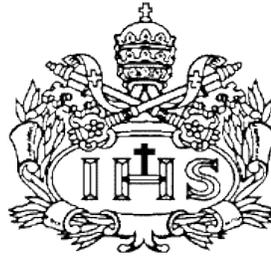


PONTIFICIA UNIVERSIDAD JAVERIANA



TRABAJO DE GRADO

**“EVALUACIÓN DE LA EFICIENCIA DE ALGORITMOS GENÉTICOS Y SISTEMAS
MULTIAGENTES PARA EL PROBLEMA DE PROGRAMACIÓN DE LA PRODUCCIÓN
EN UNA MÁQUINA ($1/prmp/\sum W_jT_j$)”**

Estudiantes

DANIEL PÁEZ PARRA

ANDREA CONSUELO VARGAS MÉNDEZ

Profesor Director Asignado

INGENIERO JOSÉ FERNANDO JIMÉNEZ GORDILLO

Bogotá D.C.

Abril de 2013

TABLA DE CONTENIDO

I. INTRODUCCIÓN.....	5
PRELIMINARES.....	6
I. PLANTEAMIENTO DEL PROBLEMA.....	7
II. JUSTIFICACIÓN DEL PROBLEMA	8
III. OBJETIVOS	10
IV. ALCANCE Y RESTRICCIONES.....	11
V. RECURSOS.....	13
CAPÍTULO I.....	14
PROBLEMA: PROGRAMACIÓN DE LA PRODUCCIÓN PARA UNA MÁQUINA	14
I. ANTECEDENTES	15
II. MARCO TEÓRICO	26
III. DEFINICIÓN DE LAS PERTURBACIONES.....	36
CAPÍTULO 2.....	39
ALGORITMO DE SOLUCIÓN BASADO EN UN MÉTODO TRADICIONAL DE OPTIMIZACIÓN (ALGORITMO GENÉTICO).....	39
I. DESCRIPCIÓN DEL ALGORITMO GENÉTICO UTILIZADO.....	40
II. DEFINICIÓN OPERACIONES DEL ALGORITMO GENÉTICO	42
III. DESCRIPCIÓN DE LA SOLUCIÓN	52
CAPÍTULO 3.....	59
ALGORITMO DE SOLUCIÓN BASADO EN SISTEMAS MULTIAGENTES	59
I. DESCRIPCIÓN DEL SISTEMA MULTIAGENTE.....	60
II. DEFINICIÓN OPERACIONES DEL SISTEMA MULTIAGENTE	65
III. DESCRIPCIÓN DE LA SOLUCIÓN	70
CAPÍTULO 4.....	78
METODOLOGÍA DE COMPARACIÓN.....	78
I. DESCRIPCIÓN DE LA METODOLOGÍA	79
II. DESCRIPCIÓN DE LA SOLUCIÓN Y ANÁLISIS DE RESULTADOS.....	82

CAPÍTULO 5	84
EVALUACIÓN Y ANÁLISIS DEL SISTEMA COMPLEJO PARA EL PROBLEMA DE PROGRAMACIÓN DE LA PRODUCCIÓN EN UNA MÁQUINA (1/ prmp/wjTj)”	84
I. CONCLUSIONES.....	85
II. RECOMENDACIONES	87
BIBLIOGRAFÍA	88
ANEXOS	95

TABLA DE CONTENIDO GRÁFICOS

Gráfico 1: Metodología para el análisis de decisiones. (Castillo Hernández, 2006).....	20
Gráfico 2: Modelo básico de interacción entre agentes y entorno (Wooldridge, 2002)	28
Gráfico 3: Esquematación Single Machine Scheduling. Realizado por Daniel Páez y Andrea Vargas.....	35
Gráfico 4: Resultados ranqueo. Elaborado por Daniel Páez y Andrea Vargas	46
Gráfico 5: Algoritmo genético para programar la producción con inclusión de perturbaciones. Realizado por Daniel Páez y Andrea Vargas.....	47
Gráfico 6: Cantidad de corridas del algoritmo genético. Elaborado por Daniel Páez y Andrea Vargas.....	47
Gráfico 7: Procesos realizados por el algoritmo genético. Realizado por Daniel Páez y Andrea Vargas.....	49
Gráfico 8: Procesos realizados para programar la producción con perturbaciones. Elaborado por Daniel Páez y Andrea Vargas.....	51
Gráfico 9: Frecuencia de la tardanza total ponderada para la Instancia 1. Elaborado por Daniel Páez y Andrea Vargas	53
Gráfico 10: Frecuencia del tiempo de ejecución. Elaborado por Daniel Páez y Andrea Vargas	55
Gráfico 12: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas	57
Gráfico 13: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas	58
Gráfico 14: Cuatro Formas básicas de arquitecturas de control (Dilts, Boyd, & Whorms, 1991)	62
Gráfico 15: Descripción de la perspectiva estructural utilizada. Realizado por Daniel Páez y Andrea Vargas	65
Gráfico 16: Campo de fuerza. Realizado por Daniel Páez y Andrea Vargas	66
Gráfico 18: Procesos realizados para programar la producción con perturbaciones. Elaborado por Daniel Páez y Andrea Vargas.....	69

Gráfico 19: Frecuencia de la tardanza total ponderada para la Instancia 1. Elaborado por Daniel Páez y Andrea Vargas	72
Gráfico 21: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas	76
Gráfico 22: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas	77
Gráfico 23: Modelo general de un proceso o sistema (Montgomery, 2004).....	79

TABLA DE CONTENIDO TABLAS

Tabla 1: Revisión bibliográfica para la solución del problema de programación de la producción en una máquina (Elaborado por Daniel Páez y Andrea Vargas)	25
Tabla 2: Descripción de las meta-heurísticas más utilizadas. Elaborado por Daniel Páez y Andrea Vargas.....	27
Tabla 3: Arquitectura para Agentes Inteligentes (Wooldridge, 2002).....	28
Tabla 4: Entorno de la máquina (α) (Pinedo, 2008)	31
Tabla 5: Restricciones y Constantes (β) (Pinedo, 2008)	33
Tabla 6: Funciones objetivo y medidas de desempeño (γ) (Baker & Trietsch, 2009)	35
Tabla 7: Tipos de perturbaciones. Elaborado por Daniel Páez y Andrea Vargas	37
Tabla 8: Posibles combinaciones criterios de selección, reproducción y mutación. Elaborado por Daniel Páez y Andrea Vargas.....	44
Tabla 9: Promedio tardanza total ponderada y tiempo de ejecución para cada una de las instancias (luego de 100 corridas). Elaborado por Daniel Páez y Andrea Vargas	54
Tabla 10: Resumen de las características de la arquitectura de control, ventajas y desventajas (Dilts, Boyd, & Whorms, 1991)	62
Tabla 11: Promedio tardanza total ponderada y tiempo de ejecución para cada una de las instancias (luego de 100 corridas). Elaborado por Daniel Páez y Andrea Vargas	74
Tabla 12: Cuadro de Anova. (Montgomery, 2004)	81
Tabla 13: Tabla Anova para la tardanza total ponderada de la instancia 125. Realizado por Daniel Páez y Andrea Vargas	82
Tabla 14: Tabla Anova para la tardanza total ponderada de la instancia 125. Realizado por Daniel Páez y Andrea Vargas	82
Tabla 15: Ranking de algoritmos. Elaborado por Daniel Paez y Andrea Vargas	98

I. INTRODUCCIÓN

El afán por satisfacer las necesidades humanas y la competitividad de los mercados globalizados, ha impuesto nuevos retos a las industrias del sector productivo, lo cual hace necesario, que a diario se tomen decisiones para articular los procesos de manufactura a las necesidades y exigencias del mercado actual, teniendo en cuenta el dinamismo constante en los procesos productivos.

Adicionalmente, la toma de decisiones implica un proceso de planeación y evaluación, con el fin de determinar cuáles estrategias se van a llevar a cabo dependiendo de los objetivos enmarcados por la estrategia organizacional. La planeación de la producción en la industria, requiere organizar los trabajos de una manera óptima para aprovechar al máximo los recursos. Para esto, la industria cuenta con herramientas analíticas especializadas conocidas comúnmente dentro del campo de la investigación de operaciones como scheduling.

Con este trabajo de grado, se pretende dar solución a un problema de programación de la producción en una máquina, mediante un método tradicional (meta-heurísticas) y un sistema Multiagente, en los que se contemplen las perturbaciones que pueden sufrir los sistemas complejos de manufactura en un ambiente real. En efecto, se pretende minimizar las penalizaciones en las que se pueda incurrir por el incumplimiento en las fechas de entrega establecidas para las diferentes tareas.

El propósito es comparar los resultados arrojados por los dos métodos, con el fin de determinar cuál método es mejor para implementar en el sector industrial, de tal forma que no requiera gran tiempo de procesamiento computacional y se pueda llegar a respuestas factibles y aceptables.

Es por esto que los sistemas de programación de la producción representan la mejor forma de lograr un desarrollo sostenido, en el que se pueda alcanzar un mayor nivel de productividad y competitividad en el sector productivo.

PRELIMINARES

I. PLANTEAMIENTO DEL PROBLEMA

Tras el análisis de los nuevos requerimientos del mercado global, del sector industrial, y la revisión bibliográfica del problema de Single Machine Scheduling, se observa que se pueden utilizar diferentes modelos que se adaptan a las especificaciones del problema. Como respuesta a esto, los investigadores se preocupan por garantizar que estas herramientas de planeación representen la realidad de la situación, hagan una asignación de recursos y tareas de manera eficaz y eficiente y, que la respuesta computacional sea lo más eficiente posible.

Teniendo en cuenta esto, se ha encontrado la oportunidad de realizar un trabajo de comparación que dé solución al problema de Single Machine Scheduling mediante el uso de meta-heurísticas y los sistemas Multiagente. En primer lugar las metaheurísticas, al ser modelos más de planeación que de ejecución, no están diseñadas para soportar perturbaciones, por lo que se encuentra la oportunidad de desarrollar un sistema Multiagente. Sin embargo, este sistema Multiagente aunque acepta perturbaciones, puede lograr óptimos localizados y no óptimos globales, por lo que se dificulta poder encontrar una solución óptima para los procesos de manufactura y distribución simultáneamente.

En consecuencia, se pretende determinar los cambios de eficiencia en el problema de Single Machine Scheduling, de acuerdo a las perturbaciones que sufre el sistema. Se plantea la siguiente pregunta de investigación ¿Es posible que en el problema de Single Machine Scheduling, sea más eficiente el uso de meta-heurísticas que la utilización de sistemas Multiagentes al momento de considerar perturbaciones dentro del sistema?

II. JUSTIFICACIÓN DEL PROBLEMA

El desarrollo y seguimiento de una metodología para el análisis de decisiones hacen parte del proceso de planeación de una organización. La globalización ha hecho que las organizaciones tiendan a desarrollar estrategias de planeación a lo largo de la cadena de abastecimiento y a adoptar modelos que le permitan optimizar sus procesos productivos; la planeación se debe entender como el esfuerzo sistemático y formal de una compañía para establecer sus, objetivos y estrategias básicas, con el fin de desarrollar planes detallados para poner en práctica las políticas y (Steiner, 2004). (Amaya Amaya, 2007) definen la planeación como el proceso mediante el cual quienes toman decisiones en una organización obtienen, procesan y analizan información interna y externa, con el fin de evaluar la situación presente de la empresa, así, como su nivel de competitividad, con el propósito de anticipar y decidir sobre el direccionamiento de la institución hacia el futuro.

Otros autores como (Pinedo, 2008), afirman que la planeación de la producción es un proceso de toma de decisión que tiene como función asignar recursos a tareas en un periodo de tiempo, con el fin de optimizar uno o más objetivos estratégicos de la empresa. Esta toma de decisiones, debe tener en cuenta que dentro de un proceso productivo existe un gran número de actividades que deben ser controladas para no incurrir en costos de demora innecesarios. Una adecuada planeación en la producción puede garantizar una asignación óptima de recursos para la realización de las diferentes tareas, teniendo en cuenta que cada una de estas deben cumplir con un tiempo específico de procesamiento, tanto en una producción continua (donde se asignan metas de producción por semana o diarias, partiendo del supuesto de que todas las máquinas utilizadas en el proceso están perfectamente sincronizadas (Kanawaty, 1996)) como en una producción discontinua (donde se debe determinar el orden de las operaciones correspondiente a cada producto o pieza y el tiempo necesario para cada operación, con inclusión de los tiempos de preparación (Kanawaty, 1996)).

Por esta razón, si el sector industrial quiere sobrevivir en un mercado global y competitivo debe planear y estructurar sus procesos de manufactura y, configurar las estrategias de productividad y competitividad, con el fin de buscar la eficiencia operacional y responder a las necesidades del mercado dinámico (Gunasekaran & Ngai, 2012). Sin embargo, para lograrlo, debe competir basado en las métricas como el costo de manufactura, la calidad del producto, la flexibilidad de la operación, la confiabilidad del proceso y la capacidad de respuesta (Gunasekaran & Ngai, 2012).

La presente investigación se propone debido a la importancia que tiene programar la producción en una máquina mediante el uso de meta-heurísticas y sistemas Multiagentes con el fin de determinar, mediante una metodología en la que se comparen

los resultados obtenidos, cuál método es más eficiente teniendo en cuenta las ventajas y desventajas que cada uno de estos presenta.

Por un lado, se tiene el uso de la modelación tradicional de la Investigación de operaciones, por medio de los algoritmos de búsqueda, específicamente las meta-heurísticas, con la cual se provee un análisis metodológico cuantitativo y cualitativo para soportar la toma de decisiones y alcanzar la optimización en el proceso, mediante la generación de unos lineamientos adecuados en la toma de decisiones que se caracteriza por contar con metodologías centralizadas, jerárquicas y globales (Trentesaux, 2009), en las que el sistema se optimiza por mecanismos centrales con reglas generales en búsqueda de la optimización global del sistema, por lo que un número creciente de industrias quieren ahora, sistemas de control que proporcionen soluciones satisfactorias, adaptables y resistentes en lugar de soluciones óptimas que deben cumplir con varias hipótesis (Trentesaux, 2009). Teniendo en cuenta esto, el uso de las metaheurísticas se quedaría corto a la hora de abordar un problema de planeación de la producción en una máquina, pues la modelación tradicional no responde a posibles perturbaciones en el sistema, como lo exige el comportamiento dinámico del mercado.

Por otro lado, existen los modelos computacionales basados en la simulación e interacción de agentes: los Sistemas Multiagente. Estos son un conjunto de agentes inteligentes (autónomos, independientes, perceptivos, racionales y funcionales) que representan cualquier tipo de entidad dentro de un sistema y se organizan bajo unos principios de comunidad que buscan un objetivo específico (Mařika & Lařanskýa, 2007) (Lee & Kimb, 2008), es decir, bajo un sistema descentralizado y distribuido en agentes, el cual permite perturbaciones y la toma de decisiones, por medio de interacciones y negociaciones, para lograr un ambiente de cooperación, colaboración y capacidad de aprendizaje al momento de solucionar problemas (Nwana & Ndumu, 1997). Sin embargo, aunque la modelación se enfatiza en la agilidad y reconfiguración del sistema, es débil, pues permite la existencia de un comportamiento miope (Trentesaux, 2009), ya que debido a la autonomía de cada agente, el sistema general puede proporcionar operaciones óptimas por entidad y no se lograría una optimización total del sistema (Trentesaux, 2009) lo cual dificultaría la optimización integral del sistema.

Finalmente, la resolución de este problema puede arrojar datos importantes que permitan determinar cuál método, tradicional o Multiagente, es mejor a la hora de programar la producción en una máquina en una situación real, dado que pueden ocurrir eventos imprevistos que hagan necesaria la reprogramación de la producción para seguir garantizando la eficiencia y eficacia de los procesos. Adicionalmente, por las características de los métodos de solución, no se espera obtener un óptimo global sino un óptimo local satisfactorio, por lo cual, este tipo de problemas son susceptibles a mejoras y el trabajo propuesto puede servir como punto de partida para posteriores investigaciones.

III. OBJETIVOS

OBJETIVO GENERAL

Realizar la comparación y evaluación de dos tipos de modelación de sistemas complejos (algoritmo genético y sistemas Multiagentes) para el problema de programación de la producción de una máquina con la finalidad de establecer cuál método de solución es más eficiente

OBJETIVOS ESPECÍFICOS

1. Definir un sistema complejo de manufactura, específicamente en la programación de la producción para una sola máquina, que considere perturbaciones reales específicas.
2. Diseñar un algoritmo genético que resuelva el problema de programación de la producción en una sola máquina y tenga un manejo pertinente a las perturbaciones del sistema.
3. Diseñar y desarrollar un sistema Multiagente que sea capaz de modelar un escenario en el que existan perturbaciones inesperadas que alteran la programación de la producción en una sola máquina.
4. Diseñar una metodología de comparación que permita evaluar los sistemas de modelación desarrollados (métodos tradicionales y sistemas Multiagentes) para solucionar el problema de programación de la producción en una sola máquina.
5. Evaluar y analizar el sistema complejo de manufactura, por medio de una metodología de comparación, según los resultados obtenidos de la modelación tradicional y Multiagentes anteriormente desarrollada.

IV. ALCANCE Y RESTRICCIONES

En este trabajo de grado se pretende realizar un estudio, que establecerá por medio de una metodología de comparación desarrollada dentro del alcance del proyecto, cuál modelo es mejor para simular el problema de programación de la producción en una máquina entre un método tradicional de optimización (algoritmos genéticos) y un sistema Multiagente.

Se pretende realizar un estudio preliminar de los diferentes métodos tradicionales de solución a este problema estudiados previamente, con el fin de determinar, según expertos, cuál de los modelos presenta un mejor desempeño y es óptimo. Este análisis tiene en cuenta como punto de partida la revisión de la literatura. Observando el comportamiento y algunos de los requerimientos prácticos que se han utilizado y tenido en cuenta para el uso de cada método de solución, las características propias de cada problema son: función objetivo y restricciones.

Cabe resaltar que se tomará como caso de estudio la programación de producción en una máquina, debido a que éste caso es el más sencillo y los demás casos suelen ser analizados desde una máquina antes de generalizar a más máquinas.

El análisis estará enfocado desde la perspectiva real de una empresa en la cual, la programación de la producción suele presentar perturbaciones, estas se refieren a tareas inesperadas, que alteran el orden prestablecido y pueden causar demoras en los tiempos de entrega, cuellos de botellas, entre otros. El trabajo tendrá en cuenta un número finito de perturbaciones aleatorias y un número de tareas finitas a ser realizadas por una máquina en la cual las características se definirán durante el desarrollo del mismo.

La metodología de comparación es desarrollada dentro del trabajo, con el fin de seleccionar una que se acople y sea relevante dentro del desarrollo del mismo, con base en los resultados arrojados por el método tradicional de optimización y el sistema Multiagente respectivamente, y los entornos definidos más adelante tales como, número de tareas a realizar, perturbaciones, etc.

Los resultados de este trabajo de grado serán de utilidad para posteriores estudios que conlleven al diseño de nuevas estructuras, esquemas y modelos para la programación de la producción en una máquina, que respondan a las necesidades de fomento de los niveles de competencia en sistemas complejos de manufactura.

Por otro lado, debido a que el presente trabajo se sustenta en la investigación, se pudieron establecer algunas restricciones durante su ejecución. En primer lugar, no es posible asegurar que las respuestas alcanzadas con la meta-heurística, sean las mejores (óptimas globales) ni siempre las mismas, debidos a la ejecución del algoritmo con las mismas variables de entrada.

En ese caso, no es posible afirmar, más allá de una metodología de comparación con el sistema Multiagente, que la solución es buena o mala. Por lo cual es necesario tener medidas de desempeño e indicadores que permitan determinar qué tan buena o mala es respectivamente.

V. RECURSOS

Los recursos utilizados en este trabajo (con su respectiva descripción) fueron los siguientes:

1. **Recursos Bibliográficos:** Base de datos de la Pontificia Universidad Javeriana, revistas y recursos bibliográficos: Se requieren como base para definir el estado del arte y el marco teórico en el cual se puede incorporar un trabajo investigativo acerca de la programación de la producción.
2. **Recurso Computacional:** Se divide en dos aspectos, físico y lógico. En el primero, se hace referencia a la necesidad de un computador con características de procesamiento mínimas para la ejecución de las instrucciones lógicas. El segundo, se refiere a los programas requeridos para la programación, visualización y registro de resultados de cada una de las etapas propuestas.
3. **Recurso Humano:** Finalmente, para el desarrollo del trabajo se requiere de dos estudiante de Ingeniería Industrial con habilidades en la programación, además del director del proyecto y un estudiante de doctorado de la Universidad de Valenciennes, Francia, personas expertas en el tema de Scheduling y Multiagente, respectivamente, que brindaron apoyo y orientación durante la investigación.

CAPÍTULO I

PROBLEMA: PROGRAMACIÓN DE LA PRODUCCIÓN PARA UNA MÁQUINA

I. ANTECEDENTES

Los avances en la tecnología de la información, la globalización de los mercados, las operaciones descentralizadas, y la conciencia cada vez mayor hacia las preocupaciones ambientales, han obligado a las industrias a replantear tanto sus estrategias de productividad y calidad como las técnicas utilizadas (Gunasekaran & Ngai, 2012), y a rediseñar las operaciones generales de administración que, junto con las expectativas de los clientes, han ocasionado la integración de los procesos de negocio en la empresa y con las empresas extendidas o los proveedores, (Gunasekaran & Ngai, 2012).

El mercado mundial está requiriendo productos de alta calidad, bajo costo, altamente personalizables y con ciclos cortos de producción (Leitão, 2009); por esto, se han tenido que implementar una serie de estrategias, que incluyen just-in-time (JIT), gestión de calidad total (TQM), sistemas flexibles de manufactura (FMS), fabricación integrada por ordenador (CIM), la fabricación ágil (AM), la producción ajustada (LP), la reingeniería de procesos de negocio (BPR), fabricación de respuesta rápida (QRM) y gestión de la cadena de suministro (SCM) (Gunasekaran & Ngai, 2012), para satisfacer las necesidades y sobrevivir en un mercado competitivo.

