ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

INSTITUTO DE CIENCIAS MATEMÁTICAS
ESCUELA DE GRADUADOS

PROYECTO DE GRADUACIÓN

PREVIO A LA OBTENCIÓN DEL TITULO DE:
MAGÍSTER EN CONTROL DE OPERACIONES Y GESTIÓN LOGÍSTICA

TEMA

USO DE METAHEURISTICAS PARA LA OPTIMIZACION DE LA SECUENCIA DE PRODUCCION Y LA ASIGNACION DE MANO DE OBRA EN UNA EMPRESA MANUFACTURERA

AUTORES:

Héctor Vinicio Sevilla Quevedo
Carlos Francisco Zurita Noriega

Guayaquil - ECUADOR

AÑO 2010

DEDICATORIA

A mi hijo Mateo y a mi hija Romina, que en su tierna edad también contribuyeron al éxito de la culminación de este trabajo. Su inagotable amor me infundieron de aliento, paciencia y sobretodo de energía para llegar a la meta.

A mis padres por haberme dado la vida e inculcado el respeto y la responsabilidad, mi triunfo es su triunfo.

Finalmente quiero agradecer a mi esposa María Elena, por su inagotable paciencia durante el tiempo que duro el programa de Maestría y gracias a su permanente apoyo hoy puedo con orgullo finalizar esta etapa tan exitante de mi vida académica y profesional. Solo puedo decirte que te amo.

Vinicio Sevilla

DEDICATORIA

A Myriam, la mejor mamá del mundo, cuyo ejemplo, preocupación, esfuerzo y apoyo más allá de lo imaginable hicieron posible que yo pueda obtener este logro. Todo lo que obtengo es muy poco por mí y mucho debido a ti. No hay palabras mi viejita para agradecerte todo lo que has hecho y sigues haciendo por mí a pesar de que no lo merezca ... Te amo mucho.

Carlos Zurita

AGRADECIMIENTOS

A mi familia por haberme apoyado siempre. A mi tía Ñata(+) por su infinito amor, alegría y paciencia. A mis amigos de la Maestría Carlos, Santiago y Jorge, gracias por los buenos momentos compartidos.

A todas aquellas personas que me ayudaron a concluir este trabajo. Un agradecimiento especial a mi director de tesis, Fernando Sandoya, quien fue un guia y soporte en la elaboración de este trabajo.

A Dios por haberme dado la salud para poder emprender y concluir todos los retos propuestos.

Vinicio Sevilla

AGRADECIMIENTOS

A mi abuela Elba, mi abuelo Manuel (+), mi tia Fabi y mi tio Carlos Augusto, gracias por haber sido un apoyo para mamá en mi crecimiento y educación.

A José Luis y Carlita, gracias por darme muchos momentos de alegría y cariño.

A Johana gracias por tu calidez y ternura mi amor.

A mis amigos de la Maestria, Vinicio y Santiago gracias por su enseñanza, apoyo y buenos momentos compartidos.

A mis mejores amigas de la vida, Alejandra y Mónica E. las mejores personas que alguien pueda conocer, gracias por su amistad.

A todos los amigos que a pesar de que soy ingrato siempre los recuerdo con mucho cariño, gracias por todo.

A mi Dios, gracias por haberme permitido conocer a tantas personas maravillosas con quien estar agradecido . . .

Carlos Zurita

DECLARACION EXPRESA

La responsabilidad por los hechos y doctrinas expuestas en este Proyecto de Graduación, así como el Patrimonio Intelectual del mismo, corresponde exclusivamente al ICM (Instituto de Ciencias Matemáticas) de la Escuela Superior Politecnica del Litoral.

Ing. Vinicio Sevilla Quevedo **Ing. Carlos Zurita Noriega**

TRIBUNAL DE GRADO

MSc. Fernando Sandoya
DIRECTOR DE TESIS

MSc. Washington Armas
PRESIDENTE TRIBUNAL

M.A.E. Pedro Echeverría
VOCAL

AUTORES DEL PROYECTO

Ing. Vinicio Sevilla Quevedo **Ing. Carlos Zurita Noriega**

INDICE GENERAL

INDICE DE FIGURAS

INDICE DE TABLAS

RESUMEN

La presente investigación ha sido desarrollada para optimizar las actividades de secuenciamiento de producción y asignación de mano de obra en una empresa específica, que por cuestiones de confidencialidad de la información usada denominaremos empresa ABC. Actualmente, en la Empresa ABC la Planificación Finita de Producción, el Secuenciamiento de la Producción y la asignación de mano de obra a las diferentes tareas se realiza manualmente. Esta tarea toma aproximadamente 2 días de trabajo a la semana de los siguientes elementos de la dirección de la empresa: el Jefe de Planificación, el Jefe de Producción y el Gerente de Producción. Las principales restricciones que se toman para la planificación son:

- Número de personal disponible
- Ingreso de materia prima fresca
- Horas hombre necesarias por día

Debido a que se emplean múltiples recursos es importante que la Planificación sea resuelta de manera óptima, de forma que se puedan ahorrar todos los recursos asociados al problema. Para poder resolver el problema de manera óptima en esta tesis, debido a que se trata de un problema NP-duro, se plantea un algoritmo genético, que implementado en el lenguaje de propósito general C++, permite resolver el problema de secuenciación de órdenes de trabajo, en inglés: Job Shop Scheduling (JSSP). El algoritmo propuesto se fundamenta en el principio de selección natural y utiliza mutación clonal, y se demostró que funciona de manera eficiente en la resolución de los problemas de prueba planteados en esta tesis. También se incorpora un mecanismo que permite explorar la vecindad con la selección de la mejor solución, de esta manera se logra una mejora significativa de resultados.

Adicionalmente para probar la validez del algoritmo construido se prueba resolviendo problemas estándar encontrados en la literatura del secuenciamiento de producción. De estas pruebas se concluye que el algoritmo propuesto tarda más tiempo en resolver los problemas estándar que los métodos actuales, sin embargo para resolver el problema propuesto en la Empresa ABC toma aproximadamente 3 minutos obteniéndose una mejor solución a la actual y permitiendo que las personas de planificación y producción puedan dedicar su tiempo a otras tareas.

OBJETIVOS GENERALES

El objetivo principal de ésta investigación es desarrollar una metaheurística que resuelva el problema de secuenciamiento de la producción y asignación de mano de obra en una empresa manufacturera. Para ello se ha investigado el estado del arte en los problemas de secuenciamiento de la producción (Job Shop Scheduling Problem). con el fin de encontrar la metaheurística más apropiada para resolver el problema planteado.

El algoritmo desarrollado tiene que ser mucho más eficiente que el método tradicional usado que consiste en una planificación manual basada en la experiencia de las personas del área de producción de la empresa.

OBJETIVOS PARTICULARES

Los objetivos particulares de la presente investigación son:

- Desarrollar un algoritmo genético que secuencie la producción de la empresa ABC.
- Calibrar los parámetros del algoritmo genético usando una técnica experimental de aproximación de un solo factor.
- Disminuir las horas extras no planificadas e improductivas con un mejor secuenciamiento de la producción.
- Facilitar la planificación para el responsable en la Empresa ABC de forma que sea más fácil controlar la programación de piso.
- Poder manejar escenarios y hacer cambios de planes en función de la información real del cumplimiento de planificación.

INTRODUCCION

La mejora de los procesos productivos en las empresas manufactureras ha sido un factor clave para el éxito de los negocios. Estos procesos se han vuelto muy complicados debido a que son procesos sistémicos en los que el objetivo a cumplir es fabricar una cierta cantidad determinada de un mix de productos deseados al menor costo posible con recursos limitados y en un rango de fechas requeridas. En estos procesos intervienen recursos como personal, máquinas, materia prima e insumos.

El objetivo es determinar el Plan de Producción y ejecutarlo en una secuencia tal que se minimicen tiempos improductivos, tardanza en las órdenes, u otros criterios deseados. Adicionalmente hay que tomar en cuenta que los recursos que se utilizan en las diversas operaciones productivas son limitados y costosos. El problema se complica dado que existen comúnmente dependencias entre las diversas operaciones. De ésta forma resulta relevante para los sistemas de manufactura, el estudio de los problemas de planificación o scheduling en los cuales se trata de determinar la secuencia y las fechas exactas en las cuales deben ser realizadas las tareas planificadas.

Dentro del área del Scheduling, el problema del Job Shop (JSSP) es el que más aplicación práctica ofrece, pues permite incrementar la eficiencia de los procesos de manufactura. Desde los primeros años de la década de 1960, los problemas de Job Shop, han sido de mucho interés para los investigadores en métodos de optimización, ya que son problemas difíciles del tipo NP-duro. En el pasado se han desarrollado un sinnúmero de Heurísticas específicas para este problema. De éstas, el procedimiento de sustitución del cuello de botella (Shifting Bottleneck Procedure) se convirtió en uno de los más usados. Hace pocos años el desarrollo de nuevas técnicas universales de búsqueda han tenido auge gracias a los avances computacionales y podrían convertirse en métodos satisfactorios para resolver el problema del job shop (JSSP). Los Algoritmos Genéticos, el Recocido Simulado, Búsqueda TABU, Búsqueda Variable Local Paralela entre otras son las nuevas técnicas que se están utilizando para poder resolver el problema del Job Shop.

El objetivo principal de éste estudio es encontrar un algoritmo que permita encontrar una mejor solución al método actualmente propuesto y sin ocupar el tiempo valioso del personal de Producción y Planificación de la Empresa ABC.

CAPITULO 1

REVISION BIBLIOGRAFICA

1.1. Planificación de la Producción

Oliver Wight quien es considerado actualmente un experto en planificación de la producción, debido al concepto propuesto de excelencia operacional a través del plan de ventas y operaciones, estableció que la planificación de producción debe tomar en cuenta dos problemas principales los cuales son ***prioridades*** y ***capacidad***. En otras palabras determinar la secuencia de producción y en que máquina se hará.

Cabe indicar que en las empresas de producción existen múltiples tipos de planificación, incluyendo la planificación de ventas y operaciones, el plan maestro de producción, la planificación de capacidad, la planificación de requerimiento de materiales, la planificación de recursos empresariales, la planificación y control de manufactura y la planificación finita o secuenciamiento detallado en el piso de producción. Este último tipo de planificación indica específicamente los tiempos de inicio y finalización de cada una de las operaciones en los centros de trabajo existentes, además es el más complejo debido a todas las restricciones que deben ser analizadas para obtener un secuenciamiento óptimo de producción.

> "La Planificación finita se define como la asignación real de fechas de inicio y finalización de las operaciones o grupo de operaciones para indicar cuando deberían ser ejecutadas para completar la orden de producción a tiempo"[1]

[1] Cox et al. APICS Dictionary. 1992 17 pag.

1.2. Nacimiento de la Planificación de la Producción

Con el nacimiento de la Revolución Industrial en la segunda mitad del siglo XVIII, aparecieron formalmente talleres centralizados donde se fabricaban los productos requeridos. Esta Revolución Industrial provocó cambios socioeconómicos, tecnológicos y culturales en la humanidad, debido principalmente a que la economía basada en el trabajo manual fue reemplazada por la basada en el trabajo industrial. Estos talleres fueron el inicio de las fábricas modernas. Los primeros talleres eran pequeños, en los cuales se fabricaba pocos productos en grandes lotes. Las mejoras de productividad provenían de disminuir las operaciones de calibración que consumían mucho tiempo por lo que no había un enfoque en la planificación de producción. Al finalizar el siglo XIX las empresas manufactureras se empezaron a preocupar de maximizar la productividad de los costosos equipos de la fábrica. Es decir el principal objetivo era mantener la utilización de línea (producción real versus producción teórica en porcentaje) en niveles altos, para esto los administradores empezaron a dar importancia a la planificación de producción.

> "Al Inicio de la Epoca Industrial la Planificación, cuando era usada, consistía únicamente de un listado del inicio y finalización del trabajo y no proporcionaba ninguna información de la duración total de cada orden o del tiempo requerido para completar las operaciones individuales"[2]

El listado de inicio y finalización de trabajo fue ampliamente usado antes del nacimiento de los métodos formales de planificación (aunque todavía puede ser encontrado en algunas fábricas); en otras palabras la planificación de la producción era efectuada por una persona con técnicas muy rudimentarias y básicas para el efecto. Cabe indicar que los métodos informales como el basado en el despacho todavía no han desaparecido.

> "La administración de inventario y producción en muchas compañías actualmente consiste únicamente en lanzar órdenes y despachar"[3]

[2]Roscoe y Freark. Organization for Production. 1971. 15 pag.
[3]Wight, O. Production and Inventory Management in the Computer Age.1984.25 pag.

1.3. El Nacimiento de los Sistemas Formales

Para el principio del siglo XX todo cambió, debido a que las empresas manufactureras empezaron a fabricar una amplia variedad de productos, la misma que incrementó la complejidad a los administradores. La administración científica fue la respuesta para mejorar el control de esta complejidad. Las fábricas crecieron significativamente, con lo cual la reducción de costos fue por primera vez el objetivo fundamental de los responsables. Por estas razones empezó el trabajo de alcanzar economías de escala y costos unitarios de producción bajos.

1.3.1. La oficina de control de producción

Inicialmente los operarios planeaban libremente las actividades de su trabajo según la forma en que ellos creían conveniente. Frederick Taylor propuso que los administradores realicen la planificación de actividades, de ésta manera separó la planificación de la ejecución de la producción con lo que se justificó el uso de métodos formales de planificación y la creación de la oficina de control de producción.

La filosofía de Taylor se fundamentaba en tres aspectos:

- Las leyes científicas gobiernan cuanto puede producir un trabajador por día.
- Es función de la gerencia descubrir y utilizar estas leyes en la operación de sistemas productivos.
- Es función del trabajador cumplir los deseos de la gerencia sin cuestionarlos.

A partir de los postulados de Taylor, las gerencias requirieron personal para planificar, administrar el inventario y monitorear las operaciones. La oficina de control de producción creaba un plan maestro de fabricación basado en las órdenes en firme y en la capacidad de la planta. El objetivo principal de esta oficina era crear órdenes de trabajo en función de los requerimientos más urgentes y abastecer de material al piso de producción.

1.3.2. Henry Gantt y su diagrama

Debido a que aún existía mucha informalidad y no se tenían métodos estandarizados entre las oficinas de producción de las diferentes fábricas, Gantt creó un diagrama innovador, que lleva su nombre, para el control de la producción (1). El diagrama de Gantt es una herramienta visual para indicar la secuencia de los trabajos que serán efectuados en un conjunto de máquinas y para monitorear su avance.

Gantt creó muchos tipos de diagramas que representan diferentes formas de estudiar los sistemas de producción y analizar métricas. Además diseñó sus diagramas de forma que los planificadores pudiesen rápidamente conocer el uso ideal y el uso real de los recursos a través del tiempo.

Gantt creó muchos tipos de diagramas que representan diferentes formas de estudiar los sistemas de producción y analizar métricas. Además diseñó sus diagramas de forma que los planificadores pudiesen rápidamente conocer el uso ideal y el uso real de los recursos a través del tiempo.

Tabla 1.1: *Principales diagramas creados por Gantt*

Tipo de diagrama	**Unidad**	**Cantidad que va a ser medida**	**Representaciones de tiempo**	**Fuentes**
Balance diario de trabajo	Parte u operación	Cantidad producida	Filas para cada día;las barras muestran datos de inicio y datos de fín.	Gantt, 1903; Rathe, 1961
Registro de personal	Trabajador	Cantidad de trabajo hecho cada día y semana, medidos como tiempo	3 o 5 columnas por cada día en dos semanas	Gantt, 1981; Rathe, 1961
Registro de maquinaria	Máquina	Cantidad de trabajo hecho cada día y semana, medidos como tiempo	3 o 5 columnas por cada día en dos semanas	Gantt, 1919, 1981; Rathe, 1961
Diagrama dee ubicación	Máquina	Avance de tareas asignadas, medidas como tiempo	3 o 5 columnas por cada día en dos semanas	Clark, 1942
Diagrama de trabajo de Gantt	Tipo de máquina	Tareas programadas y el total de trabajo por fecha	Una columna por cada día para dos meses	Mitchell, 1939
Diagrama de trabajo de Gantt	Orden de trabajo	Trabajo completado para una fecha, medido como tiempo	Una columna por cada día para dos meses	Mitchell, 1939
Diagrama de secuenciamiento	Tareas en un trabajo	Inicio y fin de cada tarea	El eje horizontal dividido en 45 días	Muther, 1944
Diagrama de avance	Producto	Cantidad producida cada mes	5 columnas para cada mes para cada año	Gantt, 1919, 1981; Rathe, 1961
Diagrama de órdenes de trabajo	Orden de trabajo	Cantidad producida cada mes	5 columnas para cada mes para cada año	Gantt, 1919, 1981; Rathe, 1961

1.3.3. La planificación basada en computadora

A pesar que la planificación de la producción era una tarea compleja, los esfuerzos iniciales de usar algoritmos computacionales se centraron en la planificación de grandes proyectos, principalmente de construcción o desarrollo de sistemas. La planificación de la producción basada en la computadora comenzó en la segunda mitad de la década de los 60 aproximadamente.

Existen tres factores que contribuyeron al uso de computadoras para la planificación de la producción, estos son:

- IBM desarrolló el sistema de control e información de la producción en 1965.
- La implementación de este sistema y otros similares guiaron al conocimiento práctico del uso de computadoras.
- Los investigadores compararon estas experiencias y desarrollaron nuevas ideas en la administración de la producción.

Los primeros sistemas computacionales de planificación fueron proyectos de investigación realizados en universidades los cuales usaron computadoras centrales y terminales. Las terminales recibían datos acerca de la finalización de las tareas o el movimiento de material; la computadora central actualizaba la información, incluía registros para las máquinas y empleados, materiales consumidos y colas en las estaciones de trabajo. A partir de esta información, la computadora creaba para cada estación una lista de despacho con las tareas pendientes que debían ser procesadas. Para crear una lista de despacho, el sistema usaba una regla que consideraba uno o más factores incluyendo el tiempo de proceso, fecha de finalización, tardanza, número de operaciones restantes o costos de producción.

La planificación basada en computadora evolucionó desde proyectos de investigación hacia sistemas comerciales. Uno de los primeros sistemas comerciales fue el Sistema de Administración Logística (LMS por sus siglas en inglés), desarrollado por IBM para sus instalaciones de producción de semiconductores. El sistema LMS comenzó a operar alrededor de 1980 como una herramienta para modelar los recursos productivos. Los módulos capturaban información del piso, recuperaban las prioridades del plan de producción diario y finalmente tomaba decisiones de despacho.

En conclusión el sistema tenía dos objetivos básicos:

- Soporte pasivo a las decisiones: Proporcionando información detallada a los usuarios acerca de la producción; y,
- Despacho proactivo: Proporcionando sugerencias de producción en función de reglas establecidas con anterioridad.

Este software usó estrategias estándar para generar una planificación sugerida que luego podía ser modificada por el personal de acuerdo a sus necesidades. La ventaja más grande de usar computadoras es la reducción drástica de tiempo usado en generar un plan de producción. De esta forma se desarrollaron los sistemas comerciales que planifican y controlan la producción.

A medida que los sistemas basados en computadora se volvían populares los programadores mejoraron su funcionalidad enfocándose finalmente en sistemas que podían tomar decisiones por sí solos.

1.3.4. Evolución a los sistemas de Planificación finita de la producción

La aplicación de las computadoras se extendió a otras funciones de planificación de la producción, las cuales se listan a continuación en orden cronológico:

- La Planificación de Requerimiento de Materiales (MRP): Traslada la demanda de artículos terminados en una planificación basada en el tiempo de compra y de producción de insumos y semielaborados respectivamente.
- La Planificación de Requerimientos Empresariales (ERP): Método para la planificación y el control sincronizado de todos los recursos involucrados en la cadena de valor de una empresa. Los ERP se componen generalmente de módulos como Contabilidad, Finanzas, Manufactura, Producción, Transporte, Ventas y Distribución, Recursos Humanos y Materiales.
- La Planificación Finita o Secuenciamiento de Producción: Toma en cuenta las condiciones reales del piso incluyendo restricciones de capacidad, herramientas,

personal, fechas de entrega comprometidas y los requerimientos de las órdenes que ya han sido lanzadas. Esta planificación es una pieza clave del sistema ERP II, mismos que incluyen además Gestión de Cadena de Suministro (SCM), Gestión de Relaciones con el Cliente (CRM) y e-business.

Los Sistemas Finitos tienen 3 componentes primarios que son:

- Base de datos.
- Motor de planificación
- Interfaz de usuario.

El sistema de planificación puede compartir una base de datos con otro sistema (como un MRP) o tener su base propia la cual se actualiza automáticamente desde el otro sistema. La interfaz de usuario ofrece diferentes formas de ver la planificación, entre las más importantes están los diagramas de Gantt, listas de despacho y gráficos de uso de recursos. El motor de planificación genera sugerencias para lo cual usa planificación basada en reglas, optimización o simulación.

Los sistemas de Planificación Finita de Producción se han vuelto una herramienta fundamental para el éxito de las empresas manufactureras. De hecho se ha establecido el uso obligatorio de estos sistemas para poder obtener una ventaja competitiva que permita a las empresas actuales sobrevivir a la dura competencia y a los cambios en los factores externos que afectan la rentabilidad.

A pesar de la importancia de los Sistemas de Planificación Finita de Producción solamente el 30 % de las Empresas en Estados Unidos los usan como parte de sus operaciones. En países del tercer mundo la cifra se reduce a incluso menos del 5 %, lo cual demuestra que en estos países existe una gran oportunidad de mejora.

Los factores determinantes para que las empresas no usen éste sistema de planificación son:

- Desconocimiento de la existencia y los beneficios que éste sistema brinda. Adicionalmente, falta de bases matemáticas en optimización combinatoria para poder desarrollar un sistema dentro de la propia empresa.

- Costos de implementación altos y accesibles solo a grandes empresas. Una implementación cuesta en promedio 150.000 USD teniéndose programas que pueden llegar incluso al millón de dólares.

- Tiempos de implementación altos, toma en promedio 4 meses.

- Pocos proveedores quienes ofrecen sistemas compatibles sólo con los ERP de las mismas firmas desarrolladoras. Para usar un ERP de una firma y un secuenciador de otra se requerirán desarrollar las interfases necesarias; esto se traduce en mayores costos y tiempos de implementación.

- La mayor desventaja es que estos sistemas son verdaderas ***cajas negras***, es decir el usuario no sabe el algoritmo que está usando el sistema para encontrar la solución. Debido al desconocimiento en optimización combinatoria, el usuario no puede demostrar la validez de la respuesta. En conclusión no se conoce si se está comprando a alto precio un sistema antiguo que no genera buenas soluciones.

1.4. Clasificación de secuenciamiento de la producción

Con el interés de los investigadores enfocado en encontrar una Planificación Finita de la Producción óptima, se propusieron varios problemas estándar de secuenciamiento de la producción. Debido a la gran cantidad de entornos productivos existentes se plantearon una variedad de problemas, muchos de los cuales se volvieron teóricos sin aplicación práctica pero fueron la base para desarrollar métodos de solución a la planificación finita.

Todos los problemas de secuenciamiento de producción tienen características comunes, sin embargo se los puede clasificar bajo los siguientes criterios:

- Proceso de llegada de las tareas: Divididos en estáticos (se conoce exactamente el inicio de las tareas) o dinámicos (el inicio de las tareas depende de una función probabilística).

- Políticas de inventario: Divididos en abiertos (los productos siguen la política de producción bajo pedido) o cerrados (los productos siguen una política de producción para inventario). En la vida real se encuentran sistemas híbridos (mezcla de las dos políticas).

- Atributos de las tareas: Pueden ser determinísticos (los tiempos de proceso y la disponibilidad de máquinas es conocida previamente) o estocásticos (no se cumple lo anterior).

