1000000) como la representación para anular esta opción y que este algoritmo no la seleccione.

Se tuvo un total de 1220 variables para los 13 casos.

AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT
Plotter 1	Plotter 2	Plotter 3	Detallado	Calandra	Costura 1	Costura 2	Costura 3	Costura 4	Bordado 1	Bordado 2	Empaque
X526	X587	X648	X709	X770	X831	X892	X953	X1014	X1075	X1136	X1197
X527	X588	X649	X710	X771	X832	X893	X954	X1015	X1076	X1137	X1198
X528	X589	X650	X711	X772	X833	X894	X955	X1016	X1077	X1138	X1199
X529	X590	X651	X712	X773	X834	X895	X956	X1017	X1078	X1139	X1200
X530	X591	X652	X713	X774	X835	X896	X957	X1018	X1079	X1140	X1201
X531	X592	X653	X714	X775	X836	X897	X958	X1019	X1080	X1141	X1202
X532	X593	X654	X715	X776	X837	X898	X959	X1020	X1081	X1142	X1203
X533	X594	X655	X716	X777	X838	X899	X960	X1021	X1082	X1143	X1204
X534	X595	X656	X717	X778	X839	X900	X961	X1022	X1083	X1144	X1205
X535	X596	X657	X718	X779	X840	X901	X962	X1023	X1084	X1145	X1206
X536	X597	X658	X719	X780	X841	X902	X963	X1024	X1085	X1146	X1207
X537	X598	X659	X720	X781	X842	X903	X964	X1025	X1086	X1147	X1208
X538	X599	X660	X721	X782	X843	X904	X965	X1026	X1087	X1148	X1209
X539	X600	X661	X722	X783	X844	X905	X966	X1027	X1088	X1149	X1210
X540	X601	X662	X723	X784	X845	X906	X967	X1028	X1089	X1150	X1211
X541	X602	X663	X724	X785	X846	X907	X968	X1029	X1090	X1151	X1212
X542	X603	X664	X725	X786	X847	X908	X969	X1030	X1091	X1152	X1213
X543	X604	X665	X726	X787	X848	X909	X970	X1031	X1092	X1153	X1214
X544	X605	X666	X727	X788	X849	X910	X971	X1032	X1093	X1154	X1215
X545	X606	X667	X728	X789	X850	X911	X972	X1033	X1094	X1155	X1216
X546	X607	X668	X729	X790	X851	X912	X973	X1034	X1095	X1156	X1217
X547	X608	X669	X730	X791	X852	X913	X974	X1035	X1096	X1157	X1218
X548	X609	X670	X731	X792	X853	X914	X975	X1036	X1097	X1158	X1219
X549	X610	X671	X732	X793	X854	X915	X976	X1037	X1098	X1159	X1220

Ilustración 14. Matriz de variables (elaboración propia).

El problema de asignación con la matriz de tiempos fue comparado con herramientas como Solver, AG y Plant Simulation.

No fue posible resolver los 13 casos con la herramienta de Solver, ya que no cuenta con la capacidad de trabajar con tantas variables, por lo que solo se utilizaron 2 de los 13 casos, los cuales al resolver por medio de simplex (LP) la respuesta de la herramienta fue que no es posible resolver ya que no es un problema lineal.

Posteriormente se probó con Gradiente, el cual, si logro encontrar un óptimo de 35100 segundos, aunque el tiempo computacional fue muy elevado, y utilizando el Algoritmo Genético se redujo el tiempo computacional significativamente, encontrando un óptimo de 29700, el cual es mejor que el encontrado por Gradiente.

Se pudo observar que utilizar la herramienta Solver de Excel no es conveniente ya que el problema no es lineal y cuenta con demasiadas variables.

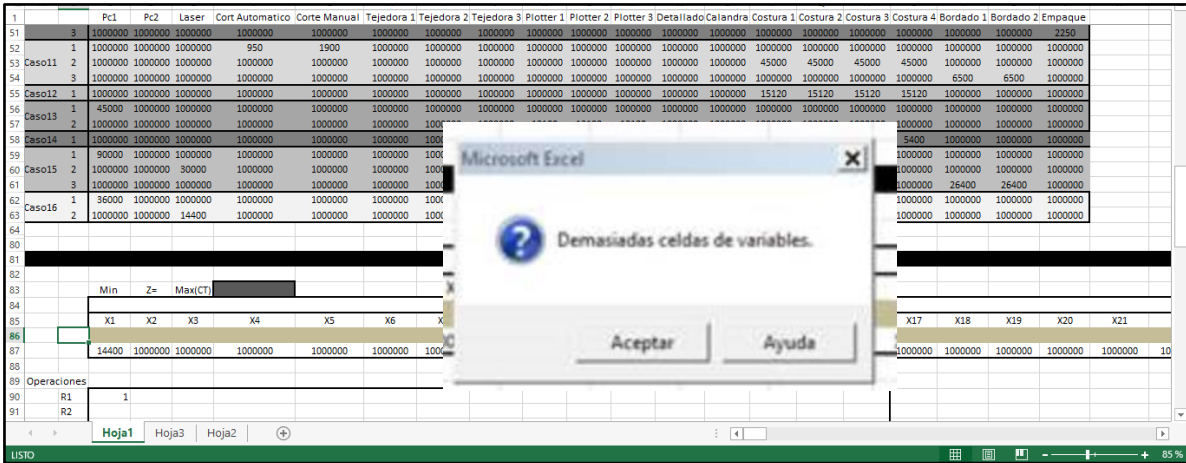


Ilustración 15. Resolviendo por Solver (3 casos).

TR	OP	Pc1	Cort Automatico	Corte Manual	Plotter 1	Plotter 2	Plotter 3	Calandra	Costura 1	Costura 2	Costura 3	Costura 4	Bordado 1	Bordado 2	Empaque		
Caso1	1	X1	X11	X21	X31	ASIGNACIÓN											
	2	X2	X12	X22		ASIGNACIÓN											
	3	X3	X13	14400	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000		
	4	X4	X14														
	5	X5	X15														
	6	X6	X16	Min	Z=	Max(CT)	29700										
Caso2	1	X7	X17														
	2	X8	X18														
	3	X9	X19		1												
	4	X10	X20														
		M1					M2										
		X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16
		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		14400	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	1000000	3600	1000000	1000000	1000000	2

Ilustración 16. Resolviendo por Solver (GRADIENTE).

Resultados de Solver

No se cumplen las condiciones de linealidad requeridas por este Solver de PL.

Conservar solución de Solver

Restaurar valores originales

Volver al cuadro de diálogo de parámetros de Solver

Informes de esqema

Aceptar Cancelar Guardar escenario...

No se cumplen las condiciones de linealidad requeridas por este Solver de PL.

Cree un informe de linealidad para ver dónde está el problema o cambie al motor GRG.

Ilustración 17. Resolviendo por Solver (SIMPLEX).

Resultados de Solver

No se cumplen las condiciones de linealidad requeridas por este Solver de PL.

Conservar solución de Solver

Restaurar valores originales

Volver al cuadro de diálogo de parámetros de Solver

Informes de esqema

Aceptar Cancelar Guardar escenario...

No se cumplen las condiciones de linealidad requeridas por este Solver de PL.

Cree un informe de linealidad para ver dónde está el problema o cambie al motor GRG.

Ilustración 18. Resolviendo por Solver (ALGORITMO GENÉTICO).

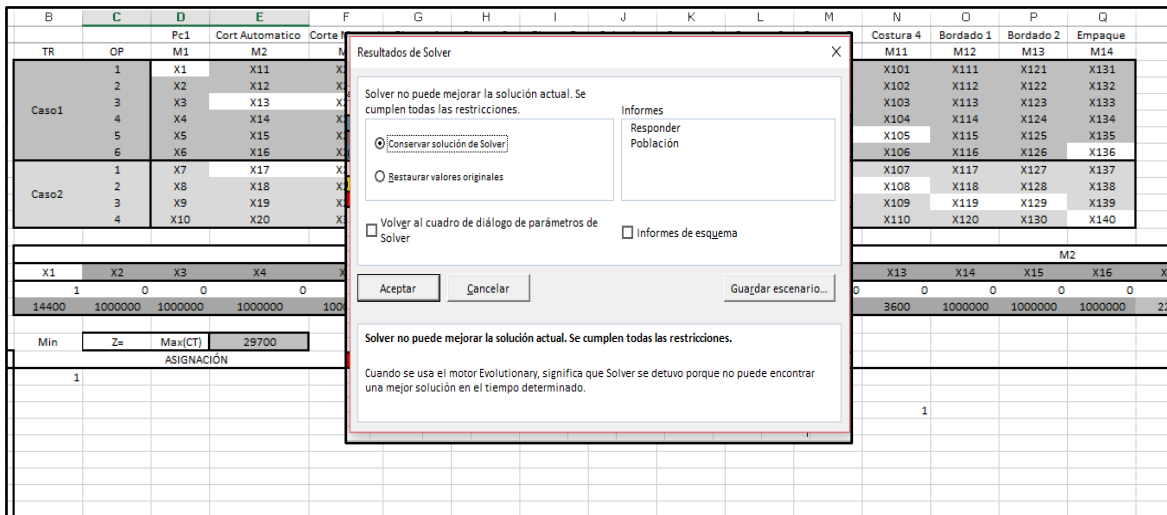


Ilustración 19. Resolviendo por Solver (ALGORITMO GENÉTICO).

Habiendo encontrado que Solver tiene limitaciones en la cantidad de variables, se recurre al uso de otras herramientas mencionadas en la literatura para la optimización, en donde se encuentre a los Algoritmos genéticos como mejor alternativa de solución.

Plant Simulation de Tecnomatix, es una herramienta de simulación de eventos discretos que ayuda a la creación de modelos logísticos (de producción) para su optimización.

El software cuenta con un módulo de Algoritmos genéticos (GA wizard) la cual integra algoritmos genéticos en el modelo de simulación y así optimizar la secuencia y reducir el tiempo de operaciones.

4.1.4 Simulación

Descripción del modelo de simulación

La simulación se construyó en base al layout de la planta, ésta se divide en 2 pisos, para esto en la simulación se construyen 2 frames o 2 capas de simulación para representar cada uno de los niveles de la planta. En cada uno de los niveles se encuentran diferentes áreas productivas, mismas que se describen en consecuente.

En el primer piso se encuentra localizadas las áreas de Diseño, Bordado, Serigrafía, Tejido y Empaque. En el área de diseño se cuenta con 2 equipos (computadoras) para crear los diseños los cuales son enviados electrónicamente a los plotters 1 localizado en el primer piso y 2 en el segundo, a la maquina láser que se encuentra en el primer piso en esta misma área, y a tejedoras que al igual se encuentran el primer piso, sin embargo están en un área separada, los diseños creados son mandados según sea la operación que corresponda, esta área cuenta con 1 persona que es la responsable de hacer todas las operaciones.

El área de bordado cuenta con 2 máquinas multi-bordadoras las cuales son operadas por una persona. El área de serigrafía cuenta con 3 máquinas tejedoras las cuales son operadas por una sola persona, esta a su vez realiza el mantenimiento de todas las máquinas de las empresas, ya que las maquinas tejedoras operan automáticamente, y solo requieres de supervisión esporádica de su proceso.

Por último el área de empaque se localiza en el primer piso, esta cuenta con una persona que realiza el embalaje del producto final. En la *ilustración 20* se muestra el layout conceptual con las áreas descritas anteriormente.

En el segundo piso se encuentra el área de corte, sublimado, detallado, plotter y costura, las cuales se describen a continuación, el área de corte cuenta con dos máquinas, una de ellas manual la cual es utilizada para lotes pequeños y una maquina automática para lotes de gran tamaño, estas son operadas por una persona y no pueden ser operadas al mismo tiempo.

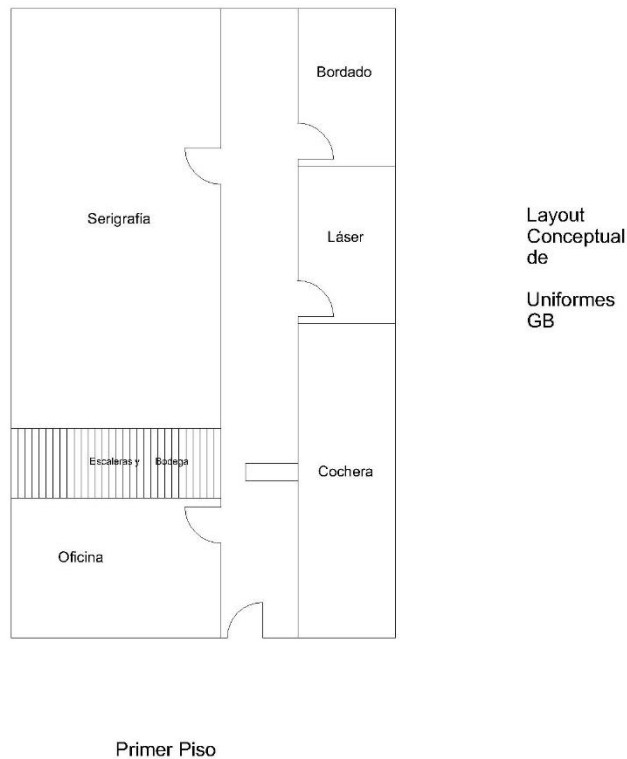


Ilustración 20 Layout Conceptual del primer piso.