De esta manera, las empresas se ven obligadas a competir basadas en las métricas de competencia como el costo de manufactura, la calidad del producto, la flexibilidad de la operación, la confiabilidad del proceso y la capacidad de respuesta rápida (Gunasekaran & Ngai, 2012), ya que, una vez que las necesidades básicas de los clientes en términos de calidad y precio se mostraron satisfechas, comenzaron a esperar que sus necesidades individuales fueran satisfechas a través de la personalización en masa. En respuesta, las empresas comenzaron a implementar la fabricación integrada por ordenador (CIM), el sistema de fabricación flexible (FMS), y las estrategias ágiles de fabricación y tecnologías (Gunasekaran & Ngai, 2012).

Dadas las circunstancias y los nuevos requerimientos de las industrias, se han comenzado a producir, adicionalmente, importantes transformaciones conceptuales y metodológicas ligadas íntimamente al estudio de los fenómenos no lineales (Miramontes, 1999). Como parte de esta nueva visión, se ha puesto en evidencia que, en los sistemas complejos existen diversos eventos (perturbaciones) que surgen espontáneamente, a partir de interacciones entre los elementos que lo constituyen (Miramontes, 1999), cuyo estudio no sólo rompe las barreras entre la física, la química y la biología, sino de la psicología, la sociología, la economía y antropología (Bar-yam, 2003).

Los sistemas complejos, entonces, están formados por un conjunto grande de componentes individuales que interactúan entre sí y que comparten una estructura interconectada, por medio de la cual pueden intercambiar información y modificar los estados internos (Miramontes, 1999) que se ven afectados por la presencia de

fluctuaciones y la ocurrencia de perturbaciones. Tales sistemas pueden ser estructuralmente simples, aunque tal simplicidad no impide que exhiban comportamientos dinámicos diversos y no triviales (Miramontes, 1999). El proceso de interacciones puede generar comportamientos colectivos y globales (Bar-yam, 2003); es decir, conductas que no están definidas en los elementos individuales, pero que emergen como un proceso colectivo y que no pueden ser reducidas ni explicadas tomando aisladamente a los elementos constituyentes (Miramontes, 1999).

A pesar de que hemos pasado de la era industrial a la era de la información, la industria manufacturera sigue siendo una parte importante de la economía global. (Fowler & Rose, 2004). Dada la complejidad de los nuevos sistemas, las empresas han tenido que modificar la forma en que administran sus operaciones e interactúan con otras organizaciones (Gaither & Frazier, 2003). Una mejor dirección de las operaciones en la organización puede agregarle un valor sustancial, mejorando su competitividad y su rentabilidad a largo plazo (Gaither & Frazier, 2003). Para esto, la gerencia de la producción, concentra su atención en lograr hacer las cosas del modo más compatible con los objetivos de la organización, usando medidas de efectividad orientadas hacia el proceso y sistemas en su naturaleza, en combinación con medidas de eficiencia orientadas hacia las operaciones (Valencia, 2003).

Al orientar la efectividad hacia el proceso y el sistema, es necesario considerar todos los elementos que integran el sistema complejo, en vez de realizar supuestos que faciliten la resolución de problemas de producción; en efecto, la modelación debe permitir entender el sistema tanto en su habitual operación, como en la ocurrencia de perturbaciones, las cuales a su vez pueden generar problemas de capacidad, repetición de ciclos y demoras. Al orientar la eficiencia hacia las operaciones, es importante tener en cuenta que la producción de bienes y/o servicios, comprende ordinariamente la parte más grande de los recursos (humanos, materiales, financieros y técnicos) y gastos de una organización, y que el proceso de transformación también consume una cantidad considerable de tiempo; por esto, la dirección adecuada de esta función es crítica para lograr los objetivos organizacionales. (Valencia, 2003). Para alcanzar dicha eficiencia y eficacia, la gerencia de la producción debe planear, programar, controlar y evaluar el proceso productivo (Valencia, 2003).

En primer lugar, la planeación se debe entender como el esfuerzo sistemático y formal de una compañía para establecer sus propósitos, objetivos y estrategias básicas, con el fin de desarrollar planes detallados para poner en práctica las políticas y estrategias y lograr los objetivos y propósito (Steiner, 2004). Por otro lado, (Amaya Amaya, 2007) lo define como el proceso mediante el cual quienes toman decisiones en una organización obtienen, procesan y analizan información pertinente interna y externa, con el fin de evaluar la situación presente de la empresa, así como su nivel de competitividad. Esto con el propósito de anticipar y decidir sobre el direccionamiento de la institución hacia el futuro.

Adicionalmente, el intervalo de tiempo en la planificación de la producción debe incluir la secuencia de las actividades y la programación de posibles cuellos de botella, pues es uno de los factores determinantes para el cumplimiento de pedidos. Sin embargo, éste depende de la disponibilidad de recursos y la cantidad de pedidos imprevistos que puedan presentarse; por lo tanto, la planeación adecuada permite que los ajustes que se deban realizar, para cumplir con la demanda, causen el menor impacto posible en la producción y en la consecución de los objetivos establecidos (Hartmut & Christoph, 2008).

Un tema esencial en la dirección de la producción para la consecución de los objetivos es la preparación de planes para la fabricación de un producto o una gama de productos en la cantidad y de la calidad deseadas para cumplir las fechas de entrega convenidas (Kanawaty, 1996). Sin embargo, hasta en los mejores planes se pueden presentar eventos inesperados o perturbaciones. Es preciso establecer un control sobre los progresos de las operaciones, que pueda revelar desviaciones de los planes y poner en marcha medidas correctivas (Kanawaty, 1996). De esta manera, la planeación de la producción se ha convertido en una necesidad para las compañías ya que les permite sobresalir en la industria y adquirir una ventaja competitiva en el mercado.

Dada la importancia de la planeación de la producción dentro de una empresa, se debe acudir a técnicas y sistemas de modelación para poder analizar y entender adecuadamente las problemáticas de la manufactura; por ejemplo, la planeación agregada busca determinar los volúmenes y los tiempos oportunos de producción para un futuro intermedio, a menudo con una anticipación de 3 a 18 meses. Los administradores de operaciones buscan determinar la mejor forma de satisfacer la demanda pronosticada ajustando los índices de producción, los niveles de mano de obra, los niveles de inventario, el trabajo en tiempo extra, las tasas de subcontratación y otras variables controlables. En general, el objetivo de la planeación agregada es minimizar los costos para el período de planeación. Sin embargo, existen otros aspectos estratégicos más importantes que el costo bajo. Estas estrategias son suavizar los cambios en los niveles de empleo, reducir los niveles de inventarios o satisfacer un nivel de servicio alto (Render & Heizer, 2004).

La planeación agregada es el primer paso para estructurar un Material Requirement Planning (MRP). Es la herramienta más usada cuando se habla de planeación de la producción, ya que permite planificar las compras de material para la fabricación de los productos en cada periodo (Inza, 2006); después de que un horario se genera, es necesario que todas las materias primas y los recursos están disponibles en los tiempos especificados (Pinedo, 2008). El MRP se basa en un plan maestro de producción o MPS cuyo objetivo principal es determinar qué materiales y en qué cantidad se necesitan, y cuándo se necesitan (Inza, 2006). No obstante, estos métodos no son suficientes para garantizar que se realice una asignación adecuada de recursos y se cumplan todas las actividades que afectan el proceso de producción de una empresa y su

respectiva cadena de valor, por lo cual la programación de la producción es el siguiente paso de la gerencia para determinar el orden en el que se deben llevar a cabo las tareas.

En un entorno de fabricación, la función de programación tiene que interactuar con otras funciones de toma de decisión (Pinedo, 2008). Sin embargo, antes de que existiera la programación de la producción en la mente de los gerentes, las fábricas no sabían cuándo empezaba un trabajo, dónde estaba, cómo se movía en la planta y cómo debía ser realizado (Wiers, 1997). Esto quizá, fue el acontecimiento que hizo de la programación de la producción un sistema de planeación y control para mejorar la productividad y rentabilidad, y satisfacer las necesidades del cliente (Vollmann, Berry, & Whybark, 1997), ya que las fechas disponibles de todos los trabajos tienen que ser determinadas conjuntamente por la planificación de la producción, la programación del sistema y el sistema MRP.

Posteriormente, la dirección de producción se debe encargar de verificar los procesos para que se cumplan de acuerdo a lo planeado y programado, por medio de sistemas de control de la producción. La actividad fundamental que se realiza con estos sistemas, es monitorear los costos reales y el rendimiento de la programación de la producción con el fin de corregir el desarrollo de un proceso, de modo que se apegue al plan trazado y se puedan identificar las variaciones que presenta el sistema. (Ballard, 2000). Adicionalmente, se busca controlar la cantidad de materiales que se debe adquirir, la cantidad de piezas y ensambles que se deben fabricar, y el trabajo necesario para la producción de los productos, regulándolos de tal manera, que se alcance la posición óptima en cuanto a beneficios, dentro del marco de las metas que la empresa se ha fijado (Gaither & Frazier, 2003).

Así pues, el control de producción tiene que establecer medios para una continua estimación de la demanda del cliente, la situación del capital, la capacidad productiva y la mano de obra; además debe tomar en cuenta no sólo el estado actual de los factores, sino que debe proyectarlos a futuro (Ballard, 2000). En efecto, la responsabilidad del gerente en la producción, es crear una secuencia perfecta de manera que cada participante y cada pieza, funcionen de acuerdo con un plan común y horario de actividades coordinado (Gaither & Frazier, 2003). Para regular el flujo del trabajo en los sistemas de manufactura, regularmente se utilizan los sistemas push, pull, agotamiento de depósito y el procedimiento enfocado en los cuellos de botella (Gaither & Frazier, 2003).

El sistema de agotamiento puede utilizarse en una producción enfocada al producto o al proceso, requiere de poca información compleja sobre clientes, proveedores y producción. Por otra parte, este tipo de sistemas puede llevar a inventarios excesivos y es bastante inflexible para responder a las necesidades de los clientes, por lo cual, tiende a funcionar mejor cuando la demanda de los productos es verdaderamente aleatoria. Además, las políticas de pedidos de materia prima y de inventarios de productos terminados son vitales (Gaither & Frazier, 2003). Por otro lado en el sistema push, los lotes de materiales se empujan hacia las puertas traseras de las fábricas uno después de

otro, lo que a su vez empuja a otros lotes a través de todas las etapas de la producción (Gaither & Frazier, 2003). Estos flujos de materiales se planean y controlan mediante una serie de programas de producción que indican cuándo cada lote de cada producto en particular debe salir de cada una de las etapas de la producción (Ballard, 2000).

En el sistema pull los productos pasan directamente de etapas de arriba de la producción a etapas de abajo, con pocos inventarios entre etapas, por lo que las materias primas y las piezas se extraen de la parte trasera de la fábrica hacia la parte delantera, donde se convierten en productos terminados. Con inventarios reducidos a su mínimo, cada uno de los materiales debe cumplir los estándares de calidad, cada pieza debe llegar exactamente en el momento prometido y cada máquina debe funcionar adecuadamente (Ballard, 2000). Finalmente, en los sistemas de control enfocados a los cuellos de botella, los lotes de producto llegan más rápido de lo que pueden terminarse; por lo tanto, los sistemas de control se enfocan en las operaciones, máquinas o etapas que dificultan la producción, porque tienen una capacidad restringida y limitante (Gaither & Frazier, 2003).

Finalmente, la gerencia de la producción debe evaluar el éxito de la gestión mediante indicadores de eficiencia, eficacia y efectividad. Estos indicadores, deben ser definidos en el marco del pensamiento estratégico de la organización para así, poder orientar el quehacer de cada uno de los procesos y poder optimizar recursos en todas las actividades que integran el proceso de productivo (Hurtado, 2005).

Teniendo en cuenta que la dirección de la producción tiene como objetivo mejorar la eficiencia, eficacia y efectividad con que los diferentes componentes del sistema pueden controlarse y/o modificarse, existen métodos, como la Investigación de Operaciones, que permiten encontrar las relaciones óptimas que mejor operen el sistema (Prawda, 2004). Ésta área, es una rama de las matemáticas encargada de aplicar métodos científicos y técnicas de decisión para establecer las mejores u óptimas soluciones (Rao S. S., 2009). (Prasad, 1998) define la investigación de operaciones como una disciplina que usa métodos matemáticos, estadísticos y de ciencias computacionales para soportar la toma de decisiones de un sistema. Sin embargo, (Prawda, 2004) define la investigación de operaciones desde una perspectiva más amplia, afirmando que es la aplicación, por grupos interdisciplinarios, del método científico a problemas relacionados con el control de las organizaciones o sistemas, a fin de que se produzcan las soluciones que mejor sirvan a los objetivos de toda la organización.

Los métodos modernos de optimización estudiados en la investigación de operaciones, también llamados no tradicionales, han surgido como unas poderosas y populares herramientas para resolver problemas complejos de optimización e ingeniería en las últimas décadas (Rao S. S., 2009), pues al poner en práctica la investigación de operaciones en la solución de un problema real se puede mejorar la coordinación entre los múltiples componentes de la organización, se puede mejorar el control del sistema y se incrementa la posibilidad de tomar mejores decisiones (Prawda, 2004).

Es por esto que el uso de las ciencias de la decisión en la investigación de operaciones, se vuelve un proceso trascendental dentro del funcionamiento de una empresa, ya que busca mejorar la calidad decisional por medio del análisis estructurado y teniendo en cuenta las particularidades del problema. La estructuración del problema se debe realizar con base en las características específicas y teniendo en cuenta una metodología como la propuesta por (Castillo Hernández, 2006) **(Gráfico 1)**. Adicionalmente, se deben identificar las mejores alternativas de solución con base en criterios definidos mediante el estudio y la caracterización de los componentes de un procesos de decisión: la incertidumbre, las alternativas, los criterios y las preferencias (Castillo Hernández, Toma de decisiones en las empresas: Entre el arte y la técnica, 2006).



Gráfico 1: Metodología para el análisis de decisiones. (Castillo Hernández, 2006)

En consecuencia, la estadística y la probabilidad desempeñan un papel fundamental en los procesos y en los modelos de decisión. Muchas de las variables que afectan una decisión son de tipo estocástico; es decir, no es posible conocer previamente con certeza su resultado al final del período (incertidumbre), por lo que suponen riesgo. Sin embargo, aunque la incertidumbre dificulta la modelación del sistema cuando se presentan perturbaciones, el sistema complejo se puede analizar, conocer y modelar. Esto se debe a que se han desarrollado aproximaciones probabilísticas, que permiten representar resultados de las decisiones a través de variables con sus respectivas distribuciones de probabilidad, sus estadísticas y niveles de confianza (Castillo Hernández, 2006).

Aunque estos modelos y herramientas cuantitativas son de gran ayuda para tomar decisiones, no son suficientes para resolver sistemas complejos de decisión, pues el problema se debe estructurar de manera diferente con base en sus características específicas, y teniendo en cuenta que las decisiones hacen parte del proceso de planeación de una organización (Castillo Hernández, 2006), con el fin de garantizar que se realice una asignación adecuada de recursos y se cumplan todas las actividades que afectan el proceso de producción de una empresa y su respectiva cadena de valor. Dadas estas circunstancias, la optimización y los algoritmos de búsqueda, se vuelven herramientas de gran utilidad para obtener el mejor resultado bajo circunstancias dadas (Rao S. S., 2009).

En primer lugar, dentro del proceso de diseño, construcción y control de cualquier sistema de ingeniería, los encargados deben tomar muchas decisiones tanto técnicas como gerenciales. El fin último de todas estas decisiones es bien maximizar el beneficio deseado, o minimizar los esfuerzos requeridos (materiales, mano de obra, capital, máquinas, tiempo, etc.), los cuales se pueden representar como funciones de ciertas variables de decisión (Rao S. S., 2009). Por esta razón, se puede decir que el proceso de optimización consiste en maximizar o minimizar el valor de una función.

Otros autores como (Diwekar, 2008) sugieren que en el diario vivir se toman decisiones que esperamos, minimicen o maximicen un juego de objetivos que se tienen, tales como escoger el camino más corto para minimizar el tiempo requerido para llegar a algún lugar determinado, encontrar la mejor casa que pueda satisfacer las expectativas del comprador teniendo en cuenta las constantes de costo, o encontrar el precio más económico en una tienda para un producto. La gran mayoría de estas decisiones se toman basadas en la experiencia que se tiene del sistema sin tener que recurrir a modelos matemáticos de decisión. Sin embargo, a medida que los problemas se complican y se le agregan más variables de decisión, hace que al ser resueltas simultáneamente se vuelve complicado tomar decisiones basadas en la optimización o conocimiento previo, por lo cual los algoritmos de búsqueda se vuelven la herramienta utilizada para la resolución de estos problemas.

Los algoritmos de búsqueda son aquellos que están diseñados para localizar un elemento concreto dentro de una estructura de datos. Consiste en solucionar un problema de existencia o no de un elemento determinado en un conjunto finito de elementos. Estos algoritmos, trabajan con asignaciones totales, que incluyen todas las variables. Dada una asignación total inicial, el proceso itera para obtener cada vez mejores asignaciones, hasta llegar a un óptimo. (Larrosa & Meseguer, 2003). Dentro de los algoritmos de búsqueda, podemos encontrar las heurísticas y las meta-heurísticas como métodos de solución.

Las heurísticas, por otro lado, en los últimos años han centrado la atención en la búsqueda local como método para resolver problemas de combinación lineal. La búsqueda por vecindario es una de estas categorías. Empezando con una solución arbitraria, este método construye una secuencia de soluciones repetidamente moviéndose de la solución presente a una solución en un vecindario apropiadamente definido, continuando la transición hasta encontrar la mejor solución, la cual representa un óptimo local (Crauwels & Wassenhovet, 1996). Adicionalmente otros métodos son esquina noroeste, costo mínimo, aproximación de Vogel, asignación, método Húngaro, transporte y transbordo, que al igual que el vecindario más cercano solo generan un óptimo local (subconjunto de soluciones factibles).

Sin embargo, con todas las iteraciones que se realizan, la búsqueda local puede quedar atrapada en óptimos locales. Para escapar de estos óptimos locales, se han desarrollado estrategias que incluyen criterios aleatorios, reinicios desde otros puntos del

espacio de búsqueda, permitir movimientos a estados con una cierta probabilidad, entre otros (Larrosa & Meseguer, 2003). Uno de estos métodos de solución son las meta-heurísticas las cuales permiten un mayor rango de acción.

Las meta-heurísticas, son métodos de solución que adquieren una mayor flexibilidad; un algoritmo genético, por ejemplo, es utilizado en muchos problemas diferentes (Larrosa & Meseguer, 2003). Este algoritmo consiste en un proceso iterativo en el cual mediante unos parámetros se van eliminando soluciones a través de las generaciones, hasta llegar al más apto para sobrevivir, es decir, a la solución óptima (Sortrakul & Cassady, 2007). Sin embargo, ésta no es la única meta-heurística, existe una gran cantidad debido a que se pueden adaptar a las necesidades requeridas, entre ellas se destacan La Colonia de Hormigas, Búsqueda Tabú, GRASP, PSO, entre otras.

Tanto los métodos tradicionales como las metaheurísticas son métodos que se pueden aplicar específicamente para solucionar el problema de programación de la producción, pues estos problemas se pueden abordar desde perspectivas diferentes, con diversas metodologías dependiendo del objetivo a solucionar y de la empresa. Sin embargo, la modelación debe permitir entender el sistema tanto en su operación habitual como en la ocurrencia de eventos imprevistos, por lo que en las últimas décadas también se ha desarrollado investigaciones acerca de la utilización de sistemas inteligentes de control (Trentesaux, 2009).

Algunos autores como (Wanga, Wangb, Vogel, Kumarc, & Chiud, 2009) propusieron un sistema inteligente Multiagente, en el cual se integra eficiencia y eficacia ya que el modelo es capaz de operar tanto el nivel dinámico (cadena de suministros) como el nivel operacional (proceso productivo) a pesar de las perturbaciones. En éste modelo Multiagente, los agentes interactúan por refinamiento, usando una constante de propagación para encontrar la más eficiente y efectiva configuración para la cadena de abastecimiento que satisfaga los requerimientos del cliente. El óptimo global pues, pretende proveer garantías a un nivel más alto que el operacional, el nivel estratégico. Por otro lado, el sistema Multiagente propuesto por (Lima, Zhangb, & Gohc, 2009) potencia la efectividad dimensional con un particular énfasis en la capacidad de respuesta de los planes de producción en los cuales se encuentre perturbaciones (imprevistos). La idea del modelo es usar un proceso iterativo que funcione como una licitación, de tal manera que basado en un algoritmo recocido simulado (Simulated Annealing (SA)) se minimice el total de costos de producción.

Adicionalmente, (Alkhateeb, Maghayreh, & Doush, 2011) definen un sistema Multiagente como una composición de múltiples entidades inteligentes que interactúan entre ellas. Estos sistemas pueden ser usados para resolver problemas que son difíciles o imposibles para una entidad individual o un sistema monolítico. Tienen la característica de ser sistemas abiertos y extensivos, lo que permite la implementación de componentes autónomos y proactivos de software. Por otro lado (Shoham & Leyton-Brown, 2010) definen los sistemas Multiagentes como aquellos que incluyen múltiples entidades

autónomas que contienen bien sea información divergente, intereses divergentes o ambos.

Sin embargo, los niveles operacional y estratégico no son los únicos modelados por un sistema Multiagente. Por ejemplo, (Sabarc, Montreuil, & Frayret, 2009) propuso un sistema Multiagente para los problemas de planeación de personal, en donde se modela el ritmo en una ensambladora multi-producto. El objetivo principal es generar una asignación diaria de los empleados a los puestos de trabajo de acuerdo con la actividad requerida en el momento, teniendo en cuenta factores humanos tales como competencias, habilidades y preferencias, para cada empleado con el fin de proveer una planeación de mano de obra óptima. El desarrollo se hace por medio de la definición de políticas, en las cuales, los agentes tienden a alcanzar un equilibrio de producción mediante la cooperación, buscando completar cada vez más actividades. Los resultados experimentales mostraron que la implementación del sistema Multiagente produjo alta calidad, soluciones efectivas en tiempos computacionales cortos y un futuro potencial para ser aplicado en el tiempo real en un sistema distribuido de control. El proceso de optimización utilizado fue la búsqueda del equilibrio mediante la conformación de políticas que garanticen que cuando el equilibrio sea alcanzado, el óptimo sea alcanzado de igual manera.