- Configuración de la producción: Esto depende si la producción es de un solo paso o de múltiples pasos. En el primer caso cada tarea se hace en una máquina mientras que en el segundo caso, cada tarea consta de una serie de operaciones que pueden ser procesadas en diferentes máquinas.

Otra clasificación y con seguridad la más usada por los Investigadores Teóricos es la basada en el flujo de tareas a través de los recursos, dependiendo de esto se clasifican en:

- Problemas de secuenciamiento de flow shop.

- Problemas de secuenciamiento de job shop.

- Problemas de secuenciamiento de open shop.

- Otros problemas de secuenciamiento.

1.4.1. El Problema de secuenciamiento del Flow Shop (FSSP)

El Flow Shop Scheduling Problem es un problema de secuenciamiento de la producción que de forma general se describe así:

Se tiene un conjunto de N tareas $j = (J_1, J_2, J_3, \ldots, J_n)$. que tienen que ser procesadas en un conjunto M de máquinas $m = (M_1, M_2, M_3, \ldots, M_m)$. En el FSSP general todas las tareas J deben ser procesadas secuencialmente en todas las máquinas M, en el mismo orden. Se asume que una operación se inicia solamente si se ha terminado la anterior y que los tiempos de cambio y calibración están incluidos en los tiempos de proceso. Adicionalmente, cada tarea debe ser procesada en solamente una máquina a la vez, y cada máquina puede procesar una tarea a la vez. El problema consiste en encontrar la secuencia de tareas factible que optimice un criterio específico determinado (tiempo total de proceso, número de tareas atrasadas o tiempo de tardanza total).

Existen muchas variaciones del FSSP como:

- El piso de producción con cero almacenamiento: Una tarea j que ha sido procesada en una máquina M_i no puede avanzar a la máquina M_{i+1} hasta que ésta termine de procesar la tarea predecesora a j. En éste caso la tarea j permanece en la máquina M_i, retrasando también el trabajo de ésta máquina.
- El piso de producción con bloqueo: Es similar a la anterior pero en éste caso puede existir un almacenamiento intermedio.

Aunque todas estas variantes del FSSP tienen muchas aplicaciones industriales los investigadores se han enfocado principalmente en una versión reducida del problema FSSP general, donde se asume que las tareas deben ser procesadas en la misma secuencia por cada una de las M máquinas.

1.4.2. El problema del secuenciamiento del Job Shop (JSSP)

El Job Shop Scheduling Problem es un problema de secuenciamiento de la producción que de forma general se describe así: Se tiene un conjunto de máquinas $m = (M_1, M_2, M_3, \ldots, M_m)$ las cuales deben realizar un conjunto de tareas $j = (J_1, J_2, J_3, \ldots, J_n)$. En el JSSP general, cada tarea j debe ser procesada en una máquina a la vez y requiere de una serie de operaciones $O_{1j}, O_{2j}, \ldots, O_{nj}$. Estas operaciones deben ser secuenciadas en un orden predeterminado diferente para cada tarea. Se tiene como restricción que una operación se inicia solamente si se ha terminado la anterior ($O_{ij} \longrightarrow O_{i,j+1}$) y que los tiempos de cambio y calibración están incluidos en los tiempos de proceso que son conocidos previamente p_{ij}. Adicionalmente, cada tarea debe ser procesada en solamente una máquina a la vez, y cada máquina puede procesar una tarea a la vez. La dimensión de una instancia de JSSP es n x m, donde n es el número de tareas a realizar y m el número de máquinas disponibles. El problema radica en encontrar una secuencia de operaciones factible para cada máquina de forma que se optimice un criterio específico, generalmente el tiempo total de proceso. Inicialmente, las soluciones del JSSP se representaron mediante Diagramas de Gantt, lamentablemente estos diagramas no indican las relaciones existentes entre actividades; estas restricciones llevaron al desarrollo de otros modelos de representación. Se desarrollaron Diagramas de Gantt más avanzados como el Diagrama de Gantt en tres dimensiones (tareas, máquinas y tiempo) para mejorar la representación de los problemas de secuenciamiento de la producción.

Otro método de representación y el más usado actualmente es la representación mediante grafos disyuntivos. Un grafo disyuntivo está dado por: $G = (V, A, E)$, donde:

G: Grafo disyuntivo.

V: Conjunto de nodos del grafo que representa las operaciones a ser procesadas en el conjunto de máquinas m.

A: Conjunto de arcos conjuntivos que representan las restricciones de procedencia entre las operaciones de cada tarea, de forma que $(O_{kl}, O_{km})\epsilon A$, indica que la operación O_{kl} es la inmediata predecesora de la operación Okm.

E: Conjunto de arcos dirigidos y que representan las restricciones de capacidad, enlazando un par de arcos disyuntos que comparten una misma máquina. Así, dos operaciones O_{kl}, O_{il} que son ejecutadas en la misma máquina l no pueden hacerlo simultáneamente.

El nodo inicial ficticio S es llamado la fuente y el nodo ficticio F es llamado sumidero. El peso positivo para cada nodo representa el tiempo de procesamiento d_{kl}. El nodo inicial se conecta con la primera operación de cada tarea, mediante un arco que tiene por longitud el tiempo de liberación (r_i) y de igual forma la última operación de cada tarea se conecta con el nodo final. El objetivo del problema es minimizar el tiempo total de proceso satisfaciendo las restricciones mediante la eliminación de ciertos arcos disyuntivos y convertir los que quedan en conjuntivos según el método de secuenciamiento que se quiera utilizar. En la Figura 1.1 se presenta un grafo disyuntivo para un problema de job shop (4×3) (cuatro tareas y tres máquinas).

Con estos argumentos se puede definir el JSSP matemáticamente de la siguiente manera:

Min. t_n(camino mas largo entre S y F)

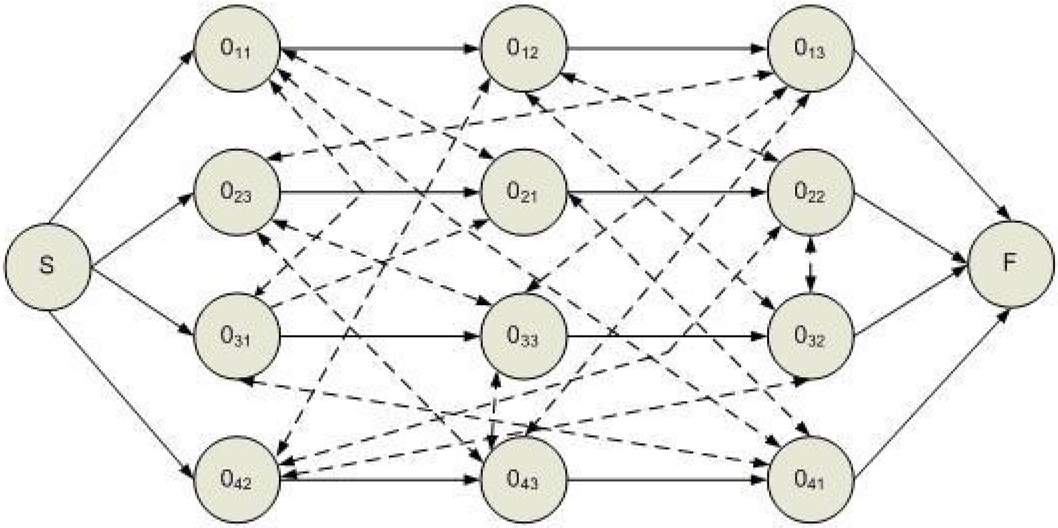

Figura 1.1: *Un grafo disyuntivo representando un problema de Job Shop*

S.A.

$t_{km} + t_{kl} \geq d_{kl}$

$t_{kl} + t_{il} \geq d_{il} \vee t_{il} + t_{kl} \geq d_{kl}$

$t_{kl} \geq r_k$

En donde:

t_n**:** Tiempo de inicio del nodo sumidero, tiempo total o makespan del proceso.

t_{km}**:** Tiempo de inicio de la tarea k en la máquina m.

d_{kl}**:** Duración de la tarea k en la máquina l.

r_k**:** Tiempo de liberación de la tarea k [3].

Cálculo del Makespan en el Job Shop (JSSP)

Para el problema del JSSP se pueden tener diferentes criterios de optimización como minimización del makespan, minimización de la tardanza promedio o maximización de las órdenes entregadas a tiempo. Generalmente estos objetivos se contraponen por lo que es recomendable estudiar a fondo el problema a resolver e identificar el criterio que se quiere optimizar. Se indica a continuación la forma de calcular el makespan para el posterior desarrollo experimental del trabajo. El makespan es el mayor tiempo de proceso de todas las tareas a realizar siguiendo un secuenciamiento factible, para entender mejor el cálculo se presenta un ejemplo numérico: se cuenta con 4 tareas y 4 máquinas. En el siguiente cuadro se indica el número de operaciones y los tiempos invertidos por tarea (para el efecto se ha numerado secuencialmente cada operación).

Tabla 1.2: *Datos para cálculo del makespan*

I	1	2	3	4	5	6	7	8	9	10
di	2	2	1	2	7	4	5	2	3	7
J(i)	1	1	1	2	2	3	3	3	4	4
u(i)	M1	M4	M2	M4	M2	M1	M4	M2	M4	M3

- I Operación
- Di Tiempo de proceso de la operación i
- J(i) Tarea a la que pertenece la operación i
- u(i) Máquina donde se procesa la operación i

Se presenta a continuación la Figura 1.2 que contiene un grafo disyuntivo donde se presenta una de las soluciones factibles.

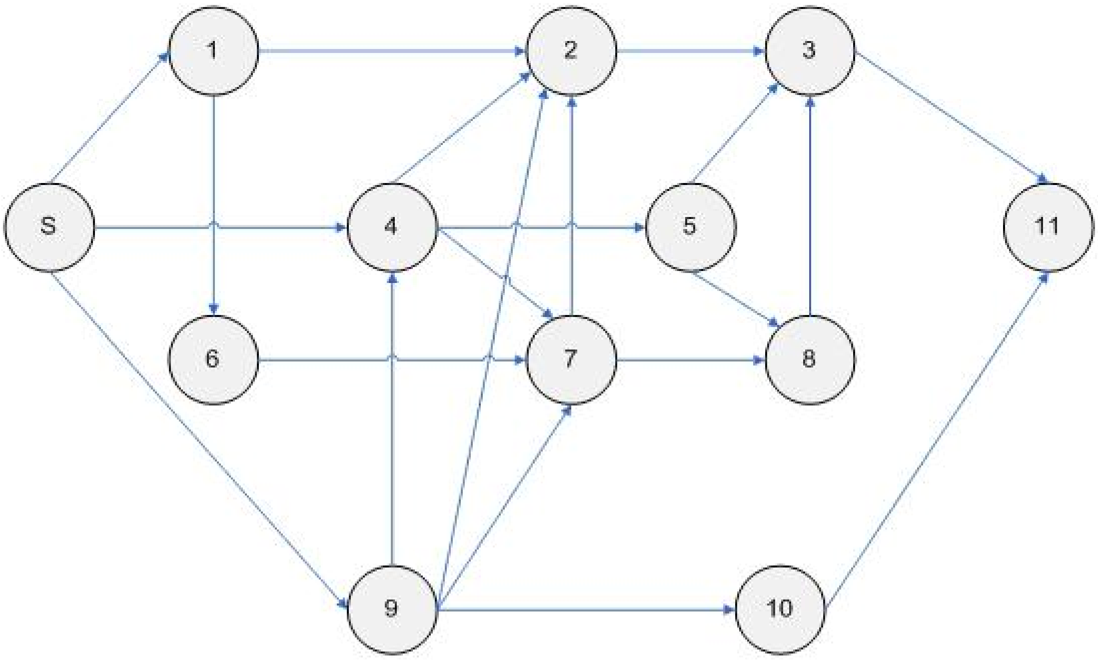

Figura 1.2: *Grafo disyuntivo para el cálculo del makespam*

En la Figura 1.3 se aprecia una propuesta de secuenciamiento.

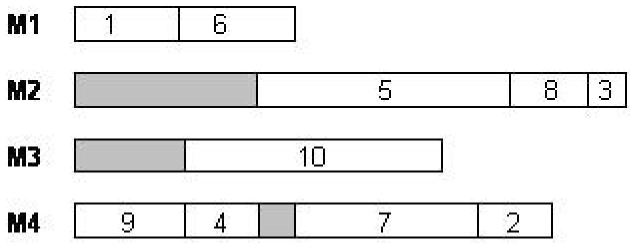

Figura 1.3: *Grafo disyuntivo para el cálculo del makespam*

La ruta más larga para el cálculo del makespan es cumplir las operaciones 9-4-5-8-3 lo que da un tiempo total de 15. El makespan obtenido en el ejemplo es el camino en el que una operación es inmediatamente continuada con la siguiente operación en la misma máquina o por la siguiente operación de la tarea en una máquina diferente. El problema de obtener el menor makespan se reduce a obtener una selección de arcos disjuntos que minimicen la longitud del camino más largo.

1.4.3. El problema del secuenciamiento del Open Shop (OSSP)

El Open Shop Scheduling Problem es un problema de secuenciamiento de la producción, que de forma general se describe así:

En el OSSP no se tiene una secuencia predefinida de operaciones entre las tareas y por lo tanto el OSSP tiene un espacio de soluciones mucho mayor al JSSP con dimensiones similares $n \times m$. Probablemente, el mejor ejemplo de un open shop es un taller de reparación de autos donde la secuencia de operación/reparación no está definida estrictamente. En general, en el OSSP, cada tarea debe ser procesada en cada máquina y consiste en una serie de mi operaciones las cuales tienen que ser secuenciadas en cualquier orden. De forma similar al FSSP y al JSSP cada tarea puede ser procesada en una máquina únicamente a la vez.

1.5. Complejidad computacional de los problemas de secuenciamiento de la producción

La complejidad computacional estudia todos los recursos requeridos durante el procesamiento de un algoritmo usado para resolver un problema determinado; en éste caso un problema de secuenciamiento de la producción. Los recursos más analizados son:

- Tiempo: relacionado con el número y tipo de pasos para la ejecución del algoritmo.
- Espacio: relacionado con la cantidad de memoria usada para la ejecución del algoritmo.

De acuerdo al orden de complejidad los problemas se clasifican en:

Clase P: Los algoritmos de complejidad polinómica se dice que son tratables en el sentido de que suelen ser abordables en la práctica.

Clase NP: Son problemas complejos intratables, los cuales se caracterizan por el hecho de que puede aplicarse un algoritmo polinómico para comprobar si una posible solución es válida o no. Esta característica lleva a un método de resolución no determinista, que consiste en aplicar heurísticos para obtener soluciones hipotéticas que se van desestimando (o aceptando) a ritmo polinómico. Los problemas de esta clase se denominan NP (la N de no-deterministas y la P de polinómicos).

Clase NP Completos: Se conoce una amplia variedad de problemas de tipo NP, de los cuales destacan algunos de ellos de extrema complejidad. Estos problemas se caracterizan por ser todos ïguales.[en] el sentido de que si se descubriera una solución P para alguno de ellos, esta solución sería fácilmente aplicable a todos ellos. Realmente, tras años de búsqueda exhaustiva de dicha solución, es un hecho aceptado que no debe existir, aunque nadie ha demostrado, todavía, la imposibilidad de su existencia. Una alternativa para resolver los problemas NP-completos son los algoritmos heurísticos que si bien no encuentran la solución óptima a un problema NP si determinan buenas soluciones. Ejemplos de problemas NP-completos son el problema del agente viajero (TSP), el problema de coloreamiento de un grafo y el problema de la satisfacibilidad.

Clase NP Duros: Esta clase la cual es la más compleja puede ser descrita como aquella que contiene los problemas de decisión que son al menos tan difíciles como un problema NP, sin embargo para los problemas NP- duros no es posible encontrar un algoritmo que realice una búsqueda exhaustiva. Cabe indicar que todos los problemas reales"de secuenciamiento de la producción son de clase NP-duros (para problemas cuyo número de máquinas es mayor a tres) debido a que toman en cuenta muchos factores de un proceso sistémico.

1.6. Metodologías de solución para los problemas de secuenciamiento de la producción

Los problemas de secuenciamiento de la producción son problemas complejos y sistémicos donde las soluciones son representadas mediante variables discretas. Estos problemas pueden ser resueltos de dos formas:

- Mediante algoritmos exactos o completos para problemas de menor complejidad (teóricos y máximo 3 máquinas).
- Mediante algoritmos de aproximación para problemas más complejos y que consisten en encontrar soluciones cercanas a la óptima. Se puede usar algoritmos heurísticos o meta-heurísticos.

1.6.1. Algoritmos Exactos

Los algoritmos exactos garantizan el poder encontrar la solución óptima en un tiempo de procesamiento razonable. Lamentablemente para los problemas NP-duros como los de secuenciamiento de producción no existe ningún algoritmo exacto para resolverlos en un tiempo polinomial. Para problemas complejos de la vida práctica los algoritmos exactos necesitan tiempos computacionales exponenciales lo que lleva a soluciones no prácticas. La familia de los métodos exactos es considerablemente grande pero los métodos exactos más comunes para los problemas de secuenciamiento de la producción son los algoritmos branch and bound, programación entera mixta y métodos de descomposición.

1.6.2. Algoritmos de Aproximación

> "Debido a los altos tiempos de cómputo necesarios para resolver algoritmos exactos en el secuenciamiento de la producción los investigadores se centraron

en métodos de aproximación. En estos métodos se sacrifica la garantía de encontrar una solución óptima en función de obtener soluciones cercanas a la óptima en tiempos computacionales razonables y prácticos. La forma básica de los algoritmos de aproximación es llamada "heurística". La clasificación más básica de los métodos heurísticos son: métodos constructivos y métodos de búsqueda local"[4].

Algoritmos Constructivos

Los algoritmos constructivos generan soluciones mediante la adición gradual de partes a la solución inicial (la cual es el conjunto vacío). En los problemas de secuenciamiento de la producción, estas partes son operaciones. Las heurísticas constructivas son generalmente los algoritmos de aproximación más rápidos aunque con algunas implementaciones especiales aumenta el tiempo de procesamiento computacional. La gran ventaja en requerimientos de tiempo computacional está compensada con el hecho que generalmente obtiene soluciones de menor calidad cuando se las compara con las técnicas de búsqueda local. Entre las heurísticas constructivas más usadas en el secuenciamiento de la producción están las variantes de las reglas de despacho". En la tabla 1.3 se encuentran las más comunes.

Debido a la facilidad de la implementación y los bajos requerimientos computa-

Tabla 1.3: *Reglas de despacho más comunes*

Regla	Descripción
SOT	Una operación con el menor tiempo de proceso en la máquina considerada
LOT	Una operación con el mayor tiempo de proceso en la máquina considerada
LRPT	Una operación con el mayor tiempo de procesamiento de las tareas remantes
SRPT	Una operación con el menor tiempo de procesamiento de las tareas remantes
LORPT	Una operación con la cola más alta y menor tiempo del proceso
Random	La operación para la máquina considerada es aleatoriamente escogida
FCFS	La primera operación en la cola de trabajos esperados para la misma máquina
SPT	Un trabajo con el menor tiempo total de proceso
LPT	Un trabajo con el mayor tiempo total de proceso
LOS	Una operación con el mayor tiempo de proceso de operación posterior
SNRO	Una operación con el menor número de operaciones posteriores
LNRO	Una operación con el mayor número de operaciones posteriores

cionales las reglas de despacho son las heurísticas más comunes para los problemas de secuenciamiento de la producción. Aunque se comportan bien en ciertos casos no existe una regla que se ajuste bien a todos los problemas de secuenciamiento; es

[4]Fatos Xhafa and Ajith Abraham. Metaheuristics for Scheduling in Industrial and Manufacturing Applications. 12 p.

decir que algunas de las reglas de despacho generan soluciones óptimas solamente en ciertos problemas.

Métodos de Búsqueda Local

Los algoritmos de búsqueda local parten con una solución inicial (la mayoría de veces generada por una heurística constructiva o aleatoria) e iterativamente tratan de reemplazar parte de la solución con una mejor en un conjunto definido de soluciones del vecindario de interés. Con la finalidad de reemplazar partes de la solución inicial los métodos de búsqueda local desarrollan una serie de movimientos para formar nuevas soluciones en el mismo vecindario. Los movimientos más comunes son los siguientes: **2-Opt** (actúa sobre una serie de tareas de longitud al azar en una máquina), el cambio $1 - 1$ (intercambia dos tareas de una misma máquina) y el cambio $1 - 0$ (transfiere una tarea de su posición en una máquina a otra posición en la misma máquina). Por supuesto el número de posibles movimientos y los vecindarios correspondientes son prácticamente infinitos. El principal inconveniente que presentan los métodos de búsqueda local es que fácilmente se quedan atrapados en óptimos locales debido a su naturaleza de búsqueda. Más específicamente en la búsqueda local realizando los movimientos apropiados se puede obtener un método muy efectivo en la exploración del vecindario a partir de una solución inicial pero no existe ningún mecanismo que lleve a buscar mejores soluciones en otros vecindarios. Para remediar este problema se han desarrollado los métodos de búsqueda local modernos **(búsqueda local modificada o VNS)** con estrategias adicionales para buscar soluciones en otros vecindarios.

1.6.3. Metaherísticas

Debido al arduo interés de los investigadores se creó una nueva familia de algoritmos de aproximación, la cual se ha constituido por las enormes ventajas que presenta, en la mejor forma de resolver los problemas de optimización combinatoria y especialmente los problemas de secuenciamiento de la producción. Este nuevo tipo de algoritmos básicamente combina los métodos heurísticos de una forma mejor estructurada. El principal objetivo de la nueva metodología es explorar de una forma más efectiva y eficiente el espacio de búsqueda basado en movimientos lógicos y conocimiento del efecto de estos movimientos de forma que facilita el escape de los óptimos locales. Estos métodos metaheurísticos son mucho mejores que los heurísticos principalmente en que generan soluciones robustas; sin embargo requieren mayores recursos computacionales, información del problema y son más difíciles de implementar.

> "Las metaherísticas son estrategias de un nivel avanzado las cuales permiten tener un mejor desempeño que las heurísticas tradicionales. El principal objetivo es evitar las desventajas del desarrollo iterativo y, en particular, el profundo descenso permitiendo a la búsqueda local escapar de los óptimos locales. Esto se puede conseguir permitiendo movimientos que llevan a soluciones de menor calidad o generando nuevas soluciones iniciales para la búsqueda local de forma más inteligente que dando soluciones iniciales al azar. Muchos de los métodos pueden ser interpretados como la introducción de un bias de forma que se obtienen rápidamente soluciones de alta calidad. Este bias de puede encontrar de varias formas y se clasifica en bias descendente (basado en la función objtivo), bias de memoria (basado en las decisiones previas) o bias de experiencia (basado en el desempeño primario). Muchas de las metaheurísticas usan decisiones probabilisticas durante la búsqueda. Pero la principal diferencia con la búsqueda al azar es que la aleatoriedad en los algoritmos metaheurísticos es usada de forma inteligente"[5].

Las metaheurísticas son estrategias que **guían** el proceso de búsqueda. Los algoritmos exactos son demasiado lentos incluso cuando exploran espacios de soluciones muy pequeños, por otro lado las heurísticas realizan búsquedas **ciegas** y no proporcionan soluciones robustas que puedan ser explicadas de forma lógica.

Recientes investigaciones han demostrado que los mejores métodos para los problemas de secuenciamiento de la producción son los que combinan técnicas de búsqueda local con metaheurísticas de forma que se permita el escape de los óptimos locales aceptando movimientos que **empeoran** la solución y de esta forma soluciones de menor calidad. Aunque el permitir movimientos que degradan la calidad de la solución perecería contradictorio, estos movimientos apoyan a las metaheurísticas para escapar de los óptimos locales con la esperanza que las nuevas soluciones inferiores estén en la vecindad de un óptimo global y por ende, una técnica de búsqueda local en esta vecindad de cómo resultado una solución óptima global. Las estrategias usadas para escapar de los óptimos locales pueden dividirse en estrategias de diversificación (exploración efectiva de todos los vecindarios posibles del espacio de

[5] Blum C. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. 270 p.

soluciones) y de intensificación (basados en un conocimiento particular que permite explorar una subespacio de soluciones más pequeño).