El área de sublimado cuenta con una calandra, máquina que imprime la tinta del papel en tela, por lo que sublimado depende del área de plotter y corte para poder trabajar, esta área cuenta con 3 personas, donde 2 operan directamente en la máquina y un líder que se encarga de proveer el material, entregar el producto terminado en sublimado y mantener limpia el área.

El área de detallado es donde se da acabado final a las telas que salen del área de tejido o de costura, esta operación es realizada por una sola persona.

El área de plotter que cuenta con 2 máquinas en el segundo piso y una maquina en el primer piso en área de Diseño, éstas imprimen el diseño que proporciona el cliente o se crea en el área correspondiente, una persona es la encargada de esta área la cual debe estar pendiente del proceso de impresión ya que el papel se puede arrugar, romper o manchar de tinta, aparte debe cortar el papel pieza por pieza o en tiras (según el tamaño) para que sea más sencillo manipular para el área de sublimado.

Por último el área de costura la cual cuenta con 4 máquinas costureras y cada máquina es operada por una persona. Las áreas anteriormente descritas se pueden observar en la *ilustración 21*.

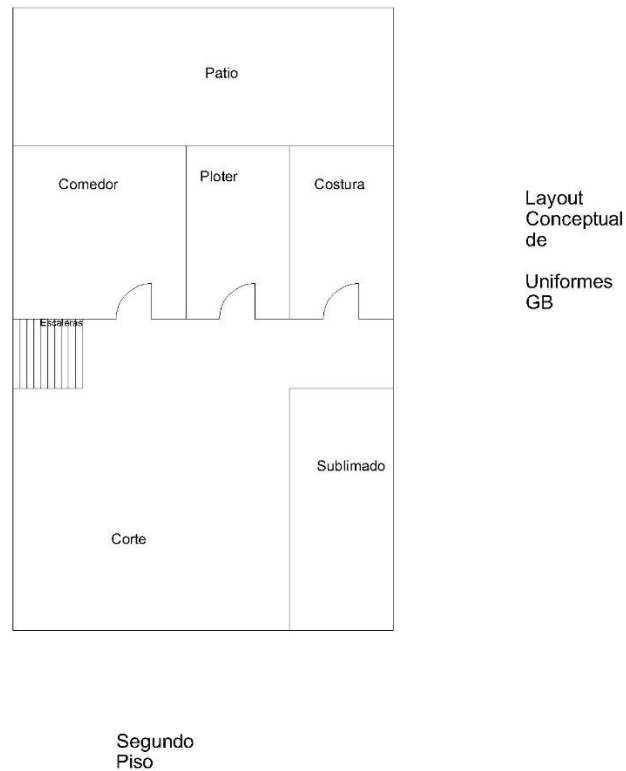


Ilustración 21 Layout conceptual de planta segundo piso

El horario general de la planta es de 8:00 am a 5:00 pm; Los horarios de comida son de media hora y se manejan por grupos, donde los horarios son: el área de costura, corte, detallado, empaque y bordado tienen de 12:30 pm a 1:00 pm para comer, posteriormente, el área de diseño, tejido, láser, plotter y sublimado de 1:00 pm a 1:30 pm.

Dentro de la simulación se utilizan 20 single proc(estaciones de trabajo que procesan pieza por pieza), para representar a las máquinas y cada estación con sus 19 buffers (Estación de espera de piezas temporales) son almacenes de los cuales se dejan o recogen piezas, cuenta con estaciones de trabajo para cada máquina y los trabajadores que corresponden a cada área, esto se logra mediante un workerpool representa la sala del total de trabajadores en la planta, así mismo se coloca los caminos por los cuales transitarán los operadores (footpath) para

indicar el recorrido que debe seguir, y un bróker (mediador de servicios requeridos) el cual se encarga de asignar, contando con 6 workerpool y cada uno cuenta con su propio bróker.

Para la simulación se colocan horarios a cada una de las estaciones, para esto el software cuenta con un calendario llamado “Shift-Calendar”, el cual realiza el control de horarios para las diferentes estaciones, además de esto esta simulación cuenta con 8 Flow Controls (permite modelar estrategias comunes para dividir y reunir el flujo de materiales en el espacio de trabajo, no procesa las entidades que se mueven, sólo las distribuye entre los objetos) los cuales ayudan a establecer el flujo de material, es decir, designa el material a cada una de las estaciones dependiendo de la secuencia de producción. Se cuenta con un source (produce Mu’s en una sola estación, más no produce tiempo), el cual es como un proveedor de material o piezas que entran al sistema, es a donde inicialmente llegan las ordenes de los clientes y al terminar el proceso de producción el producto final llega a un drain (saca las piezas de trabajo que ya han sido procesadas y enviadas al proceso por el source) que viene siendo el área de venta.

Se utilizaron 2 tablas para administrar la información de tiempos y de llegadas, y en cuanto a la eficiencia de trabajadores se tiene ya establecido el 100%.

Validación.

Para validar la simulación se utilizaron 2 de los 16 casos que se utilizaron en la experimentación, el caso 1 y 12.

En el caso 1 se trabajaron 8 uniformes deportivos; Se realiza el diseño del uniforme en Diseño 1, se imprimen las 8 piezas de diferentes tamaños en el área de plotter y simultáneamente se corta la tela para posteriormente pasar al área de sublimado las piezas de corte y de plotter; Ya impreso el papel en la tela pasan a costura para dar el acabado final y después al área de empaque para venta.

En el caso 12 se realizaron 70 tiras personalizadas que son utilizadas para los uniformes donde solo son procesados en el área de costura.

Resultados de simulación

Simulation time: 4:12:04.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	3:30:15.5221	1	0	100.00%	0.00%	0.00%	85.61%	
Drain	caso12	4:12:02.6667	1	0	100.00%	0.00%	0.00%	99.98%	

Ilustración 22. Tiempos y resultados de simulación de caso 1 y 12.

El tiempo total de terminación del caso 1 y 12 es de 4 horas, 12 minutos y 4 segundos.

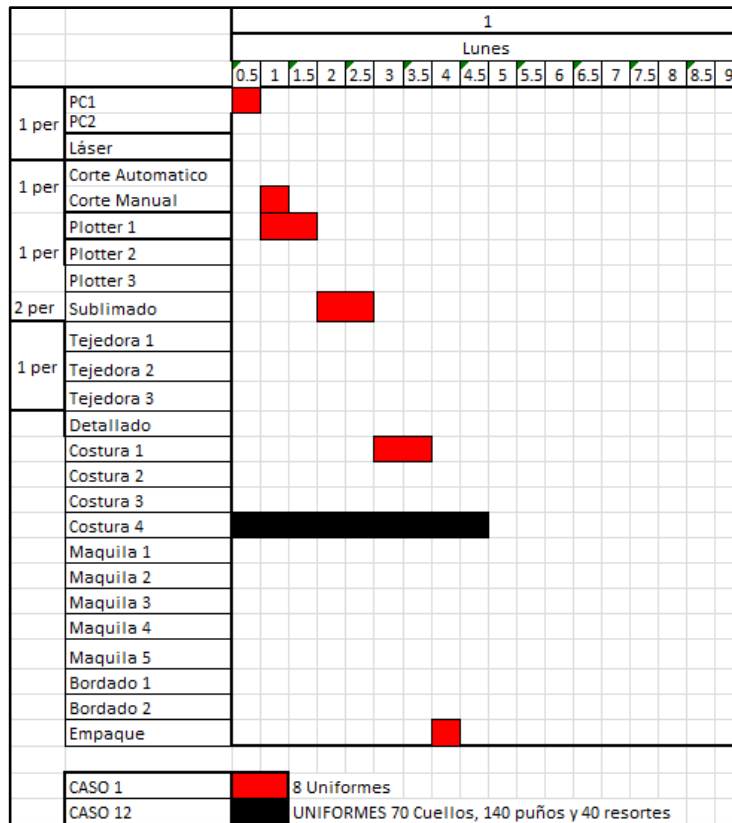


Ilustración 23. Job shop (tipo gantt) de caso 1 y 12.

Resultados de Job shop tipo Gantt.

Su tiempo de terminación total es de 4 horas y media (redondeando).

Primero se comparó el resultado obtenido en el diagrama de Gantt con el de la simulación, en esta parte se concluye que tienen un mismo tiempo de terminación por lo que de esta forma validamos que la simulación arroja resultados cercanos a la realidad.

Posteriormente se comparó la simulación con el modelo matemático en base a la representación lo más real posible, se pudo observar que el modelo matemático no toma en cuenta diversas consideraciones como horas de comidas, idas al baño, etc..., por lo que no se obtiene una representación lo más real posible de la realidad, por lo que el modelo es complejo de resolver por sus múltiples variables que elevan su dificultad para ser resueltos y se tienen muchas posibles combinaciones, por lo que la herramienta Solver, no pudo ejecutarse por el gran número de variables.

4.2 RESULTADOS

4.2.1. Simulación Basal

Habiendo validado la simulación se procede a realizar la misma, utilizando los casos de estudio, para esto se incluye una tabla con los casos que se utilizaron en la experimentación.

	time 1	object 2	integer 3
string	Delivery Time	MU	Number
1	0.0000	.MUs.caso1	1
2	0.0000	.MUs.caso2	1
3	0.0000	.MUs.caso5	1
4	0.0000	.MUs.caso6	1
5	0.0000	.MUs.caso7	1
6	0.0000	.MUs.caso8	1
7	0.0000	.MUs.caso9	1
8	0.0000	.MUs.caso9	1
9	0.0000	.MUs.caso9	1
10	0.0000	.MUs.caso10	1
11	0.0000	.MUs.caso12	1
12	0.0000	.MUs.caso13	1
13	0.0000	.MUs.caso14	1
14	0.0000	.MUs.caso15	1
15	0.0000	.MUs.caso16	1

Ilustración 24. Casos de estudio con su cantidad de piezas y tiempos de entrada al proceso.

La simulación comienza con fecha de miércoles 28 de junio de 2017 a las 08:00 horas como se puede observar en la *Ilustración 25*.

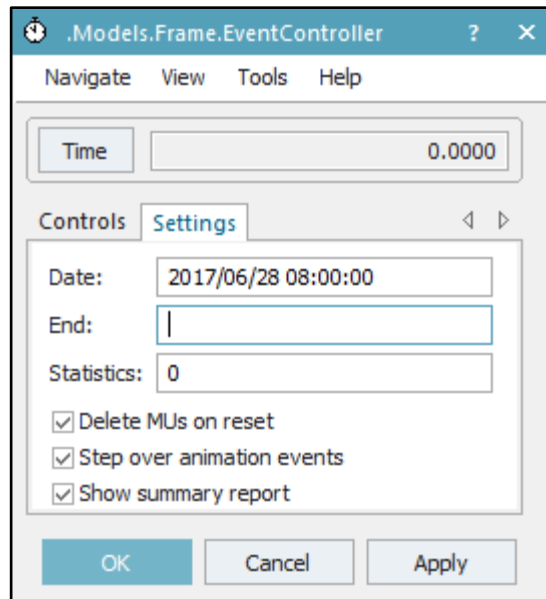


Ilustración 25. Fecha en que comienza la simulación y el tiempo que transcurrió.

Habiendo definido las máquinas y sus cargas con sus tiempos respectivos, así como las diferentes rutas del material se procede a realizar la simulación, en donde se puede observar en la *Ilustración 26* que para el término del total de los casos toma un tiempo de 29 días, con 07 horas y 41 minutos con 56 segundos.