Como se puede ver, la opción de modelar situaciones con sistemas Multiagente tiene una enorme ventaja debido a la flexibilidad y simultaneidad que el mismo provee. En cuanto a la programación de la producción también existen investigaciones que usan sistemas Multiagentes como (Caridia & Cavalieri, 2007) quienes proponen un sistema que a diferencia del tradicional centralizado o multinivel jerárquico, asume la presencia de numerosas entidades que toman decisiones distribuidas dentro de un sistema de manufactura, e interactúan y cooperan entre ellas con el fin de alcanzar un rendimiento óptimo. Entre los resultados se puede evidenciar el impacto positivo de los sistemas Multiagentes dentro de la práctica empresarial tanto en manufactura como en la cadena de suministros.

Con el esfuerzo de la investigación de operaciones para resolver problemas específicos de programación de la producción, la literatura clasificó los problemas para poder desarrollarlos teóricamente en Single Machine, Identical machines in parallel, Machines in parallel with different speed, Flow Shop, Job Shop, Open shop entre otros (Pinedo, 2008). Sin embargo, (Pinedo, 2008) argumenta que el caso Single Machine es el caso más sencillo de todas las posibles configuraciones de máquinas, y a su vez es un caso especial de todos los demás. Éste consiste en asignar un orden específico de tareas a una sola máquina. Al igual que el número de problemas que se pueden presentar en la planeación de la producción, cuando se resuelven casos puntuales existe un gran número de funciones objetivo dependiendo de lo que se quiera controlar, como la minimización de la tardanza ponderada (Sortrakul & Cassidy, 2007), por lo que las maneras de

solucionarlas varían desde programación lineal básica hasta algoritmos genéticos y meta heurísticas más complejas como se puede apreciar en tabla 1.

Para realizar esta tabla de resumen literario se tienen en cuenta varias categorías. La primera, es el año debido a la importancia de tener información actualizada y reciente, pues este tema de investigación solo se ha llevado a cabo en las últimas décadas. La segunda, son los autores por las disposiciones de derechos de autor. Por último, se tiene un breve resumen de cada fuente para ayudar a la comprensión y fácil búsqueda de los mismos

Con esta tabla se puede concluir que todos los autores buscan que sus modelos o métodos de solución a los problemas, tengan tres atributos que son fundamentales. El primero es la representación de la realidad, es decir que la investigación se puede llevar a un plano real y poner a prueba. En segundo lugar, el fin de cualquier problema de programación de producción o de optimización se reduce a encontrar la respuesta más eficiente y eficaz para la situación planteada, por esta razón la eficiencia de la respuesta es un atributo importante. Y por último, la rapidez computacional, esto se debe a que para la mayoría de estos problemas se requiere un gran número de horas destinadas únicamente a éste proceso, por esta razón entre menor sea el tiempo computacional, y si la solución logra ser óptima, el modelo será aún mejor.

Teniendo en cuenta el resumen literario realizado en la **Tabla 1**, en donde se pudo observar que el algoritmo genético es un modelo eficiente para solucionar el sistema de programación de la producción en una máquina, se concluye que este modelo es óptimo para solucionar el problema planteado ya que cumple con los tres atributos fundamentales nombrados anteriormente: permite representar la realidad, es eficiente y es rápido computacionalmente. Así pues, con el desarrollo de esta metaheurística, se pretende evaluar si el algoritmo genético sigue siendo un modelo eficiente a la hora de evaluar el problema de programación de la producción con la inclusión de perturbaciones que tiene prioridad, dado que en la literatura no se ha desarrollado esta problemática específica.

Por otro lado, el caso de single machine también se puede resolver por medio de sistemas de Multiagente que se han desarrollado y probado. Por ejemplo (Agnetis, Pacciarelli, & Pacifici, 2007) consideraron el problema de single machine cuando diversos agentes (entidades), cada uno a cargo de una serie de trabajos sin preferencias, compiten por realizar sus tareas respectivas en una única máquina. Cada agente tiene como objetivo minimizar cierta función del costo, la cual depende del tiempo total de procesamiento solamente. Las funciones que se tuvieron en cuenta fueron el máximo de funciones regulares (por tarea), número de tareas atrasadas y tiempo total de procesamiento ponderado. Es claro que las diferentes combinaciones de funciones con las entidades llevan a una variedad de problemas computacionalmente complejos. Además se tiene una restricción en la que el costo por cada entidad no puede superar un límite dado por el autor. De esta manera se logra encontrar la mejor combinación de entidades y recursos simultáneamente.

AÑO	AUTOR	MÉTODO	FUNCIÓN OBJETIVO	¿CÓMO LO RESOLVIO?
2002	Gagne, Price And Gravel	Colonia de Hormigas; Algoritmo Genético; Recocido Simulado; Búsqueda Local; Ramificación y acotamiento	Minimizar la tardanza total	Se comparó un algoritmo de colonia de hormigas con las demás meta heurísticas cambiándole solamente un criterio en la regla de transición
2007	Mazdeh, Sarhadi And Hindi	Ramificación y Acotamiento	Minimizar el tiempo de producción y los costos de entrega	Mediante la buena coordinación entre la parte productiva y la logística de entrega se logró estructurar las propiedades del problema para poder solucionarlo por medio de la meta heurística propuesta
2008	Stecco, Cordeau And Moretti	Búsqueda Tabú	Minimizar el tiempo de finalización del último trabajo	Se basa en cambios y movimientos de intercambio, y un sustituto de la función objetivo se utiliza para acelerar la evaluación de barrio.
2003	Merkle and Middendorf	Colonia de hormigas	Minimizar la tardanza total ponderada	En el espacio de decisión, las hormigas se guían por la información global de feromona y se muestran las diferencias absolutas de calidad entre las alternativas antes de que sean utilizados por las hormigas.
2007	Sortrakul and Cassady	Algoritmo genético	Minimizar la tardanza total ponderada	Basado en un esquema de selección determinado y una estrategia reproductiva, se seleccionan cromosomas como padres y se aparearon. Las crías, evolucionan de acuerdo con la probabilidad de mutación. La próxima generación, se crea a partir de la combinación de padres e hijos.
2011	Singh, Valente And Moreira	Algoritmo Genético Híbrido	Minimizar el costo de la tardanza y adelantamiento cuadrático	Se realizaron 3 mejoras de procedimientos adicionales al algoritmo, las dos mejoras menos costosas se introdujeron en el algoritmo para mejorar la creación de las siguientes generaciones, mientras que la más costosa se usó para mejorar la solución arrojada por el algoritmo.
2005	Panneerselvam	Greedy	Minimizar la tardanza total	Se desarrolla un heurística simple con n puestos de trabajo independientes, cada uno tiene su tiempo de procesamiento y fecha de vencimiento
2005	Liao And Juan	Colonia de hormigas	Minimizar la tardanza ponderada	Se introdujo un nuevo parámetro para el rastro de feromona inicial y el ajuste del calendario de aplicación de búsqueda local.
2010	Khorshidian, Javadian, Zandieh, Rezaeian And Rahmani	Algoritmo genético	Minimizar la tardanza total	Se realizó un nuevo modelo basado en el problema de programación justo a tiempo donde se permite interrupciones y el tiempo de inactividad.
2008	Valente and Gonçalves	Algoritmo genético	Minimizar la suma de adelantamiento y el costo cuadrático de tardanza	El enfoque del algoritmo genético propuesto codifica los cromosomas como un vector compuesto de n genes.
2007	Chang, Chen And Fan	Algoritmo genético	Minimizar la suma total de la tardanza y el adelantamiento	Cromosomas artificiales que se generan de acuerdo a una matriz de probabilidad que se transforma a partir de genes, son inyectados en el algoritmo genético para acelerar la convergencia de los procesos evolutivos.
2004	Sortrakul, Cassady And Nachtmann	Algoritmo Genético	Minimizar el tiempo de procesamiento	Desarrolla una heurística especial para el caso específico basando en algoritmos genéticos.
2008	RibAnghinolfi, Boccalatte, Paolucci and Vecchiola	Colonia de Hormigas	Minimizar el promedio ponderado de la tardanza total	Para lograr que se adapte el modelo, se introduce un mecanismo de feromona global dentro del código.
2011	Li, Yang And Ren	Recocido Simulado	Minimizar el consumo total de recursos.	El algoritmo tiene dos operadores (derecho e izquierdo) y dos operaciones (insertar e intercambiar) para generar el vecindario y dar respuesta al problema inicial, en donde la fecha de entrega del trabajo es una función decreciente lineal del consumo de recursos
2009	Ribiero	Algoritmo Genético	Resolver el problema de una máquina sencilla con penalidades por tardanza o adelantamiento	Se genera una población inicial por medio de GRASP. Un algoritmo polinómico permite determinar la fecha óptima de inicio de procesamiento. Perduran los mejores individuos de cada generación para garantizar la optimalidad y mejorar la calidad de la solución final.

Tabla 1: Revisión bibliográfica para la solución del problema de programación de la producción en una máquina (Elaborado por Daniel Páez y Andrea Vargas)

II. MARCO TEÓRICO

Algunos autores como (Gigerenzer & Gaissmaier, 2011) definen una meta-heurística como una estrategia que ignora parte de la información, con el objetivo de tomar decisiones más rápidamente, mesuradamente, y/o precisas que métodos más complejos. A continuación se presentan las metaheurísticas más usadas y su descripción (tabla 2):

META-HEURISTICA	EXPLICACIÓN
Colonia de Hormigas	La fuente inspiradora de este método es el comportamiento foráneo de las colonias reales de hormigas. Este comportamiento es llevado a colonias de hormigas artificiales para la búsqueda de soluciones aproximadas para problemas discretos, problemas continuos, e importantes problemas de telecomunicaciones, tales como ruteo de líneas (Blum, 2005).
PSO	A diferencia de los algoritmos de búsqueda tradicionales, las técnicas evolucionarias de computación trabajan en una población de soluciones potenciales (puntos) en un campo de búsqueda. Mediante cooperación y competencia a través de las soluciones potenciales, estas técnicas frecuentemente logran encontrar óptimos más rápido que los demás métodos. PSO (<i>Particle swarm optimization</i>) fue motivada por el comportamiento social en una manada de pájaros. En PSO cada partícula (individualmente) ajusta su "vuelo" de acuerdo a su posición y experiencia de vuelo. La mejor posición previa (reflejada en la función objetivo) se almacena, la rata a la que cambia cada partícula su posición (velocidad), y la inercia ponderada son las restricciones principales (Shi & Eberhart, 1998).
Búsqueda Tabú	En su forma básica, la búsqueda tabú se puede ver como una modificación de la búsqueda por vecindarios. Cada vez que un vecindario se genera y una nueva semilla se selecciona, se llama un movimiento. En lugar de parar cuando un óptimo local se encuentra, esta estrategia acepta una nueva semilla inclusive si la solución es peor que la actual. Cuando esto sucede se llama a éste, movimiento tabú para saber que la solución es mala y que ya se ha realizado, y se registra en una lista tabú. Por consiguiente en cada etapa, el algoritmo selecciona la mejor solución de aquellas en el vecindario que no están en la lista tabú (Baker & Trietsch, 2009).
Recocido Simulado	Recocido es un término utilizado en mecánica de materiales. Se refiere al proceso de enfriar un material lentamente hasta que llegue a una zona estable. En este método se puede pensar en cada etapa como siendo llevada a una temperatura inferior que en la etapa anterior. El valor de la función objetivo es análogo con la temperatura del material que es enfriado. Al comienzo en la búsqueda (altas

	temperaturas) existe cierta flexibilidad para moverse a una peor solución, pero a medida que el material se enfría dicha flexibilidad se va reduciendo. Así, el valor de la función objetivo tiende a fluctuar bastante al comienzo, pero se estabiliza al final (Baker & Trietsch, 2009).
Algoritmos Genéticos	Generalmente un Algoritmo Genético, mantiene una lista de soluciones tentativas por cada etapa, y se realizan iteraciones algorítmicas con el fin de generar mejores soluciones por unas determinadas restricciones. Se combinan dos secuencias dentro del conjunto de soluciones, con base en criterios de apareamiento, ya que los nuevos aspirantes se pueden ver como la nueva generación. Así, en cada etapa se seleccionan los mejores pares de solución, se aparean y de repente mutaciones (cambios aleatorios a alguno de los criterios) pueden ocurrir (Baker & Trietsch, 2009).

Tabla 2: Descripción de las meta-heurísticas más utilizadas. Elaborado por Daniel Páez y Andrea Vargas

Por otro lado, (Wooldridge, 2002) define un agente como un sistema computarizado que está situado en algún entorno, y que es capaz de realizar acciones autónomas en dicho entorno con el fin de alcanzar sus objetivos de diseño. Un agente en su entorno realiza básicamente dos acciones dependiendo del sistema. Recoge información como entrada del entorno, y produce una acción como salida que lo afecta. La interacción es usualmente continua y constante.

Con relación a los entornos (Wooldridge, 2002) plantea la siguiente clasificación de sus propiedades:

Accesible vs Inaccesible. Un entorno accesible es aquél en el cual un agente puede obtener información completa, precisa y actualizada sobre el estado del mismo. La gran mayoría de entornos de la vida real (incluyendo, por ejemplo, el mundo físico del día a día y el internet) son inaccesibles desde esta definición.

Determinístico vs No determinístico. Un entorno determinístico es aquél en el cual cada una de las acciones tiene un efecto garantizado, no existe incertidumbre sobre el estado que resultará de la acción ejecutada.

Estático vs Dinámico. Un entorno estático es aquél que puede ser asumido como invariable, excepto por las acciones realizadas por el agente. Por otro lado, en un entorno dinámico existen otros procesos operativos que lo afectan, además de que se pueden percibir cambios, o perturbaciones, que están más allá del control del agente.

Discreto vs Continuo. En un entorno discreto existen un número finito de acciones, mientras que en el continuo son infinitos.

El modelo básico de interacción entre agentes y entorno funciona de la siguiente manera: Se inicializa el entorno en algún estado prestablecido, dependiendo del sistema a simular, en este momento el agente empieza eligiendo una acción a realizar para ese estado. Como resultado de dicha acción, el entorno puede responder con un número diferente de estados. Sin embargo, solamente un estado será el resultado de la acción y, por supuesto, el agente no sabe cuál estado resultará como salida del sistema, y así sucesivamente (Wooldridge, 2002) (**Gráfico 2**).



Gráfico 2: Modelo básico de interacción entre agentes y entorno (Wooldridge, 2002)

Una serie r de un agente en un entorno consiste en una secuencia de estados y acciones intercaladas. En la siguiente tabla (**Tabla 3**) se caracteriza el sistema Multiagente de acuerdo a (Wooldridge, 2002):

ARQUITECTURA PARA AGENTES INTELIGENTES	
NOTACIÓN	NOMBRE Y SIGNIFICADO
Ag	Conjunto finito de todos los agentes, que por cierto se asume como un determinístico.
E	Conjunto finito de todos los estados del entorno, este conjunto se asume como no determinístico.
Ac	Conjunto finito de todas las posibles acciones disponibles que transforman un estado del entorno.
R	Conjunto finito de todas las posibles series (sobre E y Ac)
R^{Ac}	El subconjunto de aquellas series que terminen como una acción
R^E	El subconjunto de aquellas series que terminen como un estado del entorno.
$Env = (E, e_0, T)$	Es la definición formal de un entorno específico, donde E es el conjunto de todos los estados del entorno, e_0 es el estado inicial y T es la función de transformación del entorno.

Tabla 3: Arquitectura para Agentes Inteligentes (Wooldridge, 2002)

Formalmente una secuencia

$$(e_0, \alpha_0, e_1, \alpha_1, e_2, \dots)$$

representa una serie de un agente Ag en un entorno $Env = (E, e_0, T)$ si:

- e_0 es el estado inicial de Env
- $\alpha_0 = Ag(e_0)$
- para $u > 0$,

$$e_u \in \tau((e_0, \alpha_0, \dots, \alpha_{u-1})),$$

donde

$$\alpha_u = Ag((e_0, \alpha_0, \dots, e_u))$$

Para el caso de programación de la producción, (Pinedo, 2008) define *scheduling* como un proceso de toma de decisiones que es usado regularmente en las industrias manufactureras y de servicio. Consiste en asignar recursos a tareas en periodos de tiempo determinados, con el fin de optimizar uno o más objetivos. El caso más sencillo es en el que hay un solo recurso, o máquina, y todos los tiempos de procesamiento son determinísticos (Baker & Trietsch, 2009).

El problema básico cumple los siguientes parámetros según (Baker & Trietsch, 2009):

- Existe un número j de tareas de operación sencilla simultáneamente preparadas para ser procesadas (en el tiempo cero).
- Las máquinas pueden procesar solamente una tarea a la vez.
- Los tiempos de procesamiento son determinísticos y se conocen con anterioridad.
- Las máquinas están continuamente listas para la siguiente tarea (cero averías).
- Las máquinas nunca están en espera si hay un trabajo esperando a ser realizado.

En todos los problemas de *scheduling* el número de tareas se asume como finito. El número de tareas se denota por n y el número de máquinas como m . Generalmente el subíndice j se refiere a una tarea mientras que i se refiere a una máquina. Si la tarea requiere un número de etapas de procesamiento u operaciones, entonces la pareja (i, j) se refiere la etapa de procesamiento u operación de la tarea j en la máquina i (Pinedo, 2008).

Cada tarea tiene atributos dentro del modelo de una máquina, por lo que es bastante útil poder distinguir la información. (Pinedo, 2008) caracteriza el sistema de la siguiente forma:

Tiempo de procesamiento (Processing Time) (p_{ij}) Representa el tiempo de procesamiento necesario de la tarea j en la máquina i . Cabe resaltar que el subíndice se omite si el tiempo de procesamiento de la tarea j no depende de la máquina en la que sea procesada, o si solo puede ser procesada en una máquina específica.

Fecha de entrada (Release Date) (r_j) Se refiere a la fecha (momento) en el que la tarea j entra al sistema. El tiempo más temprano en el cual el trabajo j puede empezar su procesamiento.

Fecha de entrega (Due Date) (d_j) La fecha de entrega (d_j) de la tarea j representa el envío acordado (la fecha que se acordó con el cliente) o la fecha (tiempo) de completación. Se permite completar la tarea después de la fecha de entrega, pero entonces existirá una penalización. Cuando una fecha de entrega debe ser cumplida, se convierte en una fecha o tiempo límite y se denota \bar{d}_j .

Peso (Weight) (w_j) El peso de la tarea j es básicamente un factor de prioridad, que denota la importancia de la tarea j con relación a otras tareas del sistema.

Fecha de inicio (Start date) (S_j) La fecha de inicio (S_j) de la tarea j representa el momento en el que la tarea entra en la máquina para ser procesada

Generalmente un problema de programación de la producción se clasifica y describe por una tripleta ($\alpha | \beta | \gamma$). El campo α describe el entorno de la máquina y tiene solo un dato entrante. El campo β provee detalles de procesamiento, características y constantes, y puede no tener entradas, una entrada sencilla o múltiples entradas de información. Y por último el campo γ que describe la función objetivo a ser minimizada, en la mayoría de los casos solo tiene una entrada de información (Pinedo, 2008).

En las siguientes tablas (**Tabla 4**, **Tabla 5**) se resumen las posibilidades de entradas de información para cada campo anteriormente expuesto propuestas por (Pinedo, 2008) y (Baker & Trietsch, 2009):

ENTORNO DE LA MÁQUINA (α)	
Notación	Nombre y Significado
1	Una máquina. Es el caso más simple de todos los entornos, y a su vez es un caso especial de todos los demás entornos más complicados
(P_m)	Máquinas idénticas en paralelo. Existen m máquinas idénticas en paralelo. La tarea j puede ser procesada en cualquiera de las m máquinas.

(Q_m)	Máquinas en paralelo con diferentes velocidades. Existen m máquinas en paralelo con diferentes velocidades. La velocidad de la máquina i esta dada por v_i
(R_m)	Máquinas no relacionadas en paralelo. Este entorno es una generalización del anterior. Existen m diferentes máquinas en paralelo. La máquina i puede realizar la tarea j a una velocidad v_i .
(F_m)	Taller de flujo. Existen m máquinas en serie. Cada tarea tiene que ser procesada en cada una de las máquinas. Todas las tareas tienen que seguir la misma ruta. Una vez terminada una operación en una máquina la tarea se une a una fila de espera de la siguiente máquina, y así sucesivamente siguiendo la disciplina <i>primera en entrar primera en salir</i> .
(FF_c)	Taller de flujo flexible. Es una generalización del taller de flujo y máquinas en paralelo. En lugar de m máquinas en serie, hay c estaciones en serie, en donde en cada estación existe un número de máquinas idénticas en paralelo. Cada tarea debe pasar de estación en estación hasta terminar el trabajo y en cada estación cualquier máquina puede realizar el trabajo requerido.
(J_m)	Taller de trabajo. En un taller de trabajo con m máquinas, cada tarea tienen su propia y predeterminada ruta de procesamiento. Es necesario hacer una diferenciación entre talleres de trabajo donde una tarea pase por una máquina una sola vez, y los talleres de trabajo donde una tarea puede pasar por una máquina más de una vez.
(FJ_c)	Taller de trabajo flexible. Es una generalización del taller de trabajo y máquinas en paralelo. En lugar de m máquinas en serie existen c estaciones de trabajo, en cada una de éstas existe un número de máquinas idénticas en paralelo. Cada tarea tiene una ruta específica por el taller, la tarea j puede ser procesada por cualquier máquina de la estación de trabajo en la que se encuentre. Cabe resaltar que una tarea puede pasar una o más veces por cada estación de trabajo.
(O_m)	Taller abierto. Hay m máquinas. Cada trabajo tiene que ser procesado en cada una de las m máquinas. Sin embargo, algunos tiempos de procesamiento pueden ser cero, no hay restricciones en lo que respecta a las rutas de cada tarea. El programador está autorizado para definir una ruta para cada tarea.