Clasificación de las Metaherísticas

La clasificación más usada se basa en la cantidad de soluciones con las que puede trabajar cada metaheurística. En base a esto se clasifican en:

Búsqueda de punto simple: mejoran una solución específica explorando su vecindario con una serie de movimientos determinados.

Basados en población: combinan un número determinado de soluciones con la finalidad de generar nuevas soluciones que combinen las mejores características de las antiguas y que sean de mejor calidad. Estos son generalmente métodos iterativos que reemplazan gradualmente las soluciones anteriores con las nuevas mejoradas.

Prácticamente todas las metaheurísticas se pueden aplicar para la resolución de problemas de secuenciamiento de la producción. Sin embargo los métodos más usados por la efectividad en este tipo de problemas son:

Algoritmos evolutivos: Este tipo de algoritmos se dividen principalmente en 3 categorías: algoritmos genéticos, estrategias evolutivas y programación evolutiva. Uno de los algoritmos mas recientes en esta categoría es el algoritmo evolutivo diferencial (DEA).

Colonia de hormigas: Se inspira en el comportamiento que rige a las hormigas de diversas especies para encontrar los caminos más cortos entre las fuentes de comida y el hormiguero.

Búsqueda dispersa (Scatter Search): Similar a los algoritmos genéticos pero usan estrategias sistemáticas en lugar de aleatorias

Redes Neuronales: Constituyen una tecnología de inteligencia artificial avanzada que simula el aprendizaje del cerebro y los procesos de toma de decisiones.

Búsqueda Local Básica: Explora una vecindad de una solución con un conjunto de movimientos con la finalidad de obtener un óptimo local.

Búsqueda Local Explorativa: Puede ser GRASP (greedy randomized adaptative search procedure), VNS (variable neighboorhood search) y ILS (iterated local search).

Recocido Simulado: Basado en los cambios energéticos de un sistema de partículas conforme disminuye la temperatura hasta que llega a un estado estable.

Búsqueda Tabú: Utiliza un procedimiento de búsqueda local para moverse iterativamente buscando una mejor solución hasta satisfacer un criterio determinado. El procedimiento se fundamenta en una lista tabú.

Aceptación de umbral: Una modificación de la búsqueda local para permitir aceptar soluciones de menor calidad pero escapar de óptimos locales.

1.6.4. Métodos Híbridos

En los años recientes los investigadores se han concentrado en desarrollar métodos que combinan las fortalezas de las metaheurísticas y reducen sus debilidades. A estos métodos se les conoce como metaheurísticas híbridas. Debido a estas características es posible encontrar mejores soluciones con éstos métodos. Se puede distinguir tres formas de hibridización:

Cambio de Componentes: Combinación de metaheurísticas con técnicas de búsqueda local.

Búsqueda cooperativa: Intercambio de información entre dos o más metaheurísticas.

Métodos de Integración: Combina metaheurísticas y programación con restricciones. Dependiendo del tipo de problema a resolver debe escogerse una de las tres alternativas.

1.6.5. Metaheurísticas para el Job Shop Scheduling Problem

Para resolver el JSSP se han usado varias metaheurísticas, como los algoritmos genéticos. Según los investigadores éste método no proporciona soluciones cercanas a la óptima. Es decir los algoritmos genéticos fallan en la búsqueda de mejores regiones de una vecindad. El uso de colonia de hormigas para resolver este problema es muy limitado ya que el método funciona mejor con variables continuas por lo que generalmente se lo usa combinado con otras metaheurísticas. Los mejores métodos en la resolución de estos problemas se encuentran con el uso de búsqueda variable local (VNS). Sin embargo en términos de aplicaciones prácticas los métodos que dan mejores soluciones son muy complejos de implementar y requieren muchos recursos computacionales por lo que el uso de algoritmos genéticos los cuales son muy simples en comparación con los otros métodos es una buena alternativa.

1.7. Algoritmos Genéticos

Los algoritmos genéticos propuestos por Hollan en 1975 son metaheurísticas que usan estrategias de búsqueda general que permiten resolver problemas de naturaleza combinatoria. Los algoritmos genéticos simulan el proceso de evolución de los organismos vivos y están basados en el principio de ***supervivencia del más apto*** para formar la siguiente generación de posibles soluciones. Estas soluciones se construyen utilizando operadores probabilísticos, como el cruce genético, mutación y supervivencia del más apto (selección). A medida que avanza el proceso evolutivo los individuos más aptos sobreviven, lo que representa las mejores soluciones, mientras que los individuos menos aptos desaparecen. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de éstas.

1.7.1. Espacio de Búsqueda en Algoritmos Genéticos

El espacio de búsqueda es el conjunto de todas las posibles soluciones de un problema concreto. El espacio de búsqueda puede ser bien definido, pero en la mayoría de las ocasiones sólo se conocen algunos puntos. A cada posible solución se le puede asociar una aptitud o puntuación que representa la bondad de ésta solución para el problema. En la naturaleza esto equivaldrá al grado de efectividad de un organismo para competir por unos determinados recursos. Cuanto mayor sea la adaptación de un individuo al problema, mayor será la probabilidad de que el mismo sea seleccionado para reproducirse, cruzando su material genético con otro individuo seleccionado de igual forma. Este cruce producirá nuevos individuos descendientes de los anteriores (los cuales comparten las mejores características de sus padres) y que reemplazan la solución anterior. Un algoritmo genético devolverá la mejor solución de entre todas las posibles que tenga en un momento dado. Cuando se usa un algoritmo genético las posibles soluciones generan otras soluciones a medida que el algoritmo evoluciona. Las características de los algoritmos genéticos son:

- Operan en una población y no en puntos aislados.
- Operan en un espacio de soluciones codificadas y no directamente en el espacio de búsqueda.
- Solamente requiere información del valor de la función objetivo de cada individuo
- Usa transiciones probabilísticas y no determinísticas.

1.7.2. Conceptos Básicos de los algoritmos genéticos

Individuo o cromosoma: Un individuo determina una posible solución del problema que se pretende resolver mediante el algoritmo genético. Las variables de decisión a ser optimizadas generalmente se representan como un vector. Un cromosoma se compone de un número determinado de genes, los cuales pueden ser representados por un alfabeto binario o uno de mayor cardinalidad.

Población: Conjunto de individuos con los que se trabaja en el algoritmo genético. En un algoritmo genético los individuos que constituyen la población van cambiando pero generalmente el tamaño de la misma permanece constante.

Función de ajuste: Se trata de una función evaluadora de la calidad de un individuo como solución al problema. Permite ordenar a los individuos de la población de acuerdo a la bondad de los mismos.

Cruce: Es una de las operaciones fundamentales que intervienen en todo algoritmo genético. Como norma general se aplica después de un proceso de selección de dos individuos y consiste en una combinación de los mismos para obtener como resultado otros dos nuevos individuos.

Mutación: Constituye otra operación fundamental en un algoritmo genético. En este caso se selecciona un individuo, el cual sufre una pequeña modificación aleatoria en su codificación obteniéndose otro individuo nuevo.

1.7.3. Estructura básica de un algoritmo genético

La ejecución básica de un algoritmo genético se presenta en la Figura 1.4. El proceso empieza con la obtención de la población inicial ya sea mediante una heurística o aleatoriamente. Luego se selecciona los individuos para el cruce en función del valor de la función objetivo, para esto se calcula el valor de la función objetivo de todos los individuos. En ésta fase los individuos con bajos valores de la función objetivo son eliminados y reemplazados por individuos ***más aptos*** de forma que la población siempre conserva el tamaño original. El siguiente paso consiste en realizar el cruce entre los individuos (los más aptos tienen más probabilidad de ser seleccionados para el cruce). Posteriormente se realiza una mutación (alteración aleatoria de un gen de un individuo o de la población) a los cromosomas de acuerdo a una probabilidad determinada. Al igual que en la naturaleza la mutación tiene una pequeña probabilidad de ocurrencia y afecta únicamente un pequeño número de individuos. Luego de la mutación se tiene la siguiente población del mismo tamaño que la inicial y se verifica el cumplimiento del criterio de búsqueda.

1.7.4. Parámetros de los algoritmos genéticos

Porcentaje de Cruce (Pc): Indica con que frecuencia se cruzarán los individuos. Si este es 0 %, los hijos serán como los padres y sólo serán alterados por la mutación. Si éste es 100 % todos los individuos nuevos serán creados mediante cruce de los padres de la generación previa. Cuanto más se crucen los individuos se supone que los hijos serán mejores. Sin embargo es recomendable, que algunos individuos pasen sin modificar a la siguiente generación. Si

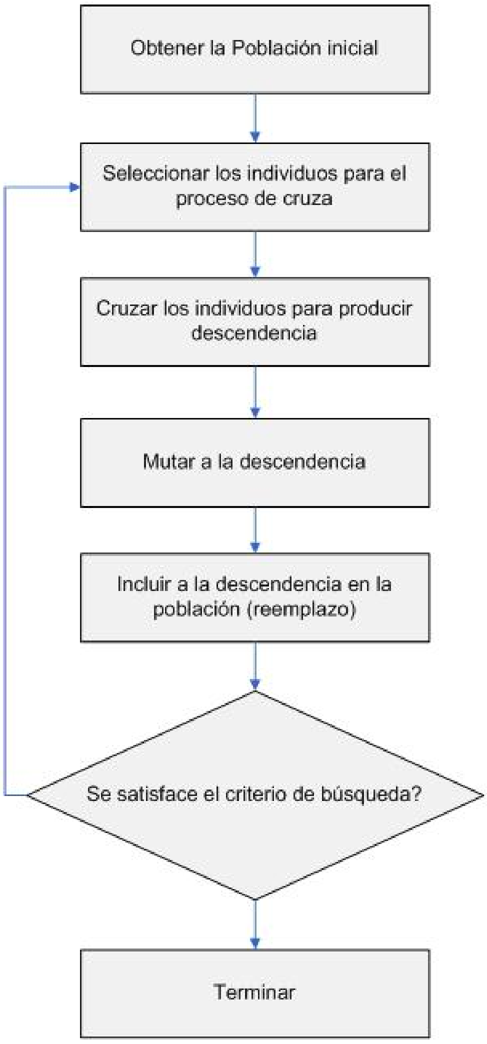

Figura 1.4: *Ejecución general de un Algoritmo Genético*

el porcentaje de cruce es muy bajo la convergencia será muy lenta y se desperdiciará tiempo hasta llegar a la solución, por otro lado si el porcentaje de cruce es muy alto se puede converger en una mala solución de forma temprana. Dependiendo del problema puede estar ente 60 % a 95 %.

Porcentaje de Mutación (Pm): Establece la probabilidad con la cual los individuos serán mutados. Si éste porcentaje es 0 % los individuos generados después de aplicarse el cruce no sufrirán ningún cambio por el contrario si es de 100 % todos lo individuos de la población sufrirán cambios o mutaciones. La mutación trata de impedir que la búsqueda del genético caiga en óptimos locales por eso es conveniente que ocurra ocasionalmente. No es bueno, sin embargo, que la mutación ocurra continuamente, ya que la búsqueda del genético pasa de ser ***inteligente*** a búsqueda aleatoria. El porcentaje de mutación generalmente toma valores cercanos a $\frac{1}{L}$, donde L es la longitud del cromosoma.

Tamaño de la Poblacion: Establece cuántos individuos habrá en cada una de las generaciones. Si el tamaño de la población es muy bajo, el algoritmo genético tiene pocas posibilidades de evolucionar por el cruce y los individuos nuevos se parecerán mucho a sus padres. Tampoco un tamaño excesivo es adecuado porque se llega a un punto en el que los resultados no mejoran. Lo ideal es, en función del problema y la codificación, establecer un límite adecuado del tamaño de la población.

Número de Generaciones: Es el número de iteraciones del proceso, con el paso de las generaciones la población del algoritmo genético evolucionará obteniendo cada vez mejores individuos. Conviene, al igual que con el tamaño de la población, fijar un número de generaciones adecuado para conseguir el resultado deseado.

Tamaño del Individuo: Dependerá del número de elementos que constituyan una solución. Varía con el problema a resolver.

1.7.5. Procesos de un algoritmo genético

Codificación del Problema: Es esencial distinguir en una solución a un problema entre el genotipo y el fenotipo. El genotipo es la representación interna que se utiliza para trabajar con la solución, mientras que el fenotipo es la solución en sí misma. Los operadores del algoritmo trabajarán con el genotipo. Existen algunas formas de representar el genotipo:

> **Representación Binaria:** La representación binaria es la más común. En ella, un cromosoma es una cadena de bits 0 ó 1. Las primeras investigaciones en genéticos utilizaron este tipo de codificación debido a su sencillez. Es una representación indirecta.
>
> **Representación basada en Permutaciones:** Una permutación de los elementos de un determinado conjunto Z induce un orden total en ellos.

Tabla 1.4: *Representación Binaria*

CROMOSOMA A	1011001011001010111100101
CROMOSOMA B	1111111000000110000011111

Esta relación de orden total puede ser representada mediante una matriz de 1 y 0 llamada matriz de precedencia. El elemento de la matriz a_{ij} colocado en la fila i y la columna j es 1 (Verdadero) si y solo si el símbolo etiquetado como i antecede al símbolo etiquetado como j en la secuencia. Esto da lugar a una primera forma de manipular una permutación, como una colección no ordenada de relaciones de precedencia. En ciertas ocasiones esta representación es demasiado exhaustiva y conviene relajarla un poco utilizando el concepto de adyacencia. En esta representación una permutación es un vector de $\mid Z \mid$ (cardinal del conjunto que estamos considerando) pares, cada uno de ellos indicando cual es el inmediato predecesor de cada uno de ellos. La forma mas natural para representar una permutación es la basada en la posición, es decir una lista ordenada de los elementos del conjunto Z. También se puede considerar una representación basada en bloques, siendo un bloque un subconjunto de elementos contiguos. De todas estas posibles representaciones se elegirá la basada en la posición por considerarse mas sencilla de usar y suficientemente eficaz para el problema a resolver. Es una representación indirecta.

Tabla 1.5: *Representación basada en Permutaciones*

CROMOSOMA A	1 5 3 2 6 4 7 9 8
CROMOSOMA B	8 5 6 7 2 3 1 4 9

Representación Directa: En este tipo de representación se usa directamente en problemas que utilizan valores difíciles de representar. Por ejemplo si se usan número reales. Se trata de problemas en los que una codificación binaria sería demasiado complicada. Cada cromosoma es una secuencia de valores. Los valores son cualquier cosa relacionada con el tipo problema. (Números reales, caracteres u otros tipos de objetos).

Tabla 1.6: *Representación Directa*

CROMOSOMA A	1.2324 5.3243 0.4556 2.3293 2.4545
CROMOSOMA B	ABDJEIFJDHDIERJFDLDFLFEGT
CROMOSOMA C	(atrás), (atrás), (derecha), (hacia delante), (izquierda)

Selección: Es el proceso de escoger dos padres de la población para cruzarse. Luego de que se ha escogido una codificación adecuada para el problema es vital decidir la forma de escoger a los individuos que crearan descendencia para la siguiente generación. De acuerdo con la teoría de la evolución de Darwin, sólo los mejores individuos se cruzan, es decir el propósito de la selección es enfocarse en los mejores individuos con la esperanza que su descendencia tenga una mejor adaptación. Basándose en esto existen varios métodos que son utilizados por los algoritmos genéticos sin embargo pueden clasificarse en dos tipos la selección proporcional y la selección basada en orden. En la selección proporcional se eligen individuos en función de sus valores relativos de ajuste respecto al de otros individuos en la población. Por otro lado en la selección basada en orden se toma en cuenta un ranking para los individuos. La selección debe ser equilibrada con el cruce y la mutación debido a que una selección muy rigurosa puede hacer que se pierda diversidad de los individuos. Los métodos más conocidos de selección son:

> **Selección por la Regla de la Ruleta:** Los padres se seleccionan de acuerdo a su adaptación. Los mejores individuos son los que tienen mayores posibilidades de ser elegidos. Intuitivamente el proceso construye una ruleta en la que cada uno de las porciones representa a un individuo. El tamaño de cada porción es proporcional a la adaptación del individuo, así los individuos buenos se llevarán las mayores porciones y al revés ocurrirá con los peores. A continuación se describe el proceso de selección por ruleta:

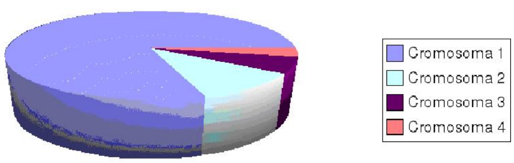

Figura 1.5: *Representación de la selección por ruleta*

Ahora, se gira la ruleta N veces, donde N es el número de individuos de la población. Existe un algoritmo para realizar este proceso:

- Calcular la suma total acumulada de la función de evaluación de todos los individuos de la población actual.
- Generar un número aleatorio entre 0 y la suma total.
- Recorrer la población sumando los valores de la función de ajuste hasta que la suma sea mayor o igual a r(probabilidad elegida), en éste punto se selecciona el individuo que aportó al final para que se supere o iguale el valor de r.

Este tipo de selección depende de la variación entre los valores de la función de los individuos, cuando son muy diferentes no funciona bien. Por ejemplo si un cromosoma ocupa el 90 % de la ruleta el resto de los cromosomas tienen muy pocas posibilidades de ser elegidos.

Selección por Ranking: Los individuos son ordenados de acuerdo a su valor de la función objetivo en un ranking, de esta manera si se tiene N cromosomas el individuo con peor ajuste se le asignará un 1 y el que tenga el mejor ajuste se le asignará la posición N. Con éste método se tiene una convergencia más lenta pero previene de quedar atrapado en un óptimo local. En la figura 1.6 y la figura 1.7 se indica cómo cambia la situación antes y después del ranking.

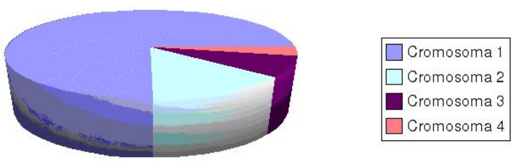

Figura 1.6: *Situación antes del ranking (ruleta)*

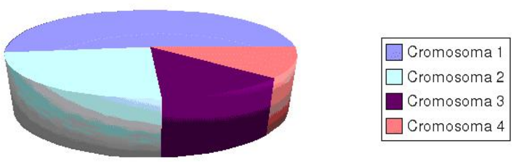

Figura 1.7: *Situación depués del ranking (ruleta)*

Ahora todos los cromosomas tienen la oportunidad de ser seleccionados. Sin embargo este método puede hacer que el genético converja lentamente a la solución, ya que los mejores individuos no presentan gran diferencia de los peores. A este esquema se le pueden aplicar otros criterios; un ejemplo es el muestreo estocástico universal, que trata de evitar que los individuos con más adaptación copen la población. En lugar de dar la vuelta a una ruleta con una ranura, se da la vuelta a la ruleta con N ranuras, tantas como la población. De esta forma, la distribución estadística de descendientes en la nueva población es más parecida a la real.

Selección por Torneo K/L: La selección por Torneo KŁconsiste en seleccionar K individuos de la población aleatoriamente y de estos K individuos se seleccionan los L que tengan mejor adaptación. Este proceso se repite todas las veces necesarias hasta formar la nueva población. Este es uno de los métodos de selección más utilizados actualmente. Se utiliza también en algunos algoritmos en el momento de la aceptación.

Elitismo: Este concepto expresa la idea de que el mejor individuo de la actual generación pase sin ser modificado a la siguiente generación. De esta forma no se perderá el mejor cromosoma. Al resto de la población se le aplica la reproducción normalmente. Por otra parte existen algoritmos genéticos llamados elitistas debido a que convergen muy rápidamente a la solución.

Cruce: El cruce trata de crear una generación de individuos nuevos (descendencia) pidiendo información a sus padres, por lo que luego del cruce la población se enriquece con mejores individuos. Inicialmente puede parecer que el operador de cruce se limita a obtener información básica pero realmente se fundamente en otros factores, de los cuales depende el tipo de operador.

Cruce en un Punto: Usado en los algoritmos genéticos tradicionales. Se copian los genes del primer padre hasta el punto de corte y se rellena con el resto de elementos que hagan la solución válida en el orden en que aparecen en el segundo padre considerando la cadena de genes como cíclica. En el caso de que se haya utilizado una codificación binaria simplemente se copian el resto de genes del segundo padre. En la figura 1.8 se observa un ejemplo del cruce en un punto.

$$11001011{+}11011111 = 11001111$$

Figura 1.8: *Cruce en un punto*

Cruce en dos Puntos: El añadir puntos de cruce reduce el desempeño del algoritmo genético debido a que se tiende a romper la cadena cíclica de los padres. Sin embargo la ventaja de adicionar puntos es que se explora el espacio del problema de forma más efectiva. Para realizar el cruce en dos puntos se copian los genes del primer padre comprendidos entre los dos puntos de cruce y se rellenan los que faltan con los del segundo padre considerando la cadena de genes como cíclica. En las figuras 1.9 y 1.10 se observa un ejemplo del cruce en dos puntos.

$$11001011 + 11011111 = 11011111$$

Figura 1.9: *Cruce en dos puntos*

$$11001011 + 11011111 = 11011111$$

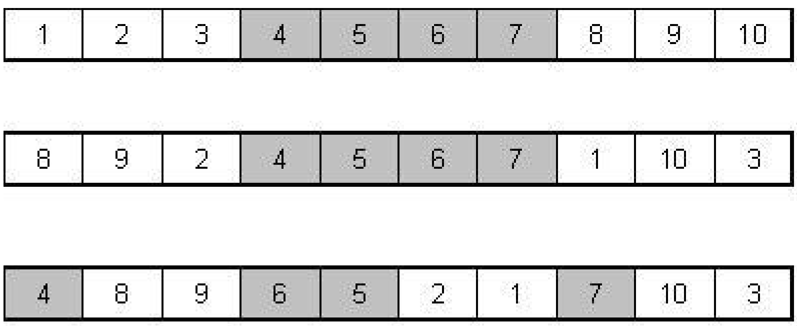

Figura 1.10: *Cruce en dos puntos*

Cruce uniforme: Cada gene de la descendencia se crea mediante la copia del gen correspondiente de uno de los padres. Se escoge aleatoriamente si el gen i-ésimo del hijo se toma del primer o del segundo padre. En la figura 1.11 se observa un ejemplo del cruce uniforme.

$$11001011 + 11011101 = 11011111$$

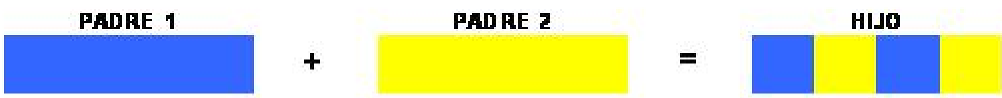

Figura 1.11: *Cruce Uniforme*

Cruce aritmético: Se realizan operaciones aritméticas con los genes de los padres que dan como resultado la codificación genética del hijo. En la figura 1.12 se observa un ejemplo del cruce aritmético.

$$11001011 + 11011111 = 11001001$$

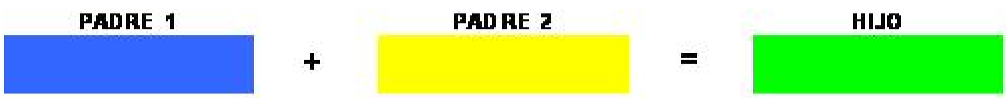

Figura 1.12: *Cruce Aritmético*

Cruce de tres padres: Se escogen aleatoriamente tres padres. Una parte del primer padre se compara con la misma parte del segundo padre. Si son iguales se elige esta parte para la descendencia, de lo contrario se toma la parte del tercer padre. En el cuadro 1.7 se observa un ejemplo del cruce de tres padres.