Simulation time: 29:07:41:56.0000

Cumulated Statistics of the Parts which the Drain Deleted									
Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	4:00:19.1386	1	0	100.00%	0.00%	0.00%	74.90%	
Drain	caso10	16:07:01:05.1327	1	0	99.62%	0.00%	0.38%	27.24%	
Drain	caso12	4:12:02.6667	1	0	100.00%	0.00%	0.00%	99.98%	
Drain	caso13	1:00:00:34.4720	1	0	97.89%	0.00%	2.11%	31.24%	
Drain	caso14	1:03:30:02.6667	1	0	100.00%	0.00%	0.00%	43.64%	
Drain	caso15	15:04:30:28.2832	1	0	99.59%	0.00%	0.41%	26.34%	
Drain	caso16	16:00:10:28.2832	1	0	49.89%	0.00%	50.11%	13.06%	
Drain	caso2	1:05:30:23.0442	1	0	100.00%	0.00%	0.00%	45.75%	
Drain	caso5	3:01:01:05.1327	1	0	98.62%	0.00%	1.38%	34.24%	
Drain	caso6	4:06:00:58.0829	1	0	99.01%	0.00%	0.99%	34.32%	
Drain	caso7	3:06:09:41.7053	1	0	97.43%	0.00%	2.57%	34.11%	
Drain	caso8	4:05:41:41.7053	1	0	70.98%	0.00%	29.02%	22.32%	
Drain	caso9	29:07:41:54.7496	1	0	92.11%	0.00%	7.89%	22.84%	

Ilustración 26. Resultado de simulación de los 13 casos.

De la simulación se pueden obtener resultados como los que se muestran en las Ilustraciones 30 y 31, caso 1 termina primero con un tiempo en ser procesado de 4 horas, después caso 12,13, 14, 2, 5, 7,8 y 6 terminan en 1 día aproximadamente (promediando), los casos 15 y 16 toman entre 15 y 16 días, sin embargo, el caso 9 es el que más tiempo toma en terminar con 29 días, 7 horas y 42 segundos.

Drain	Production							Transport							Storage							
	Working	Set-up	Waiting	Stopped	Failed	Paused	Sum	Working	Set-up	Waiting	Stopped	Failed	Paused	Sum	Working	Set-up	Waiting	Stopped	Failed	Paused	Sum	
caso1	78.15%	0.00%	21.85%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
caso10	27.24%	0.00%	0.13%	0.00%	0.00%	72.25%	99.62%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.38%	0.00%	0.00%	0.00%	0.00%	0.38%
caso12	99.98%	0.00%	0.02%	0.00%	0.00%	0.00%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
caso13	29.40%	0.00%	1.97%	0.00%	0.00%	60.76%	92.13%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	7.87%	0.00%	0.00%	0.00%	0.00%	7.87%
caso14	43.64%	0.00%	0.00%	0.00%	0.00%	56.36%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
caso15	26.42%	0.00%	0.05%	0.00%	0.00%	73.35%	99.81%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.19%	0.00%	0.00%	0.00%	0.00%	0.19%
caso16	13.62%	0.00%	0.05%	0.00%	0.00%	34.21%	47.87%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	52.13%	0.00%	0.00%	0.00%	0.00%	52.13%
caso2	45.75%	0.00%	0.02%	0.00%	0.00%	54.23%	100.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
caso5	14.37%	0.00%	1.23%	0.00%	0.00%	38.59%	54.18%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	45.82%	0.00%	0.00%	0.00%	0.00%	45.82%
caso6	20.10%	0.00%	0.19%	0.00%	0.00%	64.53%	84.83%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	15.17%	0.00%	0.00%	0.00%	0.00%	15.17%
caso7	6.88%	0.00%	0.36%	0.00%	0.00%	25.66%	32.89%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	67.11%	0.00%	0.00%	0.00%	0.00%	67.11%
caso8	9.02%	0.00%	0.77%	0.00%	0.00%	29.92%	39.71%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	60.29%	0.00%	0.00%	0.00%	0.00%	60.29%
caso9	26.53%	0.00%	1.11%	0.00%	0.00%	72.23%	99.87%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.13%	0.00%	0.00%	0.00%	0.00%	0.13%

Ilustración 27. Porcentaje de trabajo, esperas, etc.... de los casos en cuanto a producción, transporte y el almacenado.

Drain	Life Time				Exit Time				Throughput		
	Mean Value	Standard Deviation	Minimum	Maximum	Mean Value	Standard Deviation	Minimum	Maximum	Total	Per Hour	Per Day
caso1	4:00:19.1386	0.0000	4:00:19.1386	4:00:19.1386	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso10	16:07:01:05.1327	0.0000	16:07:01:05.1327	16:07:01:05.1327	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso12	4:12:02.6667	0.0000	4:12:02.6667	4:12:02.6667	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso13	1:00:00:34.4720	0.0000	1:00:00:34.4720	1:00:00:34.4720	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso14	1:03:30:02.6667	0.0000	1:03:30:02.6667	1:03:30:02.6667	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso15	15:04:30:28.2832	0.0000	15:04:30:28.2832	15:04:30:28.2832	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso16	16:00:10:28.2832	0.0000	16:00:10:28.2832	16:00:10:28.2832	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso2	1:05:30:23.0442	0.0000	1:05:30:23.0442	1:05:30:23.0442	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso5	5:01:01:05.1327	0.0000	5:01:01:05.1327	5:01:01:05.1327	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso6	6:06:00:58.0829	0.0000	6:06:00:58.0829	6:06:00:58.0829	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso7	5:06:09:41.7053	0.0000	5:06:09:41.7053	5:06:09:41.7053	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso8	6:05:41:41.7053	0.0000	6:05:41:41.7053	6:05:41:41.7053	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497
caso9	29:07:41:54.7496	0.0000	29:07:41:54.7496	29:07:41:54.7496	0.0000	0.0000	0.0000	0.0000	1	0.0014210624	0.034105497

Ilustración 28. Tiempo promedio por caso.

Lo ideal, para reducir el tiempo de terminación de todos los casos, sería repartir la carga de trabajo equitativamente a las máquinas iguales, pero el simulador tiene como política enviar las piezas a la máquina más disponible.

Como trabajo futuro, el tiempo de procesamiento puede mejorar si los casos se dividen en lotes, repartiendo así cierto número de piezas del mismo caso a las máquinas iguales que estén disponibles, y esto es conocido como balanceo de líneas.

Se experimentó, además, con 19 y 25 casos respectivamente, realizando la simulación basal y después la optimizada, donde se pueden apreciar los resultados a continuación.

4.2.1.1. Simulación basal con 19 casos de estudio.

Simulation time: 39:03:37:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	4:00:19.1386	1	0	100.00%	0.00%	0.00%	74.90%	
Drain	caso10	18:07:01:05.1327	1	0	99.66%	0.00%	0.34%	24.26%	
Drain	caso12	4:12:02.6667	1	0	100.00%	0.00%	0.00%	99.98%	
Drain	caso13	1:00:00:34.4720	1	0	97.89%	0.00%	2.11%	31.24%	
Drain	caso14	1:03:30:02.6667	1	0	100.00%	0.00%	0.00%	43.64%	
Drain	caso15	15:08:45:05.1327	1	0	98.44%	0.00%	1.56%	26.03%	
Drain	caso16	18:01:25:05.1327	1	0	55.30%	0.00%	44.70%	11.57%	
Drain	caso17	25:06:43:54.7496	1	0	72.34%	0.00%	27.66%	17.24%	
Drain	caso18	7:04:19:09.1327	1	0	81.42%	0.00%	18.58%	17.87%	
Drain	caso19	19:02:34:39.0386	1	0	72.15%	0.00%	27.85%	17.35%	
Drain	caso2	1:05:30:23.0442	1	0	100.00%	0.00%	0.00%	45.75%	
Drain	caso20	15:07:15:05.1327	1	0	91.14%	0.00%	8.86%	22.79%	
Drain	caso21	35:00:00:28.2832	1	0	65.30%	0.00%	34.70%	16.73%	
Drain	caso22	39:03:37:28.2832	1	0	46.22%	0.00%	53.78%	10.94%	
Drain	caso5	6:04:15:05.1327	1	0	38.46%	0.00%	61.54%	16.86%	
Drain	caso6	7:07:33:05.1327	1	0	84.88%	0.00%	15.12%	19.94%	
Drain	caso7	5:06:09:41.7053	1	0	98.41%	0.00%	1.59%	21.13%	
Drain	caso8	6:08:15:03.7994	1	0	78.94%	0.00%	21.06%	14.91%	
Drain	caso9	29:07:41:54.7496	1	0	85.57%	0.00%	14.43%	22.91%	

Ilustración 29. Resultados de simulación basal con 19 casos.

Drain	Production						Sum
	Working	Set-up	Waiting	Stopped	Failed	Paused	
caso1	74.90%	0.00%	25.10%	0.00%	0.00%	0.00%	100.00%
caso10	24.26%	0.00%	0.12%	0.00%	0.00%	75.28%	99.66%
caso12	99.98%	0.00%	0.02%	0.00%	0.00%	0.00%	100.00%
caso13	31.24%	0.00%	2.10%	0.00%	0.00%	64.56%	97.89%
caso14	43.64%	0.00%	0.00%	0.00%	0.00%	56.36%	100.00%
caso15	26.03%	0.00%	0.14%	0.00%	0.00%	72.27%	98.44%
caso16	11.57%	0.00%	0.12%	0.00%	0.00%	43.61%	55.30%
caso17	17.24%	0.00%	0.55%	0.00%	0.00%	54.55%	72.34%
caso18	17.87%	0.00%	0.00%	0.00%	0.00%	63.54%	81.42%
caso19	17.35%	0.00%	0.18%	0.00%	0.00%	54.63%	72.15%
caso2	45.75%	0.00%	0.02%	0.00%	0.00%	54.23%	100.00%
caso20	22.79%	0.00%	0.00%	0.00%	0.00%	68.35%	91.14%
caso21	16.73%	0.00%	0.06%	0.00%	0.00%	48.51%	65.30%
caso22	10.94%	0.00%	0.11%	0.00%	0.00%	35.17%	46.22%
caso5	16.86%	0.00%	0.35%	0.00%	0.00%	21.25%	38.46%
caso6	19.94%	0.00%	1.99%	0.00%	0.00%	62.94%	84.88%
caso7	21.13%	0.00%	1.98%	0.00%	0.00%	75.30%	98.41%
caso8	14.91%	0.00%	1.96%	0.00%	0.00%	62.07%	78.94%
caso9	22.91%	0.00%	0.28%	0.00%	0.00%	62.38%	85.57%

Ilustración 30. Porcentaje de trabajo, esperas, pausas, etc... de los resultados de 19 casos de estudio.

Para la simulación basal de 19 casos, se toma un tiempo de terminación de todos los casos de 39 días, 3 horas, 37 minutos y 30 segundos, siendo el caso 22 el que más tiempo se lleva producirse, ya que cuenta con mucho tiempo de espera, como se puede observar en la ilustración 33 es el caso que menor porcentaje de trabajo tiene, siendo este de 10.94%.

4.2.1.2 Simulación basal con 26 casos de estudio.

En la simulación basal con 26 casos, se puede observar que el tiempo de terminación de todos los casos es de 56 días, 3 horas, 30 minutos y 30 segundos, siendo el caso 29 el que toma más tiempo en producirse.

Simulation time: 56:03:30:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	5:41:41.0501	1	0	79.10%	0.00%	20.90%	52.68%	
Drain	caso10	18:07:01:05.1327	1	0	99.66%	0.00%	0.34%	24.26%	
Drain	caso12	4:12:02.6667	1	0	100.00%	0.00%	0.00%	99.98%	
Drain	caso13	1:00:00:34.4720	1	0	97.89%	0.00%	2.11%	31.24%	
Drain	caso14	1:03:30:02.6667	1	0	100.00%	0.00%	0.00%	43.64%	
Drain	caso15	15:08:45:05.1327	1	0	98.44%	0.00%	1.56%	26.03%	
Drain	caso16	18:01:25:05.1327	1	0	55.30%	0.00%	44.70%	11.57%	
Drain	caso17	25:06:58:54.7496	1	0	72.40%	0.00%	27.60%	17.24%	
Drain	caso18	7:04:19:09.1327	1	0	81.42%	0.00%	18.58%	17.87%	
Drain	caso19	19:02:34:39.0386	1	0	72.15%	0.00%	27.85%	17.35%	
Drain	caso2	1:05:30:23.0442	1	0	100.00%	0.00%	0.00%	45.75%	
Drain	caso20	15:07:15:05.1327	1	0	91.14%	0.00%	8.86%	22.79%	
Drain	caso21	35:00:00:28.2832	1	0	65.30%	0.00%	34.70%	16.73%	
Drain	caso22	39:03:37:28.2832	1	0	46.22%	0.00%	53.78%	10.94%	
Drain	caso23	25:06:21:54.7496	1	0	36.89%	0.00%	63.11%	8.06%	
Drain	caso24	1:01:55:02.6667	1	0	100.00%	0.00%	0.00%	40.19%	
Drain	caso25	6:01:29:47.0386	1	0	14.39%	0.00%	85.61%	4.08%	
Drain	caso26	22:04:56:54.7496	1	0	20.30%	0.00%	79.70%	5.88%	
Drain	caso27	21:04:08:41.7053	1	0	13.58%	0.00%	86.42%	4.26%	
Drain	caso28	57:02.6667	1	0	100.00%	0.00%	0.00%	99.92%	
Drain	caso29	56:03:30:28.2832	1	0	50.02%	0.00%	49.98%	12.76%	
Drain	caso5	6:06:21:05.1327	1	0	38.25%	0.00%	61.75%	16.63%	
Drain	caso6	7:07:33:05.1327	1	0	76.33%	0.00%	23.67%	19.94%	
Drain	caso7	5:06:09:41.7053	1	0	98.41%	0.00%	1.59%	21.13%	
Drain	caso8	7:00:51:03.7994	1	0	79.77%	0.00%	20.23%	13.44%	
Drain	caso9	29:07:41:54.7496	1	0	85.57%	0.00%	14.43%	22.91%	

Ilustración 31. Resultados de simulación con 26 casos de estudio.