Tabla 4: Entorno de la máquina (α) (Pinedo, 2008)

RESTRICCIONES Y CONSTANTES (β)	
Notación	Nombre y Significado
(r_j)	Fecha de entrada. Si este símbolo aparece en este campo, la tarea j no puede empezar su procesamiento antes del tiempo establecido. Si no aparece, la tarea puede empezar en cualquier momento. En contraste, la fecha de entrega no se especifica en este campo, sino en la función objetivo.
$(prmp)$	Perturbaciones. Lo que indica esta característica es que no es necesario que una máquina termine una tarea una vez empieza, El programador está autorizado en interrumpir el procesamiento de una tarea en cualquier momento y poner otra tarea diferente. El porcentaje de procesamiento de la tarea que fue interrumpida no se pierde, cuando la perturbación termina y la tarea anterior es puesta en la máquina (o en otra máquina para el caso de máquinas en paralelo), la máquina solo necesita terminar el trabajo restante.
$(prec)$	Constante de prioridad. Esta característica solo aplica para una máquina o máquinas en paralelo, requiriendo que una o más tareas sean realizadas antes de que otra tarea empiece su procesamiento. Si cada tarea tiene como mínimo un predecesor y un sucesor se denominan cadenas de tareas. Si cada tarea tiene por lo menos un sucesor, se denomina <i>intree</i> . Si cada tarea tiene un predecesor, se denomina <i>outtree</i> .
(s_{jk})	Tiempos dependientes de alistamiento. Representa la secuencia de tiempos de alistamiento que incurren entre el procesamiento de la tarea j y la k ; s_{0k} denota el tiempo de alistamiento para la tarea k , si ésta es la primera en la secuencia. Por consiguiente s_{j0} el tiempo de limpieza después de la tarea j , si ésta es la última, (estos tiempos también pueden ser cero). Si estos tiempos dependen de la máquina se debe incluir el subíndice i para especificar la máquina, si no, se asume que todas las tareas tienen el mismo tiempo.
$(fmls)$	Familias de tareas. Las n tareas pertenecen en este caso a F diferentes familias de tareas. Las tareas de la misma familia pueden tener diferentes tiempos de procesamiento, pero pueden ser procesados en una máquina uno tras otro sin incurrir en tiempos de alistamiento y limpieza. Sin embargo, si la máquina cambia de una familia a otra, es necesario tener en cuenta el tiempo de limpieza y alistamiento.

$(batch(b))$	Procesamiento por lotes. Una máquina puede ser capaz de procesar un número de tareas, digamos b , simultáneamente; Esto quiere decir que puede procesar un lote de hasta b tareas al mismo tiempo. Los tiempos de procesamiento en el lote pueden no ser los mismos, y el lote completo finaliza solamente cuando la última tarea del mismo sea terminada, implicando que el tiempo de finalización del lote está determinado por la tarea con el mayor tiempo de procesamiento.
$(brkdown)$	Averías. Esta característica implica que la máquina pueda no estar continuamente disponible. Si existen máquinas idénticas en paralelo, el número de máquinas disponibles es entonces una función del tiempo.
(M_j)	Restricciones de elección. El símbolo M_j puede aparecer en un entorno de máquinas en paralelo (P_m). Cuando es así, no todas las m máquinas son capaces de realizar la tarea j . Esta restricción especifica qué máquinas pueden realizar dicha tarea, y cuáles no.
$(block)$	Bloqueos. Los bloques son un fenómeno que puede ocurrir en los talleres de flujo. Si dicho taller tiene un buffer limitado entre dos máquinas consecutivas, el bloqueo puede ocurrir cuando dicho buffer está lleno, la máquina anterior no puede entregar el trabajo que acaba de terminar. Un bloqueo ($block$) implica permutación ($prmu$).
$(prmu)$	Permutación. Una constante que puede aparecer en los talleres de flujo es que las colas de la máquina siguiente opera de acuerdo a la filosofía primeras en entrar primeras en salir (FIFO). Esto implica que el orden (o permutación) en el que las tareas son realizadas se mantiene desde la primera máquina y es mantenido a través del taller.
(nwt)	No esperas. El requisito de no esperas es otro fenómeno que puede ocurrir en talleres de flujo. Las tareas no pueden esperar entre dos máquinas consecutivas. Esto implica que el momento en el que inicia una tarea puede ser retrasado con el fin de garantizar que no existan pausas entre máquinas.
$(rcrc)$	Recirculación. Esto sucede cuando una tarea pueda visitar una máquina o centro de trabajo más de una vez. Ocurre en talleres de trabajo flexibles.

Tabla 5: Restricciones y Constantes (β) (Pinedo, 2008)

Dentro del campo γ , que solo tiene una entrada generalmente, está la función objetivo que minimiza una función del tiempo de compleción de las tareas, que por supuesto dependen de la programación de producción (Baker & Trietsch, 2009). El tiempo

de cumplimiento de la operación o proceso de la tarea j en una máquina i se denota como C_{ij} . Y el tiempo en el que la tarea j sale del sistema (que obviamente es el tiempo de cumplimiento en la última máquina) se denota como C_j . La función objetivo puede ser también una función de la fecha de entrega (Pinedo, 2008).

Las funciones más usadas e importantes, así como las medidas de desempeño del sistema, las define (Baker & Trietsch, 2009) como sigue (**Tabla 6**):

FUNCIONES OBJETIVO Y MEDIDAS DE DESEMPEÑO (γ)	
Notación	Nombre y Significado
F_j	Tiempo de flujo. El tiempo que la tarea j gasta en el sistema $F_j = C_j - r_j$
L_j	Retraso. La cantidad de tiempo en la que el tiempo de compleción excede la fecha de entrega, $L_j = C_j - d_j$ Una tardanza negativa significa un servicio previo al solicitado, mientras que una tardanza positiva significa un servicio posterior al solicitado. Dependiendo de las situaciones del problema, pueden existir penalidades por retrasos en la fecha de entrega.
T_j	Tardanza. El retraso de la tarea j si no logra alcanzar la fecha de entrega, o cero en otro caso. $T_j = \max\{0, L_j\}$
F	Tiempo de flujo total. $F = \sum_{j=1}^n F_j$
T	Tardanza total. $T = \sum_{j=1}^n T_j$
U	Número de trabajos tardíos o penalidades. $U = \sum_{j=1}^n \delta(T_j)$, donde $\delta(x) = 1$ si $x > 0$ y $\delta(x) = 0$ de lo contrario
C_{max}	Makespan. Se define como $\max(C_1, \dots, C_n)$, es equivalente también al tiempo de compleción de la última tarea en salir del sistema, un makespan mínimo indica, usualmente, una buena utilización de la(s) máquina(s).
$\sum (w_j C_j)$	Tiempo de compleción ponderado total. Esta es una posible función objetivo y puede dar una noción del stock que maneja el sistema o los costos de inventario en los que se incurren con la programación de la producción dada.

$\sum w_j(1 - e^{-rc_j})$	<p>Tiempo de compleción ponderado total con descuento. Esta es una función de costo más general que la anterior. Donde los costos van teniendo un descuento a una tasa de r, $0 < r < 1$, por unidad de tiempo. Esto quiere decir que si la tarea j no es completada para el tiempo t, se incurre en un costo adicional $w_j r e^{-rt} dt$ durante el periodo de tiempo $[t, t + dt]$. Pero si la tarea es terminada para el tiempo t, el costo en el que se incurre en el periodo de tiempo $[0, t]$ es $w_j(1 - e^{-rt})$. El valor de r, es usualmente cercano a 0.</p>
$\sum (w_j T_j)$	<p>Tardanza total ponderada. Esta es también una función de costo más general que el tiempo de compleción ponderado.</p>
$\sum (w_j U_j)$	<p>Número de trabajos tardíos ponderado. Esta medida de desempeño no es solo tenida en cuenta en casos teóricos, es frecuentemente un objetivo en la práctica industrial.</p>

Tabla 6: Funciones objetivo y medidas de desempeño (y) (Baker & Trietsch, 2009)

Basados en la revisión literaria realizada que proporciona un estado del arte adecuado, se realiza un esquema (**Gráfico 3**) que permite visualizar el entorno de la máquina y se concluye que se va a trabajar con los siguientes parámetros del problema: Una sola máquina, en la cual se va a minimizar la tardanza total ponderada (función objetivo), y dentro de las restricciones y constantes se adoptará el modelo que permite perturbaciones. En conclusión, para el tema del trabajo, se puede usar la siguiente notación: $1/prmp/\sum w_j T_j$

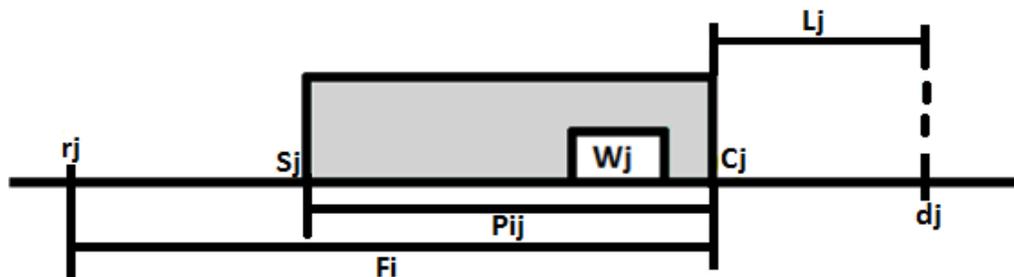


Gráfico 3: Esquematización Single Machine Scheduling. Realizado por Daniel Páez y Andrea Vargas

III. DEFINICIÓN DE LAS PERTURBACIONES

Debido al ambiente de negocios dinámico y altamente competitivo, las empresas manufactureras están bajo presión para reaccionar antes los tratados de libre comercio, el rápido desarrollo tecnológico, y los continuos cambios en las demandas de los consumidores (Islam & Tedford, 2012). Sin embargo, los sistemas actuales de manufactura no son lo suficientemente flexibles para poder afrontar estos cambios (Park & Tran, 2012), por lo cual deben comenzar a implementar sistemas de programación de la producción más robustos, que les permita reaccionar ante posibles imprevistos que se puedan presentar.

Uno de los problemas más comunes en las industrias manufactureras son las perturbaciones en las líneas de producción (Padhia, Wagnera, Niranjana, & Aggarwal, 2012). Estas son definidas por (Islam & Tedford, 2012) como los eventos no planeados e indeseables que causan una desviación o desmejora en el desempeño del proceso, mientras que (Islam, Bagum, & Abul, 2012) las definen como algo indeseado o no planeado que afecta el rendimiento del sistema y lo lleva a incurrir en pérdidas, afectando, según (Zhonga, Daib., Quc, Hud, & Huang, 2013), la efectividad y eficiencia de la empresa. Por su parte, (Pinedo, 2008), ha definido las perturbaciones como todas aquellas tareas que llegan de improviso luego de haber planificado y programado la producción.

Adicionalmente, es importante tener en cuenta que en la mayoría de sistemas de producción, sólo entre el 50% y 60% del tiempo total de producción es utilizado para manufactura (Padhia, Wagnera, Niranjana, & Aggarwal, 2012), el resto del tiempo lo ocupan perturbaciones, las cuales, como se había mencionado anteriormente, interfieren con la efectividad, eficiencia y adicionalmente con la calidad, costos y clima laboral (Padhia, Wagnera, Niranjana, & Aggarwal, 2012).

Teniendo en cuenta todas las implicaciones que las perturbaciones tienen dentro de la programación de la producción, uno de los retos más grandes que existen actualmente para las industrias manufactureras, es determinar cómo lidiar de manera efectiva con estos imprevistos, pues es difícil que en una línea de producción no haya perturbaciones, por lo cual, se han desarrollado diversos enfoques y herramientas que permiten manejar de una manera adecuada dichas perturbaciones (Darmoula, Pierreval, & Hajri-Gabouj, 2012), con el fin de reducir sus efectos al mínimo posible.

Dentro de los estudios que han hecho diversos autores como (Houa & Li, 2012), (Islam & Tedford, 2012) y (Ericsson & Dahlén, 1997), han clasificado las perturbaciones en internas y externas. Las internas hacen referencias a problemas que se puedan presentar durante la ejecución de la producción y dentro de las instalaciones de la fábrica; mientras que las externas, se refieren a grandes cambios en el entorno que pueden

causar disturbios en los proceso de producción. Algunas de estas perturbaciones se presentan a continuación en la **Tabla 7**.

CATEGORÍA	PERTURBACIÓN
Externas	<ul style="list-style-type: none"> • Llegada de una nueva orden inesperada • Cambio de prioridad en las ordenes • Modificación de órdenes de producción (fechas, cantidades) • Cancelación de una orden • Falta de materias primas (retrasos del proveedor) • Fluctuaciones en la demanda • Competencia • Regulaciones gubernamentales • Obstáculos financieros • Desempleo • Catástrofes ambientales (incendios, terremotos, huracanes, tsunamis, derrames de petróleo, etc) (Hu, Gurnani, & Wang, 2013) • Interrupciones de energía eléctrica (Padhia, Wagnera, Niranjamb, & Aggarwalc , 2012)
Internas	<ul style="list-style-type: none"> • Rupturas o averías de la máquina • Rupturas o averías de las herramientas • Problemas con la manipulación de materiales • Falla en el equipamiento auxiliar • Escasez de materiales • Falla en equipos de control y medición • Tiempo de ejecución de operarios (Padhia, Wagnera, Niranjamb, & Aggarwalc , 2012) • Reproceso por productos defectuosos inesperados. • Inspecciones (Padhia, Wagnera, Niranjamb, & Aggarwalc , 2012) • Mantenimiento correctivo por falta de programación de mantenimiento preventivo (Weea & Widyadana, 2012) • Falta de entrenamiento y capacitación del personal en el uso de herramientas (Ben-Daya, 2002) • Programación incorrecta de datos (Huang , Shi, & Mak, 2000) • Falta de planes de contingencia en caso de fallas menores (Huang , Shi, & Mak, 2000)

Tabla 7: Tipos de perturbaciones. Elaborado por Daniel Páez y Andrea Vargas

No obstante, debido a cómo funcionan actualmente las industrias manufactureras, muchas veces la producción se tiene que parar por completo debido a una perturbación. Después de esto, se asigna un tiempo para la recuperación (se realizan cambios en la programación original para poder satisfacer con los requerimientos del cliente), y luego de superada la perturbación, se restaura la producción de acuerdo con el programa original de ejecución. Teniendo en cuenta esto, se evidencia una oportunidad clara de modelar la programación de la producción en una máquina incluyendo perturbaciones.

El modelo que se va a desarrollar, por lo tanto, con perturbaciones para la elaboración de este proyecto es definido como lo plantea (Pinedo, 2008): una vez iniciada la producción, llega un trabajo j en el tiempo t que tiene prioridad y debe ser ejecutado, pero para su ejecución se debe terminar el trabajo que se esté llevando a cabo, y una vez esto ocurra, la perturbación será realizada. Se definió así, ya que el planificador está autorizado para interrumpir el procesamiento de un trabajo en cualquier punto en el tiempo, y de esta forma realizar un trabajo diferente en la máquina; es decir, se interrumpe la tarea j , se realiza la perturbación, y al finalizarla, se continúa con la realización de la tarea j ; esto implica tiempos de preparación de la máquina que se pueden evitar. Por esto, se decide permitir el ingreso de la perturbación en la máquina, una vez la tarea j , que se esté realizando, se finalice.

Al ejecutar la tarea j y la perturbación, se debe realizar una nueva programación de la producción, en la que solo se incluyan las tareas que hacen falta realizar, teniendo en cuenta la fecha de entrega acordada con el cliente o due date (r_j) y el tiempo de procesamiento (p_{ij}).

CAPÍTULO 2

ALGORITMO DE SOLUCIÓN BASADO EN UN MÉTODO TRADICIONAL DE OPTIMIZACIÓN (ALGORITMO GENÉTICO)

I. DESCRIPCIÓN DEL ALGORITMO GENÉTICO UTILIZADO

El desarrollo de un algoritmo genético es óptimo para solucionar el problema de programación de la producción en una máquina, ya que permite representar la realidad, es eficiente y rápido computacionalmente, de acuerdo a los estudios previos realizados por autores como: (H. Khorshidian a, 2011), (Amir Hamidinia, 2012), (Tang, 1999), (Navadon Sortrakul, 2007)

Adicionalmente, otra ventaja de la utilización de los algoritmos genéticos como lo afirma (Arranz de la Peña & Parra Tuyol, 2005) es que son intrínsecamente paralelos, es decir, operan de forma simultánea con varias soluciones, en vez de trabajar de forma secuencial. Esto significa que mientras técnicas tradicionales exploran el espacio de soluciones en una dirección y llegan a un óptimo local, los algoritmos genéticos simplemente desechan este óptimo local y siguen por otros caminos.

Otra ventaja es que no necesitan conocimientos específicos sobre el problema que intentan resolver. Realizan cambios aleatorios en sus soluciones candidatas y luego utilizan la función de aptitud para determinar si esos cambios producen una mejora. (Arranz de la Peña & Parra Tuyol, 2005). Además usan operadores probabilísticos, en vez de los típicos operadores determinísticos de las otras técnicas.

Teniendo claro esto, es importante tener en cuenta una serie de parámetros para el estudio de los algoritmos genéticos:

- **Datos de entrada:** Los datos iniciales para la realización del algoritmo genético son extraídos de OR library, una colección de conjuntos de datos de prueba para una variedad de problemas OR (Congram & Potts, 2004). En esta biblioteca, son presentados el Processing time (P_{ij}), weights (W_j) y el due date (d_j), los cuales fueron obtenidos como se indica a continuación para 40 trabajos diferentes en 125 instancias.
 - ✓ P_j : Fue generado con una distribución uniforme (1,100)
 - ✓ W_j : Fue generado con una distribución uniforme (1,10)
 - ✓ D_j : Fue generado con una distribución uniforme $[P(1-TF-RDD/2), P(1-TF+RDD/2)]$, donde RDD son los rangos relativos oportunos (RDD=0.2, 0.4, 0.6,0.8,1.0) y TF es el factor promedio de tardanza (TF=0.2, 0.4, 0.6,0.8,1.0) (Congram & Potts, 2004).
 - ✓ Adicionalmente, se debe generar un valor de release date (r_j) que indica el momento en el que la tarea j entra al sistema y puede comenzar a realizarse. Esta se obtuvo con una distribución uniforme $(0, \frac{3}{4} \sum P_j)$. Posteriormente, se escogen 20 tareas con $r_j=0$, para esto se genera un aleatorio entre 0 y 1 y se determina que:
Si aleatorio $\leq 0,5$ entonces $r_j = 0$

Si aleatorio > 0,5 entonces $r_j =$ distribución uniforme $(0, \frac{3}{4} \sum P_j)$.

- **Tamaño de la Población:** Indica el número de cromosomas que se debe tener en la población para una generación determinada. En caso de que esta medida sea insuficiente, el algoritmo genético tiene pocas posibilidades de realizar reproducciones, con lo que se realizaría una búsqueda de soluciones escasa y poco óptima. Por otro lado si la población es excesiva, el algoritmo genético será excesivamente lento (Arranz de la Peña & Parra Tuyol, 2005). Por lo cual, se decide trabajar con una población inicial $n=50$
- **Terminación:** El algoritmo, al ser un método de búsqueda estocástico puede en principio correr por siempre, por lo tanto se necesita un criterio de terminación que indique un límite de evaluación de la reproducción (Glover, 2003). Para este caso, el criterio de terminación que se establece es de 100 generaciones factibles, es decir, 100 combinaciones diferentes para realizar 40 operaciones en una máquina, analizado en 125 instancias de producción.
- **Función objetivo:** Para el algoritmo en estudio se establece como función objetivo la minimización de la tardanza total ponderada, lo cual hace referencia al retraso de la tarea j si no logra alcanzar la fecha de entrega, o cero en otro caso $T_j = \max\{0, L_j\}$ multiplicado por un factor de ponderación W_j que equivale al peso, es decir, a la importancia de la tarea j con relación a otras. (Pinedo, 2008)

II. DEFINICIÓN OPERACIONES DEL ALGORITMO GENÉTICO

En un algoritmo genético tras parametrizar el problema en una serie de variables, se codifican en un cromosoma al que se le aplicarán todos los operadores utilizados por un algoritmo genético (Arranz de la Peña & Parra Tuyol, 2005), ya que las soluciones codificadas en un cromosoma compiten para ver cual constituye la mejor solución. Adicionalmente, el ambiente constituido por las otras soluciones, ejercerá presión selectiva sobre la población, de forma que sólo los mejores hijos (aquellos que resuelvan mejor el problema) sobrevivan a las siguientes generaciones, igual que en la evolución de las especies (Arranz de la Peña & Parra Tuyol, 2005).

Por lo tanto, (Arranz de la Peña & Parra Tuyol, 2005) afirma que un algoritmo genético consiste en hallar de qué parámetros depende el problema, codificarlos en un cromosoma, y aplicar los métodos de la evolución: selección, reproducción y mutación que generen diversidad.

1. **Codificación de los cromosomas:** Los cromosomas de alguna manera deberán contener información acerca de la solución que representan. En este caso, se va a utilizar una codificación numérica (Arranz de la Peña & Parra Tuyol, 2005) que busca determinar el orden en que se deben realizar 40 trabajos.
2. **Selección:** Los métodos de selección se utilizan para determinar ¿Qué padres deben ser escogidos para reproducirse? (Talbi, 2009), es decir, cuales son los individuos más capacitados para sobrevivir y crear una nueva generación más facultada. Por lo tanto, se tiene que crear la nueva población teniendo en cuenta que los buenos rasgos de los mejores individuos se transmitan a ésta (Arranz de la Peña & Parra Tuyol, 2005). En este trabajo se van a utilizar los siguientes métodos de selección:
 - a) **Selección por rueda de ruleta:** Con este método, se asignará a cada individuo una probabilidad de selección proporcional a su aptitud relativa (Talbi, 2009). Cada cromosoma tendrá una parte de la ruleta, mayor o menor de acuerdo a la aptitud f_i (Arranz de la Peña & Parra Tuyol, 2005) del individuo p_i en la población de P , por lo que su probabilidad de ser seleccionado es $p_i = \frac{f_i}{\sum_{j=1}^n f_j}$. La selección de μ individuos es realizado por μ giros independientes de la ruleta (Talbi, 2009). Obviamente, el cromosoma con mayor puntuación saldrá con mayor probabilidad.
 - b) **Selección por torneo:** Consiste en escoger de forma aleatoria un número k de individuos de la población, donde k es el tamaño del grupo del torneo. El torneo se aplica μ veces para seleccionar μ individuos de entre los k miembros del grupo y el que tenga mejor puntuación puede reproducirse (Talbi, 2009).