Tabla 1.7: *Cruce de tres padres*

Padre 1	1	1	0	1	0	0	0	1
Padre 2	0	1	1	0	1	0	0	1
Padre 3	0	1	1	0	1	1	0	0
Hijo	0	1	1	0	1	0	0	1

Cruce de combinación parcial (PMX o Partial Mapped Cross: Este método de cruce fue propuesto por Goldberg y se aplicó inicialmente al problema del agente viajero En peste problema los cromosomas son secuencias de enteros; donde cada entero representa una ciudad diferente. El método propuesto garantiza que todas las ciudades estén presentes solamente una vez en la descendencia, el procedimiento se tiene a continuación:

- Se elige aleatoriamente dos puntos de cruza.
- Se intercambia estos dos segmentos en los hijos que se generan (igual a la cruza de dos puntos).
- Del resto de las cadenas que conforman los hijos se obtienen haciendo mapeo entre los dos padres:
  - Si un valor no está contenido en el segmento intercambiado permanece igual.
  - Si está contenido en el segmento intercambiado, entonces se sustituye por el valor que tenga dicho segmento en el otro padre. En siguientes figuras 1.13, refpmx1, refpmx2 y refpmx3 se observa un ejemplo.

9	8	4	5	6	7	1	3	2	10
8	7	1	2	3	10	9	5	4	6

Figura 1.13: *PMX: Los padres son*

X	X	X	2	3	10	X	X	X	X
X	X	X	5	6	7	X	X	X	X

Figura 1.14: *PMX: Los hijos son*

Para completar a los dos hijos se copia los valores que no están en el segmento intercambiado. Otro ejemplo es el siguiente: se tiene dos padres P1

9	8	4	2	3	10	1	X	X	X
8	X	1	5	6	7	9	X	4	X

Figura 1.15: *PMX: Completar los hijos*

= (1,2,4,6,3,7,5,8) y P2 = (5,4,1,7,2,6,8,3) y el segmento seleccionado al azar de P1 para ser insertado en P2 es el (4,6,3) esto establece una relación

9	8	4	2	3	10	1	6	5	7
8	10	1	5	6	7	9	2	4	3

Figura 1.16: *PMX: Se mapean los valores restantes*

con el segmento (1,7,2) que ocupa las mismas posiciones en P2. Entonces la secuencia de operaciones transformarían a P2 en (5,4,4,6,3,6,8,3) y luego eliminando las repeticiones quedaría (5,*,4,6,3,*,8,*) donde los asteriscos corresponden a los elementos repetidos que deben eliminarse de la configuración. Reemplazando queda (5,1,4,6,3,7,8,2), el número 5 ocupa la posición del número 1 por ésta razón el primer asterisco es llenado con el número 1, el número 6 ocupa la posición del número 7 y el último asterisco es llenado con el número 2. En la figura refpmx4 se muestra graficamente explicado.

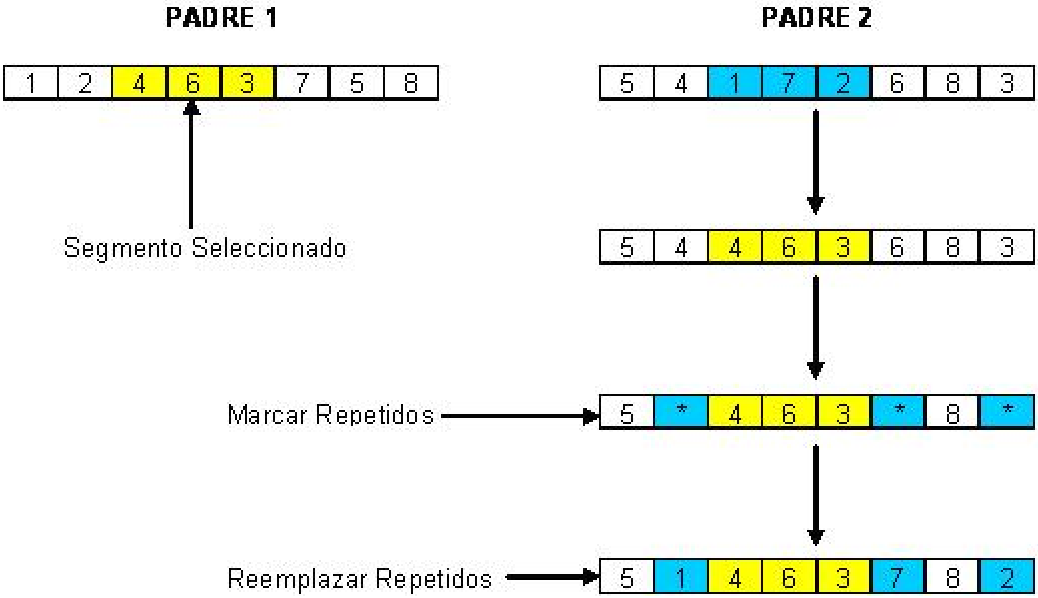

Figura 1.17: *PMX: Ejemplo adicional*

Cruce JOX (Job Order Cross: el operador de cruce jox permite trabajar sobre las representaciones de problemas que usan la codificación de permutaciones. El operador de cruce genera dos hijos, para ello es necesario seleccionar de forma aleatoria una operación de entre todas las existentes. La operación o gen seleccionado mantendrá su posición en los dos cromosomas hijos; y, los genes diferentes al seleccionado se intercambian entre los dos cromosomas padres para generar dos nuevos hijos.

Job
3

Pos	0	1	2	3	4	5	6	7	8
P1	3	1	2	1	2	3	2	3	1
P2	2	1	3	3	2	1	1	3	2

H1	3	2	1	2	1	3	1	3	2
H2	1	2	3	3	1	2	2	3	1

Figura 1.18: *JOX*

Mutación: La mutación ocurre luego del cruce y tiene la finalidad de prevenir que el algoritmo se atrape en un óptimo local. La mutación se encarga de mantener la diversidad genética en la población ya que introduce nuevas estructuras genéticas en la población mediante la modificación aleatoria de ciertos bloques. Existen diversas formas de mutación, las cuales dependen de la codificación elegida, las mismas que se describen a continuación:

Inversión de genes: Se seleccionan genes aleatoriamente y se invierte su valor. Se utiliza en representaciones de bits, cambiando 0 por 1 o viceversa.

$$11001001 \rightarrow 10001001$$

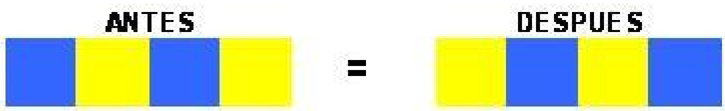

Figura 1.19: *Inversión de Genes*

Cambio de orden: Se seleccionan dos genes aleatoriamente y se intercambian sus posiciones. Se utiliza en representaciones basadas en permutaciones.

$$(123456897) \rightarrow (183456297)$$

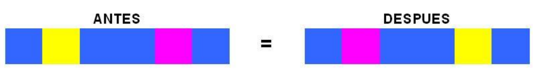

Figura 1.20: *Cambio de Orden*

Modificación de Genes: Se realizan pequeñas modificaciones en los genes. Por ejemplo en una codificación basada en números reales se realizan sumas de números muy pequeños positivos o negativos.

$$(1{,}295{,}682{,}864{,}115{,}55) \rightarrow (1{,}295{,}682{,}734{,}225{,}55)$$

Aceptación: Después de realizar el cruce y la mutación de los individuos de la población se tiene que dos padres se cruzaron y se obtuvieron dos hijos, pero no todos pueden regresar a la población original. El operador de aceptación se encarga de decidir los miembros actuales, de llegar a haber alguno, que serán reemplazados por las nuevas soluciones. Existen algunos criterios de aceptación:

Aceptación total: Es la manera más utilizada, todos los hijos generados son aceptados reemplazando a los padres y pasan a formar parte de la nueva población.

De mejora: Los hijos pasan a la nueva población si son mejores que los peores individuos de la población actual.

Estado constante (Steady State): Se cambia un miembro de la población a la vez. Para llevar a cabo la selección se escoge un miembro de la población de acuerdo a su función de aptitud. Se copia éste individuo y se procede a mutar la copia. Se selecciona un segundo miembro de la población el cual es reemplazado por la cadena mutada.

Por torneo: De los cuatro individuos, dos hijos y dos padres, solo los dos más aptos, sean hijos o padres regresan a la población.

1.8. Programación en C++

C++ es un lenguaje de programación muy potente que fue creado por Bjarne Stroustrup en la década de los 80. Inicialmente fue una extensión del lenguaje C que se llamó ***C con clases***. Inmediatamente adquirió popularidad ya que potenció la flexibilidad y eficacia que tenia el lenguaje C y se lo llamó así para referenciar el carácter del operador de incremento de C (++). Actualmente el C++ es un lenguaje versátil, potente y general por lo que permite programar sistemas operativos, compiladores, aplicaciones de bases de datos, procesadores de texto, juegos, entre otros.

Una de las ventajas del lenguaje C++ es que posee una gran cantidad de operadores lo cual le permite tener una expresividad notable cuando se implementan aplicaciones científicas o matemáticas; adicionalmente la sintaxis de clases y objetos permite manipular convenientemente diversas estructuras de datos y operaciones; las excepciones permiten procesar de un modo claro los casos de error.

C++ es un lenguaje consistente y sin ambigüedades pero si tiene desventajas como que deja algunos aspectos a criterio del implementador como el tamaño del tipo de datos, por lo que se requiere de conocimientos medianos cuando se empieza a programar en este lenguaje.

Para facilitar la programación y tener mayor facilidad para programar aplicaciones en Internet, se han creado programas como JAVA, PHP y .NET. Sin embargo, la potencia del Lenguaje C++ hace que hoy en día es uno de los lenguajes de programación más usados en el mundo.

CAPITULO 2

DESCRIPCION Y FORMULACION DEL PROBLEMA

En la presente Tesis de Grado inicialmente se estudia el proceso de planificación actual para luego proponer la optimización pertinente.

2.1. Método de Planificación Actual

La Empresa ABC usa las siguientes fases en el método de planificación actualmente usado:

- Generación de un Plan Maestro de Producción (MPS) a partir del presupuesto comercial, este plan indica la cantidad de cada producto que debe ser fabricada semanalmente con un horizonte anual. Esta parte es realizada por el ERP que existe en la Empresa ABC y requiere de una afinación final.

- Semanalmente se realiza la planificación finita con un horizonte de un mes y un período congelado de dos semanas. Para ésta planificación finita se inicia con el MPS generado anteriormente y se distribuye los productos a generarse diariamente en función de las restricciones de ingreso de materia prima. Una vez que se ha generado el plan diario el jefe de planificación, jefe y gerente de producción generan manualmente la secuencia de producción y asignación de mano de obra requerida para las tareas en función de la experiencia sin tomar en cuenta ninguna técnica de optimización formal. Toda ésta labor toma aproximadamente dos días semanalmente lo cual es un tiempo alto y finalmente no se obtiene un plan óptimo de producción.

2.2. Situación real de la Empresa ABC

Como se indicó anteriormente la tarea de planificación finita en la Empresa ABC toma 2 días de tiempo del gerente de producción y jefes de planificación y producción. Adicionalmente no se tiene una secuencia óptima de producción, para conocer más a fondo la situación actual se obtiene la siguiente información.

2.2.1. Tiempo de Para en Celdas de Producción

En cada celda de producción se tiene información del tiempo diario improductivo y las razones para estos tiempos de paro. Se presenta a continuación la información del tiempo de paro no programado con las causas de los últimos 3 meses.

Tabla 2.1: *Tiempo de para promedio diario*

Causa	**Tiempo (min)**
Secuenciación de la producción	121
Daño de máquina	44.4
Falta de personal	31.6
Otros	20.3
Total	**217.3**

Cabe indicar que para obtener esta información, se entregó a los operadores líderes de cada célula un formato para registrar los tiempos y motivos de para. Posteriormente, el supervisor de producción tabulaba los datos a diario y atribuía las causas reales. Se entiende como para de producción por mal secuenciamiento, a toda para de la célula de producción debida a que la operación anterior no estaba completa. Se puede observar en la Tabla 4.6 que en la Empresa ABC se tienen serios problemas de productividad pues se trabajan diariamente dos turnos de 9 horas aproximadamente. Es decir 1080 minutos al día de los cuales las máquinas paran en promedio 217,3 minutos lo que equivale aproximadamente al 20,1 %. La Empresa ABC, por estos problemas de productividad incurre en costos innecesarios los que se detallan a continuación:

- Pago de horas extras innecesarias al personal de producción, mantenimiento, calidad y bodega.
- Pago de recorridos no programados para el personal.
- Pago de servicios complementarios para producción como electricidad, vapor, agua y aire.

Estos gastos incrementan los costos de producción y le quitan una ventaja competitiva a la Empresa ABC. En la Figura 2.1, se presenta un diagrama de Pareto de los motivos de para con la finalidad de conocer la causa raíz de la pérdida de productividad.

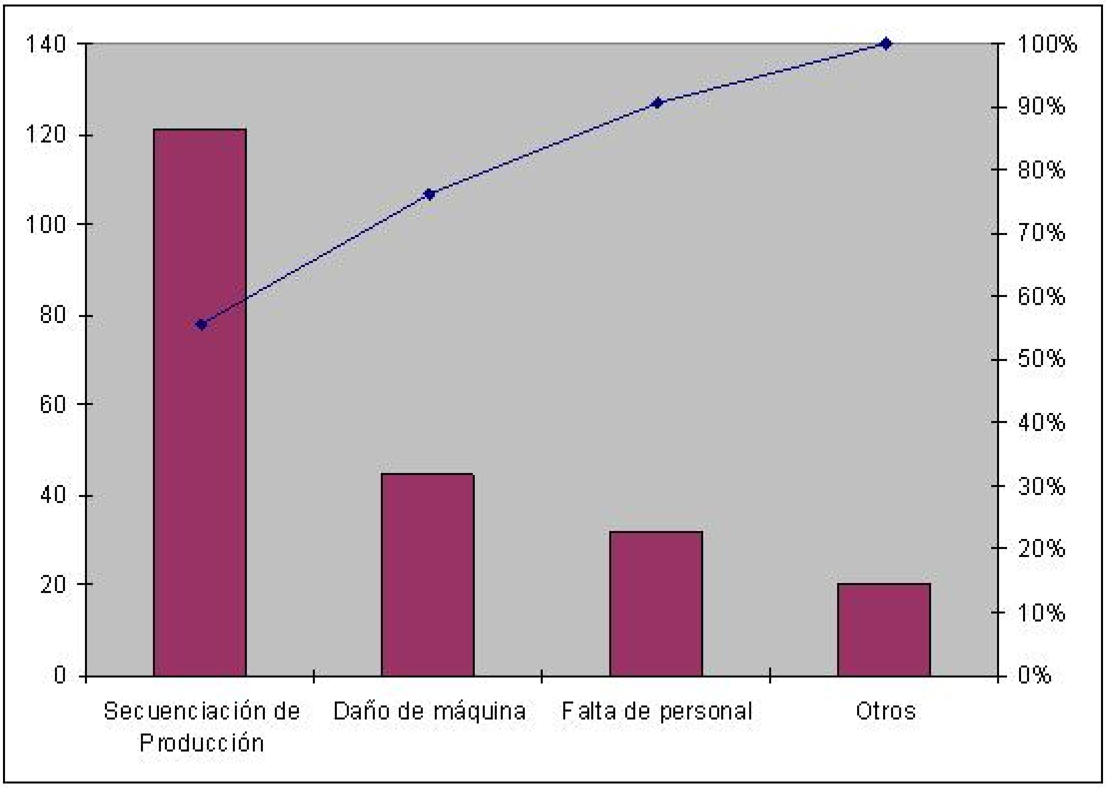

Figura 2.1: *Diagrama de Pareto para los problemas de Productividad*

Gracias a la información recopilada para el presente proyecto de titulación también se logra evidenciar que la Empresa ABC aparte de mejorar la planificación finita de producción debe enfocarse adicionalmente en solucionar los frecuentes daños de máquinas. Para esto se está trabajando en un Plan de mantenimiento Preventivo para lo cual se estableció rutas de inspección mecánica, tableros de alertas y se compró equipo para realizar Mantenimiento Predictivo como: Medidor de vibración, Cámara termográfica y estraboscopio.

En la tabla 4.6 se presenta la información detallada de para en cada célula de producción debido a un mal secuenciamiento de las operaciones.

Tabla 2.2: *Tiempo de para promedio diario*

Celda	Tiempo Promedio (min)	Causa
Pelado MP	7.2	Secuenciación de la Producción
Picado MP	11,1	Secuenciación de la Producción
Lavado MP	8.1	Secuenciación de la Producción
Procesar MP	35.7	Secuenciación de la Producción
Lavado Envases	0.8	Secuenciación de la Producción
Preparación LG	18.7	Secuenciación de la Producción
Envasado	6.2	Secuenciación de la Producción
Cerrado	5.7	Secuenciación de la Producción
Pasteurizado	23.1	Secuenciación de la Producción
Embalado	4.4	Secuenciación de la Producción

2.2.2. Horas extras pagadas no planificadas

Debido a que los trabajos ya programados por día se cumplen al 100 % el indicador que refleja la pérdida de productividad por paras no programadas son las horas extras no planificadas que se pagan diariamente. Para calcular este indicador se procede de la siguiente forma:

- Con el plan diario de producción (artículos a fabricar y cantidades) se determina la cantidad de horas que se debe trabajar cada celda de producción, para éste cálculo se usan las rutas de producción las cuales proporcionan información del número de personas y tiempo necesario por cada una de las operaciones.

- Diariamente, mediante el reloj dactilar de la Empresa ABC, se registra el número de horas que cada persona trabaja, éste tiempo se almacena en el sistema informático, el cual reconoce a que célula de producción pertenece dicho empleado. Finalmente con ésta información se obtiene la diferencia entre las horas planificadas y las no programadas determinando así el tiempo extra que fue requerido para cumplir la producción.

En la Tabla 2.4 se presenta el resumen año de las horas extras no planificadas por mes desde noviembre de 2008 pues ésta información esta disponible en la Empresa

ABC desde antes de la propuesta del presente proyecto.

Tabla 2.3: *Horas extra no planificadas por mes*

Mes	Horas extras 50 %	Horas extras 100 %	Total
Noviembre	3729	2128	5857
Diciembre	3508	2977	6485
Enero	2753	1920	4673
Febrero	2411	1754	4165
Marzo	2540	2022	4562
Abril	2016	1987	4003
Mayo	2870	1520	4390
Junio	2283	1948	4231
Julio	2229	2187	4416
Agosto	3498	1312	4810
Septiembre	3294	1829	5123
TOTAL			**52715**

Si se considera que en promedio una hora extra al 50 % cuesta \$1,76 USD y una hora extra al 100 % cuesta \$3,75 USD el valor total promedio por mes que la Empresa ABC está pagando debido a ineficiencias es \$12.340,00 USD. Es decir, al año se estima que paga \$150.000,00 USD, por lo que se observa una gran oportunidad de mejora. Cabe indicar que éste valor toma en cuenta únicamente la mano de obra directa, no se ha incluido los otros valores indicados anteriormente, por lo que se concluye que la pérdida es mucho mayor pero imposible de calcular con la información que posee actualmente la Empresa ABC.

2.3. Notación Matemática del Problema

El problema a resolver en la Empresa ABC se describe y formula a continuación:

Datos de entrada:

- Se tiene un conjunto de 10 máquinas que representan células de producción de los procesos existentes en la línea de interés.
- Se tiene un conjunto de artículos (tareas) con cantidades determinadas a producir en esa línea en el día de interés. Cada artículo tiene rutas establecidas con operaciones determinadas que dependen del producto a fabricar.

Objetivo:

- Minimizar el tiempo total de proceso de una secuencia factible. Donde el tiempo total del proceso es el máximo tiempo necesario para fabricar todos los artículos.
- $Min_{cmax} = Max_{j=1}^{N}(C_j)$. En donde C_j es el tiempo total requerido para completar la tarea j y se calcula $C_j = Sn_j, j + dn_j, j$

Restricciones:

- $O_{hj} - O_{ij} \geq d_{ij}$ para todo $(i,j) \rightarrow (h.j)\varepsilon A$
- $C_{max} - O_{ij} \geq d_{ij}$ para todo $(i,j)\varepsilon N$
- $O_{ij} - O_{ik} \geq d_{ik}$ ó $O_{ik} - O_{ij} \geq d_{ij}$ para todo $(i,k)y(i,j), i = 1, \ldots, m$
- $O_{ij} \geq 0$ para todo $(i,j)\varepsilon N$

La primera restricción asegura que la operación *(h,j)* no puede iniciar antes que se complete la operación *(i,j)*. La tercera restricción se conoce como la restricción disjunta, asegura que se respete el orden de diferentes tareas que se procesan en una misma máquina.

2.4. Estructura del algoritmo de solución propuesto

El algoritmo ha sido desarrollado en C++ usando como base el framework de algoritmos genéticos GALIB desarrollado por Matthew Wall en el Instituto Tecnológico de Massachussets (MIT) [1].

2.4.1. Codificación de los Genes

Para el caso específico del problema de job shop, existen dos diferentes técnicas para representar el problema; la primera es una representación indirecta la cual codifica las instrucciones a través de un constructor de secuencia; por ejemplo, la permutación del orden de trabajos y la priorización de las reglas de secuenciamiento, en estos casos el constructor de secuencia asegura la validez del programa producido. La segunda técnica es la representación directa la cual codifica la secuencia propiamente dicha, como ejemplo de ésta técnica se puede citar la codificación de los tiempos de finalización de cada operación o los tiempos de inicio de cada operación, en ésta técnica no toda la codificación representa una secuencia válida.

La codificación usada para resolver el problema planteado en el presente proyecto de titulación es la ***permutación ordenada de trabajos***. A continuación se explica ésta representación mediante un ejemplo didáctico:

En un problema de 3 tareas con 4 máquinas se tiene:

Tareas = {J1 , J2 , J3}

J1 = {t11(R1), t12(R2), t13(R3)}

J2 = {t21(R2), t22(R4), t23(R3)}

J3 = {t31(R1), t32(R3)}

Recursos = {R1, R2, R3, R4}

Du1j = Du2j = 2, Du3j = 3

Tmin = 0, Tmax = 10

En la Figura 2.2 se muestra un grafo disjunto que representa gráficamente el problema de ejemplo.

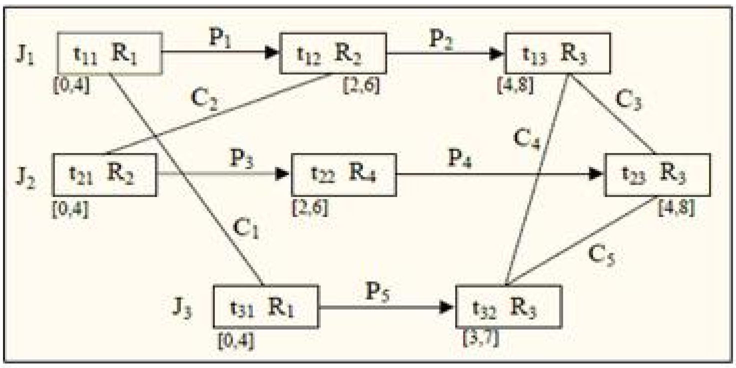

Figura 2.2: *Grafo disjunto que representa el problema del Job Shop*

En la Tabla 2.4 se presenta una solución factible al problema.

Tabla 2.4: *Una solución al problema planteado*

Recursos	0	1	2	3	4	5	6	7	8	9	10
R1	t_{31}			t_{11}							
R2	t_{21}					t_{12}					
R3				t_{32}			t_{23}		t_{13}		
R4			t_{22}								

La representación para un algoritmo genético sería:

Permutación ***Ordenada*** de las operaciones:

{t21, t31, t22, t11, t32, t12, t13, t23}

Reducción al número de tareas:

{2, 3, 2, 1, 3, 1, 1, 2}

2.4.2. Número de Generaciones

Las generaciones permiten al algoritmo genético evolucionar obteniendo cada vez mejores cromosomas, por lo tanto es importante fijar un número adecuado de generaciones para conseguir un buen resultado. De la bibliografía revisada se obtiene que un número adecuado para éste tamaño de problema es 5000, de ésta forma nos aseguramos que el problema converja a la solución óptima.