4.2.2. Resultados de simulación optimizada.

La simulación ayudo a observar el comportamiento del proceso y así mejorar la secuencia.

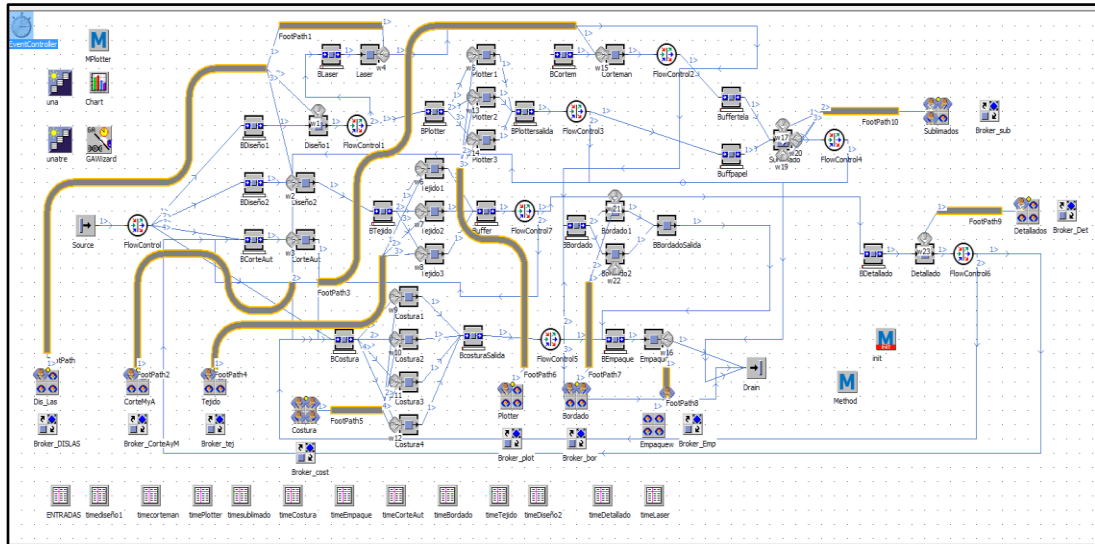


Ilustración 32. Bosquejo de simulación.

Simulation time: 17:06:40:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	3:30:15.5221	1	0	100.00%	0.00%	0.00%	85.61%	<div style="width: 85.61%; height: 10px; background-color: green;"></div>
Drain	caso10	16:07:01:05.1327	1	0	99.62%	0.00%	0.38%	27.24%	<div style="width: 99.62%; height: 10px; background-color: green;"></div>
Drain	caso12	4:12:02.6667	1	0	100.00%	0.00%	0.00%	99.98%	<div style="width: 100%; height: 10px; background-color: green;"></div>
Drain	caso13	1:01:50:34.4720	1	0	90.95%	0.00%	9.05%	29.02%	<div style="width: 90.95%; height: 10px; background-color: green;"></div>
Drain	caso14	1:03:30:02.6667	1	0	100.00%	0.00%	0.00%	43.64%	<div style="width: 100%; height: 10px; background-color: green;"></div>
Drain	caso15	17:06:10:28.2832	1	0	87.77%	0.00%	12.23%	23.18%	<div style="width: 87.77%; height: 10px; background-color: green;"></div>
Drain	caso16	7:08:20:28.2832	1	0	100.00%	0.00%	0.00%	28.45%	<div style="width: 100%; height: 10px; background-color: green;"></div>
Drain	caso2	1:05:30:23.0442	1	0	100.00%	0.00%	0.00%	45.75%	<div style="width: 100%; height: 10px; background-color: green;"></div>
Drain	caso5	10:02:41:58.0829	1	0	78.64%	0.00%	21.36%	20.27%	<div style="width: 78.64%; height: 10px; background-color: green;"></div>
Drain	caso6	17:01:51:58.0829	1	0	47.37%	0.00%	52.63%	11.80%	<div style="width: 47.37%; height: 10px; background-color: green;"></div>
Drain	caso7	7:08:09:41.7053	1	0	99.52%	0.00%	0.48%	25.92%	<div style="width: 99.52%; height: 10px; background-color: green;"></div>
Drain	caso8	10:03:41:41.7053	1	0	31.06%	0.00%	68.94%	10.13%	<div style="width: 31.06%; height: 10px; background-color: green;"></div>
Drain	caso9	11:14:37:17.1053	3	0	79.38%	0.00%	20.62%	22.38%	<div style="width: 79.38%; height: 10px; background-color: green;"></div>

Ilustración 33. Tiempo total transcurrido en la simulación con los 13 casos con el caso 9 loteado en 3 partes iguales.

La simulación sin optimización y loteada arrojó un tiempo de terminación de 17 días, 06 horas ,40 minutos y 30 segundos.

Con el fin de mejorar aún más el resultado, se dividieron los 16 casos en lotes, donde por ejemplo, el caso 9 se dividió en 3 partes iguales, así al lotear se da opciones al simulador para repartir el trabajo, aprovechando las máquinas disponibles y mejorar los tiempos.

Cada caso con sus respectivas operaciones y tiempos se optimiza mediante un Algoritmo genético incluido en el software de simulación, el cual ayudó a obtener una secuencia mejorada

con el fin de reducir el tiempo de terminación, en el módulo de Algoritmos genéticos del software se utilizaron parámetros de población, los cuales fueron:

NTGen (Número total de generaciones) = 50

Tamaño de generación o NTI (Número total de individuos) = 20.

.Models.Fram

Simulation time: 14:06:34:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	5:28:00.7552	2	0	84.68%	0.00%	15.32%	27.44%	
Drain	caso10	4:15:23:08.8717	4	0	90.74%	0.00%	9.26%	23.90%	
Drain	caso12	2:51:02.6667	2	0	73.68%	0.00%	26.32%	73.67%	
Drain	caso13	6:13:04.4720	2	0	77.78%	0.00%	22.22%	60.31%	
Drain	caso14	2:53:17.6667	8	0	51.94%	0.00%	48.06%	51.93%	
Drain	caso15	5:21:20:12.1053	6	0	42.82%	0.00%	57.18%	11.32%	
Drain	caso16	8:10:44:57.3998	4	0	24.32%	0.00%	75.68%	6.19%	
Drain	caso2	1:00:59:43.9204	4	0	15.27%	0.00%	84.73%	13.50%	
Drain	caso5	9:05:57:06.9718	3	0	25.09%	0.00%	74.91%	5.27%	
Drain	caso6	8:04:13:24.7496	2	0	32.99%	0.00%	67.01%	11.27%	
Drain	caso7	9:07:42:56.7053	2	0	10.72%	0.00%	89.28%	3.34%	
Drain	caso8	8:11:28:11.7053	4	0	20.51%	0.00%	79.49%	4.53%	
Drain	caso9	8:10:43:27.8330	3	0	91.34%	0.00%	8.66%	23.42%	

Ilustración 34. Resultados de simulación optimizada con 16 casos de estudio.

El algoritmo genético logra reducir el tiempo de procesamiento, encontrando un mejor tiempo de 14 días, 6 horas, 34 minutos y 30 segundos, como se muestra en la *Ilustración 34*.

Best solutions of the sequence problems

.Models.Frame.ENTRADAS
24, 15, 5, 4, 36, 6, 46, 37, 9, 28, 13, 40, 44, 43, 22, 19, 33, 18, 48, 16, 29, 21, 1, 45, 3, 12, 26, 2, 10, 11, 8, 47, 41, 27, 31, 30, 39, 17, 14, 38, 7, 34, 25, 23, 20, 32, 42, 35

10,8,2,2,14,2,16,14,5,12,7,15,15,15,9,9,14,8,16,8,13,9,1,16,2,6,10,1,5,6,5,16,15,12,14,13,15,8,7,14,5,14,10,10,9,14,15,1

Ilustración 35. El mejor individuo (secuencia óptima, en posiciones de acuerdo a cada caso).

En base con el resultado que se generó por medio del algoritmo genético, tomando en cuenta la posición correspondiente a cada caso de estudio, lo primero que debe entrar a

producción es el segundo lote del caso 10, siguiendo el primer lote del caso 8, después el tercer lote del caso 2, segundo lote del caso 2, el sexto lote de caso 14, el cuarto lote de caso 2, el segundo lote de caso 16, el séptimo lote de caso 14, etc., como se muestra en la *Ilustración 35*.

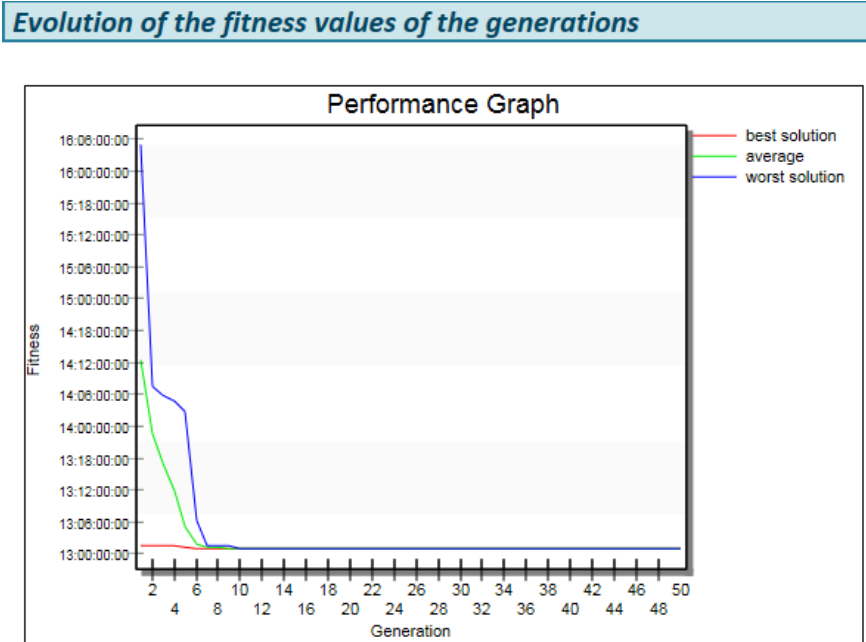


Ilustración 36. Gráfica de convergencia.

En la gráfica de convergencia, se puede observar la mejor (---), promedio (---) y peor (---) solución y como converge en la generación 10.

El algoritmo genético realiza 1694 corridas de simulación y genera 1980 individuos.

El algoritmo genético y el loteo de casos ayudaron significativamente a reducir el tiempo de terminación de 29 días, con 07 horas y 41 minutos con 56 segundos a 14 días, 6 horas, 34 minutos y 30 segundos.

4.2.2.1. Simulación optimizada con 19 casos de estudio.

Se realizó la simulación optimizada con lotes dependiendo el número de piezas por caso con un NTGen de 30 generaciones y un NTI de 20.

Simulation time: 28:07:30:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	3:01:08.1388	2	0	96.02%	0.00%	3.98%	49.69%	
Drain	caso10	7:09:51:57.0664	4	0	72.21%	0.00%	27.79%	14.96%	
Drain	caso12	2:51:02.6667	2	0	73.68%	0.00%	26.32%	73.67%	
Drain	caso13	5:02:18:38.8968	2	0	34.96%	0.00%	65.04%	3.07%	
Drain	caso14	2:53:17.6667	8	0	51.94%	0.00%	48.06%	51.93%	
Drain	caso15	15:14:44:28.2832	6	0	19.35%	0.00%	80.65%	4.27%	
Drain	caso16	25:10:20:58.2832	4	0	14.21%	0.00%	85.79%	2.05%	
Drain	caso17	18:02:32:41.4162	2	0	37.34%	0.00%	62.66%	8.26%	
Drain	caso18	6:05:22:00.6327	4	0	34.35%	0.00%	65.65%	10.58%	
Drain	caso19	11:11:52:19.0386	3	0	23.30%	0.00%	76.70%	5.33%	
Drain	caso2	2:23:26:51.1372	4	0	32.02%	0.00%	67.98%	4.72%	
Drain	caso20	6:05:02:14.9440	6	0	30.44%	0.00%	69.56%	9.36%	
Drain	caso21	14:01:48:20.7832	4	0	47.92%	0.00%	52.08%	10.27%	
Drain	caso22	17:16:00:17.6531	5	0	20.03%	0.00%	79.97%	4.85%	
Drain	caso5	12:09:32:01.6078	4	0	20.71%	0.00%	79.29%	4.62%	
Drain	caso6	9:13:10:16.7080	2	0	18.86%	0.00%	81.14%	3.74%	
Drain	caso7	4:06:08:52.7524	2	0	5.67%	0.00%	94.33%	4.05%	
Drain	caso8	3:06:19:32.6667	4	0	66.66%	0.00%	33.34%	13.76%	
Drain	caso9	13:21:47:59.9411	4	0	60.46%	0.00%	39.54%	12.83%	

Ilustración 37. Resultados de simulación optimizada de 19 casos de estudio.