3. Reproducción: Es el proceso mediante el cual una vez realizada la selección de los cromosomas se procede a realizar el cruce o intercambio de material genético entre 2 o más de éstos, con el fin de que el descendiente mejora la aptitud de sus padres (Arranz de la Peña & Parra Tuyol, 2005). Existe gran variedad de criterios de reproducción, sin embargo, para la elaboración de este trabajo se van a utilizar los siguientes:

- a) Reproducción uniforme:** En este método se genera un número aleatorio entre 0 y 1 con el fin de elegir el padre que se va a utilizar para la reproducción (Arranz de la Peña & Parra Tuyol, 2005) . Para este caso se establece que:
- Si el número aleatorio se encuentra entre 0 y 0,5 se va a elegir el 60% de los cromosomas de la madre y el 40% restante se completa con los cromosomas del padre que aún no se han incluido en la programación de la producción, con el fin de garantizar que se realicen todas las tareas.
 - Si el número aleatorio se encuentra entre 0,5 y 1 se va a elegir el 40% de los cromosomas de la madre y el 60% restante se completa con los cromosomas del padre que aún no se han incluido en la programación de la producción, con el fin de garantizar que se realicen todas las tareas.

Nota: Se decide tomar estos rangos debido a que porcentajes pequeños o iguales están asociados con pérdida de diversidad y convergencia prematura (Glover, 2003).

b) El siguiente método de reproducción que va a ser utilizado fue propuesto por la Ingeniera Eliana González, Profesora de la Pontificia Universidad Javeriana (González, 2013) y consiste en la selección de 2 cromosomas para la reproducción, en donde un cromosoma indica el orden en que se van a llevar a cabo los trabajos y el segundo, indica los trabajos. Así por ejemplo:



4. Reemplazo: Permite la evolución de la población inicial hacia la optimización de la función objetivo, ya que con esta operación se elimina el individuo menos apto de esta y se reemplaza por el hijo obtenido luego de la reproducción.

5. Mutación: Representa pequeños cambios en los individuos seleccionados de la población (Talbi, 2009). Este operador busca preservar la diversidad genética de la población explorando áreas posiblemente no abordadas del espacio de búsqueda (Arranz de la Peña & Parra Tuyol, 2005). En este caso se van a utilizar los siguientes métodos de mutación:

- a) **Mutación de intercambio:** Se aplica a codificaciones basadas en el orden. Consiste en elegir 2 coordenadas al azar e intercambiar sus valores (Will, 2010)

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

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

- b) El siguiente método de mutación fue propuesto por Daniel Páez durante la elaboración de un algoritmo genético para la solución del problema de Knapsack en el año 2011 (Páez, 2011) y consiste en seleccionar el valor de la última coordenada y desplazarlo a la primera. Esto implica que todas las demás coordenadas van a ser desplazadas una casilla:

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

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

Teniendo definidos los parámetros y las operaciones que se van a llevar a cabo para desarrollar el algoritmo genético, se crean diferentes combinaciones (**Tabla 8**) entre los criterios de selección, reproducción y mutación, para comparar los resultados arrojados por estos y seleccionar la combinación de operaciones que se aproxima más al óptimo. Los criterios que se tienen en cuenta para la selección del mejor algoritmo son:

1. El mejor valor de la Función objetivo de la última población
2. El promedio de la última población
3. La desviación de la última población
4. La rapidez de convergencia (dada por la pendiente en la gráfica del promedio de cada iteración)
5. El promedio del tiempo de ejecución de cada iteración
6. La desviación del tiempo de ejecución de cada iteración

	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo 5	Algoritmo 6	Algoritmo 7	Algoritmo 8
Selección	Ruleta	Ruleta	Ruleta	Ruleta	Torneo	Torneo	Torneo	Torneo
Reproducción	Uniforme	Uniforme	Eliana	Eliana	Uniforme	Uniforme	Eliana	Eliana
Mutación	Intercambio	Daniel	Intercambio	Daniel	Intercambio	Daniel	Intercambio	Daniel

Tabla 8: Posibles combinaciones criterios de selección, reproducción y mutación.

Elaborado por Daniel Páez y Andrea Vargas

Con estos criterios definidos, se realiza un análisis envolvente de datos o DEA por sus siglas en inglés (data envelopment analysis). Este es un modelo propuesto por primera vez por Charnes, Cooper y Rodas en 1978 (Wei , 2001), que sintetiza la investigación de operaciones, la gestión de la ciencia y la econometría, y proporciona una evaluación de la eficiencia relativa de un grupo de unidades de toma de decisiones (DMU), con número múltiple de entradas y salidas, los cuales permiten expandir el conjunto de posibilidades de producción (Wei , 2001).

Se decidió realizar un DEA y no un diseño de experimentos, dado que el diseño de experimentos permite determinar si hay un efecto significativo de un factor, independientemente de los niveles que tenga, y además aplicar una prueba donde solo se puede hacer comparación de datos por pares; mientras que el DEA permite realizar un ranqueo de todos los algoritmos a la vez y no por pares.

Adicionalmente, durante la elaboración del trabajo se querían aplicar varias herramientas aprendidas durante la carrera de Ingeniería Industrial, y el DEA resulto ser un método sencillo y que permite seleccionar entre los 8 algoritmos que se tienen, solamente uno de acuerdo a la evaluación de eficiencias relativas.

La información inicial para este modelo, está dada por la normalización de los resultados de cada uno de los criterios para cada una de las 125 instancias, y la evaluación se hace para cada una de estas, teniendo en cuenta los resultados arrojados en los 8 algoritmos.

De acuerdo a esta información, se establece que las salidas del problema son los 6 criterios de selección del mejor algoritmo, y la única entrada es la eficiencia. Teniendo claro esto, se determina la notación del modelo:

- Variables:
 - j : Criterios de selección
 - i : Algoritmos
- Parámetros
 - $1 \leq j \leq 6 \quad \forall j \in R$
 - $1 \leq i \leq 8 \quad \forall i \in R$
 - S_{ij} : Salida del algoritmo i para el criterio $j \quad \forall i \in R, \forall j \in R$
 - W_j : Peso del criterio de selección $j \quad \forall j \in R$
 - E_{ij} : Entrada del algoritmo i para el criterio $j \quad \forall i \in R, \forall j \in R$
- Restricciones
 - $0 \leq \sum_{i=1}^8 \sum_{j=1}^6 S_{ij} \times W_j$
 - $W_j \geq 0 \quad \forall j \in R$
 - $W_j \times E_{ij} = 1 \quad j = 6, i = 1$

$$\circ \left(\sum_{i=1}^8 \sum_{j=1}^6 S_{ij} \times W_j \right) \leq (W_j \times E_{ij}) \quad \forall i \in R, j = 6$$

- Función objetivo:

$$\text{Max } (Z) = \sum_{j=1}^6 W_j \times S_{ij} \quad i = 1$$

Los resultados arrojados por Solver para cada una de las 125 instancias, se organizan de mayor a menor y se realiza un ranqueo (**Anexo 1**) que permite determinar la moda de la posición uno, es decir, las veces que cada algoritmo tuvo la mejor eficiencia, obteniendo así el mejor algoritmo (**Gráfico 4**). Éste algoritmo está comprendido por la selección por ruleta, la reproducción uniforme y la mutación por intercambio.

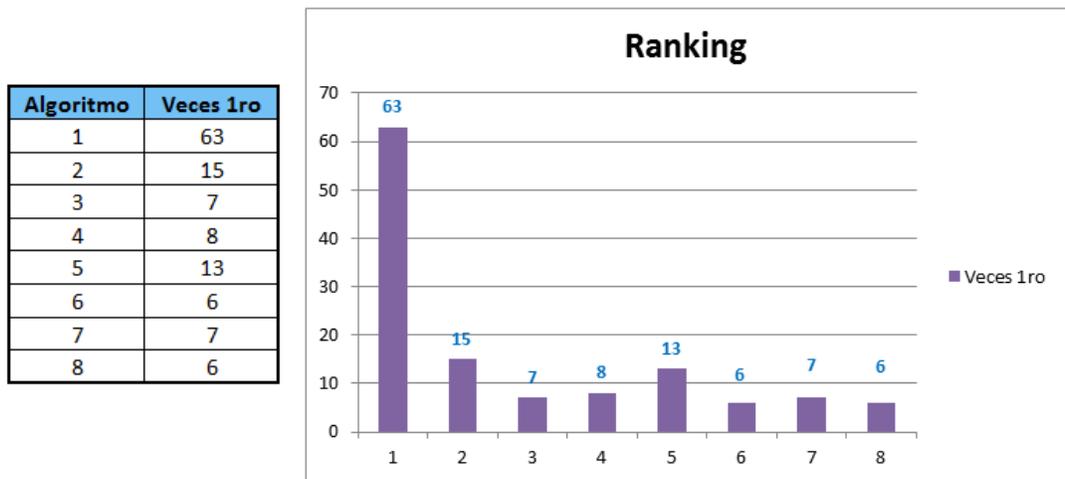


Gráfico 4: Resultados ranqueo. Elaborado por Daniel Páez y Andrea Vargas

Con esto claro, se procede a desarrollar el algoritmo genético con la inclusión de las perturbaciones, como lo indica el siguiente gráfico (**Gráfico 5**): luego de planeada, programada por el algoritmo genético (determinado el orden como se van a realizar las tareas) e iniciada la producción, llega un trabajo j en el tiempo t que tiene prioridad y debe ser ejecutado, pero para su ejecución se debe terminar el trabajo que se esté llevando a cabo. Una vez esto ocurre, se realiza la perturbación y se debe realizar una nueva programación de la producción, en la que solo se incluyan las tareas que hacen falta realizar, teniendo en cuenta la fecha de entrega acordada con el cliente o due date (r_j) y el tiempo de procesamiento (p_{ij}).

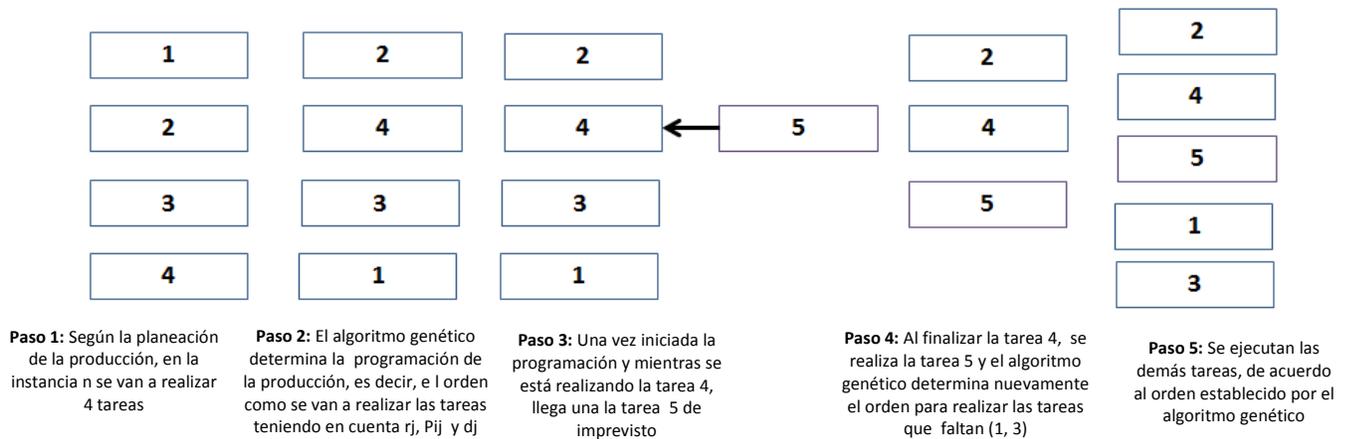


Gráfico 5: Algoritmo genético para programar la producción con inclusión de perturbaciones. Realizado por Daniel Páez y Andrea Vargas

Adicionalmente, para su realización se tienen en cuenta los siguientes parámetros:

- ✓ Datos de entrada: Extraídos de OR library
- ✓ Release date: 20 trabajos con $r_j = 0$, 20 trabajos con $r_j =$ distribución uniforme $(0, \frac{3}{4} \sum P_j)$
- ✓ Tamaño de la población inicial: 50
- ✓ Las mutaciones son determinísticas y se realizan cada 10 generaciones factibles.
- ✓ Terminación: 100 generaciones factibles
- ✓ Función objetivo: Minimización de la tardanza ponderada
- ✓ Perturbaciones: Trabajos imprevistos que se realizan una vez se termine la tarea j que se esté ejecutando.
- ✓ Para tener un nivel de confianza del 95% se determina que el número de corridas que se va a realizar con el algoritmo es 100. Esto se obtiene estableciendo:

z	1,96	Nivel de Confiabilidad
p	0,5	Probabilidad de Éxito
q	0,5	Probabilidad de Fracaso
ϵ	0,1	Error Propuesto

$$FORMULA \rightarrow n = \frac{z^2 \times p \times q}{\epsilon^2}$$

$$n = 96,04$$

Gráfico 6: Cantidad de corridas del algoritmo genético. Elaborado por Daniel Páez y Andrea Vargas

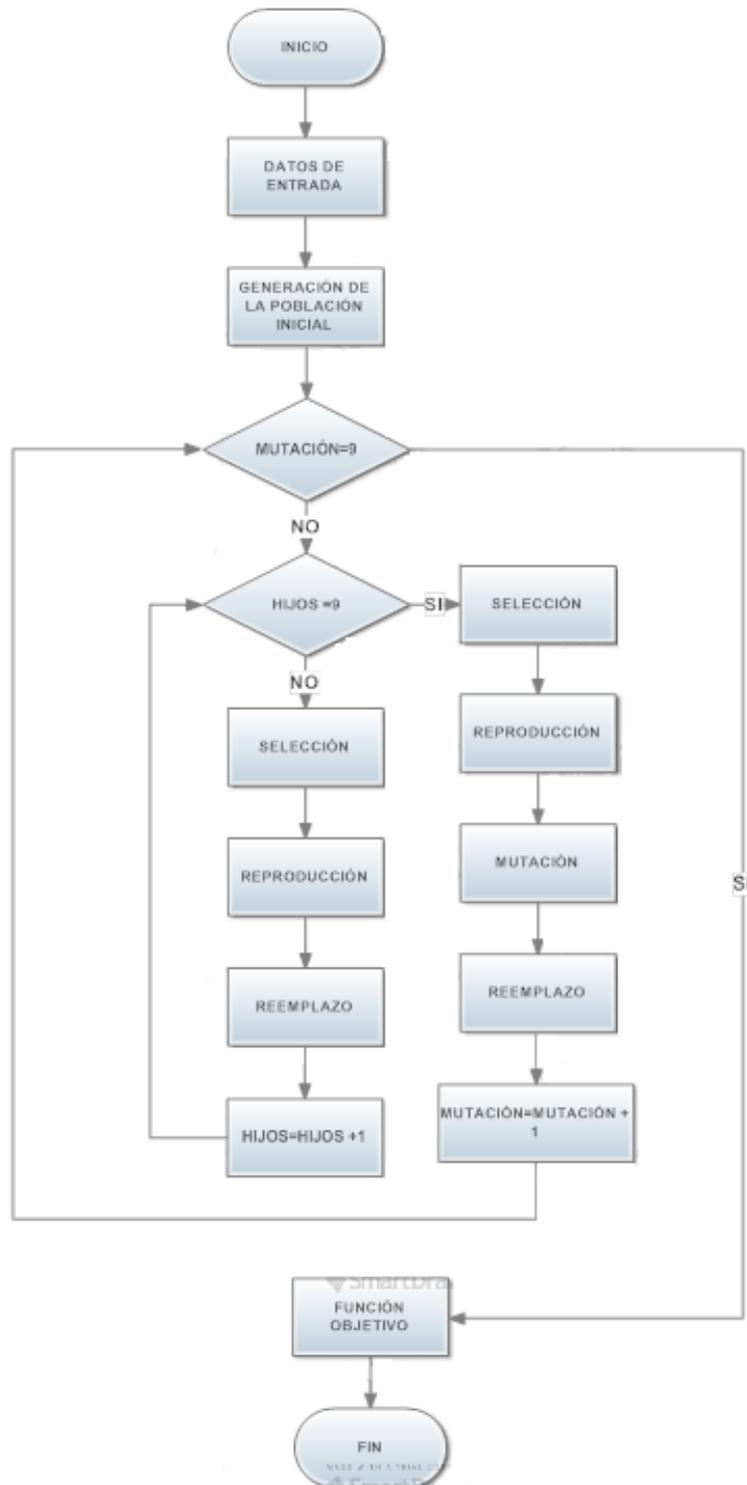
Con lo anterior se aproxima el número de encuestas a 100.

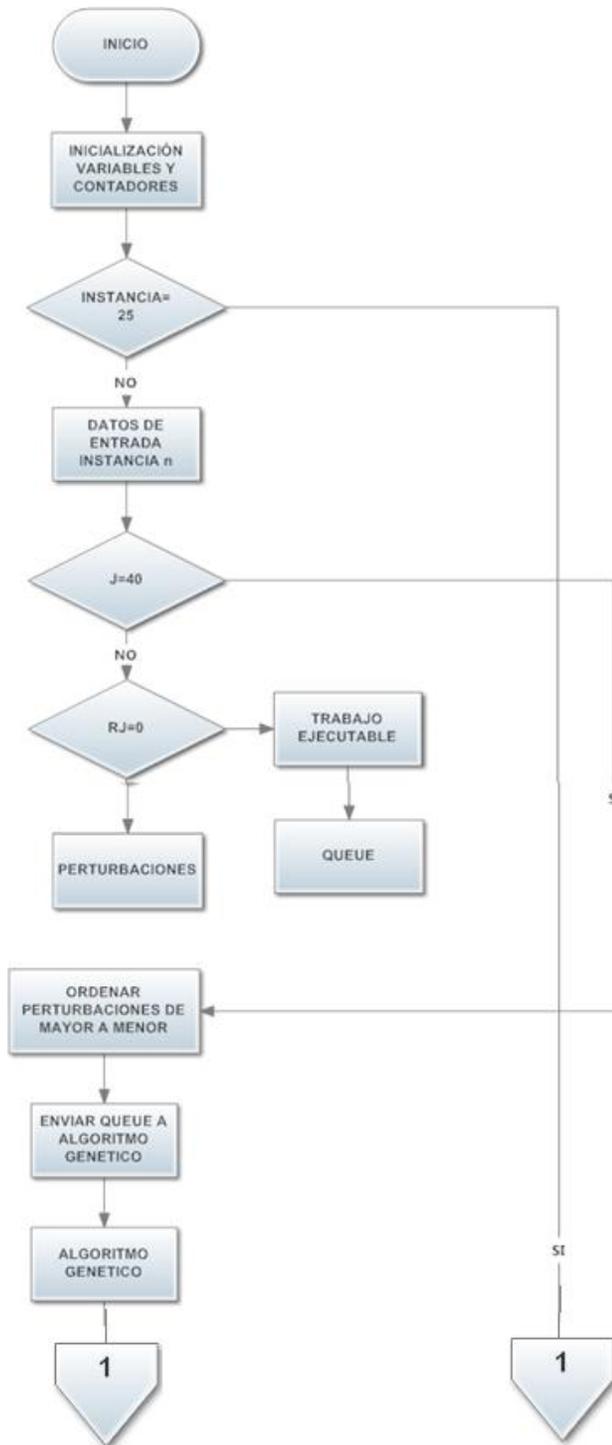
A continuación, se presentan 2 diagramas de flujo, los cuales permiten esquematizar los procesos realizados por el algoritmo genético (**Gráfico 7**), y los procesos realizados para programar la producción con perturbaciones (**Gráfico 8**), respectivamente.

Adicionalmente es importante tener en cuenta, que para garantizar el funcionamiento del algoritmo, es necesario recurrir a estrategias de reparación, es decir, a procedimientos de restauración de la factibilidad en aquellas soluciones no factibles, con el fin de evitar el uso de penalizaciones en la función objetivo y por lo tanto en la solución (Arroyo Sánchez, 2000). La reparación que se realizó, se llevó a cabo en el proceso de selección y se estableció que si el mismo individuo es seleccionado como padre y madre entonces:

- Si el individuo seleccionado para ser madre es menor a 50, entonces el individuo que se debe seleccionar para ser padre durante la reproducción va a ser igual a 50 menos la madre.
- Si el individuo seleccionado para ser la madre es el 50, entonces el individuo que se debe seleccionar para ser padre durante la reproducción va a ser un número aleatorio entre 1 y 49.