2.4.3. Tamaño de la Población

El tamaño de la población define el número de cromosomas que existirán en cada generación. Si no existen suficientes cromosomas, el algoritmo no evoluciona. De igual manera, si existen muchos cromosomas, se llega a un punto en el que los resultados no mejoran. Este tamaño es directamente proporcional al número de operaciones. Debido a que en el problema de la Empresa ABC el número de operaciones es 10 se usó la relación tamaño de la población es igual al cuadrado del número de operaciones, por lo tanto el tamaño de la población será 100.

2.4.4. Operador de Mutación

Para el algoritmo propuesto en el presente proyecto de titulación se usan dos operadores de mutación.

- Operador de mutación simple.
- Operador de mutación combinada.

La mutación simple consiste en seleccionar aleatoriamente un número entre 0 (cero) y el número total de genes del cromosoma el cual identifica la posición de mutación. En la Figura 2.3 se intercambia los valores de los genes de la posición seleccionada con la posición inmediatamente siguiente.

La mutación combinada consiste en:

Pos
3

Pos	0	1	2	3	4	5	6	7	8
P	3	1	2	1	2	3	2	3	1
H	3	1	2	2	1	3	2	3	1

Figura 2.3: *Mutación Simple*

- Seleccionar al azar un número y verificar si el número es menor a la probabilidad de mutación (%). Si es menor, continuar con el siguiente paso, caso contrario no realizar la mutación.

- Seleccionar al azar un número. Si es menor al 50 %, realizar la mutación simple, para ello se seleccionan al azar dos posiciones del cromosoma y se intercambian sus valores, sino realizar una mutación de desplazamiento con inversión de genes o sin ella, para esto se generan otros números al azar. Para el presente estudio el algoritmo de éste método de mutación se presenta a continuación:

if $nrandom \leq probMutacion$ **then**
  if $nrandom \leq 50\,\%$ **then**
    $intercambio(posicionrandom1, posicionrandom2)$
  else
    $nodos \leftarrow Random$
    $SeleccionanNodos(posicionrandom)$
    $invertir \leftarrow Random(\leq 50\,\%, 0, 1)$
    $InsertarnNodos(posicionrandom, invertir)$
  end if
end if

2.4.5. Probabilidad de Mutación

La probabilidad de mutación es un parámetro dado en el algoritmo que se encontrará mediante una prueba experimental. Para fijar el rango inicial de la probabilidad de mutación se tendrá en cuenta el siguiente criterio:

- La probabilidad de mutación trata de impedir que la búsqueda del algoritmo genético caiga en óptimos locales por eso es conveniente que ocurra ocasionalmente. No es bueno sin embargo, que la mutación ocurra continuamente, ya que la búsqueda del genético pasa de ser ***inteligente*** a búsqueda aleatoria. Para el presente estudio se comenzará con un valor de 1/L, donde L es la longitud del cromosoma.

Se procede a calibrar la probabilidad de mutación para el algoritmo propuesto, es decir, encontrar el valor de dicho parámetro con el cual se obtiene un menor makespan. Para el efecto se realizan 90 experimentos con el problema estándar FT10, donde se mantiene constante la probabilidad de reemplazo, la población, el número de generaciones, la probabilidad de cruce y la semilla para la generación de los números aleatorios. Los resultados se indican en la Tabla 2.5.

Tabla 2.5: *Tabla de calibración de probabilidad de mutación*

Prueba	Reemplz	Población	Generación	Mutación	Cruce	Seed	PMXOVER & COMBmutator MAKESPAM
1	0,86	100	5000	0,01	0,60	99	1098
2	0,86	100	5000	0,011	0,60	99	1109
3	0,86	100	5000	0,012	0,60	99	1109
4	0,86	100	5000	0,013	0,60	99	1061
5	0,86	100	5000	0,014	0,60	99	1068
6	0,86	100	5000	0,015	0,60	99	1064
7	0,86	100	5000	0,016	0,60	99	1143
8	0,86	100	5000	0,017	0,60	99	1157
9	0,86	100	5000	0,018	0,60	99	1093
10	0,86	100	5000	0,019	0,60	99	1057
11	0,86	100	5000	0,02	0,60	99	1103
12	0,86	100	5000	0,021	0,60	99	1083
13	0,86	100	5000	0,022	0,60	99	1017
14	0,86	100	5000	0,023	0,60	99	1070
15	0,86	100	5000	0,024	0,60	99	1087
16	0,86	100	5000	0,025	0,60	99	1031
17	0,86	100	5000	0,026	0,60	99	1038
18	0,86	100	5000	0,027	0,60	99	1047
19	0,86	100	5000	0,028	0,60	99	1054
20	0,86	100	5000	0,029	0,60	99	1054
21	0,86	100	5000	0,03	0,60	99	1114
22	0,86	100	5000	0,031	0,60	99	1051
23	0,86	100	5000	0,032	0,60	99	1015
24	0,86	100	5000	0,033	0,60	99	1037
25	0,86	100	5000	0,034	0,60	99	1080
26	0,86	100	5000	0,035	0,60	99	1100
27	0,86	100	5000	0,036	0,60	99	1039
28	0,86	100	5000	0,037	0,60	99	1015
29	0,86	100	5000	0,038	0,60	99	1000
30	0,86	100	5000	0,039	0,60	99	1047
31	0,86	100	5000	0,04	0,60	99	1109
32	0,86	100	5000	0,041	0,60	99	986
33	0,86	100	5000	0,042	0,60	99	1024
34	0,86	100	5000	0,043	0,60	99	1106
35	0,86	100	5000	0,044	0,60	99	1042
36	0,86	100	5000	0,045	0,60	99	1038
37	0,86	100	5000	0,046	0,60	99	987
38	0,86	100	5000	0,047	0,60	99	1159
39	0,86	100	5000	0,048	0,60	99	1030
40	0,86	100	5000	0,049	0,60	99	1032
41	0,86	100	5000	0,05	0,60	99	1073
42	0,86	100	5000	0,051	0,60	99	1034
43	0,86	100	5000	0,052	0,60	99	1109
44	0,86	100	5000	0,053	0,60	99	1082
45	0,86	100	5000	0,054	0,60	99	1022
46	0,86	100	5000	0,055	0,60	99	1042
47	0,86	100	5000	0,056	0,60	99	1012
48	0,86	100	5000	0,057	0,60	99	1046
49	0,86	100	5000	0,058	0,60	99	1064
50	0,86	100	5000	0,059	0,60	99	1068
51	0,86	100	5000	0,06	0,60	99	1089
52	0,86	100	5000	0,061	0,60	99	1033
53	0,86	100	5000	0,062	0,60	99	1035
54	0,86	100	5000	0,063	0,60	99	1112
55	0,86	100	5000	0,064	0,60	99	1045
56	0,86	100	5000	0,065	0,60	99	1059
57	0,86	100	5000	0,066	0,60	99	1020
58	0,86	100	5000	0,067	0,60	99	1133
59	0,86	100	5000	0,068	0,60	99	986

60	0,86	100	5000	0,069	0,60	99	1046
61	0,86	100	5000	0,07	0,60	99	1016
62	0,86	100	5000	0,071	0,60	99	1015
63	0,86	100	5000	0,072	0,60	99	1088
64	0,86	100	5000	0,073	0,60	99	998
65	0,86	100	5000	0,074	0,60	99	1042
66	0,86	100	5000	0,075	0,60	99	1109
67	0,86	100	5000	0,076	0,60	99	1058
68	0,86	100	5000	0,077	0,60	99	1016
69	0,86	100	5000	0,078	0,60	99	1080
70	0,86	100	5000	0,079	0,60	99	1005
71	0,86	100	5000	0,08	0,60	99	999
72	0,86	100	5000	0,081	0,60	99	985
73	0,86	100	5000	0,082	0,60	99	1034
74	0,86	100	5000	0,083	0,60	99	1155
75	0,86	100	5000	0,084	0,60	99	1017
76	0,86	100	5000	0,085	0,60	99	1085
77	0,86	100	5000	0,086	0,60	99	1041
78	0,86	100	5000	0,087	0,60	99	1015
79	0,86	100	5000	0,088	0,60	99	1068
80	0,86	100	5000	0,089	0,60	99	1060
81	0,86	100	5000	0,09	0,60	99	1001
82	0,86	100	5000	0,091	0,60	99	946
83	0,86	100	5000	0,092	0,60	99	1070
84	0,86	100	5000	0,093	0,60	99	1100
85	0,86	100	5000	0,094	0,60	99	1040
86	0,86	100	5000	0,095	0,60	99	1029
87	0,86	100	5000	0,096	0,60	99	1103
88	0,86	100	5000	0,097	0,60	99	1084
89	0,86	100	5000	0,098	0,60	99	1084
90	0,86	100	5000	0,099	0,60	99	1102

En la Figura 2.4 se presenta un gráfico del desempeño de las pruebas en función del valor de makespan.

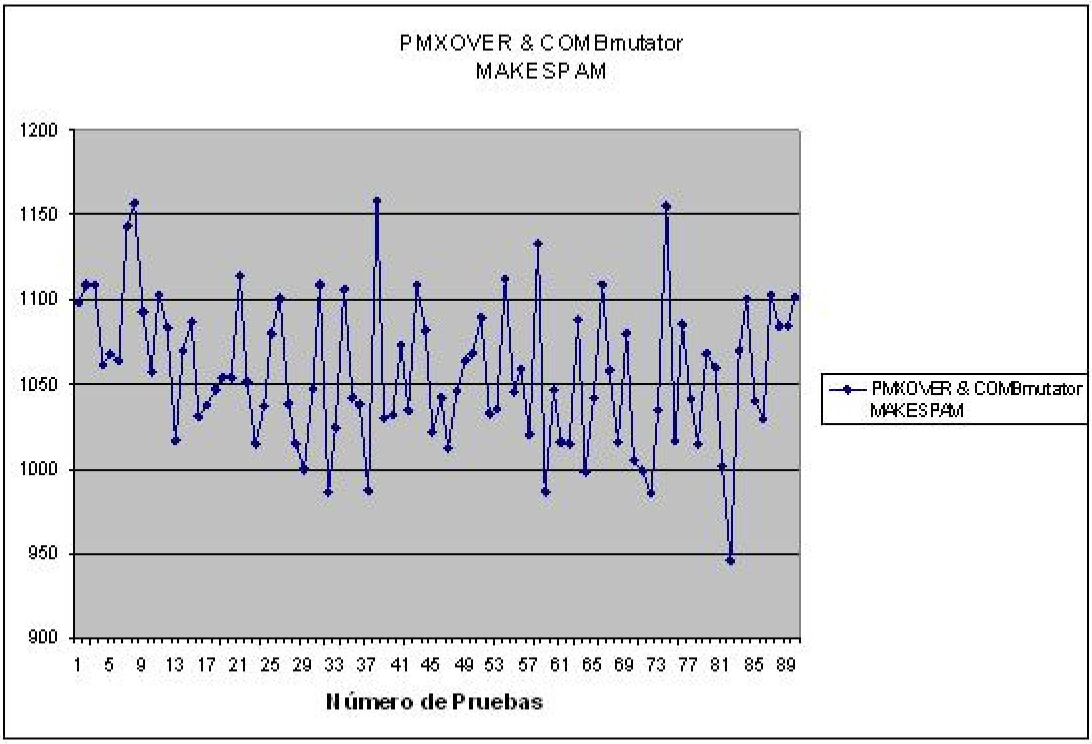

Figura 2.4: *Desempeño de la calibración del porcentaje de mutación*

El mejor valor de makespan manteniendo inalterables los otros parámetros es de 946 para una probabilidad de mutación de 0,091 por lo tanto se trabajará con ésta proabilidad para el resto de calibraciones.

2.4.6. Operador de Cruce

El algoritmo desarrollado tiene la posibilidad de usar dos operadores de cruce: el PMX y JOX. En las pruebas de los problemas estándar que se indican en el capítulo 3 se usarán los dos operadores para conocer el que presente mejor desempeño y aplicarlo en la solución de la Empresa ABC.

2.4.7. Probabilidad de Cruce

La probabilidad de cruce es un parámetro dado en el algoritmo que se encontrará mediante una prueba experimental. Para fijar el rango inicial de la probabilidad de cruce se tendrá en cuenta el siguiente criterio:

- La probabilidad de cruce debe estar entre el 60 % al 95 %, de ésta forma se asegura que el problema no llegue a convergencia de forma lenta y algunos individuos pasen a la siguiente generación sin cruzarse.

Para encontrar el valor de éste parámetro se usa el problema estándar FT10 y se usa el operador PMX. Se realizaron 36 pruebas variando la probabilidad de cruce en 1 % desde el 60 % al 95 % obteniendo para cada una el makespan.

El mejor valor de éste parámetros es 0,6 con un makespan de 946 (2 % de error) como se detalla en la Tabla 2.6:

Tabla 2.6: *Tabla de calibración de probabilidad de cruce*

Prueba	Reemplz	Población	Generación	Mutación	Cruce	Seed	PMXOVER & COMBmutator MAKESPAM
1	0,86	100	5000	0,091	0,95	99	1024
2	0,86	100	5000	0,091	0,94	99	1020
3	0,86	100	5000	0,091	0,93	99	981
4	0,86	100	5000	0,091	0,92	99	1013
5	0,86	100	5000	0,091	0,91	99	1039
6	0,86	100	5000	0,091	0,90	99	1045
7	0,86	100	5000	0,091	0,89	99	1043
8	0,86	100	5000	0,091	0,88	99	1029
9	0,86	100	5000	0,091	0,87	99	1046
10	0,86	100	5000	0,091	0,86	99	1032
11	0,86	100	5000	0,091	0,85	99	1033
12	0,86	100	5000	0,091	0,84	99	1011
13	0,86	100	5000	0,091	0,83	99	1018
14	0,86	100	5000	0,091	0,82	99	1067
15	0,86	100	5000	0,091	0,81	99	991
16	0,86	100	5000	0,091	0,80	99	991
17	0,86	100	5000	0,091	0,79	99	1055
18	0,86	100	5000	0,091	0,78	99	1063
19	0,86	100	5000	0,091	0,77	99	997
20	0,86	100	5000	0,091	0,76	99	1014
21	0,86	100	5000	0,091	0,75	99	1035
22	0,86	100	5000	0,091	0,74	99	1038
23	0,86	100	5000	0,091	0,73	99	1076
24	0,86	100	5000	0,091	0,72	99	1035
25	0,86	100	5000	0,091	0,71	99	1039
26	0,86	100	5000	0,091	0,70	99	1061
27	0,86	100	5000	0,091	0,69	99	1017
28	0,86	100	5000	0,091	0,68	99	970
29	0,86	100	5000	0,091	0,67	99	1060
30	0,86	100	5000	0,091	0,66	99	1012
31	0,86	100	5000	0,091	0,65	99	1082

32	0,86	100	5000	0,091	0,64	99	1013
33	0,86	100	5000	0,091	0,63	99	1015
34	0,86	100	5000	0,091	0,62	99	993
35	0,86	100	5000	0,091	0,61	99	1056
36	0,86	100	5000	0,091	0,60	99	946

En la Figura 2.5 se presenta un gráfico del desempeño de las pruebas en función del valor de makespan.

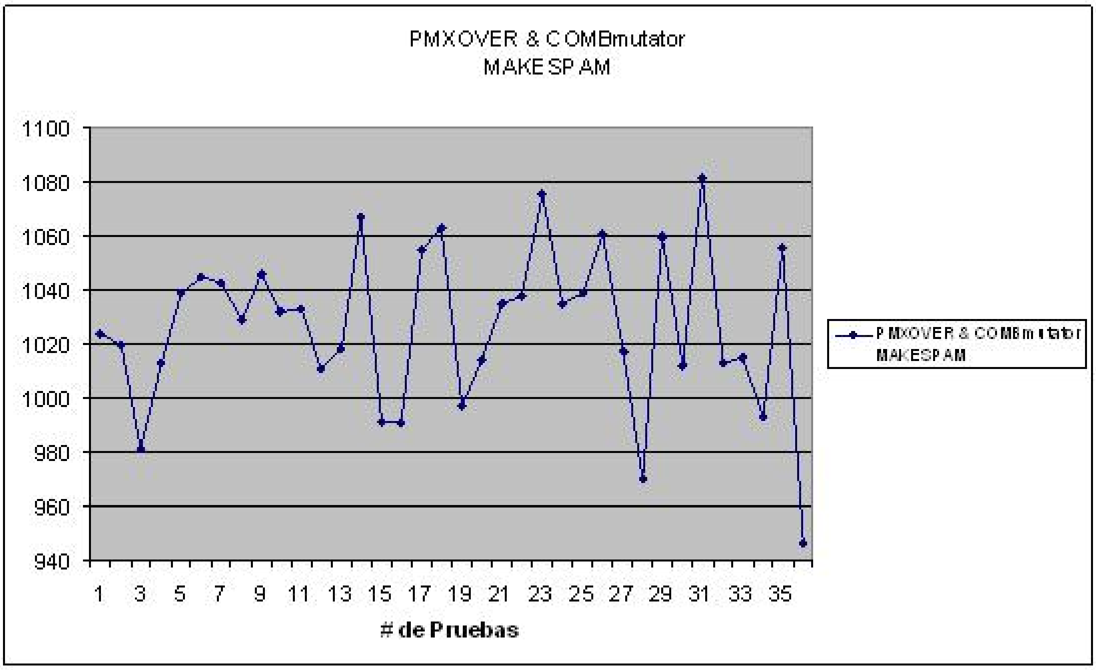

Figura 2.5: *Desempeño de la calibración del porcentaje de cruce*

El mejor valor de makespam manteniendo inalterables los otros parámetros es de 946 para una probabilidad de cruce de 0,6 por lo tanto se trabajará con ese porcentaje para el resto de calibraciones.

2.4.8. Probabilidad de Reemplazo

La probabilidad de reemplazo es necesaria ya que el algoritmo construido es del tipo STEADY-STATE, el cual utiliza el traslape de poblaciones. Para cada generación solamente una porción de ésta es reemplazada por sus hijos. Cuando la probabilidad de reemplazo es del 100 % solamente uno o dos genes son reemplazados, mientras que si la probabilidad de reemplazo es del 0 %, todos los individuos de la población son reemplazados.Para la calibración de la probabilidad de reemplazo se realizan 20 experimentos con el problema estándar FT10, donde se mantiene constante la población, el número de generaciones, la probabilidad de mutación, la probabilidad de cruce y la semilla para la generación de los números aleatorios. Los resultados se indican en la Tabla 2.7.

Tabla 2.7: *Tabla de calibración de probabilidad de reemplazo*

Prueba	Reemplz	Población	Generación	Mutación	Cruce	Seed	PMXOVER & COMBmutator MAKESPAM
1	0,8	100	5000	0,091	0,60	99	1034
2	0,81	100	5000	0,091	0,60	99	1068
3	0,82	100	5000	0,091	0,60	99	1031
4	0,83	100	5000	0,091	0,60	99	1108
5	0,84	100	5000	0,091	0,60	99	1076
6	0,85	100	5000	0,091	0,60	99	1112
7	0,86	100	5000	0,091	0,60	99	946
8	0,87	100	5000	0,091	0,60	99	1048
9	0,88	100	5000	0,091	0,60	99	997
10	0,89	100	5000	0,091	0,60	99	982
11	0,9	100	5000	0,091	0,60	99	1040
12	0,91	100	5000	0,091	0,60	99	1028
13	0,92	100	5000	0,091	0,60	99	1123
14	0,93	100	5000	0,091	0,60	99	1080
15	0,94	100	5000	0,091	0,60	99	1095
16	0,95	100	5000	0,091	0,60	99	1042
17	0,96	100	5000	0,091	0,60	99	1024
18	0,97	100	5000	0,091	0,60	99	1049
19	0,98	100	5000	0,091	0,60	99	1116
20	0,99	100	5000	0,091	0,60	99	1056

El mejor makespan vuelve a ser 946 con un porcentaje de reemplazo de 0,86 por tanto el algoritmo está calibrado y listo para ser usado en los problemas estándar y en el problema de interés.

2.4.9. Archivo de Parámetros

El archivo de parámetros contiene la información con la que va a operar el algoritmo. El nombre del archivo debe ser siempre *entrada_jss.txt* y los datos que contiene son:

- Probabilidad de cruce
- Probabilidad de mutación
- Tamaño de la población
- Número de generaciones
- Probabilidad de reemplazo
- Semilla de números aleatorios

2.4.10. Archivo de Resultados

El archivo de resultados contiene la información de la solución del problema, el nombre del archivo siempre es $resultados\_jss.txt$; contiene la siguiente información:

- MakeSpan Mínimo
- Secuencia Planificada
- Estadística de toda la ejecución del algoritmo, entre otros valores tenemos, número de cruces, número de mutaciones, promedio de todos los makespan, etc.

2.4.11. Diagrama de Flujo

En la Figura 2.6 se presenta el diagrama de flujo del algoritmo propuesto para resolver el problema de interés.

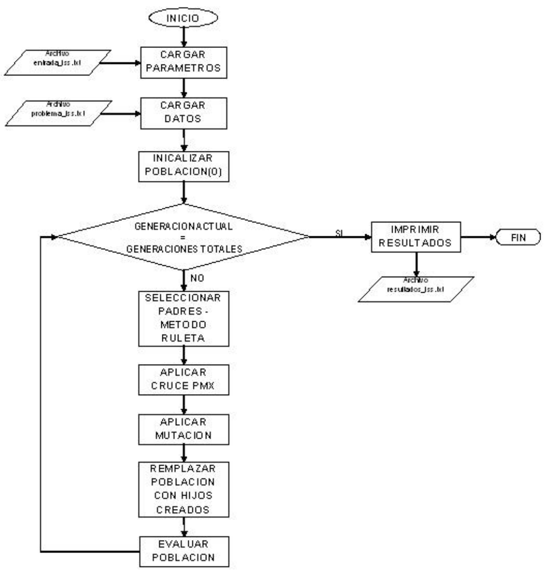

Figura 2.6: *Diagrama de Flujo*

CAPITULO 3

PRUEBAS Y RESULTADOS CON PROBLEMAS ESTANDAR

A continuación, se presenta los resultados obtenidos luego de aplicar el algoritmo genético construido a los problemas estándar FT20, FT10, LA24, LA27 y FT6.

3.1. Problemas Estándar y Hardware usado

Para la evaluación del algoritmo desarrollado durante el presente proyecto de titulación se ha seleccionado 5 problemas estándar de estudio de la comunidad científica sobre el JOB SHOP SCHEDULING PROBLEM disponibles en Internet en la librería OR [2]. Los experimentos han sido ejecutados en una LAPTOP Core 2 DUO T5500, 1.66Ghz, con una memoria RAM de 2.0 GB, bajo Windows XP Service Pack 3. Se realizó 400 ejecuciones para cada instancia (problema estándar). En cada problema estándar con los parámetros calibrados en el capítulo 2 se evaluó las 4 combinaciones posibles de operador de mutación y operador de cruce:

- Operador de cruce JOX y mutación simple
- Operador de cruce JOX y mutación combinada
- Operador de cruce PMX y mutación simple
- Operador de cruce PMX y mutación combinada

Para cada una de las 4 combinaciones se realizan 100 pruebas y se guarda la mejor solución encontrada. En la Tabla 3.1 se presentan los resultados obtenidos por el algoritmo en las instancias evaluadas y en el anexo 1 se pueden encontrar los datos completos.