La simulación optimizada arrojó un tiempo de terminación de 28 días, 07 horas, 30 minutos y 30 segundos, mejorando el tiempo de 39 días, 3 horas, 37 minutos y 30 segundos de la simulación basal sin lotes, como se puede observar en la *ilustración 37*, donde la secuencia óptima es la que se muestra a continuación.

Best solutions of the sequence problems



.Models.Frame.ENTRADAS

52, 6, 59, 41, 3, 47, 71, 29, 30, 51, 17, 61, 13, 7, 12, 39, 28, 9, 21, 58, 66, 62, 10, 27, 49, 65, 70, 38, 26, 23, 4, 60, 55, 54, 42, 68, 53, 36, 45, 43, 14, 64, 25, 35, 56, 1, 33, 18, 40, 31, 15, 69, 32, 37, 67, 72, 46, 22, 5, 16, 57, 11, 44, 24, 34, 8, 19, 63, 20, 2, 48, 50

Ilustración 38. Secuencia optima de los 19 casos de estudio loteados.

Evolution of the fitness values of the generations

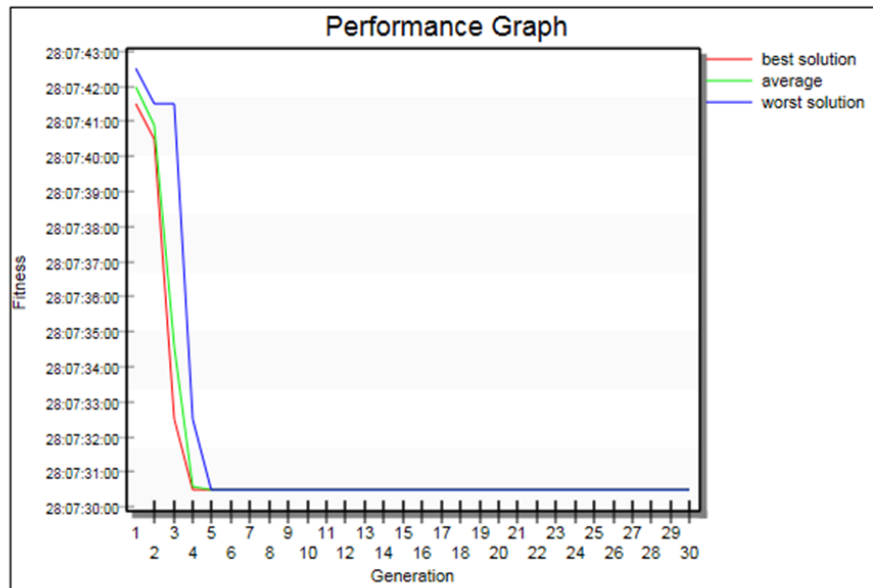


Ilustración 39. Gráfica de convergencia de los 19 casos de estudio.

En la gráfica de convergencia, se puede observar la mejor (---), promedio (---) y peor (---) solución y como converge en la generación 2 aproximadamente.

4.2.2.2. Simulación optimizada con 26 casos de estudio.

Se realizó la simulación optimizada con lotes dependiendo el número de piezas por caso con un NTGen de 50 generaciones y un NTI de 20.

Simulation time: 41:08:30:30.0000

Cumulated Statistics of the Parts which the Drain Deleted

Object	Name	Mean Life Time	Throughput	TPH	Production	Transport	Storage	Value added	Portion
Drain	caso1	10:53:28.2861	2	0	86.46%	0.00%	13.54%	13.77%	
Drain	caso10	8:04:27:05.2832	4	0	57.52%	0.00%	42.48%	13.55%	
Drain	caso12	12:03:02.6667	2	0	19.50%	0.00%	80.50%	17.43%	
Drain	caso13	12:10:56:27.4218	2	0	3.77%	0.00%	96.23%	1.25%	
Drain	caso14	11:01:36.4167	8	0	13.60%	0.00%	86.40%	13.60%	
Drain	caso15	28:21:21:45.6332	6	0	14.68%	0.00%	85.32%	2.31%	
Drain	caso16	33:09:51:37.5457	4	0	9.69%	0.00%	90.31%	1.56%	
Drain	caso17	17:10:51:09.7496	2	0	42.16%	0.00%	57.84%	9.86%	
Drain	caso18	9:00:10:43.0708	4	0	39.98%	0.00%	60.02%	7.31%	
Drain	caso19	5:20:21:19.0386	3	0	24.48%	0.00%	75.52%	3.90%	
Drain	caso2	1:23:40:15.5000	4	0	23.22%	0.00%	76.78%	7.08%	
Drain	caso20	6:00:38:34.7552	6	0	44.16%	0.00%	55.84%	9.64%	
Drain	caso21	23:23:16:30.7124	4	0	26.05%	0.00%	73.95%	6.03%	
Drain	caso22	22:12:33:06.2265	5	0	22.11%	0.00%	77.89%	3.80%	
Drain	caso23	12:02:11:27.8330	3	0	16.40%	0.00%	83.60%	4.71%	
Drain	caso24	9:28:08.6667	5	0	2.20%	0.00%	97.80%	2.20%	
Drain	caso25	4:23:49:28.0708	4	0	1.34%	0.00%	98.66%	1.24%	
Drain	caso26	8:10:21:34.0000	2	0	33.33%	0.00%	66.67%	5.09%	
Drain	caso27	4:12:45:47.6667	2	0	24.06%	0.00%	75.94%	8.86%	
Drain	caso28	10:12:32.6667	5	0	1.88%	0.00%	98.12%	1.88%	
Drain	caso29	18:06:24:09.9749	4	0	42.90%	0.00%	57.10%	9.81%	
Drain	caso5	13:17:18:50.6787	4	0	17.70%	0.00%	82.30%	2.63%	
Drain	caso6	10:13:56:59.3748	2	0	42.12%	0.00%	57.88%	7.18%	
Drain	caso7	13:09:39:52.7524	2	0	19.66%	0.00%	80.34%	3.90%	
Drain	caso8	2:19:51:04.9263	4	0	30.64%	0.00%	69.36%	11.48%	
Drain	caso9	16:18:56:42.3454	4	0	44.81%	0.00%	55.19%	11.18%	

Ilustración 40. Resultados de simulación optimizada de los 26 casos de estudio.

La simulación optimizada con 26 casos y con sus respectivos lotes, logra encontrar un mejor tiempo de 41 días, 08 horas, 30 minutos, 30 segundos, comparando con el tiempo de la simulación basal que se puede observar en la *ilustración 31* con un tiempo de 56 días, 3 horas, 30 minutos y 30 segundos, donde el tiempo del caso 29, el cual tomaba mayor lapso en producirse, mejoro significativamente.

Best solutions of the sequence problems



.Models.Frame.ENTRADAS

38, 53, 11, 94, 5, 39, 16, 84, 22, 83, 23, 58, 66, 88, 78, 76, 7, 95, 30, 25, 20, 69, 55, 2, 24, 60, 77, 40, 12, 17, 14, 33, 82, 36, 87, 74, 13, 4, 57, 68, 48, 6, 21, 79, 28, 46, 59, 56, 96, 90, 29, 34, 72, 41, 32, 15, 45, 81, 50, 44, 86, 89, 80, 71, 92, 47, 75, 61, 54, 49, 19, 8, 65, 26, 91, 3, 73, 67, 18, 37, 43, 1, 93, 85, 70, 51, 42, 52, 62, 9, 63, 10, 64, 27, 35, 97, 31

Ilustración 41. Secuencia optimizada de los 26 casos loteados.

Evolution of the fitness values of the generations

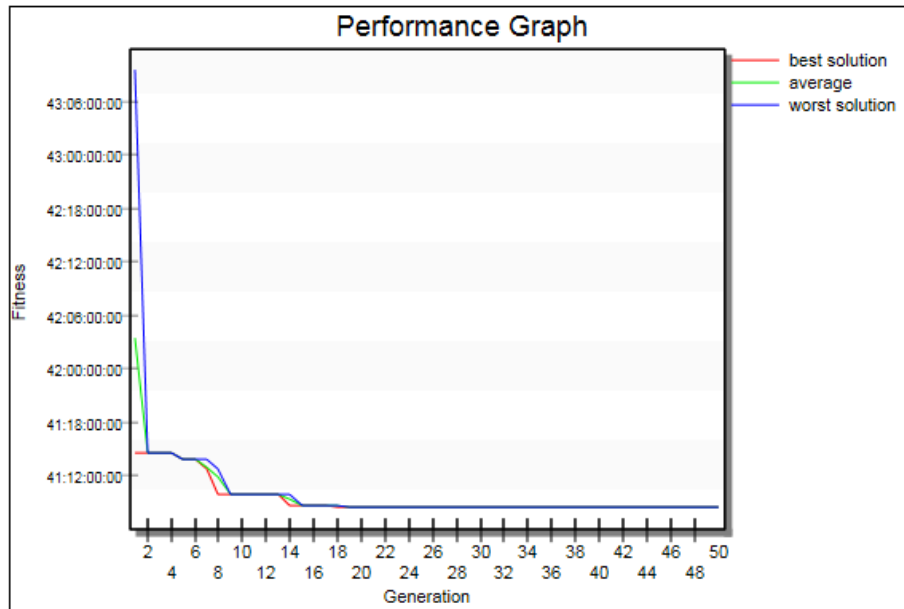


Ilustración 42. Gráfica de convergencia de los 26 casos de estudio.

En la gráfica de convergencia, se puede observar la mejor (---), promedio (---) y peor (---) solución y como converge en la generación 19 aproximadamente.

CAPITULO V

5.1 CONCLUSIONES

Para concluir esta tesis, se encontró por medio de la literatura consultada que la mayoría de autores optan por utilizar Job shop tradicional en los problemas de la planeación de la producción, por sus características en cuanto a que sus tareas no obligatoriamente pasan por todos sus procesos en el mismo orden y puede trabajar con algunas máquinas flexibles, en cambio las otras técnicas de Job shop no se adecuan al problema por ciertas restricciones en sus características, como por ejemplo el Flow shop, en donde las operaciones pasan por todos los procesos o maquinaria para llevarse a cabo en un mismo orden, el Open shop, por otra parte, clasifica por sistemas con 2 tipos de máquinas, no trabaja en serie y no hay restricciones de categorización respecto a las operaciones y por último, en el Job shop flexible existe flexibilidad dentro del proceso donde no es relevante el número de tarea u operación a ser procesado, por lo que el Job shop tradicional es el que se ajusta al problema de este trabajo.

Ya que el Job shop tradicional fue la metodología que más se adaptó al problema por su flexibilidad en los procesos, después se realizó la representación del problema por medio de un modelo matemático, incluyendo todas y cada una de las características del mismo, sus restricciones y función objetivo que incluyen todos los tiempos de procesamiento para obtener el tiempo de terminación de todos los pedidos o casos.

Dicha metodología de Job shop tradicional se utilizó para modelar tanto en la simulación como en el modelo matemático.

Como primer paso en la experimentación, se decidió manejar un diagrama de Gantt para reducir tiempos muertos que se traducen a menor efectividad, éste ayudó a reducir los tiempos de terminación de algunos pedidos, acomodando los mismos en las diferentes máquinas, pero el tiempo de terminación utilizando dicho diagrama no es exacto, ya que se trabaja con divisiones de cada media hora, dando un tiempo aproximado en redondeo, por lo que se descartó el uso de tal herramienta, posteriormente se utiliza un modelo matemático, con el cual se tiene complejidad al resolver, por la forma de representar la realidad y entre más variables se agreguen a dicho modelo más complejo se vuelve; Después se hace uso a la herramienta Solver de Excel para resolver los 13 casos con 1220 variables, la cual se trata de solucionar en 3 partes: primero con programación lineal, la cual arroja un resultado que son

demasiadas las variables a resolver, se opta por resolver solo 3 de los 13 casos, pero continua arrojando los mismos resultados, procediendo a utilizar solo 2 casos y esto arroja resultado de que no se cumple con la linealidad requerida para resolver con programación lineal; Después se soluciona por Gradiente y este arroja un respuesta de 35100 seg., por último se prueba con Algoritmos genéticos para ver si la respuesta mejoraba, y efectivamente, con 29700 seg y arrojando la respuesta más rápido que con Gradiente; Esta experimentación sirve para validar el uso de los Algoritmos genéticos.