Gráfico 7: Procesos realizados por el algoritmo genético. Realizado por Daniel Páez y Andrea Vargas





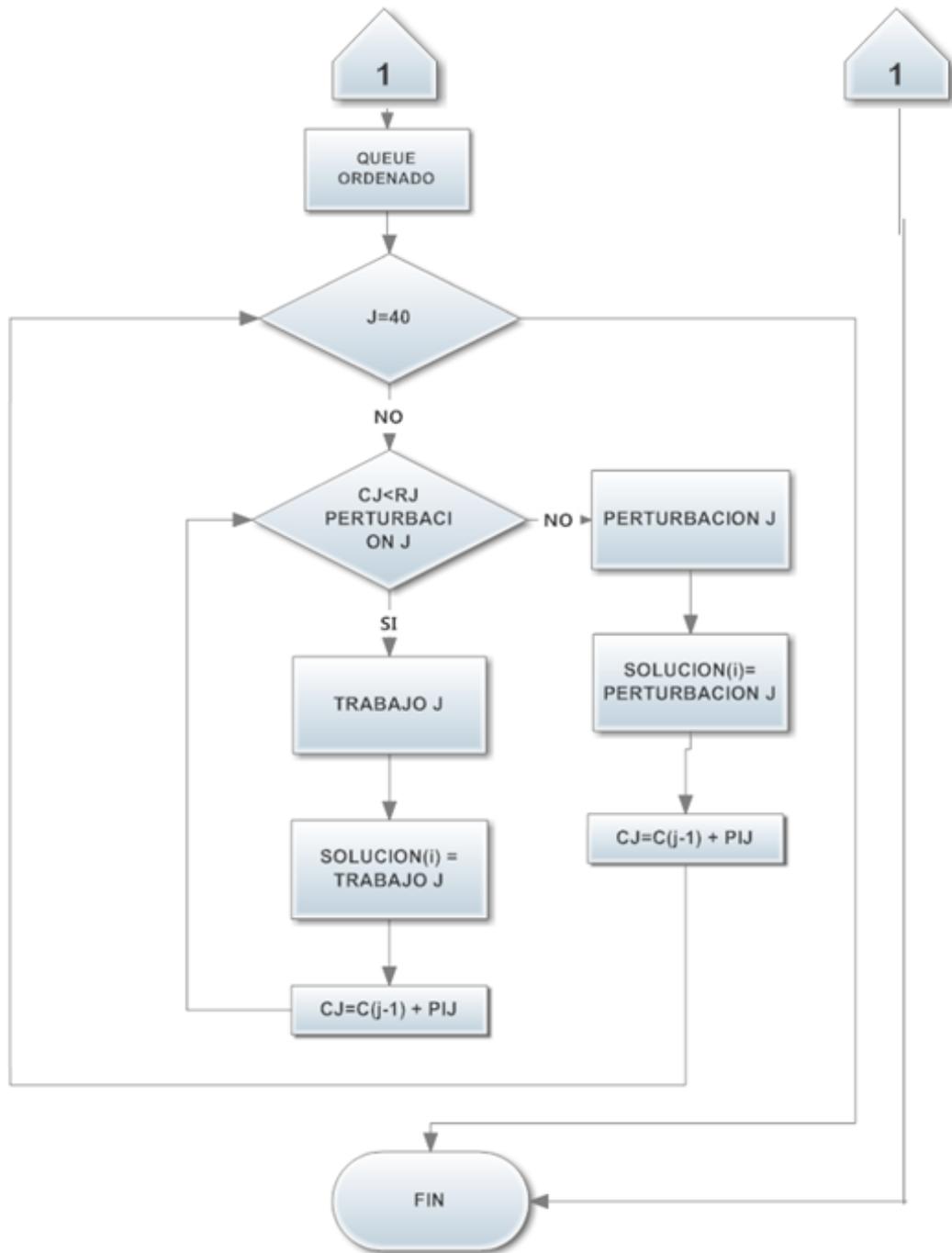


Gráfico 8: Procesos realizados para programar la producción con perturbaciones. Elaborado por Daniel Páez y Andrea Vargas

III. DESCRIPCIÓN DE LA SOLUCIÓN

Luego de haber corrido el algoritmo las 100 veces establecidas, se obtienen 2 indicadores:

1. Tardanza total ponderada
2. Tiempo total de ejecución

En la siguiente tabla (**Tabla 9**), se pueden observar tanto el promedio de la tardanza total ponderada como el promedio del tiempo de ejecución, para cada una de las 125 instancias.

Adicionalmente, los resultados (para la instancia 1 por ejemplo) se pueden ver en las siguientes gráficas, donde se indica (por intervalos) la frecuencia de la tardanza total ponderada (**Gráfico 9**) y la frecuencia del tiempo de ejecución respectivamente (**Gráfico 10**)

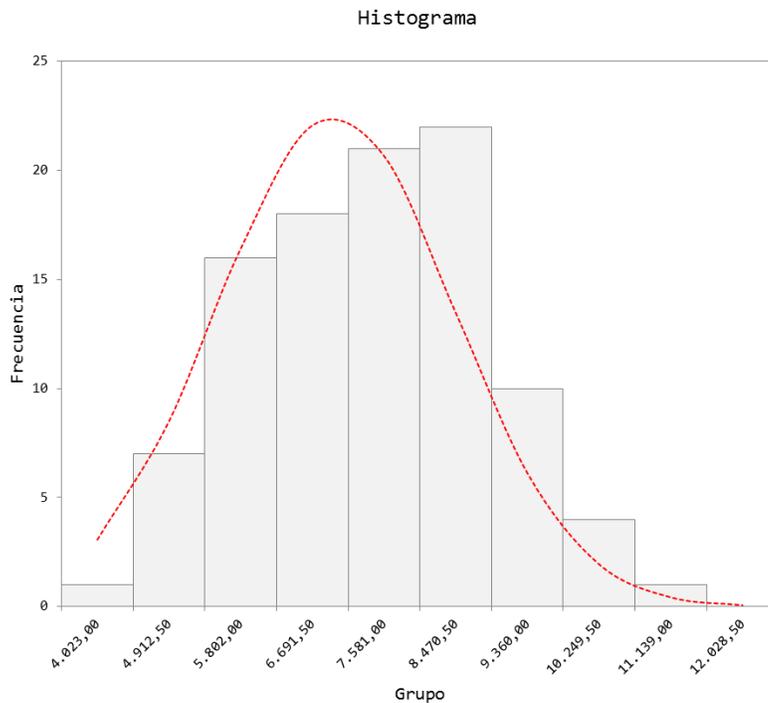
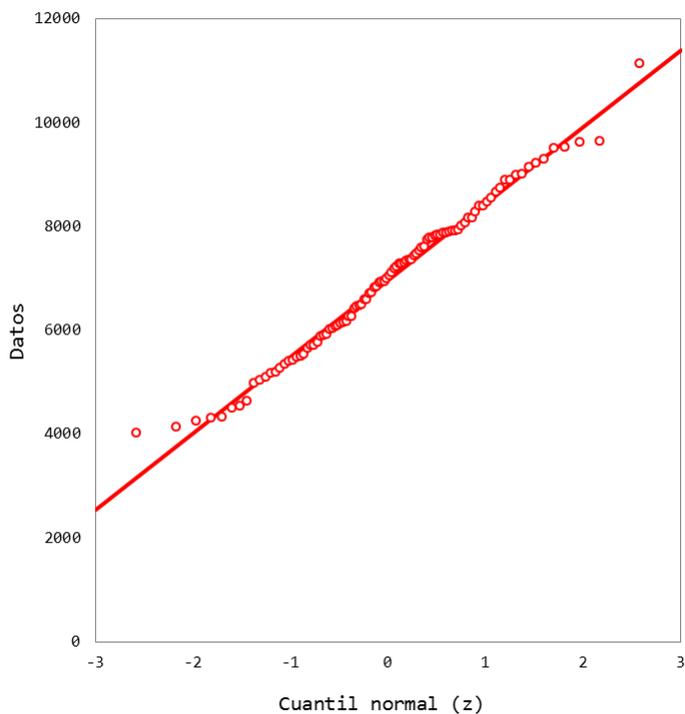


Gráfico de probabilidad normal

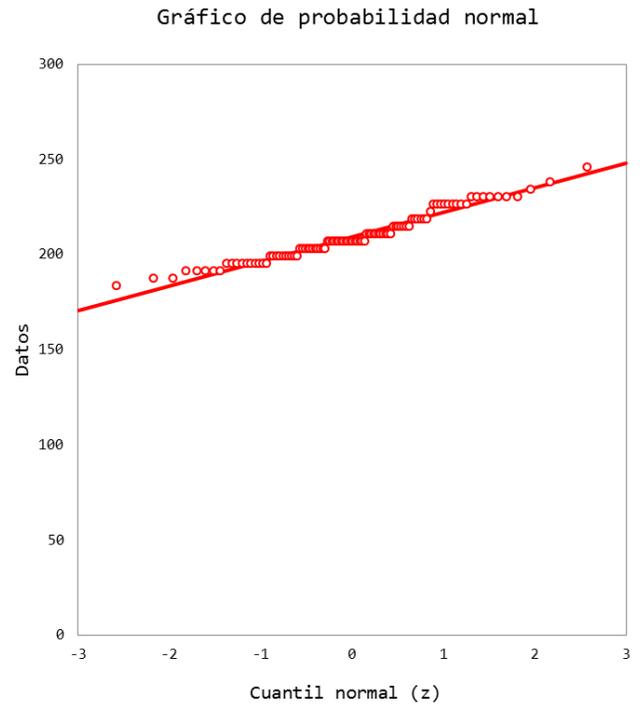
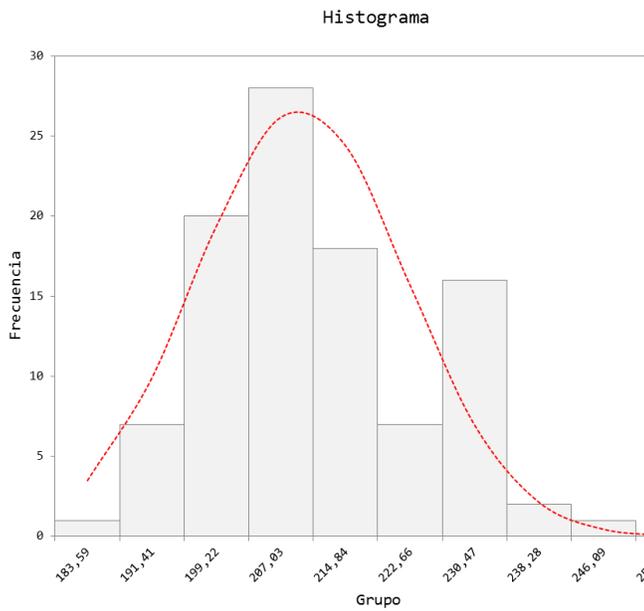


Datos Descriptivos		Grupo	Frecuencia
Cuenta	100	4023	1
Promedio	6974,3300	4912,5	7
Desviación	1478,7194	5802	16
Varianza	2186611,1122	6691,5	18
Curtosis	-0,4725	7581	21
Sesgo	0,0779	8470,5	22
Mínimo	4023,0000	9360	10
Máximo	11139,0000	10249,5	4
Rango	7116,0000	11139	1
Norm (p-valor)	0,3099	12028,5	0

Gráfico 9: Frecuencia de la tardanza total ponderada para la Instancia 1. Elaborado por Daniel Páez y Andrea Vargas

INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN
1	6974,33	209,453125	43	133894,7	151,1328125	85	18209,53	143,5546875
2	7489,41	149,765625	44	131766,69	143,359375	86	87830,09	145,9765625
3	8071,39	174,8828125	45	111893,7	193,5546875	87	84439,78	167,03125
4	12901,89	157,8515625	46	104896,69	139,84375	88	61981,36	149,921875
5	5166,33	186,25	47	181523,95	187,1875	89	93756,37	124,8046875
6	26542,94	187,578125	48	155524,6	179,3359375	90	56037,86	131,2109375
7	21329,38	184,296875	49	150712,1	153,2421875	91	118241,15	141,6015625
8	31302,26	160,6640625	50	158633,07	180,8984375	92	105684,78	132,6953125
9	30172,82	206,796875	51	27321,3	158,359375	93	107188,19	119,375
10	27772,73	188,1640625	52	12432,06	176,4453125	94	112968,29	143,125
11	68001,19	186,5234375	53	10184,33	184,7265625	95	119569,15	143,359375
12	56465,54	184,6875	54	7186,46	191,875	96	166956,58	105,3125
13	58197,25	191,71875	55	8490,56	177,3828125	97	179610,74	159,4140625
14	45715,54	164,375	56	19000,58	175,1171875	98	191702,15	149,84375
15	69187,01	176,875	57	36569,21	183,8671875	99	176392,47	141,640625
16	146039,28	167,3828125	58	37017,34	183,4765625	100	234323,78	112,6953125
17	128319,02	185,9375	59	43138,7	169,1796875	101	5342,64	151,171875
18	144976,04	180,9765625	60	32276,73	142,0703125	102	23547,41	135,859375
19	141563,37	196,7578125	61	57673,57	174,21875	103	10730,04	110,15625
20	112462,67	182,3046875	62	57795,47	188,28125	104	19054,35	157,265625
21	138203,62	168,1640625	63	80861,73	183,1640625	105	20068,99	84,8828125
22	135731,44	162,3828125	64	66178,34	130,1171875	106	50940,83	122,421875
23	199567,36	172,1875	65	56077	155,3125	107	39317,14	95,1953125
24	191073,73	175,9765625	66	132006,98	136,953125	108	53057,77	123,3984375
25	186948,01	152,7734375	67	131920,85	119,140625	109	24124,97	130,234375
26	10410,98	189,0625	68	150748,13	147,2265625	110	20801,47	142,6953125
27	5970,62	199,4140625	69	135720,74	129,296875	111	108839,4	155
28	8252,45	172,7734375	70	148580,21	149,375	112	62949,5	140,5859375
29	5167,3	179,84375	71	153571,77	134,140625	113	69183,2	132,8515625
30	4469,99	163,4765625	72	209688,83	148,9453125	114	78123,35	113,5546875
31	32998,32	203,4375	73	132699,76	144,765625	115	64415,12	138,125
32	36368,26	176,71875	74	128135,91	103,4765625	116	104483,83	107,65625
33	20339,74	180,7421875	75	149713,09	105,4296875	117	124265,9	140,546875
34	18735,43	194,0234375	76	10266,8	137,109375	118	78522,89	162,34375
35	22765,87	174,0234375	77	11509,98	148,125	119	161520,12	167,734375
36	62373,27	160,625	78	2912,6	156,875	120	129334,25	133,5546875
37	50464,01	155,8203125	79	12338,65	132,9296875	121	178924,38	134,8046875
38	61553,17	194,6484375	80	17168,83	175,546875	122	149652,07	93,90625
39	64141,9	194,53125	81	37378,74	141,2890625	123	147567,98	156,2109375
40	56031,47	185,7421875	82	26479,15	133,2421875	124	139077,54	128,5546875
41	108082,54	169,765625	83	32777,81	150,1953125	125	178356,91	162,109375
42	144799,7	177,34375	84	34658,71	160,234375			

Tabla 9: Promedio tardanza total ponderada y tiempo de ejecución para cada una de las instancias (luego de 100 corridas). Elaborado por Daniel Páez y Andrea Vargas



Datos Descriptivos		Grupo	Frecuencia
Cuenta	100	183,59375	1
Promedio	209,453125	191,40625	7
Desviación	12,8273984	199,21875	20
Varianza	164,54215	207,03125	28
Curtosis	-0,3635853	214,84375	18
Sesgo	0,46453497	222,65625	7
Mínimo	183,59375	230,46875	16
Máximo	246,09375	238,28125	2
Rango	62,5	246,09375	1
Norm (p-valor)	0,05699613	253,90625	0

Gráfico 100: Frecuencia del tiempo de ejecución. Elaborado por Daniel Páez y Andrea Vargas

Después de realizar la un test de normalidad, se puede observar en estos gráficos (Histograma y Gráfico de probabilidad Normal), que los datos tienden a formar una campana de Gauss, además podemos afirmar que con un 95% de confianza los datos tienden a una distribución normal. Esto significa que son más probables los valores cercanos a la media y conforme nos separamos de esta, la probabilidad va decreciendo de igual forma a derecha e izquierda.

Por otro lado, con el fin de realizar una descripción adecuada de los datos que incluya una medida de tendencia central, junto con información sobre la forma y la dispersión de los datos, se realizan **diagramas de cajas (Gráfico 11)**. En éstos se describe la información contenida en una muestra basándose en cinco estadísticas: Valor máximo (Máx), tercer cuartil (Q3), mediana (m), primer cuartil (Q1), valor mínimo (Mín). Su utilización es de gran ayuda, ya que permite observar:

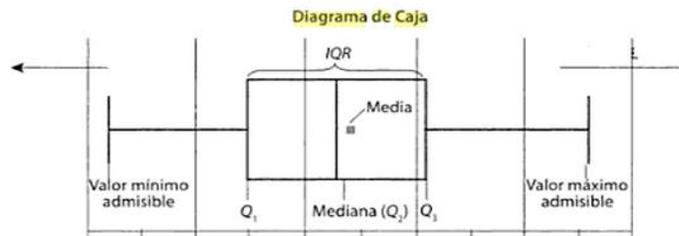


Gráfico 11: Diagrama de caja (Rial Boubeta, 2008)

- **Medidas de posición:** Los extremos de la caja representan el primer cuartil (Q_1) y el tercer cuartil (Q_3). Por tanto, el interior de la caja contiene el 50% de los datos centrales. La línea central representa el segundo cuartil, que coincide con la mediana de la distribución (Rial Boubeta, 2008).
- **Medidas de tendencia central:** Se representa la mediana y la media aritmética. La relación entre ambas medidas, sirven de referencia para examinar la asimetría de la distribución (Rial Boubeta, 2008).
- **Medidas de variabilidad:** El ancho de la caja es una medida de la variabilidad de los datos, ya que representa la amplitud intercuartílica. Por tanto, tanto mayor sea la longitud, mayor será la dispersión de la distribución de los datos (Rial Boubeta, 2008).
- **Medidas de distribución:** Es indicada por la posición de la mediana. Si la línea se sitúa en el centro de la caja, es una distribución simétrica. Si se aproxima a la derecha (Q_3), es una distribución asimétrica negativa, y si se aproxima a la izquierda (Q_1), asimétrica positiva (Rial Boubeta, 2008).
- **Valores fuera del cuerpo central de datos:** Los brazos de la caja indican aquellos valores que quedan fuera del cuerpo central de los datos. Los extremos representan el valor mínimo y el valor máximo (Rial Boubeta, 2008).

Teniendo en cuenta esto, se realizan diagramas de cajas de todas las instancias (**Anexo 2**) tanto para la tardanza total ponderada como para los tiempos de ejecución. Para la tardanza total ponderada por ejemplo (**Gráfico 12**), se tiene que: En las instancias tanto 79 como 101, los datos tienen una distribución asimétrica positiva (existe mayor concentración de los datos hacia los valores mayores a la media). En el caso de la instancia 1, esta tiene distribución asimétrica negativa (existe mayor concentración de los datos hacia los valores menores a la media), y las instancias 33 y 81 tienen una distribución simétrica (las frecuencias en ambos lados son parecidas).

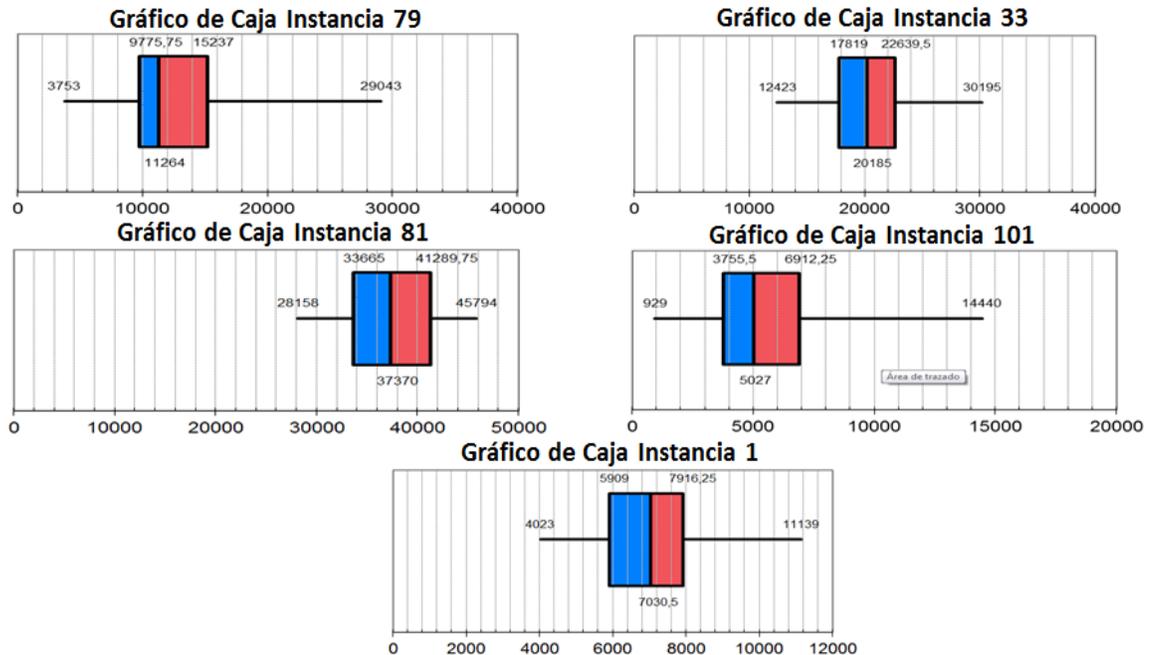


Gráfico 112: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas

En el caso del tiempo promedio de ejecución, el tiempo arrojado por el algoritmo se multiplicó por 1000 para facilitar la realización de la gráfica y se obtiene para las mismas instancias que: tienen una distribución simétrica las instancias 79, 33 y 81, la instancia 101 tiene una distribución asimétrica negativa, mientras que la instancia 1 tiene una distribución asimétrica positiva (**Gráfico 13**).