Tabla 3.1: *Mejores resultados por instancia*

Instancia	Dimensión	Optimo Conocido	jss	Error
FT20	20×5	1165	1183	2 %
FT10	10×10	930	946	2 %
LA24	15×10	935	984	5 %
LA27	20×10	1235	1323	7 %
FT6	6×6	54	55	2 %

El algoritmo construido tiene un error promedio del 3 %.

3.2. Ejemplo de decodificación y funcionamiento del algoritmo

Para entender el funcionamiento del algoritmo propuesto y con fines didácticos se realizaron pruebas con un problema estándar de 3 x 3. Para éste problema se desarrolló la respectiva codificación de la solución que se presenta en la Figura 3.1.

Posición	0	1	2	3	4	5	6	7	8
OPERACIÓN	0	1	2	3	4	5	6	7	8
TRABAJOS	0	0	0	1	1	1	2	2	2
MAQUINA	0	1	2	0	2	1	1	0	2
DURACION	3	3	3	2	3	4	3	2	1

Figura 3.1: *Codificación del problema*

Dada una solución factible obtenida durante la ejecución del algoritmo, el programa busca el orden de operaciones, las máquinas y la duración de cada operación para continuar con la ejecución del algoritmo. En la Figura 3.2 se presenta una solución:

La codificación del problema según lo explicado es:

Como se puede observar la secuencia planificada empieza con 2 lo que significa que se debe procesar la primera operación del trabajo número 2. Según la codificación propuesta ésta es la operación 6 que se realiza en la máquina 1 y tiene un tiempo de duración de 3. De igual forma se procede para el resto de la secuencia. Esta

Posición	0	1	2	3	4	5	6	7	8
OPERACIÓN	6	3	4	0	1	5	7	2	8
SOLUCION	2	1	1	0	0	1	2	0	2
MAQUINA	1	0	2	0	1	1	0	2	2
DURACION	3	2	3	3	3	4	2	3	1

Figura 3.2: *Solución del problema*

información es almacenada en arreglos auxiliares que se usarán para el cálculo del MakeSpan. Los arreglos son:

- OP Arreglo que almacena la secuencia de las operaciones
- MQ Arreglo que almacena la secuencia de máquinas
- TR Arreglo que almacena la secuencia de trabajos
- TI Arreglo que almacena los tiempos de operación

El cálculo del MakeSpan en el algoritmo desarrollado se sustenta en los tiempos de inicio y de fin de cada una de las operaciones. Para ello es necesario usar dos arreglos auxiliares que contendrán ésta información, TINICIO y TFIN. Primero inicializamos los arreglos de tiempo de inicio y fin de las operaciones

while $i < NumeroOperaciones$ **do**

  $TINICIO[i] = 0$

  $TFIN[i] = -1$

end while

El algoritmo de cálculo es el siguiente:

for $i = 0$ to NumeroOperaciones **do**

  Si es la primera operación el tiempo de fin es el tiempo de ejecución de la operación

  if $i = 0$ **then**

    $TFIN[i] = TI[i]$

  else

    Auxiliar de Máquinas

    $aux1 = -1$

    for $j = i - 1$ to 0 **do**

      if $MQ[i] = MQ[j]$ **then**

$aux1 = TFIN[j]$

Salir del for

end if

end for

Auxiliar de Trabajos

$aux2 = -1$

for $j = i - 1$ to 0 **do**

if $TR[i] = TR[j]$ **then**

$aux2 = TFIN[j]$

Salir del for

end if

end for

if $aux1 = aux2$ **then**

Es la primera operación del trabajo y la máquina aún no se ha utilizado

if $aux1 = -1$ **then**

$TINICIO[i] = 0$

$TFIN[i] = TI[i]$

else

$TINICIO[i] = aux1$

$TFIN[i] = aux1 + TI[i]$

end if

end if

if $aux1 > aux2$ **then**

La máquina está disponible y la operación inmediatamente anterior ya se ha terminado

$TINICIO[i] = aux1$

$TFIN[i] = aux1 + TI[i]$

end if

end if

end for

El mayor valor del arreglo TFIN es el MakeSpan

$mkspam = 0$

while $i < NumeroOperaciones$ **do**

  if $mkspam < TFIN[i]$ **then**

    $Mkspam = TFIN[i]$

  end if

end while

El MakeSpan para el ejemplo es 12, tal como se resume en la Figura 3.3.

TINICIO	0	0	2	2	5	8	5	8	11
TFIN	3	2	5	5	8	12	7	11	12

Figura 3.3: *Makespam*

Gráficamente la secuenciación se la representa en la Figura 3.4:

	TIEMPO											
MAQUINA	1	2	3	4	5	6	7	8	9	10	11	12
M0	3	3	0	0	0	7	7					
M1	6	6	6			1	1	1	5	5	5	5
M2			4	4	4				2	2	2	8

TRABAJO 0
TRABAJO1
TRABAJO 2

Figura 3.4: *Secuenciamiento*

Los números dentro de los trabajos son las operaciones según la secuencia planificada tal como se explicó anteriormente.

CAPITULO 4

APLICACION EN EL PROBLEMA DE ESTUDIO

En éste capítulo se presentan los resultados obtenidos luego de aplicar al problema objeto de la tesis, el algoritmo genético construido.

4.1. Descripción del Problema de Estudio

Para la resolución se tomó la planificación de un día donde existía la mayor cantidad de productos posibles y se procedió a resolver usando el algoritmo propuesto y desarrollado en el presente proyecto de titulación.

En la Tabla 4.1, se presenta el cuadro donde se tiene el problema a resolver con los tiempos requeridos en cada célula de trabajo para poder fabricar las cantidades determinadas previamente en el plan maestro de producción. Cabe indicar que los tiempos requeridos se obtienen de las rutas existentes en el ERP.

Tabla 4.1: *Problema a resolver con el algoritmo propuesto*

Producto	**Pelado**	**Picado**	**Lavado**	**Procesar**	**Lavado Envases**	**Preparac LG**	**Envasado**	**Cerrado**	**Pasteurizado**	**Embalado**
ACEITUNA VERDE PEPA 230 G DOY PACK			9	35		1	17	4		2
ACEITUNA VERDE RELLENA PIMIENTO 230 G DOY PACK			9	35		1	17	4		2
ALCACHOFA ENTERA 430 G	9	7	11	23	4	1	14	1	11	2
ALCACHOFA MITADES 400 G.	9	13	11	23	4	1	14	1	11	2
ARVEJA 420 G.			17	29	4	1	15	1	11	2
CEBOLLITAS EN VINAGRE 200 G DOY PACK			15	35		1	17			2
CEBOLLITAS EN VINAGRE 250 G.			16	36	6	1	14	1	11	2
CHAMPIÑON EN VINAGRE 4.1 KG			7	15	3	1	31	3		1
CHOCLITO 2.5 KG	37		9	9	3	1	22	3		1
CHOCLITO 240 G	5		2	2	4	1	12	2	9	3
CHOCLITO 810 G		11			4	1	16	2		2
ESPARRAGO BLANCO JUMBO FCO.350	7		3	19	4	1	14	2	8	2
ESPARRAGO BLANCO 430 G.	7		3	19	4	1	14	2	8	2
ESPARRAGO BLANCO TROZO 430 G.		3	3	19	4	1	14	2	8	2
ESPARRAGO VERDE 430 G.	7		3	19	4	1	14	2	8	2
FREJOL NEGRO 425 GR.			19	17	4	1	12	1	11	2
FREJOL ROJO 425 GR.			15	17	4	1	12	1	11	2
MENESTRA DE LENTEJA 425 G			19	18	4	1	12	1	11	2
PALMITO 410 G EO NACIONAL	5	2	4	12	4	1	9	1	12	2
PALMITO 440 G FCO NACIONAL	6	2	4	13	4	1	10	1	12	2
PALMITO 810 G EO NACIONAL	9	3	5	15	4	1	12	1	12	1

PALMITO CULTIVADO "B"410 G.	5	2	4	12	4	1	9	1	12	2
PALMITO CULTIVADO Ç"810 G.	9	3	5	15	4	1	12	1	12	1
PALMITO EN MITADES 410 G		9	4	9	4	1	9	1	12	2
PALMITO MEDALLONES 810 G		9	5	11	4	1	12	1	12	1
PALMITO MEDIA LUNA 410 G.		8	4	9	4	1	9	1	12	2
PALMITO TROZOS 410 GR.		11	4	9	4	1	9	1	12	2
PEPINILLO AGRIDULCE 220 G DOY PACK		4	2	3		1	12	4		3
PEPINILLO AGRIDULCE 4.1 KG		21	17	12	3	1	31	2		1
PEPINILLO AGRIDULCE 470 G.		6	5	5	4	1	17	2	7	2
PEPINILLO ENTERO 4.1 KG			29	12	3	1	31	2		1
PEPINILLO ENTERO 220 G DOY PACK			7	5		1	17	4		3
PEPINILLO ENTERO 470 G			14	3	4	1	12	2	7	2
PIÑAS EN ALMIBAR 600 G	23		17	31	4	1	7	1	12	2

Para la codificación del problema cada artículo a fabricar es una tarea, que se numera secuencialmente desde el 0 (cero), con lo que se tiene la información de la Tabla 4.2.

Tabla 4.2: *Codificación de las Tareas*

Producto	Tarea
ACEITUNA VERDE PEPA 230 G DOY PACK	0
ACEITUNA VERDE RELLENA PIMIENTO 230 G DOY PACK	1
ALCACHOFA ENTERA 430 G	2
ALCACHOFA MITADES 400 G.	3
ARVEJA 420 G.	4
CEBOLLITAS EN VINAGRE 200 G DOY PACK	5
CEBOLLITAS EN VINAGRE 250 G.	6
CHAMPIÑON EN VINAGRE 4.1 KG	7
CHOCLITO 2.5 KG	8
CHOCLITO 240 G	9
CHOCLITO 810 G	10
ESPARRAGO BLANCO JUMBO FCO.350	11
ESPARRAGO BLANCO 430 G.	12
ESPARRAGO BLANCO TROZO 430 G.	13
ESPARRAGO VERDE 430 G.	14
FREJOL NEGRO 425 GR.	15
FREJOL ROJO 425 GR.	16
MENESTRA DE LENTEJA 425 G	17
PALMITO 410 G EO NACIONAL	18
PALMITO 440 G FCO NACIONAL	19
PALMITO 810 G EO NACIONAL	20
PALMITO CULTIVADO ”B”410 G.	21
PALMITO CULTIVADO Ç”810 G.	22
PALMITO EN MITADES 410 G	23
PALMITO MEDALLONES 810 G	24
PALMITO MEDIA LUNA 410 G.	25
PALMITO TROZOS 410 GR.	26
PEPINILLO AGRIDULCE 220 G DOY PACK	27
PEPINILLO AGRIDULCE 4.1 KG	28
PEPINILLO AGRIDULCE 470 G.	29
PEPINILLO ENTERO 4.1 KG	30
PEPINILLO ENTERO 220 G DOY PACK	31
PEPINILLO ENTERO 470 G	32
PIÑAS EN ALMIBAR 600 G	33

4.2. Resolución del Problema de Estudio

Se colocan los datos del problema y los parámetros previamente calibrados en el archivo de texto que usa el algoritmo. Se realizan 26 pruebas, en cada una de las cuales se varía la semilla para la generación de los números aleatorios usados con los resultados presentados en la Tabla 4.3.

Tabla 4.3: *Resultados de la solución al problema*

PRUEBA	REEMPLAZO	POBLACION	GENERACIONES	MUTACION	CRUCE	SEED	MAKESPAM
0	0,86	100	5000	0,091	0,6	0	599
1	0,86	100	5000	0,091	0,6	1	587
2	0,86	100	5000	0,091	0,6	2	599
3	0,86	100	5000	0,091	0,6	3	599

4	0,86	100	5000	0,091	0,6	4	588
5	0,86	100	5000	0,091	0,6	5	584
6	0,86	100	5000	0,091	0,6	6	573
7	0,86	100	5000	0,091	0,6	7	590
8	0,86	100	5000	0,091	0,6	8	574
9	0,86	100	5000	0,091	0,6	9	606
10	0,86	100	5000	0,091	0,6	10	601
11	0,86	100	5000	0,091	0,6	11	589
12	0,86	100	5000	0,091	0,6	12	579
13	0,86	100	5000	0,091	0,6	13	614
14	0,86	100	5000	0,091	0,6	14	615
15	0,86	100	5000	0,091	0,6	15	592
16	0,86	100	5000	0,091	0,6	16	582
17	0,86	100	5000	0,091	0,6	17	596
18	0,86	100	5000	0,091	0,6	18	601
19	0,86	100	5000	0,091	0,6	19	592
20	0,86	100	5000	0,091	0,6	20	608
21	0,86	100	5000	0,091	0,6	21	591
22	0,86	100	5000	0,091	0,6	22	603
23	0,86	100	5000	0,091	0,6	23	596
24	0,86	100	5000	0,091	0,6	24	590
25	0,86	100	5000	0,091	0,6	25	603
26	0,86	100	5000	0,091	0,6	26	572

El tiempo de solución es de 2 minutos 47 segundos, obteniéndose el mejor makespan que resultó ser de 572. La secuencia de planificación se indica en la Tabla 4.4.

Esta secuencia de planificación se entiende de la siguiente forma:

- La primera tarea que se tiene que empezar a producir es la número 13, de acuerdo a la Tabla 4.2 se observa que la primera tarea a ser realizada es ESPARRAGO BLANCO TROZO 430 G. En la Tabla 4.1 se observa que la primera operación de ésta tarea es el picado con una duración de 3 minutos.

- La siguiente tarea es 13 nuevamente, de la Tabla 4.2 se observa que la siguiente operación para esa tarea es el lavado con una duración de 3 minutos. De ésta forma se analiza toda la respuesta obtenida.

En otras palabras, los números que se obtienen en el archivo de resultados equivalen a las tareas que deben ser procesadas. En la Tabla 4.1, se tiene que buscar la siguiente operación de la tarea de interés. De ésta forma, se codifica la respuesta en un diagrama de Gantt que se indica a continuación.

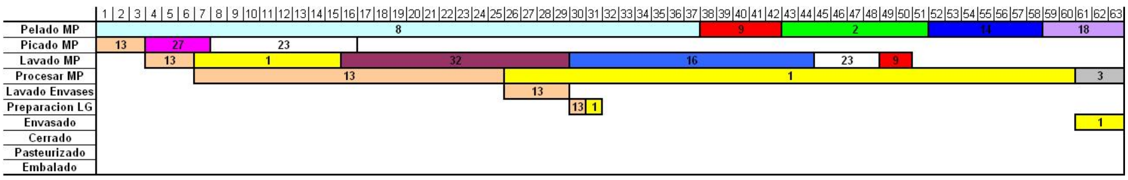

Figura 4.1: *Diagrama de Gantt de la secuencia*

Tabla 4.4: *Secuenciación de las Tareas*

13	13	8	13	27	9	13	2	1	1	32	16	23	14	18	23	18	13	1	9
10	22	32	21	23	18	18	24	18	28	32	25	1	13	8	3	1	21	11	0
21	23	18	22	20	17	23	21	19	21	16	18	8	20	16	2	21	4	20	19
25	1	21	7	13	9	32	12	16	24	23	27	29	8	15	10	28	19	8	16
10	9	13	7	19	20	7	25	30	8	26	25	10	9	26	24	15	32	32	12
20	25	27	7	15	24	11	14	28	22	7	8	2	14	14	9	28	12	22	20
2	26	33	18	31	15	28	14	23	12	10	20	28	20	15	22	12	14	23	27
7	12	19	14	27	22	6	3	26	11	4	19	26	24	3	21	31	15	31	2
17	24	12	2	2	14	16	8	11	21	30	24	4	7	30	15	5	22	20	31
31	10	11	4	2	33	27	33	11	16	11	30	0	25	12	29	29	18	11	9
30	3	19	29	0	22	26	4	26	30	31	9	27	9	19	4	11	25	24	29
28	32	17	26	17	3	4	2	17	29	26	16	4	6	3	15	32	32	6	3
29	6	25	5	17	13	25	0	6	23	22	33	3	29	17	29	0	6	30	2
5	33	19	12	3	33	28	6	0	21	20	33	17	5	17	13	18	19	24	6
3	33	5																	

4.3. Comparación de la solución propuesta con el método actual

Para ver la efectividad del algoritmo propuesto se resuelve el mismo problema con el método actual, de lo que se obtienen los resultados de la Tabla 4.5.

Tabla 4.5: *Resultados de la solución mediante el método actual*

Variable	Valor	Unidad	Observación
Tiempo de solución	2	día	Tiempo del jefe de planificación, jefe de producción y gerente de producción
Makespam	917	minutos	Adicional faltan 60 minutos de comidas

Como se puede observar el método desarrollado en el presente proyecto de titulación presenta un makespam de 572 mientras que el desarrollado por el método tradicional de planificación toma 917 minutos (es decir un 60 % adicional para éste caso específico). Este valor adicional representaba para la empresa un gasto en horas no laboradas pagadas al personal, éste gasto innecesario disminuye debido al uso del programa desarrollado. Adicionalmente, el tiempo utilizado en la resolución también es muy diferente por lo que se puede usar el nuevo método de planificación finita de la producción para la empresa de interés. El tiempo requerido para obtener un diagrama similar al de la Figura 4.1, es de aproximadamente 4 horas. Éste es un tiempo demasiado largo, por lo que para presentar el programa de producción al personal operativo se lo hará en un formato en el que se numera secuencialmente las operaciones por tarea a producir. El formato toma alrededor de 15 minutos en prepararlo y es funcional para los operadores de la planta.

4.4. Asignación de la mano de obra al secuenciameinto

El algoritmo desarrollado en el presente proyecto también sirve para la asignación de la mano de obra directa necesaria para el proceso productivo. Con el archivo de la solución se obtiene, en función de las rutas almacenadas en el ERP, las personas necesarias en cada una de las operaciones de interés.

Cabe indicar la importancia de ésta asignación de forma automática, ya que actualmente con el proceso de planificación finita, el jefe de producción tiene que asignar el personal a las células de producción y crear los horarios de ingreso del personal manualmente, tarea que le toma mucho tiempo. Ahora, al tener el secuenciamiento en un archivo plano, se puede realizar la asignación automáticamente.

4.5. Otras oportundades de mejora

Otro punto de interés es que como se observa en la Figura 4.1 existen celdas de producción que no están balanceadas. Las células que no están balanceadas son:

- Lavado de envases
- Preparación de líquido de gobierno
- Cerrado
- Embalado

Este desbalance provoca que exista demasiado tiempo improductivo en estas células de producción, por lo que se propone las siguientes estrategias de mejora:

4.5.1. Asignación de mano de obra para balanceo de líneas de producción

Las células donde se puede cambiar la asignación de mano de obra sin que afecte el proceso productivo son la de lavado de envases y la de embalaje de producto terminado. En las otras células no puede hacerse ya que el personal que está trabajando actualmente cumple funciones específicas y retirar personal afectaría gravemente el desempeño de éstas celdas.

Para la célula de lavado de envases se cuenta actualmente con 3 personas, de los datos observados en la Figura 4.1 se observa que la célula tiene un 52 % aproximadamente de tiempos improductivos, por lo que se propone las siguientes mejoras:

- Trabajo de la célula sin tiempos muertos, esto es posible si se cambia la forma de trabajo para almacenar los envases lavados hasta que el producto esté listo para ser envasado. Para esto se asignó en la planta un espacio que puede almacenar 4 pallets de envase lavado. Cabe indicar que se analizó esta decisión con el comité HACCP de la empresa y no existe problema debido a que todos los envases vienen en cartón por lo que una vez lavados se almacenan en el mismo cartón.

- Reducción de una persona de mano de obra directa. El tiempo improductivo con el nuevo método de planificación equivale aproximadamente al 52 % por lo que podrían retirarse dos personas sin embargo hay que tomar en cuenta que ahora el tiempo incrementa ya que se tiene que guardar el envase lavado en las cajas de cartón y transportar estas cajas al espacio destinado para el almacenamiento.

Para la célula embalaje se cuenta actualmente con 3 personas, de los datos observados en la Figura 4.1 se observa que la célula tiene un 69 % aproximadamente de tiempos improductivos, por lo que se propone la siguiente mejora:

- Reducción de dos personas de mano de obra directa. Esta reducción no causa problemas debido a que si la persona que se queda en la célula se retrasa en el embalaje se acumula el producto antes de pasar a bodega pero al ser la última operación no afecta al secuenciamiento planificado.

4.5.2. Cambio en el método de trabajo

Específicamente para la célula de preparación de líquido de gobierno se propone la siguiente mejora

- Unificar la formulación de líquido de gobierno para todos los productos, con esto se prepara el líquido de gobierno al inicio del turno y se lo almacena en los tanques previstos para el efecto. De ésta forma la persona de ésta célula de producción trabajaría aproximadamente 3 horas y luego pasaría a otras células.

4.6. Implementación en la Empresa ABC

El proceso de implementación en la Empresa ABC consistió en planificar usando el algoritmo propuesto y se lo empezó a usar a partir de la tercera semana luego de la finalización del desarrollo del mismo y con el balanceo de las células de producción. Con esto se pudo recopilar la información de 3 semanas con lo cual se obtiene la siguiente información.

4.6.1. Tiempo de Para en las celdas de producción

En las tres semanas se recopiló la información de para de cada celda de producción de forma similar con la que se obtuvo la información de la sección 2. Se presenta en la Tabla 4.6 el tiempo de para promedio diario.

Tabla 4.6: *Tiempo de Para promedio diario*

Causa	Tiempo (min)
Secuenciación de Producción	7,1
Daño de máquina	39,8
Falta de personal	12,2
Otros	20,3
Total	**79,4**

Se puede observar lo siguiente:

- El mal secuenciamiento de producción paso a ser la causa menor de para de no planificada en las celdas de producción. El tiempo actual de 7,1 minutos indica que todavía existe una mejor solución pero para objetos prácticos el algoritmo desarrollado es excelente.

- La falta de personal también disminuyó debido a que actualmente se tiene dos personas flotantes (provenientes del balanceo de las líneas de producción) por lo que si falta una persona inmediatamente pasan a sustituirla.

4.6.2. Horas extra pagadas no planificadas

De igual forma que el indicador calculado en la sección 2, se procede a calcular el nuevo valor del indicador con lo que se tiene los datos de la Tabla 4.7. Cabe indicar que como se contaba únicamente con datos de 3 semanas, se hizo una regla de tres para obtener el valor de 4 semanas y poder hacer el comparativo:

Tabla 4.7: *Horas extra no planificadas por mes*

Mes	Horas extras 50 %	Horas extras 100 %	Total
Noviembre	196	72	268
TOTAL			52715

Se puede observar que se tiene un valor de $268,00 USD en horas extras para el mes, lo cual da un valor anual aproximado de $7.380,00 USD el cual frente a los $150.000,00 USD que la empresa pagaba anteriormente representa un ahorro de $143.000,00 USD al año.

4.7. Conclusiones y Recomendaciones

4.7.1. Conclusiones

Con respecto al objetivo principal de la investigación, es decir la obtención de un sistema que resuelva el problema de estudio se encuentra que:

- Se desarrolló un algoritmo genético que resuelve problemas de la familia JSS debido a la facilidad de implementación y ejecución de los genéticos. Para la obtención de los parámetros del algoritmo de usó el problema estándar FT10 y se fijaron valores iniciales obtenidos de la bibliografía, se modificaron uno a uno cada parámetro y se guardaban los que permitían obtener el mejor makespan para el problema en mención. A partir de esto se obtuvo los siguientes valores de calibración:

    - Población: 100
    - Número de generaciones: 5000
    - Probabilidad de mutación: 0,091

- Probabilidad de cruce: 0,60
- Probabilidad de reemplazo: 0,86
- Semilla para generar números aleatorios: 99
- Mejor combinación de operadores: operador de cruce PMX y operador de mutación: mutación combinada.

- En el algoritmo planteado no fue necesario implementar una etapa de factibilización, debido a que la población inicial se genera con secuencias factibles y los operadores implementados garantizan secuencias legítimas. Sin embargo, para futuras investigaciones, es posible considerar alternativas de solución infactibles lo cual justificaría una etapa de factibilización y permitiría contar con un espacio de soluciones mayor.