En este trabajo se descartó el uso de distintas técnicas y herramientas para resolver problemas de planeación de la producción, ya que trabajan con inventarios y en esta investigación no se trabaja con relación a estos, y con ayuda de la literatura, se aseguró el uso de Algoritmos genéticos para optimizar el tiempo de terminación.

Para la solución del problema, primero fue necesario hacer la representación, y para esto se utilizó un modelo matemático que requirió algunas variables como: las operaciones de los casos de experimentación con sus respectivos tiempos en minutos, las restricciones de precedencia, asignación de operaciones y no negatividad y por último la función objetivo, para resolver dicho modelo se intentó realizar un código de programación en el software Matlab utilizando el pseudocódigo del algoritmo genético simple, pero al momento de hacer la representación real del problema, adaptando todas sus restricciones y características específicas de la flexibilidad en la secuencia de las operaciones propias del Job shop tradicional, el código se volvía más complejo por su extenso número de variables y las restricciones, por lo que se decidió buscar otras alternativas de solución. Dicho código se puede observar en la última parte de anexos.

Se utiliza un modelo de simulación para hacer la representación lo más real posible, por lo que se inició con una simulación basal de dicho proceso, con el fin de ver las deficiencias en el mismo, en esta parte fue posible observar tiempos muertos, tiempos de espera, espera por material, cuellos de botella, etc., posteriormente en la misma simulación con ayuda de un módulo de Algoritmos genéticos (integrado dentro del software de Tecnomatix), se logra optimizar la secuencia de entradas, para después simular ya con la respuesta mejorada y así obtener la simulación optimizada. La simulación toma en cuenta todas las variables y la realidad se representa sencillamente.

Mediante la integración de un algoritmo genético basado en simulación se logró reducir el tiempo de terminación, además de obtener una mejor representación de la realidad.

Tabla 3. Comparación de tiempos de simulación y porcentaje reducido.

Experimento	Tiempo basal	Tiempo optimizado	% Reducido
13 casos	29:07:41:56	14:06:34:30	52%
19 casos	39:03:37:30	28:07:30:30	72%
26 casos	56:03:30:30	41:08:30:30	74%

Fue comprobada la hipótesis con los 13 casos de experimentación, reduciendo más del 50% (15 días de diferencia que es un resultado significativo), donde se aumentó el número de casos de experimentación para mejorar la comprobación de hipótesis.

Como se describe en el estado del arte, los algoritmos genéticos son comúnmente requeridos para la resolución de problemas tipo taller ya que al buscar las mejores soluciones, no se estancan en óptimos locales, sino que, buscan óptimos globales y convergen rápidamente, por lo que el uso del módulo de algoritmo genético en el simulador fue más sencillo de utilizar para resolver el problema de secuenciamiento, con esto el algoritmo encontró la mejor solución de secuencia de operaciones que logro minimizar el tiempo de procesamiento de 29 días, 7 horas y 42 segundos que tomaba al utilizar PEPS a 14 días, 06 horas, 28 minutos y 26 segundos dividiendo en lotes y optimizando el secuenciamiento de operaciones, así los casos no tienen que esperar a ser procesados, porque al optimizar la secuencia de entradas se reduce el tiempo de terminación, acomodando los casos de tal forma que se reduzcan cuellos de botella, esperas y tiempos muertos. Plant simulation es el único software comercial que contiene un módulo de optimización con Algoritmos genéticos, el cual ayudó a mejorar el tiempo de terminación de los casos de experimentación.

5.2 TRABAJO FUTURO

En este trabajo no se tomaron en cuenta algunas variables, como también algunas restricciones, por lo que como trabajo futuro se piensa integrar estas variables para tener una representación más real del problema que se tiene en ■■■, tomando en cuenta la demanda, recursos disponibles y todos los pedidos que se puedan llevar a cabo en dicha empresa.

5.3 RESULTADOS Y ENTREGABLES

Se obtuvo la planeación de la secuencia a seguir de los pedidos de la experimentación, para lograr el menor tiempo de terminación.

Entregables:

- Software (interfaz gráfica) en Plant simulation.
- 2 Artículos de Congreso Academia Journals.

5.4 APORTACIÓN CIENTÍFICA

Desarrollar un mecanismo de simulación de una empresa tipo taller tradicional con infusión de un tipo taller tipo flujo con la incorporación de un Algoritmo genético.

5.5 APORTACIÓN TECNOLÓGICA

Simulador de planeación táctica de producción con interfaz gráfica para optimización de pedidos sobre tiempos de producción.

BIBLIOGRAFÍA

- Adams J, B. E. (1988). *The shifting bottleneck procedure for job shop scheduling*. . ManagSci., Vol (34), 3-331
- Aiyoshi, E., & Shimizu, K. (1981). *Hierarchical decentralized systems and its new solution by a barrier method*. . IEEE Trans Syst Man Cybern., Vol SMC (11), 444-449
- Alfalla L, R. a. (2008). *Introducción a la Dirección de Operaciones Táctico-Operativa: Un enfoque Práctico*. Madrid, España: Editorial: Del888a Publicaciones Universitarias. ISBN: 849647769X. Recuperado de <http://hdl.handle.net/123456789/23044>
- Anderson EJ, G. C. (2003). *Local search in combinatorial optimization*. John Wiley & Sons Ltd. Ediciones Priceton University Press
- Baker, K. (1974). *Introduction to sequencing and scheduling*. New york: John Wiley & Sons., Ediciones Wiley, ISBN 10 0471045551

- Bard, J., & Feo, T. (1989). *Operations sequencing in discrete parts manufacturing* *Journal of Management Scienc.* Vol(35), 249-255.
- Baumann, M. D. (2011). *Simulation zur Auswahl und Parametrisierung einer Produktionssteuerung.* ScienceDirect Volumen 7, 145-150,doi: 10.1016/j.procir.2013.05.025
- Ben-Ayed O, B. C. (1990). *Computational difficulties of bilevel linear programming.* Oper Res. 38(3):556-560 ·DOI: 10.1287/opre.38.3.556.
- Binato S., H. W. (2002). *A GRASP for Job Scheduling.* Brasil. Kluwer Academic Publishers
- Bischoff, D. M. (1997). *Análisis y simulación de procesos.* Ediciones Reverte.
- Boromi, E. a. (1984). *The N-city traveling salesman problem: statistical mechanics and the metropolis algorithm.* SIAM Review. 26(4), 551–568
<https://doi.org/10.1137/1026105>
- Bowman, E. (1959). *The schedule-sequencing problem.* Vol. 7, No. 5, pp. 621-624. Ediciones Informis.
- Brucker, P. (1988). *Scheduling algorithms.* Berlin: 2nd ed. Berlin: Springer-Verlag. *Journal of Scheduling* 4(1),65-66, ISBN 3-540-64105-X, DOI: 10.1002/1099-1425(200101/02)4:13.0.CO;2-C
- Brucker, P. K. (1998). *Parallel machine deadline batch scheduling.* *Annals of Operations Research.* 83(1998)23–40
- Cybenko, G. (1989). *Approximations by superpositions of a sigmoidal function.* *Math Control Signal Syst.* Springer-Verlag New York Inc. (1989) 2:303-314
- Campello R., M. N. (1992). *Algoritmos e Heurísticas Desenvolvimento e avaliação de performance.* . Brasil: Apolo Nacional Editores. ISBN: 85-228-0134-7
- Cao Abad, R. (2002). *Introducción a la simulación y teoría de colas.* Coruña, España: Netbiblo, S.L. ISBN: 84-9745-017-5
- Carrum Siller, E., Torres Treviño, L., & Pérez Villanueva, P. (2014). *A Hybrid Simulation Based on Multi-Objective Algorithm for Manufacturing Cells Optimization.* ISBN: 97819384960-2-8

- Castrillón, O., et al. (2016). *Scheduling of a Job Shop-Open Shop System through a Neuronal Network*. Scielo, Universidad Nacional de Colombia, Inf. tecnol. vol.27 no.5 La Serena, ISSN 0718-0764, DOI: <http://dx.doi.org/10.4067/S0718-07642016000500018>
- Chandru, V. L.-Y. (1993). *Minimizing total completion on a batch processing machine with job familie*. Elsevier. Volume 13, Issue 2, 61-65, [https://doi.org/10.1016/0167-6377\(93\)90030-K](https://doi.org/10.1016/0167-6377(93)90030-K)
- Chang, P., Damodaran, P., & Melouk, S. (2004). *Minimizing makespan on parell batch processing machines*. Springer, London. Print ISBN 978-1-4471-5348-1, DOI: https://doi.org/10.1007/978-1-4471-5349-8_25
- Chase, R., Aquilano, N., & Jacobs, F. (2006). *Operations Management for Competitive Advantage*. New York: (11th ed.). Ediciones Mc Graw-Hill/Irwin.
- Davis, L. (1985). Job shop scheduling with genetic algorithms. *First International Conference on Genetic Algorithms and their Application*. USA: Morgan – Kaufmann. J. J. Grefenstette (Ed.) San Mateo, 136–140.
- Dirk, C. (1996). *Evolutionary search and the job shop: investigations on genetic algorithms for production*. Bremen, eBook ISBN 978-3-662-11712-5, IX, 153, DOI: [10.1007/978-3-662-11712-5](https://doi.org/10.1007/978-3-662-11712-5). Ediciones Physica-Verlag Heidelberg
- Deb K., A. S. (2000). *A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II*. *Proceedings of the Parallel Problem Solving from Nature*. Springer, 849-858.
- Deb, K. P. (2002). *A fast and elitist multiobjective genetic algorithm: NSGA-II*. IEEE, 182 - 197, DOI: [10.1109/4235.996017](https://doi.org/10.1109/4235.996017)
- Domínguez M, J. A. (1995). *Dirección de Operaciones. Aspectos tácticos y operativos en la producción y los servicios*. Madrid: Editorial Mc-Graw Hill.
- Elsayed y Boucher, T. (1985). *Analysis and Control of Production Systems*. New Jersey: Ediciones Practice-Hall.

- Escamilla Fuster, J. (2016). *Eficiencia Energética y Robustez en Problemas de Scheduling*. (Tesis depositada en cumplimiento de los requerimientos para obtener el título de Doctor por la Universidad Politécnica de Valencia) Valencia.
- Fernández, E. (1993). *Dirección de la Producción*. Madrid: Editorial Civitas.
- Frutos Alazard, M., & Tohmé Hauptmann, R. (2012). *Técnicas evolutivas en problemas multi-objetivos en el proceso de planificación de la producción*. Ingeniería Industrial. ISSN 1815-5936/Vol. XXXIII/No. 1, 50-59.
- García, R. (1999). *Estudio del trabajo*. México, Ediciones Mc Graw - Hill.
- Garey, E. L., & Johnson, D. S. (1976). *The complexity of flow shop and job shop scheduling*. Mathematics of Operations Research, vol. 1, No. 2 .
- Goldratt, E. M. (1990). *What is this thing called Theory of Constraints and how should it be implemented?* Massachusetts, Ediciones North River Press: Great Barrington Massachusetts.
- Gonçalves, J., et al (2005). *A Hybrid Genetic algorithm for the Job Shop Scheduling*. Volume 167, Issue 1, 77-95, Doi: <https://doi.org/10.1016/j.ejor.2004.03.012>
- González, M. (2005). *Gestión de la producción como planificar y controlar la producción Industrial*. España, Ideaspropias Editorial S.L., ISBN 8498390141
- Guamanquispe, M. (2011). *Gestion de procesos para incrementar la capacidad de produccion en el area de tinturado y acabados en la empresa textil Andelas cía. Ltda.* (Tesis, repositorio) <http://repositorio.uta.edu.ec/jspui/handle/123456789/799>
- Gust, P., Müller, U., Feller, N., & Schiffmann, M. (2017). *Field study on the application of a simulation-based software tool for the strain-based staffing in industrial manufacturing*. AISC, volume 481, Springer, Cham, ISBN 978-3-319-41626-7, DOI: https://doi.org/10.1007/978-3-319-41627-4_1
- Ham, W. K., Ko, M., & Park, S. C. (2017). *A framework for simulation-based engine-control unit inspection in manufacturing phase*. Volume 59, 137-148, DOI: <https://doi.org/10.1016/j.conengprac.2016.12.001>
- Hérika Martínez Prado. (19 de Diciembre de 2014). *Maquilas, al tope en producción; aumentan horas de trabajo*. Ciudad Juárez, México, Reporte por NorteDigital.