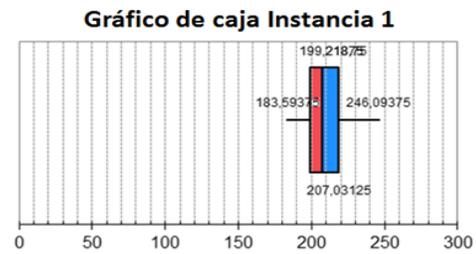
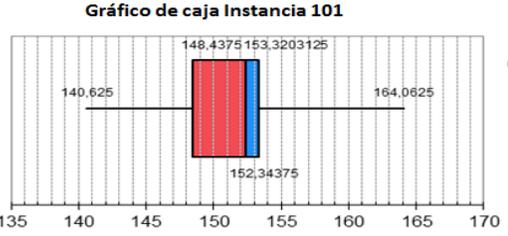
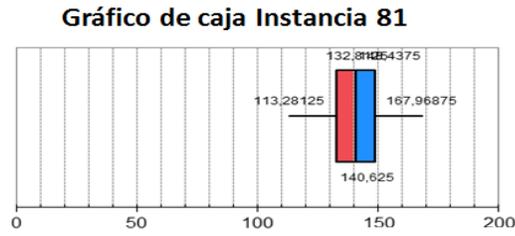
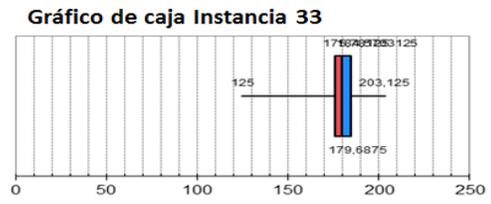
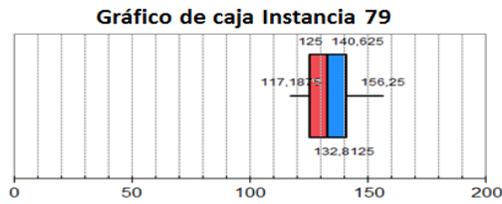


Gráfico 123: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas

CAPÍTULO 3

ALGORITMO DE SOLUCIÓN BASADO EN SISTEMAS MULTIAGENTES

I. DESCRIPCIÓN DEL SISTEMA MULTIAGENTE

Autores como (Wooldridge, 2002) definen un agente como un sistema informático que es capaz de realizar una acción independientemente de su nombre de usuario o propietario. En otras palabras, un agente puede descubrir por sí mismo lo que tiene que hacer con el fin de satisfacer sus objetivos, en lugar de que se les diga explícitamente lo que debe hacer en cada momento. Un sistema Multiagente por lo tanto, es un conjunto de agentes inteligentes (autónomos, independientes, perceptivos, racionales y funcionales) que representan cualquier tipo de entidad dentro de un sistema y se organizan bajo unos principios de comunidad que buscan un objetivo específico (Mařka & Lažanskýa, 2007) (Lee & Kimb, 2008), es decir, bajo un sistema descentralizado y distribuido en agentes, el cual permite perturbaciones y la toma de decisiones por medio de interacciones y negociaciones, que para que sean exitosas deben lograr un ambiente de cooperación, colaboración y capacidad de aprendizaje al momento de solucionar problemas (Nwana & Ndumu, 1997)

El sistema Multiagente se empieza a ejecutar tan pronto se elige una acción para ser ejecutada por los agentes. Una vez definida, el medio ambiente puede responder con un número de estados posibles; sin embargo, sólo un estado se llevará a cabo. Posteriormente, sobre la base del segundo estado, el agente de nuevo elige una acción a realizar y el entorno responde con un conjunto de estados posibles, y así sucesivamente (Wooldridge, 2002) hasta lograr la negociación entre agentes deseada.

Así pues, para el caso de la programación de la producción en una máquina, el entorno inicial está dado por una máquina i y varias tareas j que se deben llevar a cabo. La primera acción a ejecutar es determinar que trabajo j se debe realizar primero, teniendo en cuenta el tiempo de procesamiento (pij) y la fecha de entrega de las tareas (dj). Sin embargo, a medida que se finalice una tarea j , se hace una nueva revisión de las que aún falta realizar y se entra en una nueva negociación de los agentes (esto hace que las acciones realizadas por el agente juegan un papel importante en la determinación del estado actual (Wooldridge, 2002)), por lo cual las perturbaciones son incluidas en la siguiente negociación, y estas se deben repetir cuantas veces sean necesarias, hasta ejecutar todas las tareas que se deben realizar.

Sin embargo, para el desarrollo de estas negociaciones existen unos aspectos que se deben tener en cuenta a la hora de iniciar la comunicación entre agentes (Rodríguez, Insfran, & Cernuzzi, 2011):

- **Perspectiva organizacional:** La organización es representada como un sistema que tiene ciertos objetivos. Ésta alcanza estos objetivos a través de acciones coherentes, que utilizan recursos del sistema y alteran el estado deseado del sistema (Falkenberg, 1998). Algunos autores como (Zambonelli, Jennings, &

Wooldridge, 2003) afirman que cada agente tiene una serie de responsabilidades o metas en el contexto de todo el sistema y es responsable de actuar de forma autónoma. Además, las interacciones ayudan a caracterizar la estructura general de la organización y la posición del agente, y la evolución de las actividades de la organización determinan la consecución del objetivo de aplicación (Rodríguez, Insfran, & Cernuzzi, 2011).

- **Perspectiva estructural:** Ésta muestra la arquitectura del sistema en términos de entidades y la relación estática entre ellos, proporcionando de esta forma una perspectiva abstracta del sistema (Rodríguez, Insfran, & Cernuzzi, 2011). Una arquitectura hace que en un sistema todos los componentes se puedan controlar y, por lo tanto, se puedan incorporar muchas responsabilidades de toma de decisiones, incluida la programación, rotación, y la asignación de recursos. Además, la arquitectura de control determina las interrelaciones entre los componentes de control, con lo que se establece el mecanismo de coordinación de la ejecución de esas decisiones (Dilts, Boyd, & Whorms, 1991).

En la **tabla 10** se describen las cuatro formas básicas de arquitecturas de control para sistemas, identificadas por (Dilts, Boyd, & Whorms, 1991): centralizada, jerárquica, modificado jerárquico y heterárquica (**Gráfica 14**).

ARQUITECTURA	CARACTERÍSTICAS	VENTAJAS	DESVENTAJAS
Centralizada	<p>Único ordenador central.</p> <p>Todas las decisiones de control se toman en una sola ubicación.</p> <p>Una base de datos global registra todas las actividades del sistema.</p>	<p>Acceso a información global.</p> <p>Posible optimización global.</p> <p>Existe una fuente única del estado de la información del sistema.</p>	<p>Velocidad baja de respuesta.</p> <p>Dependencia de la única unidad de control.</p> <p>Dificultad de modificar el software de control.</p>
Jerárquica	<p>Múltiple, variedad de computadores</p> <p>Relación rígida patrón/esclavo entre los niveles de toma de decisiones</p> <p>El supervisor coordina todas las actividades de los subordinados</p> <p>Bases de datos</p>	<p>Implementación gradual, redundancia y reducción de los problemas de desarrollo de software.</p> <p>Adición incremental de control.</p> <p>Posibilidad de comportamientos adaptativos.</p> <p>Diferentes escalas de</p>	<p>Limitaciones computacionales de los controladores locales.</p> <p>Aumento del número de enlaces de comunicación entre niveles.</p> <p>Dificultades para tratar el control adaptativo dinámico.</p> <p>Dificultad de hacer futuras modificaciones imprevistas.</p>

	agregadas en cada nivel	tiempo permitidas Tiempos de respuesta rápidos	
Modificado jerárquico	Múltiple, variedad de computadores. Relación flexible patrón/esclavo entre los niveles de toma de decisión. El supervisor inicia la secuencia de actividades de los subordinados Los subordinados negocian para completar la secuencia	Todas las ventajas del control Jerárquico. Los sistemas locales tienen autonomía. Habilidad de transferir algunas tareas de vinculación a los controladores locales.	La mayoría de las desventajas de la forma Jerárquica. Problemas de conexión. Limitación a los controladores de bajo nivel Incremento en la dificultad del diseño del sistema de control
Heterarquica	Múltiple, pero baja variedad, de computadores. No relación patrón/esclavo. Autonomía local total. Actividades de toma de decisiones distribuida. Bases de datos locales solamente.	Autonomía local completa. Complejidad computacional reducida. Sistema tolerante a las fallas implícito. Facilidad de reconfiguración y adaptación. Difusión más rápida de información.	Principalmente limitaciones de los controladores. Requiere alta capacidad de red Falta de software disponible.

Tabla 10: Resumen de las características de la arquitectura de control, ventajas y desventajas (Dilts, Boyd, & Whorms, 1991)

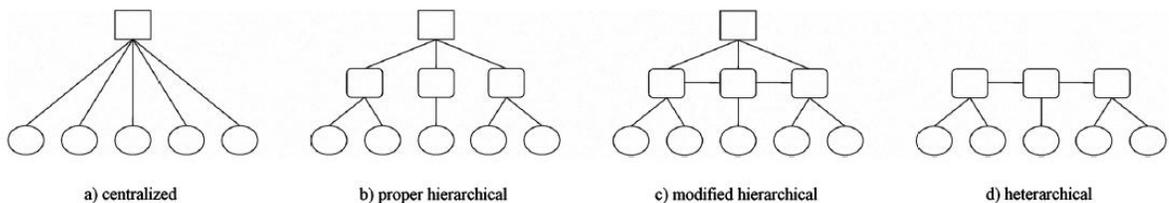


Gráfico 134: Cuatro Formas básicas de arquitecturas de control (Dilts, Boyd, & Whorms, 1991)

En esta figura, los círculos representan los controladores de equipos, los cuadrados redondeados representan controladores de estaciones de trabajo, los cuadrados representan el controlador de unidades de producción, y las líneas de conexión muestran las interrelaciones de control, a través de una red de área local.

- **Perspectiva funcional:** Un evento es algo que ocurre en el medio ambiente y para la que el papel de organización reacciona mediante la ejecución de un método o servicio. Esta perspectiva se centra en modelar los requerimientos funcionales que deben cumplir los diferentes agentes en el sistema (Rodríguez, Insfran, & Cernuzzi, 2011).
- **Perspectiva de comportamiento social:** La perspectiva del comportamiento social muestra las posibles secuencias de eventos a los que un agente puede responder, junto con aspectos de la interacción. Esto se representa a menudo como diagramas de estado o actividad (Rodríguez, Insfran, & Cernuzzi, 2011). Como se ha mencionado anteriormente, el sistema requiere características que normalmente no se requieren en los sistemas de software convencional, tales como, pro-actividad adaptabilidad, colaboración, la verdad o disposición. Estas características se expresan como conducta social.

Con el fin de estructurar y organizar adecuadamente las características de los requisitos de comportamiento social, a continuación se presenta la clasificación de las características de los agentes definidos por (A & Narasimhan, 2005). Según los autores, tres atributos principales de un agente se deben definir: (a) la autonomía, que se refiere al hecho de que un agente debe ejecutarse de forma independiente, con poco o ningún intervención humana, (b) la continuidad temporal, lo que significa que un agente debe ejecutarse continuamente en lugar de simplemente realizar una tarea y terminar, y (c) habilidades sociales, lo que significa que un agente debe poseer algún tipo de habilidades sociales, ya que las ventajas del agente residen en su comunicación interactiva con otros agentes. Además de estos atributos esenciales, un agente también se puede clasificar de acuerdo con el siguiente comportamiento social:

- a) Pro-actividad:** Se refiere a cómo el agente reacciona a su entorno, y cómo lleva a cabo sus objetivos. El agente puede reaccionar a los estímulos de su entorno mediante la asignación directamente a una acción, o puede utilizar técnicas de planificación para lograr sus metas (Rodríguez, Insfran, & Cernuzzi, 2011).
- b) Adaptabilidad:** Ésta describe la capacidad de un agente para modificar su comportamiento en el tiempo. Esta función incluye otras sub-funciones tales como el aprendizaje y la subordinación (Rodríguez, Insfran, & Cernuzzi, 2011).
- c) Movilidad:** Se refiere a la capacidad de los agentes de transportar su ejecución entre máquinas en una red. Esta forma de movimiento puede ser física, cuando el agente viaja entre máquinas en una red, o lógica, donde un

agente que se ejecuta en una sola máquina de forma remota accede desde otros lugares en Internet (Rodríguez, Insfran, & Cernuzzi, 2011)

- d) Colaboración:** La colaboración entre los agentes se basa en el éxito de una operación o acción de manera oportuna. Esto se puede lograr al ser capaces de coordinar con otros agentes el envío y recepción de mensajes a través de algún tipo de lenguaje de comunicación. Además, es posible que los agentes se colaboren sin que exista algún tipo de comunicación en tiempo real. La interacción de los agentes con los recursos y su medio ambiente puede conducir a la aparición de un comportamiento cooperativo o competitivo (Rodríguez, Insfran, & Cernuzzi, 2011).
- e) Veracidad:** Se refiere a la capacidad del agente para engañar a otros agentes a través de sus mensajes y conducta. Un agente puede así: ser sincero o mentiroso para tratar de engañar a otros agentes, con intención o sin intención respectivamente, ya sea proporcionando información falsa o actuando de una manera engañosa (Rodríguez, Insfran, & Cernuzzi, 2011).
- f) Disposición:** Se refiere a la "actitud" del agente hacia otros agentes, y su voluntad de cooperar con ellos. Un agente siempre puede intentar realizar una tarea cuando se le pide que lo haga (benevolente), puede actuar en su propio interés de colaborar con otros agentes sólo cuando es conveniente hacerlo (egoísta), o puede tratar de dañar a otros agentes o destruir de alguna manera (malévola) (Rodríguez, Insfran, & Cernuzzi, 2011).

II. DEFINICIÓN OPERACIONES DEL SISTEMA MULTIAGENTE

Teniendo en cuenta la reducida complejidad computacional, la autonomía de los agentes y la facilidad de adaptación y reconfiguración, se decide utilizar la arquitectura Heterárquica como perspectiva estructural y se define que dado que los agentes son totalmente independientes, cada tarea interactúa con la máquina de forma autónoma como lo muestra la siguiente gráfica (**Gráfica 15**): los círculos representan todos los trabajos, el cuadrado redondeado va a representar la máquina, y el cuadrado representa el entorno en el cual se encuentra la máquina. Éste está definido como la programación de la producción con la inclusión de perturbaciones que tienen prioridad, y se realizan una vez se haya terminado el trabajo que se esté ejecutando, con el fin de reducir tiempos de preparación de la máquina.

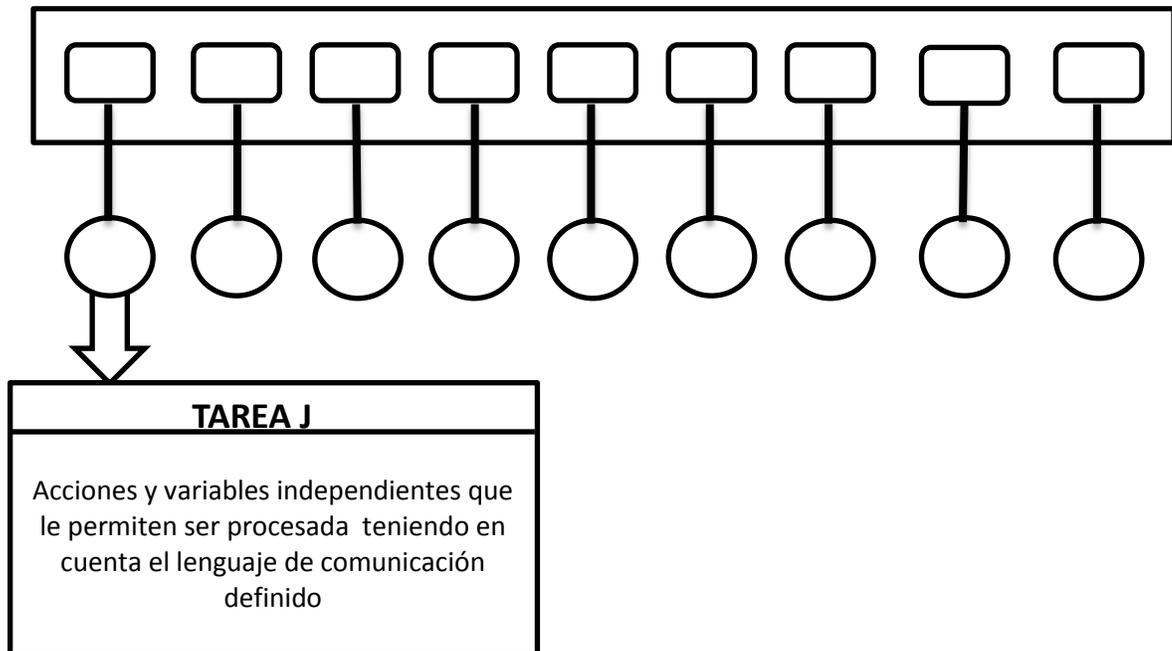


Gráfico 1514: Descripción de la perspectiva estructural utilizada. Realizado por Daniel Páez y Andrea Vargas

Adicionalmente, el lenguaje de comunicación que se utiliza entre los agentes (tareas) y la máquina (agente) se establece como un campo de fuerza (Zbib, Pach, Sallez, & Trentesaux, 2012) en el que el criterio para ser procesado es una relación entre el tiempo que la tarea lleva en cola y la proximidad con la fecha de entrega. Así pues, a medida que la tarea se vaya acercando a d_j va a tener mayor peso para ser procesada y si ya superó d_j va a existir tardanza. Dado que ésta se quiere minimizar, la tarea va a

tener mayor intensidad y prioridad para llevarse a cabo. Estos criterios se establecen teniendo en cuenta la pendiente de la recta, como se indica a continuación (**Gráfico 16**):

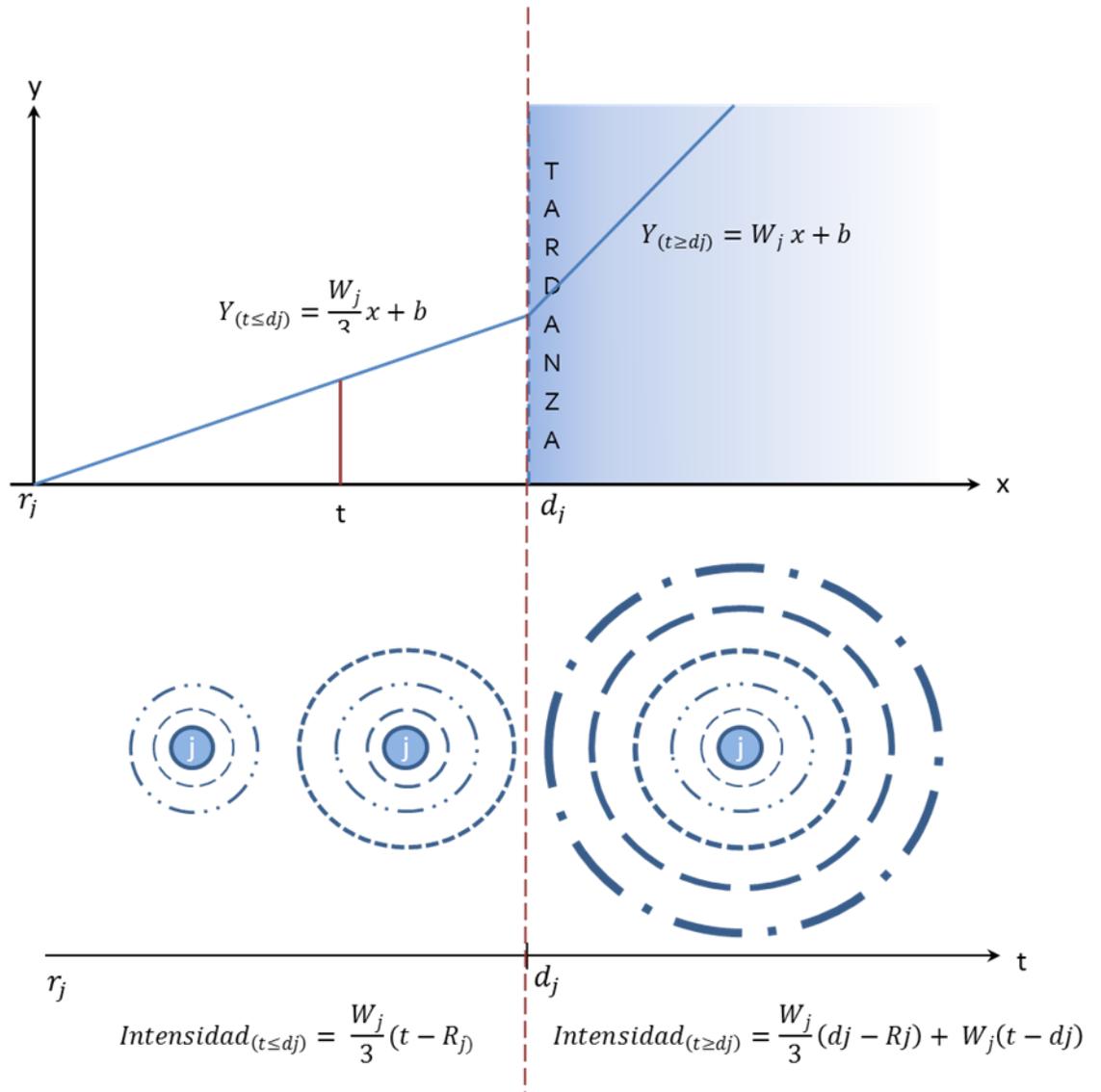


Gráfico 15: Campo de fuerza. Realizado por Daniel Páez y Andrea Vargas

Como se definió anteriormente cada tarea va a representar un agente independiente que se comunica con la máquina que también es un agente independiente. Si se mira una tarea j como una partícula la cual tiene un campo de fuerza dado por la ecuación relacionada, se puede ver que a medida que el tiempo aumenta dicha intensidad aumenta directa y proporcionalmente. Esta intensidad dentro de la simulación del sistema Multiagente se puede asemejar a la capacidad de la tarea de persuadir a la máquina que

su tiempo de entrega se aproxima, y por esto debe ser realizada para no incurrir en tardanza.

De esta manera la máquina va a estar constantemente monitoreando las señales de las tareas y su intensidad, para con base en esto programar la producción on-line. Cabe resaltar que por la arquitectura definida para el sistema las tareas se comunican únicamente con la máquina, esto hace que ésta sea la que toma las decisiones en cuanto al orden de producción. Es importante entender que la diferencia entre este lenguaje de negociación y una regla simple de programación lineal, es que para simular la negociación entre tareas y máquina, se asocia una probabilidad basada en la intensidad de la tarea, de esta manera los resultados finales entre corridas no van a ser los mismos dado a este componente estocástico. En definitiva se puede afirmar que la tarea que mejor capacidad de negociación o “persuasión” va a ser la elegida por la máquina

Adicionalmente, se puede mostrar la capacidad de adaptabilidad del sistema Multiagente, ya que al tomar decisiones on-line, en el momento en el que termina el procesamiento de una tarea, la máquina se actualiza recibiendo las señales de las tareas en cola y con base en esto se toma la decisión de qué trabajo es el siguiente.