- Se diseño el algoritmo de forma que sea compacto en cuanto a recursos usados, para esto se trabajó sobre la representación y los mecanismos de cruce y mutación utilizados. Gracias a esto se utiliza poca memoria y el tiempo de ejecución es razonable comparado con otros métodos existentes.

- El algoritmo desarrollado en el presente proyecto cumple el objetivo propuesto inicialmente, el cual era obtener un mejor plan finito de producción en un tiempo menor al que se requería en el método de planificación tradicional. En uno de los casos a resolver se pudo comparar que el algoritmo propuesto disminuyó el makespan en 60 % del valor que se obtenía mediante la forma tradicional de planificación. El tiempo requerido con el algoritmo genético propuesto oscila de dos a tres minutos, mientras que anteriormente se tardaba dos días. Cabe recalcar que éste valor es para uno de los casos propuestos en los que se pudo comparar los resultados obtenidos por los dos métodos.

- La rápida ejecución del algoritmo permitirá al planificador de la producción, realizar varios escenarios factibles y analizarlos con las particularidades que ocurren cada día de forma que cuando el desarrollo se convierte en una herramienta cuando hay imprevistos.

- Con la implementación del algoritmo genético se espera obtener un ahorro de $143.000,00 USD al año en reducción de horas extras al personal de la planta.

Mensualmente se debe monitorear este indicador para verificar que se éste usando el método de planificación propuesto.

- El algoritmo presenta un 3 % de error medio en la resolución de los problemas estándar: FT20, FT10, LA24, LA27 y FT6. El tiempo promedio de ejecución es de 2 minutos 47 segundos. Se puede concluir que a comparación de los métodos metaheurísticos modernos con errores menores al 1 % y tiempos que oscilan entre 32 a 47 segundos, el algoritmo genético desarrollado es muy bueno para la empresa de interés pero en el campo teórico está muy por debajo del desempeño de los métodos actuales.

- El algoritmo ha sido desarrollado en C++ usando como base el framework de algoritmos genéticos GALIB desarrollado por Matthew Wall en el Instituto Tecnológico de Massachussets (MIT) [1]. Los experimentos han sido ejecutados en una LAPTOP Core 2 DUO T5500, 1.66Ghz, con una memoria RAM de 2.0 GB, bajo Windows XP Service Pack 3.

4.7.2. Recomendaciones

- El algoritmo desarrollado no contempla la representación gráfica de la solución, a futuro se puede incluir el algoritmo en una aplicación de entorno Windows que facilite la operación del personal de la planta.

- Otra recomendación que se propone es extender al algoritmo desarrollado para crear un algoritmo híbrido; de la bibliografía revisada se puede construir un algoritmo genético que incluya mecanismos de la búsqueda tabú a fin de mejorar su desempeño. Otra opción es el procesamiento paralelo, el cual es la ejecución simultánea de instrucciones en una computadora. La recomendación es extender el algoritmo desarrollado para transformarlo en un algoritmo genético de procesamiento en paralelo. El método más simple de paralelizar un AG se llama paralelización global; en donde existe una sola población al igual que en el AG convencional, pero la evaluación de los individuos y los operadores genéticos se paralelizan de forma explícita.

- El algoritmo desarrollado optimiza la secuenciación diaria de producción de la planta. Se ve a futuro la necesidad de extender el algoritmo para que permita realizar la secuenciación de producción de períodos más largos de tiempo. Para ello es necesario incluir restricciones como: la rotación de personal, las fechas y cantidades de entrega de materia prima, la planificación de mantenimiento de maquinaria, entre otras.

- Existen diversos objetivos que pueden ser evaluados al buscar una secuencia óptima, como por ejemplo: cumplir con las fechas de entrega, minimizar la tardanza de trabajos, minimizar horas extras, maximizar la utilización de máquinas, minimizar el tiempo imporductivo, o minimizar inventario de trabajo en proceso. La selección del objetivo a optimizar depende de la necesidad del usuario, en este trabajo se minimizó el tiempo de procesamiento total (Cmax). Sin embargo se puede estudiar la posibilidad de mejorar el algoritmo para que pueda realizar optimización multi-objetivo.

ANEXO

							JOXOVER & SIMPmutator	JOXOVER & COMBmutator	PMXOVER & SIMPmutator	PMXOVER & COMBmutator
PRUEBA	REEMPLAZO	POBLACION	GENERACIO	MUTACION	CRUCE	SEED	MAKESPAM	MAKESPAM	MAKESPAM	MAKESPAM
1	0,86	100	5000	0,091	0,6	0	58	59	55	55
2	0,86	100	5000	0,091	0,6	1	58	58	58	55
3	0,86	100	5000	0,091	0,6	2	59	58	58	55
4	0,86	100	5000	0,091	0,6	3	55	58	55	55
5	0,86	100	5000	0,091	0,6	4	55	58	55	55
6	0,86	100	5000	0,091	0,6	5	55	55	55	55
7	0,86	100	5000	0,091	0,6	6	55	57	55	55
8	0,86	100	5000	0,091	0,6	7	59	59	55	57
9	0,86	100	5000	0,091	0,6	8	59	58	58	55
10	0,86	100	5000	0,091	0,6	9	59	59	58	55
11	0,86	100	5000	0,091	0,6	10	58	58	60	55
12	0,86	100	5000	0,091	0,6	11	59	55	57	55
13	0,86	100	5000	0,091	0,6	12	55	58	55	55
14	0,86	100	5000	0,091	0,6	13	59	55	55	55
15	0,86	100	5000	0,091	0,6	14	55	55	55	55
16	0,86	100	5000	0,091	0,6	15	55	59	55	57
17	0,86	100	5000	0,091	0,6	16	59	57	55	55
18	0,86	100	5000	0,091	0,6	17	57	55	55	55
19	0,86	100	5000	0,091	0,6	18	55	55	55	55
20	0,86	100	5000	0,091	0,6	19	59	59	58	55
21	0,86	100	5000	0,091	0,6	20	57	57	57	58
22	0,86	100	5000	0,091	0,6	21	59	60	55	55
23	0,86	100	5000	0,091	0,6	22	55	57	57	57
24	0,86	100	5000	0,091	0,6	23	55	59	57	55
25	0,86	100	5000	0,091	0,6	24	58	59	57	55
26	0,86	100	5000	0,091	0,6	25	55	58	55	55
27	0,86	100	5000	0,091	0,6	26	55	57	57	55
28	0,86	100	5000	0,091	0,6	27	59	55	60	55
29	0,86	100	5000	0,091	0,6	28	55	55	55	55
30	0,86	100	5000	0,091	0,6	29	55	55	55	55
31	0,86	100	5000	0,091	0,6	30	57	55	58	55
32	0,86	100	5000	0,091	0,6	31	59	58	55	55
33	0,86	100	5000	0,091	0,6	32	55	58	58	55
34	0,86	100	5000	0,091	0,6	33	55	55	57	55
35	0,86	100	5000	0,091	0,6	34	55	55	55	57
36	0,86	100	5000	0,091	0,6	35	55	55	55	55
37	0,86	100	5000	0,091	0,6	36	55	55	55	55
38	0,86	100	5000	0,091	0,6	37	55	55	55	58
39	0,86	100	5000	0,091	0,6	38	58	55	55	57
40	0,86	100	5000	0,091	0,6	39	55	58	59	55
41	0,86	100	5000	0,091	0,6	40	59	58	57	55
42	0,86	100	5000	0,091	0,6	41	55	59	55	55
43	0,86	100	5000	0,091	0,6	42	57	58	55	55
44	0,86	100	5000	0,091	0,6	43	58	59	59	59
45	0,86	100	5000	0,091	0,6	44	55	58	58	55
46	0,86	100	5000	0,091	0,6	45	59	55	55	55
47	0,86	100	5000	0,091	0,6	46	55	55	57	55
48	0,86	100	5000	0,091	0,6	47	55	59	55	55
49	0,86	100	5000	0,091	0,6	48	58	55	55	55
50	0,86	100	5000	0,091	0,6	49	59	55	58	58
51	0,86	100	5000	0,091	0,6	50	57	58	55	59
52	0,86	100	5000	0,091	0,6	51	55	58	55	57
53	0,86	100	5000	0,091	0,6	52	58	55	59	55
54	0,86	100	5000	0,091	0,6	53	59	59	57	58
55	0,86	100	5000	0,091	0,6	54	55	58	55	55
56	0,86	100	5000	0,091	0,6	55	58	55	55	60
57	0,86	100	5000	0,091	0,6	56	55	59	55	55
58	0,86	100	5000	0,091	0,6	57	59	55	55	55
59	0,86	100	5000	0,091	0,6	58	55	58	55	55
60	0,86	100	5000	0,091	0,6	59	59	59	57	55
61	0,86	100	5000	0,091	0,6	60	55	55	55	55
62	0,86	100	5000	0,091	0,6	61	55	59	55	55
63	0,86	100	5000	0,091	0,6	62	55	59	60	55
64	0,86	100	5000	0,091	0,6	63	58	55	57	58
65	0,86	100	5000	0,091	0,6	64	55	55	58	58
66	0,86	100	5000	0,091	0,6	65	59	58	57	55
67	0,86	100	5000	0,091	0,6	66	55	55	59	58
68	0,86	100	5000	0,091	0,6	67	59	56	55	55
69	0,86	100	5000	0,091	0,6	68	58	59	58	55
70	0,86	100	5000	0,091	0,6	69	58	58	57	57
71	0,86	100	5000	0,091	0,6	70	55	58	58	57
72	0,86	100	5000	0,091	0,6	71	59	59	59	55
73	0,86	100	5000	0,091	0,6	72	58	58	61	55
74	0,86	100	5000	0,091	0,6	73	55	55	55	57
75	0,86	100	5000	0,091	0,6	74	58	57	58	55
76	0,86	100	5000	0,091	0,6	75	55	59	55	55
77	0,86	100	5000	0,091	0,6	76	55	58	57	55
78	0,86	100	5000	0,091	0,6	77	55	55	55	57
79	0,86	100	5000	0,091	0,6	78	60	57	55	57
80	0,86	100	5000	0,091	0,6	79	55	58	55	55
81	0,86	100	5000	0,091	0,6	80	55	55	55	59
82	0,86	100	5000	0,091	0,6	81	55	59	55	55
83	0,86	100	5000	0,091	0,6	82	55	55	55	55
84	0,86	100	5000	0,091	0,6	83	58	59	58	55
85	0,86	100	5000	0,091	0,6	84	55	58	55	59
86	0,86	100	5000	0,091	0,6	85	59	55	55	60
87	0,86	100	5000	0,091	0,6	86	58	55	55	55
88	0,86	100	5000	0,091	0,6	87	55	60	55	55
89	0,86	100	5000	0,091	0,6	88	55	59	55	55
90	0,86	100	5000	0,091	0,6	89	55	55	55	55
91	0,86	100	5000	0,091	0,6	90	55	59	55	57
92	0,86	100	5000	0,091	0,6	91	59	58	59	55
93	0,86	100	5000	0,091	0,6	92	58	58	57	57
94	0,86	100	5000	0,091	0,6	93	59	59	58	58
95	0,86	100	5000	0,091	0,6	94	60	60	55	55
96	0,86	100	5000	0,091	0,6	95	59	59	57	55
97	0,86	100	5000	0,091	0,6	96	55	57	58	55
98	0,86	100	5000	0,091	0,6	97	55	59	55	55
99	0,86	100	5000	0,091	0,6	98	55	57	58	58
100	0,86	100	5000	0,091	0,6	99	55	58	55	58
101	0,86	100	5000	0,091	0,6	100	59	57	55	55

Anexo i: *Pruebas de la instancia FT6*

							JOXOVER & SIMPmutator	JOXOVER & COMBmutator	PMXOVER & SIMPmutator	PMXOVER & COMBmutator
PRUEBA	REEMPLAZO	POBLACION	GENERACIO	MUTACION	CRUCE	SEED	MAKESPAM	MAKESPAM	MAKESPAM	MAKESPAM
1	0,86	100	5000	0,091	0,6	0	1053	1135	990	1004
2	0,86	100	5000	0,091	0,6	1	1077	1046	1055	1081
3	0,86	100	5000	0,091	0,6	2	1002	1018	1071	1012
4	0,86	100	5000	0,091	0,6	3	1057	1105	1089	1043
5	0,86	100	5000	0,091	0,6	4	1071	1199	1034	1031
6	0,86	100	5000	0,091	0,6	5	1134	1021	1036	1047
7	0,86	100	5000	0,091	0,6	6	1040	1072	1091	1062
8	0,86	100	5000	0,091	0,6	7	1052	1010	1110	1057
9	0,86	100	5000	0,091	0,6	8	1125	1070	1087	1078
10	0,86	100	5000	0,091	0,6	9	1010	1051	1094	1118
11	0,86	100	5000	0,091	0,6	10	1008	1041	1016	972
12	0,86	100	5000	0,091	0,6	11	976	972	1071	1041
13	0,86	100	5000	0,091	0,6	12	1094	1048	1025	1099
14	0,86	100	5000	0,091	0,6	13	1094	1043	1047	1110
15	0,86	100	5000	0,091	0,6	14	1040	1062	1069	1096
16	0,86	100	5000	0,091	0,6	15	1119	1119	1122	1098
17	0,86	100	5000	0,091	0,6	16	1050	1058	1006	1042
18	0,86	100	5000	0,091	0,6	17	1222	1081	1046	1076
19	0,86	100	5000	0,091	0,6	18	1089	1035	1049	1026
20	0,86	100	5000	0,091	0,6	19	1015	1046	1037	1099
21	0,86	100	5000	0,091	0,6	20	1028	1002	1043	1028
22	0,86	100	5000	0,091	0,6	21	1101	1130	1024	1053
23	0,86	100	5000	0,091	0,6	22	1124	1050	1069	1075
24	0,86	100	5000	0,091	0,6	23	1021	1088	981	996
25	0,86	100	5000	0,091	0,6	24	1075	1093	1117	1094
26	0,86	100	5000	0,091	0,6	25	1081	1019	996	1055
27	0,86	100	5000	0,091	0,6	26	1094	1071	1043	1056
28	0,86	100	5000	0,091	0,6	27	1103	1092	1061	1042
29	0,86	100	5000	0,091	0,6	28	1057	1104	989	1028
30	0,86	100	5000	0,091	0,6	29	1077	1033	1046	1078
31	0,86	100	5000	0,091	0,6	30	1103	1049	1094	1020
32	0,86	100	5000	0,091	0,6	31	1014	1035	1077	1044
33	0,86	100	5000	0,091	0,6	32	1074	1045	1051	1093
34	0,86	100	5000	0,091	0,6	33	1078	1118	1057	1108
35	0,86	100	5000	0,091	0,6	34	969	1072	971	993
36	0,86	100	5000	0,091	0,6	35	1120	1098	1015	983
37	0,86	100	5000	0,091	0,6	36	1079	1018	1083	1029
38	0,86	100	5000	0,091	0,6	37	1120	1063	1026	999
39	0,86	100	5000	0,091	0,6	38	1054	1059	1073	1108
40	0,86	100	5000	0,091	0,6	39	1001	1022	1068	1088
41	0,86	100	5000	0,091	0,6	40	1048	1146	1138	1100
42	0,86	100	5000	0,091	0,6	41	980	969	1019	1076
43	0,86	100	5000	0,091	0,6	42	1056	1023	1086	1060
44	0,86	100	5000	0,091	0,6	43	1073	1086	1080	1048
45	0,86	100	5000	0,091	0,6	44	1080	1088	1027	1046
46	0,86	100	5000	0,091	0,6	45	1053	1061	1042	1024
47	0,86	100	5000	0,091	0,6	46	1027	1091	994	1058
48	0,86	100	5000	0,091	0,6	47	1105	1019	1083	1014
49	0,86	100	5000	0,091	0,6	48	1054	1122	1027	1075
50	0,86	100	5000	0,091	0,6	49	1083	1099	1011	984
51	0,86	100	5000	0,091	0,6	50	1085	1071	1021	1077
52	0,86	100	5000	0,091	0,6	51	1051	1032	1056	982
53	0,86	100	5000	0,091	0,6	52	1013	1009	1007	1057
54	0,86	100	5000	0,091	0,6	53	1014	1016	1045	1066
55	0,86	100	5000	0,091	0,6	54	1046	1029	1120	1108
56	0,86	100	5000	0,091	0,6	55	1019	1093	1005	992
57	0,86	100	5000	0,091	0,6	56	1025	1020	1058	1030
58	0,86	100	5000	0,091	0,6	57	1024	1020	1068	991
59	0,86	100	5000	0,091	0,6	58	1051	1038	1047	998
60	0,86	100	5000	0,091	0,6	59	1102	1054	1083	1071
61	0,86	100	5000	0,091	0,6	60	1026	1048	1113	1052
62	0,86	100	5000	0,091	0,6	61	1049	1034	1029	1075
63	0,86	100	5000	0,091	0,6	62	1029	1002	1041	1076
64	0,86	100	5000	0,091	0,6	63	1085	1035	1119	1042
65	0,86	100	5000	0,091	0,6	64	1084	1097	1020	1126
66	0,86	100	5000	0,091	0,6	65	1077	1121	1108	1037
67	0,86	100	5000	0,091	0,6	66	1074	1078	994	1055
68	0,86	100	5000	0,091	0,6	67	1059	1154	1089	1059
69	0,86	100	5000	0,091	0,6	68	1089	1035	1033	1047
70	0,86	100	5000	0,091	0,6	69	1030	1068	1002	1017
71	0,86	100	5000	0,091	0,6	70	1029	1039	1070	985
72	0,86	100	5000	0,091	0,6	71	1085	1053	1070	1106
73	0,86	100	5000	0,091	0,6	72	1045	1095	1052	987
74	0,86	100	5000	0,091	0,6	73	1044	1255	1055	1071
75	0,86	100	5000	0,091	0,6	74	1076	1093	1022	1071
76	0,86	100	5000	0,091	0,6	75	1056	1047	982	1019
77	0,86	100	5000	0,091	0,6	76	1018	1043	1123	1014
78	0,86	100	5000	0,091	0,6	77	1042	1046	978	1059
79	0,86	100	5000	0,091	0,6	78	1050	1087	1015	1084
80	0,86	100	5000	0,091	0,6	79	1069	1044	1092	1080
81	0,86	100	5000	0,091	0,6	80	990	1072	1029	1052
82	0,86	100	5000	0,091	0,6	81	1061	1071	1065	1084
83	0,86	100	5000	0,091	0,6	82	1004	1094	1026	1068
84	0,86	100	5000	0,091	0,6	83	1121	970	1070	1030
85	0,86	100	5000	0,091	0,6	84	1093	1041	1041	1048
86	0,86	100	5000	0,091	0,6	85	1065	1137	1098	997
87	0,86	100	5000	0,091	0,6	86	1070	1103	1000	1064
88	0,86	100	5000	0,091	0,6	87	1097	1038	1039	1014
89	0,86	100	5000	0,091	0,6	88	1035	1101	1024	1045
90	0,86	100	5000	0,091	0,6	89	1069	1093	1043	987
91	0,86	100	5000	0,091	0,6	90	983	1019	1102	1023
92	0,86	100	5000	0,091	0,6	91	1085	1007	1082	1031
93	0,86	100	5000	0,091	0,6	92	1036	1076	1052	1035
94	0,86	100	5000	0,091	0,6	93	1053	1011	1055	1013
95	0,86	100	5000	0,091	0,6	94	1006	1023	974	1000
96	0,86	100	5000	0,091	0,6	95	1057	1083	982	1037
97	0,86	100	5000	0,091	0,6	96	1051	1147	1015	1082
98	0,86	100	5000	0,091	0,6	97	1047	1040	1008	1026
99	0,86	100	5000	0,091	0,6	98	1067	1038	1109	1008
100	0,86	100	5000	0,091	0,6	99	995	1117	1045	946
101	0,86	100	5000	0,091	0,6	100	995	1117	1045	960