- Hernandez Aguirre, A., & Buckles P. Bill, C. C. (2017). *Estrategias Evolutivas : La Versión Alemana del Algoritmo Genético (Parte I)*. Recuperado de: <https://www.researchgate.net/publication/267725329>
- Hopp, W. J. (2008). *Factory physics*. New York: 3rd Edition, New York, Ediciones McGraw-Hill.
- Hua, K. S. (2014). *Uncertain random multilevel programming with application to production control problem*. Volume 19, Issue 6, 1739–1746, Springer.
- Jain, A. M. (1999). *Deterministic job-shop scheduling: past, present and future*. European Journal of Operational Research 113(2):390-434, DOI: 10.1016/S03772217(98)00113-1
- JBE, E. (2010). *Solving convex quadratic bilevel programming problems using an enumeration sequential quadratic programming algorithm*. . Volume 47, Issue 4, 615–637, Springer, DOI: 10.1007/s10898-009-9482-3
- Jiang Y, L. X. (2013). *Application of particle swarm optimization based on CHKS smoothing function for solving nonlinear bilevel programming problem*. Volume 219, Issue 9, 4332-4339, Elsevier, Science direct, DOI: <https://doi.org/10.1016/j.amc.2012.10.010>
- Jiménez, F., & Espinoza, C. (2007). *Costos industriales*. Costa Rica: Tecnológica de Costa Rica. Primera Edición, ISBN: 9977-66-183-9
- Kuhpfahl, J., & Bierwirth, C. (2016). *A study on local search neighborhoods for the job shop scheduling*. Volume 66, 44-57, Elsevier Sciencedirect, DOI: <https://doi.org/10.1016/j.cor.2015.07.011>
- Laarhoven, V. (1988). *Theoretical and computational aspects of simulated annealing*. Volume 32, Issue 3, 32(3), 504–506. DOI: <https://doi.org/10.1137/1032103>
- Liao C, Y. (1992). *An improved formulation for the job shop scheduling problem*. Oper Res, Volume 43, Issue 11, 1047–1054 , Springer, National Taiwan Institute of Technology, Taiwan
- Lödding, H. (2013). *Handbook of Manufacturing Control*. Berlin. Springer. Springer Heidelberg New York Dordrecht London, DOI: 10.1007/978-3-642-24458-2

- Mavileón, M. (2008). *Gestión de Stock*. Ediciones Díaz de Santos, ISBN: 978-84-7978-872-8
- Meisel, J., & Prado, L. (2010). *Un algoritmo genético híbrido y un enfriamiento simulado para solucionar el problema de programación de pedidos Job shop*. Escuela de Ingeniería de Antioquia, Medellín (Colombia), Número 13, *EIA*, 39-51, ISSN 1794-1237
- Molina, C. (2014). *Plan Agregado de Producción Mediante el Uso de un Algoritmo de Programación Lineal: Un caso de Estudio para la Pequeña Industria*. Revista EPN vol. 34, no. 1.
- Münzberg, B. N. (2009). *Individual Configuration of Production Control to Suit Requirements*. International conference on Manufacturing and Industrial Engineering (pág. 47). Proceeding of World Academy of science: Engineering and Technology. Leibniz University of Hanover, Garbsen, 30823 German
- Ni, H. K. (2014). *Uncertain random multilevel programming with application to production control problem*. Volume 19, Issue 6, 1739–1746, Springer
- NowickiE, S. (1996). *A fast taboo search algorithm for the job shop problem*. Manag Sci. Volume 42, Issue 6, 797 - 813, DOI: <https://doi.org/10.1287/mnsc.42.6.797>
- Osorio Gómez, J. C. (2008). *Hierarchical production planning model in flexible job shop including a preemption and sequence-dependent setup times*. Scielo, Ing. Investig. vol.28 no.2 Bogotá, ISSN 0120-5609
- Osorio, J., Castrillón, O., Toro, J., & Orejuela, J. (2008). *Modelo de programación jerárquica de la producción en un Job shop flexible con interrupciones y tiempos de alistamiento dependientes de la secuencia*. Escuela de Ingeniería Industrial y Estadística, Universidad del Valle, Colombia, Revista Ingeniería e Investigación Vol. 28 No. 2, 72-79
- Parra, J., Rodríguez, A., & Beltrán, D. (2005). *Modelo de simulación, sistemas de producción : Bovino Doble Propósito Piedemonte LLanero*. Colombia, 66, ISBN: 958-97803-0-X.

- Pask, F., Lake, P., Yang, A., Tokos, H., & Sadhukhan, J. (2017). *Sustainability indicators for industrial ovens and assessment using Fuzzy set theory and Monte carl simulation*. Volume 140, Part 3, 1217-1225, Elsevier, DOI: <https://doi.org/10.1016/j.jclepro.2016.10.038>
- Patriksson, M. & Wynter, L. (1999). *Stochastic mathematical programs with equilibrium constraints*. Volume 25, Issue 4, 159-167, Elsevier, DOI: [https://doi.org/10.1016/S0167-6377\(99\)00052-8](https://doi.org/10.1016/S0167-6377(99)00052-8)
- Pozo, J. (2015). *Consideraciones sobre la determinación de la capacidad en la industria. Una extensión a los servicios*. Ediciones Félix Valera, Revista Caribeña de Ciencias Sociales, recuperado de: <http://caribeña.eumed.net/consideraciones-sobre-la-determinacion-de-la-capacidad-en-la-industria-una-extension-a-los-servicios/>
- Ruíz, S., Castrillón, O., & Sarache, W. (2015). *Metodología selectiva de dinámica poblacional para optimizar un ambiente multiobjetivo de producción job shop*. Revista de Matemática: Teoría y Aplicaciones. Vol. 22, Nº. 1, 113-134, ISSN 2215-3373
- Santana Robles, F., Granillo, R., & González, M. I. (2017). *Redes de Petri coloreadas jerárquicas para el modelado de procesos de negocio de una cadena de suministro textil*. Springer, Vol. 4, No. 7, ISSN 2007-784X
- Santos García, J. (2007). *Organización de la Producción II Planificación de procesos productivos*. San Sebastián (Gipuzkoa-España): Pablo Callejo Goena. ISBN 84-607-9050-9
- Schönemann, M. (2017). *Multiscale Simulation approach for battery production system*. Germany: Springer Nature. Springer International Publishing, ISBN 9783319493664
- Schrage, L. (1997). *Optimization modeling with LINDO*. USA: Duxbury Press. Edición: 5ta. edición, ISBN-13: 978-0534348571
- Schroeder, R. (2011). *Administración de Operaciones. Casos y conceptos contemporáneos*. México: 2ª. Edición. Editorial. McGraw-Hill Companies. ISBN: 9786071506009
- Schuh, G., Franzkoch, B., Potente, T., & Fuchs, S. (2012). *Simulation based configuration of value stream oriented production control*. Conference Proceedings POM 21st Annual

- Conference of Simulation, *POM* (pág. 144). Vancouver, DOI: 10.1109/WSC.2012.6465080
- Schuh, G., Potente, T., & Thomas, C. (2013). *Design of Production Control's Behavior*. Elsevier Science direct, Volume 7, 145-150, DOI: <https://doi.org/10.1016/j.procir.2013.05.025>
- Schuh, G., Potente, T., Fuchs, S., & Thomas, C. (2011). *Effects of production structure and production control to logistic targets*. International Conference - Advances in Production Management Systems, University of Stavanger Stavanger. Norway: Proceedings of the APMS, volume 384, 190-199
- Shen, Z., Burnham, K. J., & Smalov, L. (2015). *Optimised Job-Shop Scheduling via Genetic Algorithm for a Manufacturing Production System*. Springer Verlag, Vol. 1089, 89-92, DOI: 10.1007/978-3-319-08422-0_13
- Skorin-Kapov, J. y. (1993). Planificación de una célula de fabricación de línea de flujo: una búsqueda tabú. *Revista Internacional de Investigación de Producción*, 1722. Volume (7), 1-75
- Supsomboon, S., & Vajasovimon, A. (2016). Simulation model for Job shop production process improvement in machine parts manufacturing. Volume (4), 611-622, ISSN 1726-4529, DOI: 10.2507/IJSIMM15(4)3.352
- Tahar, D. N. (2006). *A linear programming approach for identical parallel machine scheduling with job splitting and sequence-dependent setup times*. Elsevier Science direct, International journal of production economics, Volume 99, Issues 1–2, 63-73, DOI: <https://doi.org/10.1016/j.ijpe.2004.12.007>
- Taillard, E. (1994). *Parallel Taboo Search Technique for the Job shop Scheduling Problem*. *Journal on Computing Science*, Volumen(6), Issue 2, 108-117, DOI: <https://doi.org/10.1287/ijoc.6.2.108>
- Torres, L. (2006). *Algoritmos para aplicar la computación evolutiva*. General Escobedo, N.L.
- Tupia, M., & Mauricio, D. (2004). *Un algoritmo voraz para resolver el problema de programación de tareas dependientes en máquinas diferentes*. *Investigación de Sistemas e Informática*, Vol(1) núm 1,9-18, ISSN: 1815-0268

- Vázquez, J. (2005). *Filosofía 6-SIGMA una metodología para determinar la calidad de los productos y servicios en el sector productivo*. México, D.F.(Tesis) Instituto Politécnico Nacional
- Velázquez, G. (2008). *Administración de los sistemas de producción*. Limusa, sexta edición. ISBN: 9789681864910
- Wang, C. S. (2002). *A genetic algorithm to minimize maximum lateness on a batch processing machine*. Elsevier Science direct, Volume 29, Issue 12, 1621-1640, DOI: [https://doi.org/10.1016/S0305-0548\(01\)00031-4](https://doi.org/10.1016/S0305-0548(01)00031-4)
- Wen Yang, K., & Beck, C. (2016). *Modelo de programación entera mixta para la programación Job shop: Análisis computacional*. Elsevier. Vol (73), 165-173, DOI: <http://dx.doi.org/10.1016/j.cor.2016.04.006>
- Wiendahl H.-H., Behringer S (2006). *Stolpersteine der Liefertermin- ermittlung und -erfüllung, in: Liefertreue im Maschinen- und Anlagenbau: Stand - Potenziale - Trends*, Schuh G., Westkämper E., study results of FIR, IPA and WZL, Aachen and Stuttgart,.19-32.
- Wiendahl, H. (2008). *Ein sozio-technischer Ansatz für das industrielle Auftragsmanagement*. Institut für Industrielle Fertigung und Fabrikbetrieb (IFF) Universität Stuttgart , Beiträge zu einer Theorie der Logistik, Berlin, 275-304.
- Wiendahl, H. (2009). *Fundamentals of Production Logistics*. Springer berlin, Germany, ISBN 978-3-540-34210-6, DOI: 10.1007/978-3-540-34211-3
- Wolsey, L. (1998). *Integer Programming*. Wiley, New York, ISBN: 978-0-471-28366-9
- Xu P, W. L. (2014). *An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions*. Volume 41, 309-318, Elsevier science direct, DOI: <https://doi.org/10.1016/j.cor.2013.07.016>
- Zhang R, W. (2011). *A simulated annealing algorithm based on block properties for the job shop scheduling problem with total weighted tardiness objective*. Volume 38, Issue 5, 854-867, Elsevier science direct, DOI: <https://doi.org/10.1016/j.cor.2010.09.014>
- Zhang, R., & Raymond, C. (2016). *Solving the energy-efficient job shop scheduling problem: a multiobjective*. Elsevier Journal of Cleaner Production, Volume 112, Part 4, 3361-3375, DOI: <https://doi.org/10.1016/j.jclepro.2015.09.097>

Zhang, W., Wen, J., & Zhu, J. &. (2017). *Multi-objective scheduling simulation of flexible job-shop based on multi-population genetic algorithm*. Vol(2), 313-321, DOI: [https://doi.org/10.2507/IJSIMM16\(2\)CO6](https://doi.org/10.2507/IJSIMM16(2)CO6)

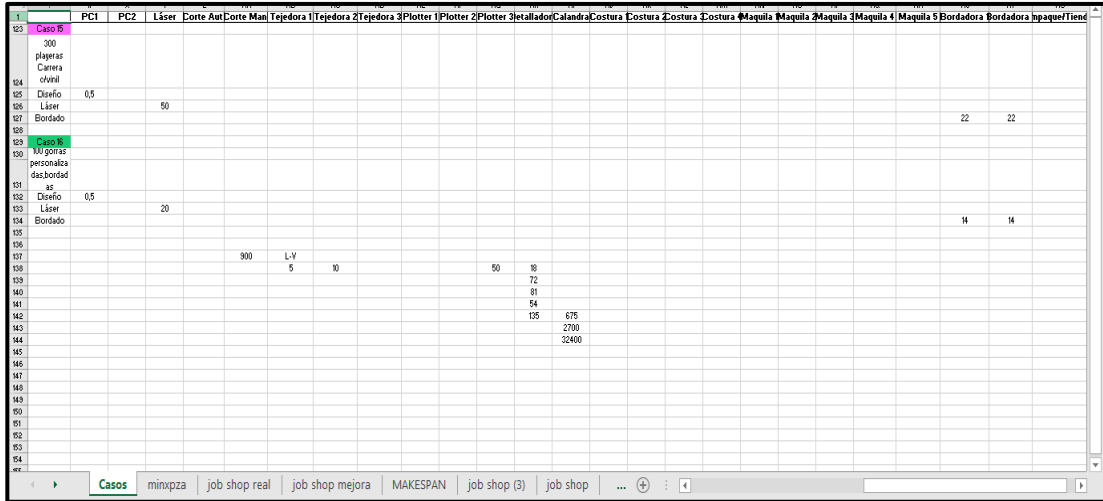


Ilustración 47. Matriz de tiempos por operación. Parte 5.