Anexo ii: *Pruebas de la instancia FT10*

							JOXOVER & SIMPmutator	JOXOVER & COMBmutator	PMXOVER & SIMPmutator	PMXOVER & COMBmutator
PRUEBA	REEMPLAZC	POBLACION	GENERACIO	MUTACION	CRUCE	SEED	MAKESPAM	MAKESPAM	MAKESPAM	MAKESPAM
1	0,86	100	5000	0,091	0,6	0	1291	1342	1334	1261
2	0,86	100	5000	0,091	0,6	1	1354	1355	1305	1344
3	0,86	100	5000	0,091	0,6	2	1263	1342	1256	1292
4	0,86	100	5000	0,091	0,6	3	1261	1255	1265	1241
5	0,86	100	5000	0,091	0,6	4	1359	1273	1306	1324
6	0,86	100	5000	0,091	0,6	5	1232	1333	1274	1289
7	0,86	100	5000	0,091	0,6	6	1350	1451	1244	1320
8	0,86	100	5000	0,091	0,6	7	1380	1493	1281	1183
9	0,86	100	5000	0,091	0,6	8	1266	1261	1310	1298
10	0,86	100	5000	0,091	0,6	9	1297	1276	1249	1283
11	0,86	100	5000	0,091	0,6	10	1388	1333	1234	1259
12	0,86	100	5000	0,091	0,6	11	1231	1340	1283	1318
13	0,86	100	5000	0,091	0,6	12	1353	1234	1282	1264
14	0,86	100	5000	0,091	0,6	13	1329	1343	1332	1260
15	0,86	100	5000	0,091	0,6	14	1271	1318	1215	1272
16	0,86	100	5000	0,091	0,6	15	1330	1361	1290	1311
17	0,86	100	5000	0,091	0,6	16	1313	1300	1283	1325
18	0,86	100	5000	0,091	0,6	17	1317	1339	1246	1274
19	0,86	100	5000	0,091	0,6	18	1233	1279	1216	1305
20	0,86	100	5000	0,091	0,6	19	1331	1318	1278	1314
21	0,86	100	5000	0,091	0,6	20	1312	1304	1304	1254
22	0,86	100	5000	0,091	0,6	21	1271	1360	1232	1277
23	0,86	100	5000	0,091	0,6	22	1340	1372	1236	1295
24	0,86	100	5000	0,091	0,6	23	1251	1245	1296	1297
25	0,86	100	5000	0,091	0,6	24	1273	1351	1247	1287
26	0,86	100	5000	0,091	0,6	25	1376	1286	1220	1304
27	0,86	100	5000	0,091	0,6	26	1346	1312	1321	1273
28	0,86	100	5000	0,091	0,6	27	1251	1291	1266	1285
29	0,86	100	5000	0,091	0,6	28	1253	1420	1368	1275
30	0,86	100	5000	0,091	0,6	29	1264	1266	1248	1273
31	0,86	100	5000	0,091	0,6	30	1433	1344	1305	1204
32	0,86	100	5000	0,091	0,6	31	1388	1349	1343	1229
33	0,86	100	5000	0,091	0,6	32	1316	1272	1247	1325
34	0,86	100	5000	0,091	0,6	33	1327	1315	1286	1325
35	0,86	100	5000	0,091	0,6	34	1209	1354	1253	1246
36	0,86	100	5000	0,091	0,6	35	1332	1300	1327	1258
37	0,86	100	5000	0,091	0,6	36	1288	1326	1314	1318
38	0,86	100	5000	0,091	0,6	37	1305	1337	1300	1297
39	0,86	100	5000	0,091	0,6	38	1279	1273	1233	1293
40	0,86	100	5000	0,091	0,6	39	1286	1262	1297	1333
41	0,86	100	5000	0,091	0,6	40	1283	1326	1263	1268
42	0,86	100	5000	0,091	0,6	41	1312	1273	1257	1292
43	0,86	100	5000	0,091	0,6	42	1241	1258	1291	1304
44	0,86	100	5000	0,091	0,6	43	1257	1330	1270	1305
45	0,86	100	5000	0,091	0,6	44	1374	1349	1252	1293
46	0,86	100	5000	0,091	0,6	45	1325	1294	1297	1292
47	0,86	100	5000	0,091	0,6	46	1345	1336	1248	1249
48	0,86	100	5000	0,091	0,6	47	1331	1268	1206	1257
49	0,86	100	5000	0,091	0,6	48	1270	1236	1274	1243
50	0,86	100	5000	0,091	0,6	49	1291	1272	1243	1284
51	0,86	100	5000	0,091	0,6	50	1361	1364	1336	1318
52	0,86	100	5000	0,091	0,6	51	1433	1381	1290	1311
53	0,86	100	5000	0,091	0,6	52	1295	1335	1209	1308
54	0,86	100	5000	0,091	0,6	53	1217	1407	1241	1276
55	0,86	100	5000	0,091	0,6	54	1265	1285	1311	1268
56	0,86	100	5000	0,091	0,6	55	1276	1338	1253	1237
57	0,86	100	5000	0,091	0,6	56	1339	1248	1305	1295
58	0,86	100	5000	0,091	0,6	57	1333	1345	1248	1261
59	0,86	100	5000	0,091	0,6	58	1269	1239	1332	1285
60	0,86	100	5000	0,091	0,6	59	1269	1326	1339	1256
61	0,86	100	5000	0,091	0,6	60	1325	1287	1282	1263
62	0,86	100	5000	0,091	0,6	61	1259	1304	1376	1225
63	0,86	100	5000	0,091	0,6	62	1334	1400	1281	1280
64	0,86	100	5000	0,091	0,6	63	1295	1327	1290	1333
65	0,86	100	5000	0,091	0,6	64	1246	1449	1292	1303
66	0,86	100	5000	0,091	0,6	65	1339	1281	1248	1284
67	0,86	100	5000	0,091	0,6	66	1322	1234	1234	1241
68	0,86	100	5000	0,091	0,6	67	1248	1201	1265	1307
69	0,86	100	5000	0,091	0,6	68	1364	1305	1325	1378
70	0,86	100	5000	0,091	0,6	69	1251	1285	1279	1253
71	0,86	100	5000	0,091	0,6	70	1290	1319	1267	1230
72	0,86	100	5000	0,091	0,6	71	1386	1281	1270	1302
73	0,86	100	5000	0,091	0,6	72	1226	1303	1249	1320
74	0,86	100	5000	0,091	0,6	73	1262	1326	1301	1312
75	0,86	100	5000	0,091	0,6	74	1337	1288	1330	1305
76	0,86	100	5000	0,091	0,6	75	1277	1343	1263	1266
77	0,86	100	5000	0,091	0,6	76	1337	1277	1323	1286
78	0,86	100	5000	0,091	0,6	77	1366	1276	1232	1308
79	0,86	100	5000	0,091	0,6	78	1331	1271	1278	1327
80	0,86	100	5000	0,091	0,6	79	1284	1337	1280	1301
81	0,86	100	5000	0,091	0,6	80	1304	1430	1327	1275
82	0,86	100	5000	0,091	0,6	81	1280	1372	1262	1316
83	0,86	100	5000	0,091	0,6	82	1274	1339	1358	1251
84	0,86	100	5000	0,091	0,6	83	1272	1210	1330	1288
85	0,86	100	5000	0,091	0,6	84	1324	1323	1316	1289
86	0,86	100	5000	0,091	0,6	85	1306	1429	1248	1311
87	0,86	100	5000	0,091	0,6	86	1339	1249	1281	1350
88	0,86	100	5000	0,091	0,6	87	1326	1258	1264	1372
89	0,86	100	5000	0,091	0,6	88	1331	1328	1269	1239
90	0,86	100	5000	0,091	0,6	89	1364	1273	1303	1255
91	0,86	100	5000	0,091	0,6	90	1246	1254	1241	1278
92	0,86	100	5000	0,091	0,6	91	1358	1250	1291	1220
93	0,86	100	5000	0,091	0,6	92	1251	1331	1233	1282
94	0,86	100	5000	0,091	0,6	93	1298	1322	1261	1311
95	0,86	100	5000	0,091	0,6	94	1403	1349	1234	1342
96	0,86	100	5000	0,091	0,6	95	1297	1232	1249	1268
97	0,86	100	5000	0,091	0,6	96	1288	1298	1269	1230
98	0,86	100	5000	0,091	0,6	97	1426	1426	1330	1289
99	0,86	100	5000	0,091	0,6	98	1310	1300	1274	1356
100	0,86	100	5000	0,091	0,6	99	1247	1286	1269	1293
101	0,86	100	5000	0,091	0,6	100	1268	1266	1189	1311

Anexo iii: *Pruebas de la instancia FT20*

							JOXOVER & SIMPmutator	JOXOVER & COMBmutator	PMXOVER & SIMPmutator	PMXOVER & COMBmutator
PRUEBA	REEMPLAZC	POBLACION	GENERACIO	MUTACION	CRUCE	SEED	MAKESPAM	MAKESPAM	MAKESPAM	MAKESPAM
1	0,86	100	5000	0,091	0,6	0	1072	1080	1078	1086
2	0,86	100	5000	0,091	0,6	1	1054	1051	1103	1070
3	0,86	100	5000	0,091	0,6	2	1035	1074	1050	1056
4	0,86	100	5000	0,091	0,6	3	1082	1083	1052	1095
5	0,86	100	5000	0,091	0,6	4	1068	1072	1055	1076
6	0,86	100	5000	0,091	0,6	5	1192	1175	1053	1095
7	0,86	100	5000	0,091	0,6	6	1021	1055	1048	1038
8	0,86	100	5000	0,091	0,6	7	1161	1062	1040	1028
9	0,86	100	5000	0,091	0,6	8	1036	1057	1093	1071
10	0,86	100	5000	0,091	0,6	9	1068	1091	1097	1008
11	0,86	100	5000	0,091	0,6	10	1094	1123	1164	1107
12	0,86	100	5000	0,091	0,6	11	1044	1049	1069	1127
13	0,86	100	5000	0,091	0,6	12	1055	1097	1051	1076
14	0,86	100	5000	0,091	0,6	13	1087	1029	1105	1020
15	0,86	100	5000	0,091	0,6	14	1161	1033	1105	1116
16	0,86	100	5000	0,091	0,6	15	1223	1057	1152	1033
17	0,86	100	5000	0,091	0,6	16	1119	1077	1107	1147
18	0,86	100	5000	0,091	0,6	17	1049	1052	1069	1083
19	0,86	100	5000	0,091	0,6	18	1026	1060	1007	1087
20	0,86	100	5000	0,091	0,6	19	1026	1085	1008	1067
21	0,86	100	5000	0,091	0,6	20	1032	1084	1109	1095
22	0,86	100	5000	0,091	0,6	21	1053	1068	1021	1124
23	0,86	100	5000	0,091	0,6	22	1014	1039	1048	1082
24	0,86	100	5000	0,091	0,6	23	1037	1143	1125	1160
25	0,86	100	5000	0,091	0,6	24	1042	1024	1077	1083
26	0,86	100	5000	0,091	0,6	25	1081	1013	1040	1101
27	0,86	100	5000	0,091	0,6	26	1046	1079	1039	1131
28	0,86	100	5000	0,091	0,6	27	1033	1106	1079	1001
29	0,86	100	5000	0,091	0,6	28	984	1051	1045	1092
30	0,86	100	5000	0,091	0,6	29	1011	1082	1099	1008
31	0,86	100	5000	0,091	0,6	30	1103	1064	1061	1122
32	0,86	100	5000	0,091	0,6	31	1084	1063	1065	1069
33	0,86	100	5000	0,091	0,6	32	1068	1100	1004	1100
34	0,86	100	5000	0,091	0,6	33	1218	1152	1092	1033
35	0,86	100	5000	0,091	0,6	34	1074	1113	1086	1045
36	0,86	100	5000	0,091	0,6	35	1027	1071	1096	1068
37	0,86	100	5000	0,091	0,6	36	1038	1115	1021	1025
38	0,86	100	5000	0,091	0,6	37	1034	1048	1043	1092
39	0,86	100	5000	0,091	0,6	38	1122	1108	1042	1078
40	0,86	100	5000	0,091	0,6	39	1051	1037	1118	1115
41	0,86	100	5000	0,091	0,6	40	1136	1087	1076	1126
42	0,86	100	5000	0,091	0,6	41	1095	1104	1037	1158
43	0,86	100	5000	0,091	0,6	42	1077	1145	1188	1070
44	0,86	100	5000	0,091	0,6	43	1061	1144	1226	1142
45	0,86	100	5000	0,091	0,6	44	1087	1056	1103	1039
46	0,86	100	5000	0,091	0,6	45	1084	1151	1030	1102
47	0,86	100	5000	0,091	0,6	46	1030	1079	1081	1087
48	0,86	100	5000	0,091	0,6	47	1057	1151	1083	1039
49	0,86	100	5000	0,091	0,6	48	1048	1088	1121	1027
50	0,86	100	5000	0,091	0,6	49	1073	1058	1081	1089
51	0,86	100	5000	0,091	0,6	50	1044	1164	1104	1047
52	0,86	100	5000	0,091	0,6	51	1035	1019	1048	1123
53	0,86	100	5000	0,091	0,6	52	1031	1069	1074	1076
54	0,86	100	5000	0,091	0,6	53	1095	1073	1014	1056
55	0,86	100	5000	0,091	0,6	54	1069	1063	1067	1059
56	0,86	100	5000	0,091	0,6	55	1160	1165	1032	1113
57	0,86	100	5000	0,091	0,6	56	1130	1095	1096	1085
58	0,86	100	5000	0,091	0,6	57	1049	1077	1087	1060
59	0,86	100	5000	0,091	0,6	58	1069	1024	1095	1087
60	0,86	100	5000	0,091	0,6	59	1036	1033	1079	1057
61	0,86	100	5000	0,091	0,6	60	1056	1044	1030	1162
62	0,86	100	5000	0,091	0,6	61	1095	1140	1080	1084
63	0,86	100	5000	0,091	0,6	62	1083	1073	1060	1046
64	0,86	100	5000	0,091	0,6	63	1038	1044	1098	1032
65	0,86	100	5000	0,091	0,6	64	1109	1062	1103	1087
66	0,86	100	5000	0,091	0,6	65	1106	1027	1141	1090
67	0,86	100	5000	0,091	0,6	66	1112	1160	1147	1092
68	0,86	100	5000	0,091	0,6	67	1113	1090	1062	1067
69	0,86	100	5000	0,091	0,6	68	1092	1059	1023	1083
70	0,86	100	5000	0,091	0,6	69	1083	1057	1097	1060
71	0,86	100	5000	0,091	0,6	70	1095	1040	1078	1097
72	0,86	100	5000	0,091	0,6	71	1077	1157	1049	1014
73	0,86	100	5000	0,091	0,6	72	1014	1055	1109	1090
74	0,86	100	5000	0,091	0,6	73	1068	1050	1131	1013
75	0,86	100	5000	0,091	0,6	74	1090	1097	1184	1099
76	0,86	100	5000	0,091	0,6	75	1106	1091	1093	1104
77	0,86	100	5000	0,091	0,6	76	1090	1088	1095	1047
78	0,86	100	5000	0,091	0,6	77	1132	1039	1178	1105
79	0,86	100	5000	0,091	0,6	78	1087	1112	1166	1114
80	0,86	100	5000	0,091	0,6	79	1163	1159	1107	1128
81	0,86	100	5000	0,091	0,6	80	1140	1116	1047	1069
82	0,86	100	5000	0,091	0,6	81	1115	1091	1047	1052
83	0,86	100	5000	0,091	0,6	82	1135	1115	1053	1071
84	0,86	100	5000	0,091	0,6	83	1053	1162	1089	1024
85	0,86	100	5000	0,091	0,6	84	1056	1209	1045	1119
86	0,86	100	5000	0,091	0,6	85	1072	1032	1105	1054
87	0,86	100	5000	0,091	0,6	86	1160	1101	1116	1054
88	0,86	100	5000	0,091	0,6	87	1112	1063	1049	1032
89	0,86	100	5000	0,091	0,6	88	1155	1039	1082	1130
90	0,86	100	5000	0,091	0,6	89	1088	996	1061	1030
91	0,86	100	5000	0,091	0,6	90	1102	1098	1093	1164
92	0,86	100	5000	0,091	0,6	91	1210	1065	1093	1028
93	0,86	100	5000	0,091	0,6	92	1093	1083	1071	1029
94	0,86	100	5000	0,091	0,6	93	1068	1071	1074	1124
95	0,86	100	5000	0,091	0,6	94	1050	1122	1047	1081
96	0,86	100	5000	0,091	0,6	95	1078	1079	1099	1067
97	0,86	100	5000	0,091	0,6	96	1151	1081	1098	1103
98	0,86	100	5000	0,091	0,6	97	1033	1116	1028	1124
99	0,86	100	5000	0,091	0,6	98	1111	1060	1014	1035
100	0,86	100	5000	0,091	0,6	99	1129	1126	1059	1099
101	0,86	100	5000	0,091	0,6	100	1128	1085	1117	1095

Anexo iv: *Pruebas de la instancia LA24*

							JOXOVER & SIMPmutator	JOXOVER & COMBmutator	PMXOVER & SIMPmutator	PMXOVER & COMBmutator
PRUEBA	REEMPLAZC	POBLACION	GENERACIO	MUTACION	CRUCE	SEED	MAKESPAM	MAKESPAM	MAKESPAM	MAKESPAM
1	0,86	100	5000	0,091	0,6	0	1447	1410	1433	1433
2	0,86	100	5000	0,091	0,6	1	1531	1420	1364	1429
3	0,86	100	5000	0,091	0,6	2	1366	1487	1420	1346
4	0,86	100	5000	0,091	0,6	3	1407	1453	1418	1429
5	0,86	100	5000	0,091	0,6	4	1508	1473	1513	1380
6	0,86	100	5000	0,091	0,6	5	1536	1574	1438	1444
7	0,86	100	5000	0,091	0,6	6	1564	1450	1370	1377
8	0,86	100	5000	0,091	0,6	7	1441	1446	1426	1421
9	0,86	100	5000	0,091	0,6	8	1404	1428	1557	1468
10	0,86	100	5000	0,091	0,6	9	1462	1394	1408	1505
11	0,86	100	5000	0,091	0,6	10	1391	1384	1426	1496
12	0,86	100	5000	0,091	0,6	11	1672	1413	1463	1476
13	0,86	100	5000	0,091	0,6	12	1441	1461	1435	1459
14	0,86	100	5000	0,091	0,6	13	1415	1378	1470	1406
15	0,86	100	5000	0,091	0,6	14	1478	1410	1424	1422
16	0,86	100	5000	0,091	0,6	15	1525	1576	1372	1408
17	0,86	100	5000	0,091	0,6	16	1377	1558	1452	1555
18	0,86	100	5000	0,091	0,6	17	1431	1513	1470	1485
19	0,86	100	5000	0,091	0,6	18	1399	1431	1405	1428
20	0,86	100	5000	0,091	0,6	19	1431	1422	1412	1384
21	0,86	100	5000	0,091	0,6	20	1481	1482	1444	1366
22	0,86	100	5000	0,091	0,6	21	1477	1550	1411	1363
23	0,86	100	5000	0,091	0,6	22	1421	1432	1485	1415
24	0,86	100	5000	0,091	0,6	23	1476	1398	1473	1369
25	0,86	100	5000	0,091	0,6	24	1446	1564	1533	1454
26	0,86	100	5000	0,091	0,6	25	1389	1341	1448	1377
27	0,86	100	5000	0,091	0,6	26	1399	1490	1408	1386
28	0,86	100	5000	0,091	0,6	27	1504	1443	1536	1388
29	0,86	100	5000	0,091	0,6	28	1540	1540	1484	1478
30	0,86	100	5000	0,091	0,6	29	1340	1503	1499	1549
31	0,86	100	5000	0,091	0,6	30	1431	1438	1411	1377
32	0,86	100	5000	0,091	0,6	31	1381	1510	1435	1433
33	0,86	100	5000	0,091	0,6	32	1414	1478	1383	1415
34	0,86	100	5000	0,091	0,6	33	1403	1404	1431	1467
35	0,86	100	5000	0,091	0,6	34	1397	1533	1358	1359
36	0,86	100	5000	0,091	0,6	35	1338	1403	1384	1413
37	0,86	100	5000	0,091	0,6	36	1509	1537	1386	1458
38	0,86	100	5000	0,091	0,6	37	1425	1442	1444	1415
39	0,86	100	5000	0,091	0,6	38	1543	1555	1439	1436
40	0,86	100	5000	0,091	0,6	39	1522	1448	1470	1400
41	0,86	100	5000	0,091	0,6	40	1567	1528	1482	1536
42	0,86	100	5000	0,091	0,6	41	1485	1435	1453	1442
43	0,86	100	5000	0,091	0,6	42	1393	1427	1416	1390
44	0,86	100	5000	0,091	0,6	43	1410	1472	1363	1467
45	0,86	100	5000	0,091	0,6	44	1432	1510	1466	1441
46	0,86	100	5000	0,091	0,6	45	1508	1473	1445	1434
47	0,86	100	5000	0,091	0,6	46	1422	1526	1389	1434
48	0,86	100	5000	0,091	0,6	47	1407	1433	1405	1500
49	0,86	100	5000	0,091	0,6	48	1464	1466	1411	1353
50	0,86	100	5000	0,091	0,6	49	1386	1410	1399	1471
51	0,86	100	5000	0,091	0,6	50	1374	1689	1408	1466
52	0,86	100	5000	0,091	0,6	51	1449	1492	1402	1442
53	0,86	100	5000	0,091	0,6	52	1555	1443	1432	1447
54	0,86	100	5000	0,091	0,6	53	1333	1514	1341	1464
55	0,86	100	5000	0,091	0,6	54	1462	1379	1391	1399
56	0,86	100	5000	0,091	0,6	55	1333	1440	1348	1403
57	0,86	100	5000	0,091	0,6	56	1429	1419	1382	1464
58	0,86	100	5000	0,091	0,6	57	1522	1426	1447	1484
59	0,86	100	5000	0,091	0,6	58	1470	1477	1428	1428
60	0,86	100	5000	0,091	0,6	59	1412	1408	1401	1478
61	0,86	100	5000	0,091	0,6	60	1460	1366	1367	1445
62	0,86	100	5000	0,091	0,6	61	1465	1475	1393	1471
63	0,86	100	5000	0,091	0,6	62	1495	1504	1472	1443
64	0,86	100	5000	0,091	0,6	63	1383	1432	1437	1541
65	0,86	100	5000	0,091	0,6	64	1428	1408	1559	1339
66	0,86	100	5000	0,091	0,6	65	1539	1548	1458	1386
67	0,86	100	5000	0,091	0,6	66	1472	1430	1442	1361
68	0,86	100	5000	0,091	0,6	67	1438	1393	1432	1323
69	0,86	100	5000	0,091	0,6	68	1374	1453	1413	1482
70	0,86	100	5000	0,091	0,6	69	1434	1411	1436	1474
71	0,86	100	5000	0,091	0,6	70	1435	1384	1494	1408
72	0,86	100	5000	0,091	0,6	71	1383	1456	1401	1487
73	0,86	100	5000	0,091	0,6	72	1499	1496	1397	1383
74	0,86	100	5000	0,091	0,6	73	1395	1465	1427	1382
75	0,86	100	5000	0,091	0,6	74	1639	1361	1501	1422
76	0,86	100	5000	0,091	0,6	75	1426	1395	1392	1492
77	0,86	100	5000	0,091	0,6	76	1571	1478	1458	1447
78	0,86	100	5000	0,091	0,6	77	1486	1439	1438	1418
79	0,86	100	5000	0,091	0,6	78	1444	1443	1536	1489
80	0,86	100	5000	0,091	0,6	79	1570	1565	1351	1355
81	0,86	100	5000	0,091	0,6	80	1501	1467	1599	1542
82	0,86	100	5000	0,091	0,6	81	1415	1489	1489	1555
83	0,86	100	5000	0,091	0,6	82	1376	1485	1387	1421
84	0,86	100	5000	0,091	0,6	83	1480	1373	1454	1424
85	0,86	100	5000	0,091	0,6	84	1411	1408	1344	1459
86	0,86	100	5000	0,091	0,6	85	1400	1398	1377	1448
87	0,86	100	5000	0,091	0,6	86	1386	1390	1473	1417
88	0,86	100	5000	0,091	0,6	87	1392	1469	1529	1421
89	0,86	100	5000	0,091	0,6	88	1539	1433	1376	1493
90	0,86	100	5000	0,091	0,6	89	1450	1506	1526	1459
91	0,86	100	5000	0,091	0,6	90	1427	1423	1376	1368
92	0,86	100	5000	0,091	0,6	91	1422	1482	1396	1478
93	0,86	100	5000	0,091	0,6	92	1459	1564	1417	1404
94	0,86	100	5000	0,091	0,6	93	1499	1450	1384	1475
95	0,86	100	5000	0,091	0,6	94	1384	1455	1378	1416
96	0,86	100	5000	0,091	0,6	95	1445	1396	1403	1473
97	0,86	100	5000	0,091	0,6	96	1399	1404	1383	1464
98	0,86	100	5000	0,091	0,6	97	1395	1415	1427	1386
99	0,86	100	5000	0,091	0,6	98	1388	1445	1370	1443
100	0,86	100	5000	0,091	0,6	99	1510	1404	1401	1437
101	0,86	100	5000	0,091	0,6	100	1447	1396	1431	1402

Anexo v: *Pruebas de la instancia LA27*

BIBLIOGRAFIA

[1] Instituto tecnológico de massachussets. Web site: http://lancet.mit.edu/ga/, 2009.

[2] Or-library librería de investigación de operaciones. Web site: http://people.brunel.ac.uk/ mastjjb/jeb/orlib/jobshopinfo.html, 2009.

[3] J. Balas. The shifting bottleneck procedure for job shop scheduling. pags. 394–401, Management Science Vol 34, USA, 1998.

[4] C. Blum. Meta-heuristics in combinatorial optimization: Overview and conceptual comparison. In *Ann Discrete Math.*, pags. 269–308, ACM Comput Surv, 2003.

[5] C. Blum. Hybrid metaheuristics. studies in computational intelligence. In *vol 114*, pags. 17–37, 2008.

[6] M. Bramlette. Initialization, mutation and selection methods in genetic algorithms for function optimization. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, pags. 100–107, 1991.

[7] P. Brucker. Complex scheduling. springer science and business. pags. 180–210, Media Inc. New York, 2006.

[8] P. Brucker. Scheduling algorithms fifht edition. springer science and business. pags. 170–190, Media Inc. New York, 2007.

[9] D. F. E.S. Roscoe. Organization for production. In *fifth edition, Richard D. Irwin Inc. Homewood Illinois*, 15 pags., 1971.

[10] J. Herrmann. Handbook of production scheduling. In *Handbook of Production Scheduling*, pags. 2–20, Springer Science and Business Media Inc., New York, 2006.

[11] C. J.F. Blackstone, J.H. Spencer. Apics dictionary. In *American Production and Inventory Control Society*, pags 17-35, Virginia, 2001.

[12] R. L. LaForge. Manufacturing scheduling and supply chain integration: a survey of current practice. In *American Production and Inventory Control Society*, pags. 1–16, Falls Church, Virginia, 1998.

[13] J. Lenstra. Computational complexity of discrete optimization problems. In *Ann Discrete Math.*, pags. 121–140, 1979.

[14] M. Mitchell. An introduction to genetic algorithms complex adaptive systems. mit press. In *Massachusetts Institute of Techology. Cambridge, Massachusetts*, 12-69 p, 1996.

[15] O. Wight. Production and inventory management in the computer age. In *Van Nostrand Reinhold Company*, pag 24, New York, 1984.

[16] F. Xhafa. Metaheuristics for scheduling in industrial and manufacturing applications.