Job Shop

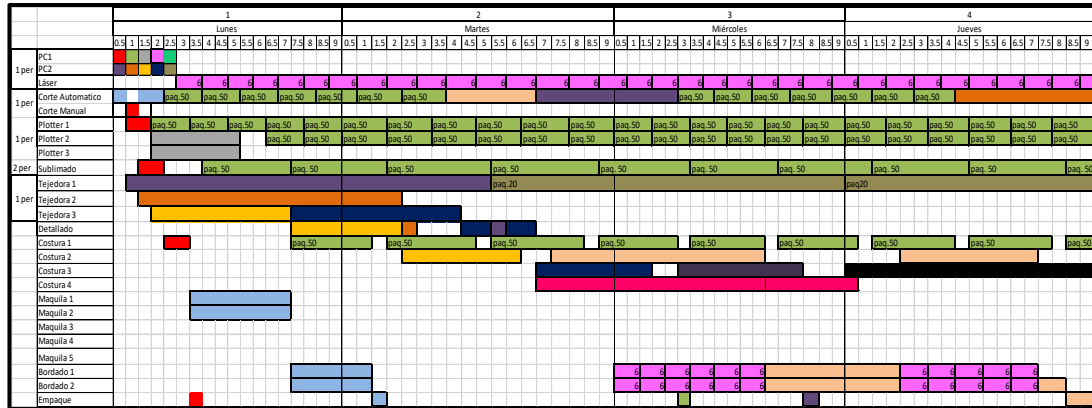


Ilustración 48. Job shop real. Parte 1.

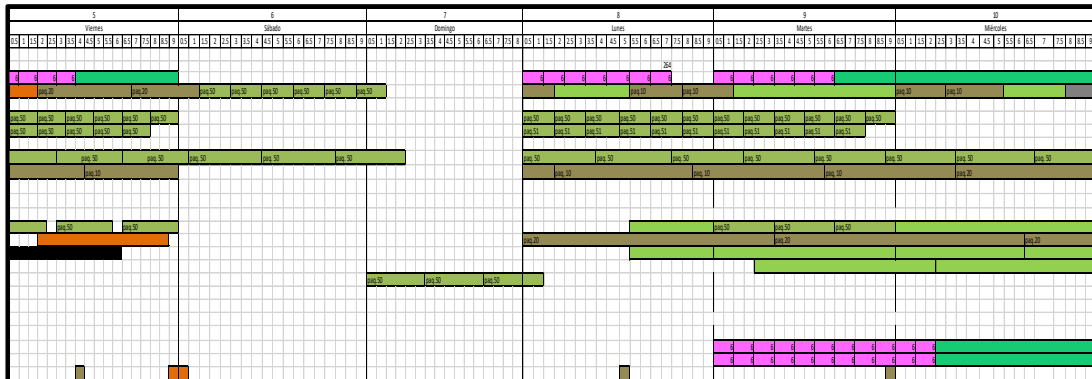


Ilustración 49. Job shop real. Parte 2.

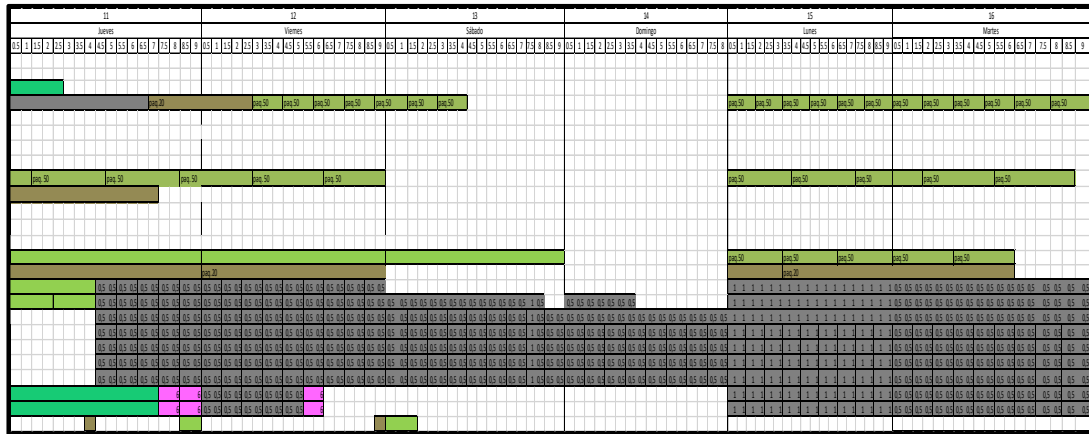


Ilustración 50. Job shop real. Parte 3.

Job Shop mejora

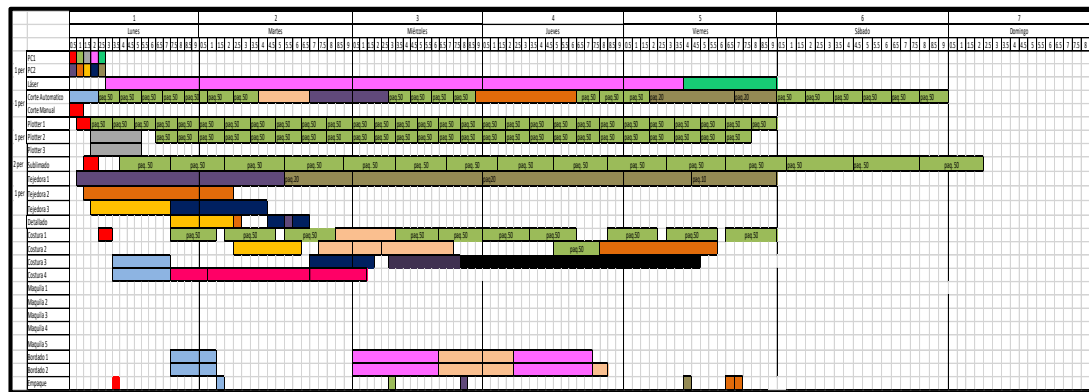


Ilustración 51. Job shop mejora. Parte 1.

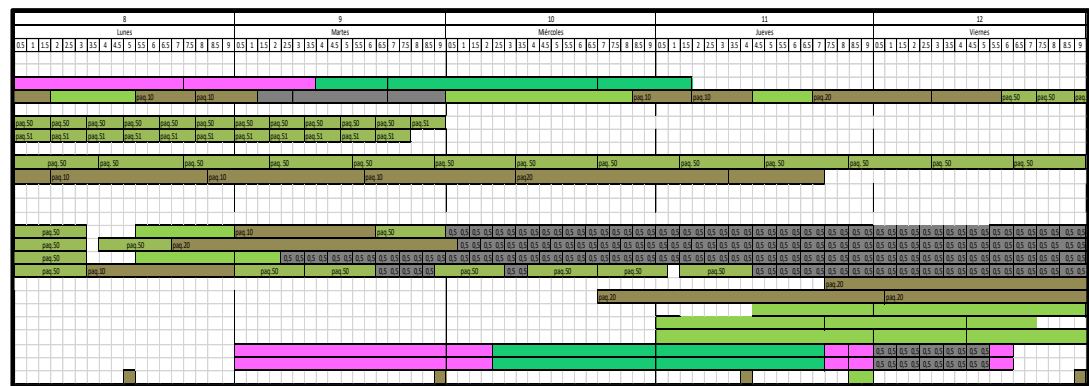


Ilustración 52. Job shop mejora. Parte 2.

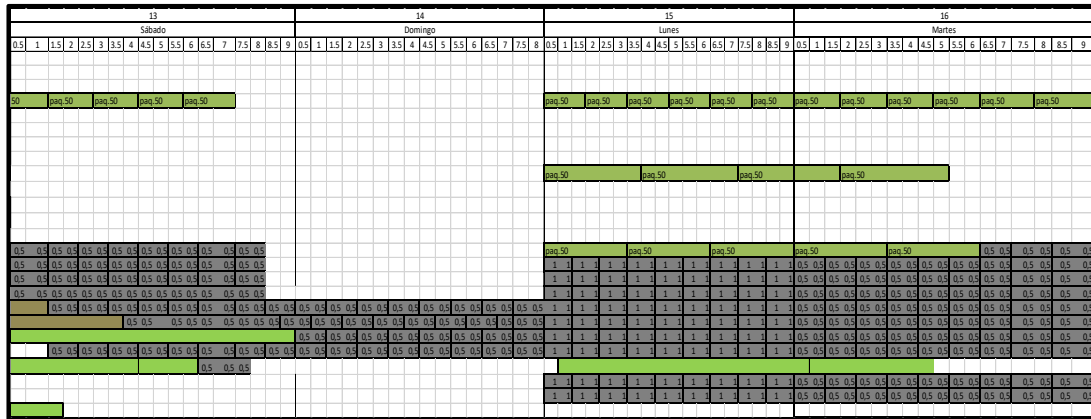


Ilustración 53. Job shop mejora. Parte 3.

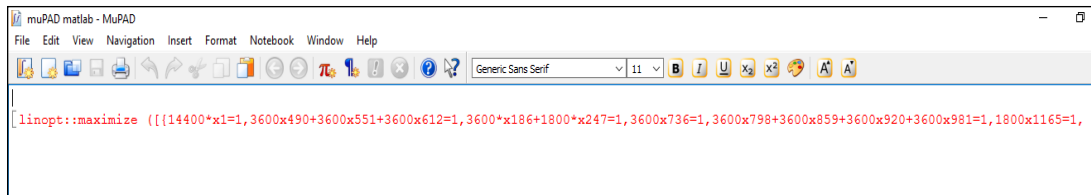


Ilustración 54. Experimentación de herramienta MuPad de Matlab.

Pseudocódigo de Algoritmo genético, programado en Matlab:

```
P=[1 1 0 0 0 0 1 1 1 1 0 1 1 0 0 1 1 0 1 0 1 0 0 0 1 1 0 1 0 1 0 1
    0 1 0 1 0 0 1 1 1 1 0 1 1 1 0 0 1 0 0 0 0 1 1 1 0 0 0 0 1 0 1 0
    1 1 0 1 0 0 0 0 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 0 0 1 1 0 0 1];
[NTI NTPR]=size(P);

Tiempo=[1 3 3 4 2 9 8 4 3 8 5 1 3 1 6 5 4 2 4 1 9 2 3 8 1 1 7 4 3 2 5 1];
nop=8; %Numero de operaciones
nmaquinas=4; % Numero de maquinas
operaciones=[3 3 2];

for i=1:NTI
    for j=1:NTPR
        fet(i,j)=P(i,j)*Tiempo(1,j);
    end
    Fe(i,:)=sum(fet(i,:))
End

Asignacion

for k=1:NTI
    temp=1;
    for w=1:nmaquinas
        feopl(k,w)=P(k,temp);
        temp=temp+nop; %ciclo para aumentar y checar en la posicion
```

```

end
if sum(feop1(k,:))>1
    Fe(k,:)=1000000;
else
end
end

Precedencia restriccion 1

r=8
for k=1:NTI
    temp=1
    temp2=2
    for w=1:2:r
        while temp<NTPR
            R2(k,w)=P(k,temp)
            R2(k,temp2)=P(k,temp+1)
            temp=temp+nop
            temp2=temp2+2
        end
    end
end
end
%
%precedencia tiempo restriccion 2

r=8
temp=1
temp2=2
c=2
for p=1:2:r
    R22(:,p)=Tiempo(:,temp)
    R22(:,c)=Tiempo(:,temp+1)
    temp=temp+nop
    temp2=temp+1
    c=c+2
end
end

```