Se puede ver además, que dado a que el sistema Multiagente es más liviano y no tiene un componente evolutivo puede que la solución no sean óptimos locales tan altos como los del algoritmo, pero definitivamente el tiempo promedio de procesamiento será menor, ya que mientras este sistema se actualiza on-line el método tradicional se ejecuta cada vez que existe una perturbación.

Teniendo definido esto, los parámetros que se establecen para la realización del sistema Multiagente son:

- ✓ Datos de entrada: Extraídos de OR library
- ✓ Release date: 20 trabajos con $r_j = 0$, 20 trabajos con $r_j =$ distribución uniforme $(0, \frac{3}{4} \sum P_j)$
- ✓ Función objetivo: Minimización de la tardanza ponderada
- ✓ Perturbaciones: Trabajos imprevistos que se realizan una vez se termine la tarea j que se esté ejecutando.
- ✓ Para tener un nivel de confianza del 95% se determina que el número de corridas que se va a realizar con el algoritmo es 100. Esto se obtiene estableciendo **(Gráfico 17)**:

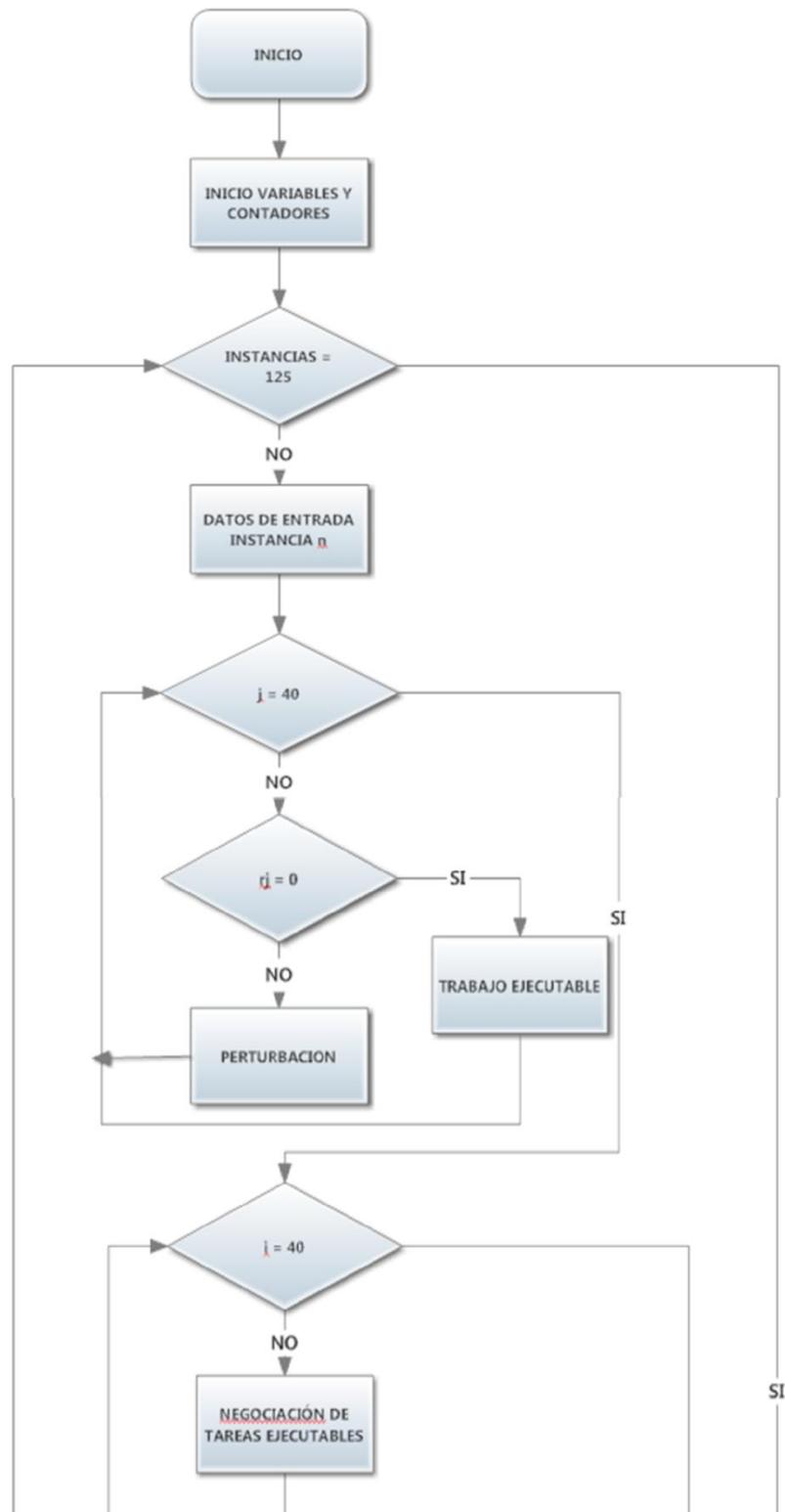
z	1,96	Nivel de Confiabilidad
p	0,5	Probabilidad de Éxito
q	0,5	Probabilidad de Fracaso
ϵ	0,1	Error Propuesto

$$FORMULA \rightarrow n = \frac{z^2 \times p \times q}{\epsilon^2}$$

$$n = 96,04$$

Gráfico 17: Cantidad de corridas del sistema multiagente. Elaborado por Daniel Páez y Andrea Vargas

A continuación, se presenta el diagrama de flujo que permite esquematizar los procesos realizados por el sistema Multiagente (**Gráfico 18**):



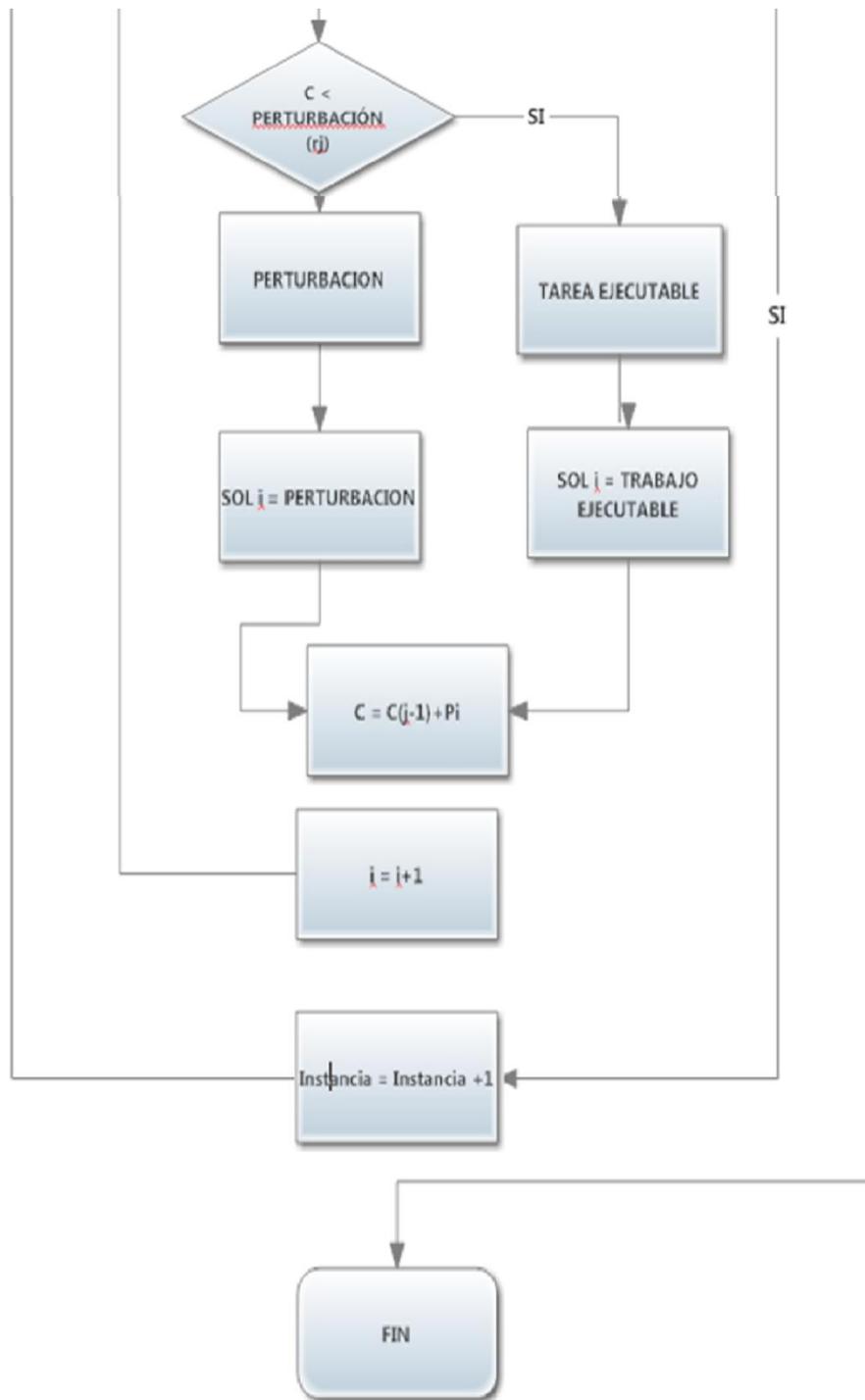


Gráfico 168: Procesos realizados para programar la producción con perturbaciones. Elaborado por Daniel Páez y Andrea Vargas

III. DESCRIPCIÓN DE LA SOLUCIÓN

Luego de haber corrido el sistema Multiagente las 100 veces establecidas, se obtienen 2 indicadores:

1. Tardanza total ponderada
2. Tiempo total de ejecución

En la siguiente tabla (**Tabla 11**), se pueden observar tanto el promedio de la tardanza total ponderada como el promedio del tiempo de ejecución, para cada una las 125 instancias.

Adicionalmente, los resultados (para la instancia 1 por ejemplo) se pueden ver en las siguiente gráficas, donde se indica (por intervalos) la frecuencia de la tardanza total ponderada (**Gráfico 19**) y la frecuencia del tiempo de ejecución respectivamente (**Gráfico 20**)

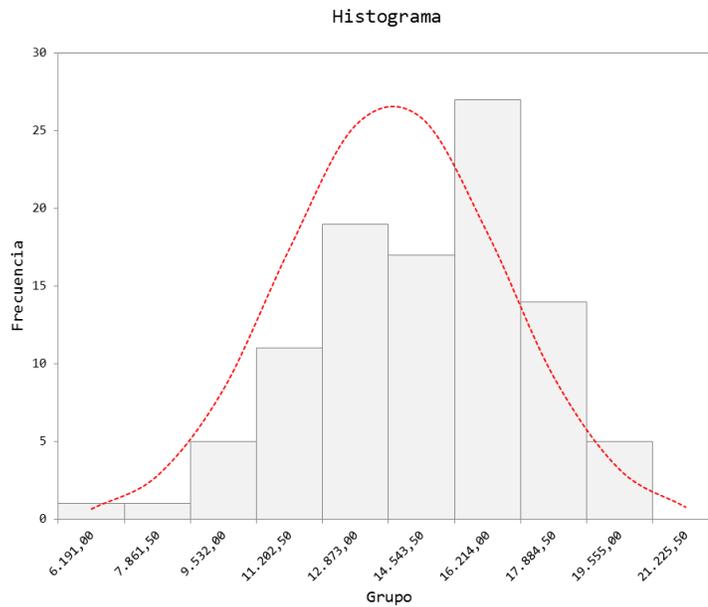
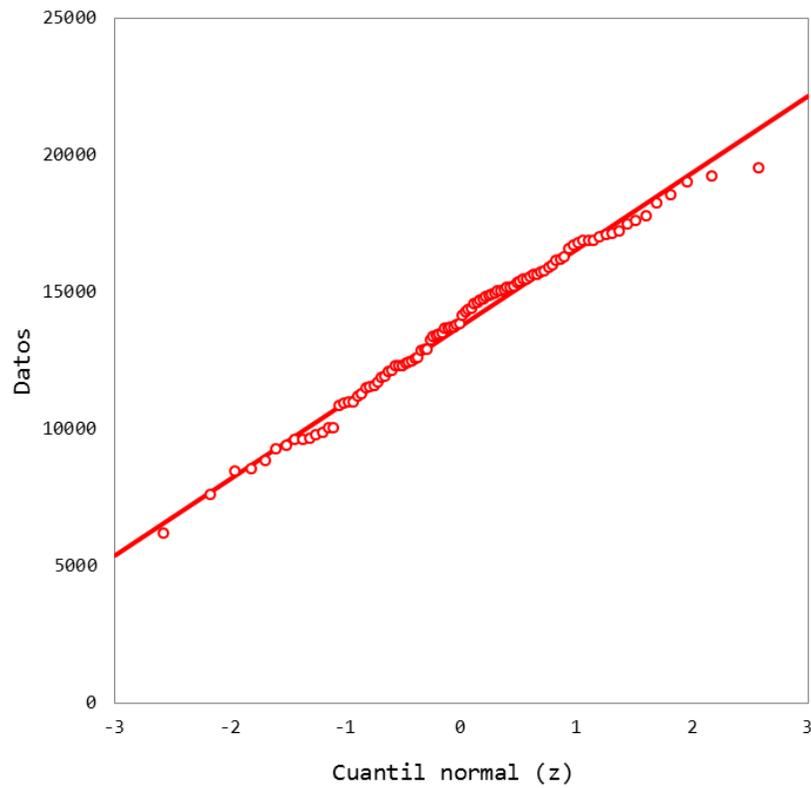


Gráfico de probabilidad normal



Datos Descriptivos		Grupo	Frecuencia
Cuenta	100	6191	1
Promedio	13793,76	7861,5	1
Desviación	2794,13672	9532	5
Varianza	7807200,02	11202,5	11
Curtosis	-0,4215696	12873	19
Sesgo	-0,2815577	14543,5	17
Mínimo	6191	16214	27
Máximo	19555	17884,5	14
Rango	13364	19555	5
Norm (p-valor)	0,15103667	21225,5	0

# CLASE	LIM INFERIOR	LIM SUPERIOR	FRECUENCIA
1	6191	7527.4	1
2	7527.4	8863.8	3
3	8863.8	10200.2	10
4	10200.2	11536.6	7
5	11536.6	12873	15
6	12873	14209.4	15
7	14209.4	15545.8	21
8	15545.8	16882.2	16
9	16882.2	18218.6	9
10	18218.6	19555	3

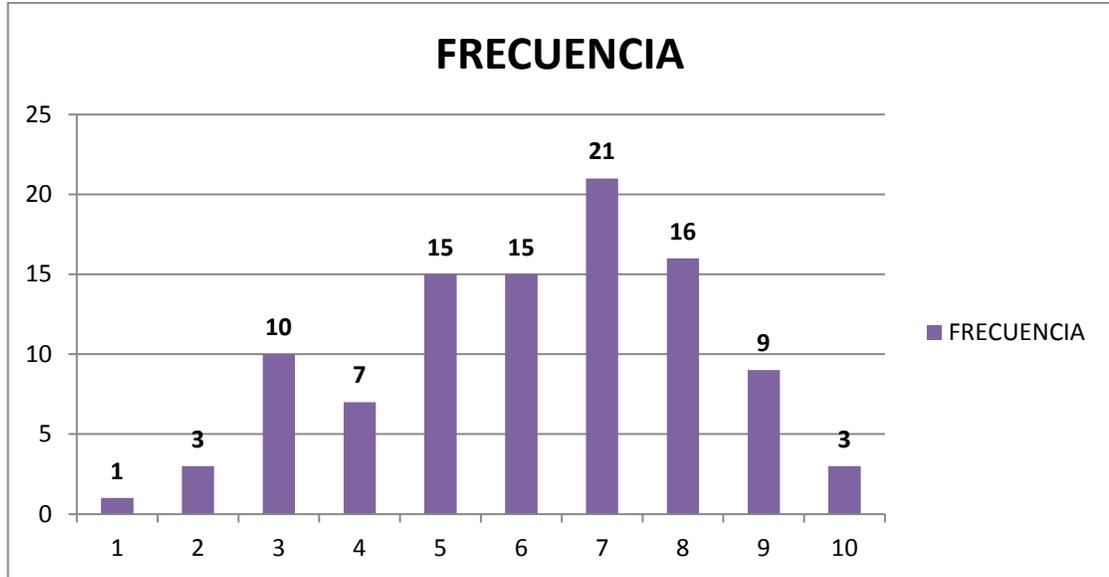
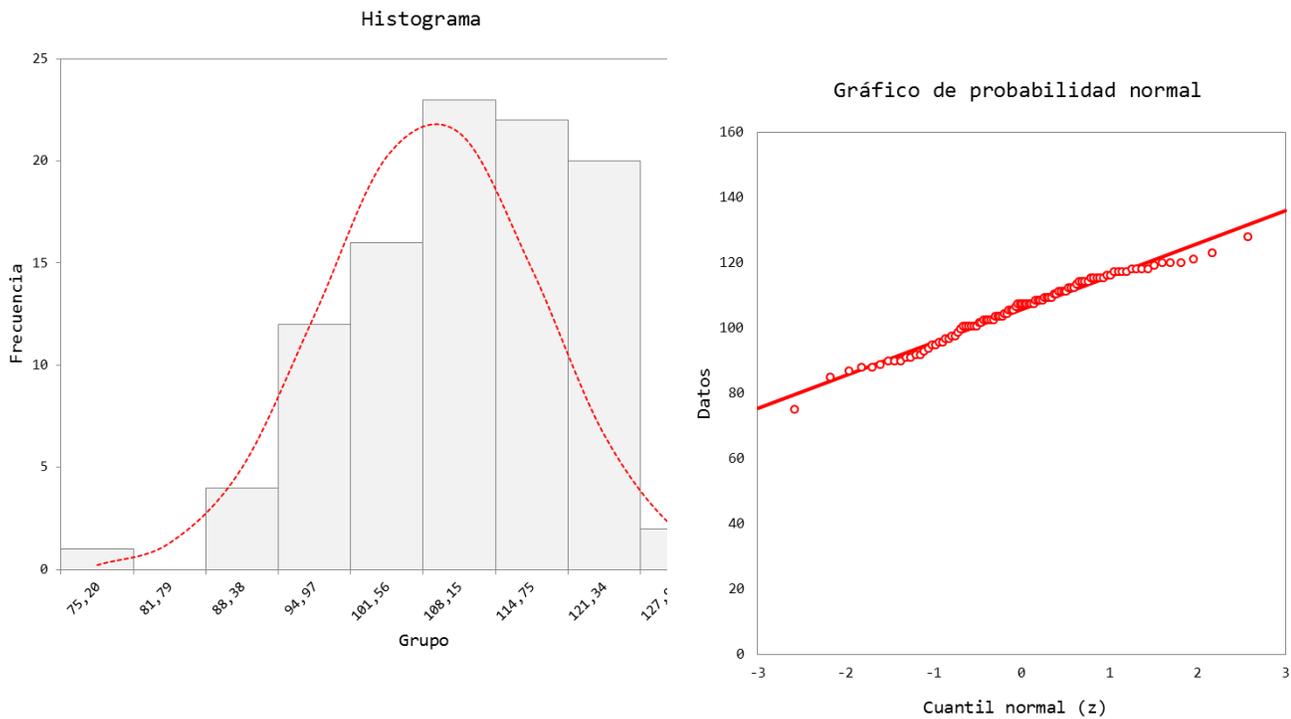


Gráfico 17: Frecuencia de la tardanza total ponderada para la Instancia 1. Elaborado por Daniel Páez y Andrea Vargas

INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN
1	13793,76	105,7128906	43	160566,53	81,80664063	85	30438,39	78,03710938
2	10389,27	86,89453125	44	164333,13	81,63085938	86	109587,25	88,19335938
3	13911,68	83,44726563	45	128067,27	80,22460938	87	99273,92	82,08984375
4	14991,58	85,3515625	46	137147,36	89,05273438	88	81859,4	82,16796875
5	9786,38	83,10546875	47	214917,68	79,39453125	89	122299,26	83,0078125
6	38865,75	89,95117188	48	179972,09	87,95898438	90	69069,62	88,22265625
7	34692,33	92,32421875	49	180063,64	80,9765625	91	147055,71	86,46484375
8	37844,15	86,15234375	50	185288,35	88,83789063	92	135945,59	88,70117188
9	45843,57	84,765625	51	31958,72	82,32421875	93	129315,26	85,49804688
10	41683,04	84,55078125	52	19957,77	84,78515625	94	129439,51	88,15429688
11	85424,77	87,15820313	53	20089,34	80,69335938	95	151737,34	83,26171875
12	63050,57	88,671875	54	16945,34	84,52148438	96	205428,17	82,95898438
13	91249,44	83,49609375	55	21766,85	76,86523438	97	206188,88	83,13476563
14	69012,34	86,72851563	56	25124,74	81,10351563	98	215669,33	86,34765625
15	86315,34	82,9296875	57	51263,52	83,21289063	99	206227,13	78,671875
16	180059,93	89,35546875	58	60455,13	81,58203125	100	269927,65	85,625
17	155370,04	86,15234375	59	60761,73	76,70898438	101	17354,27	84,56054688
18	160038,21	84,38476563	60	37565,69	85,04882813	102	30260,8	83,06640625
19	169976,57	81,24023438	61	72569,8	89,15039063	103	25220,39	82,23632813
20	145369,9	81,29882813	62	65898,61	80,09765625	104	29962,67	84,0625
21	162109,28	80,67382813	63	91548,63	87,39257813	105	20014,42	86,50390625
22	159477,54	86,37695313	64	91507,02	86,93359375	106	57312,15	82,91992188
23	239146,57	78,72070313	65	81060,28	79,54101563	107	72960,09	83,97460938
24	227794,85	79,375	66	158211,48	78,59375	108	61288,48	81,42578125
25	218568,57	82,82226563	67	167940,92	83,33007813	109	47807,8	81,92382813
26	16587,49	81,11328125	68	180726,45	84,43359375	110	41330,7	88,29101563
27	13673,51	80,77148438	69	161608,78	86,5625	111	130938,48	83,31054688
28	10958	85,43945313	70	177587,58	84,70703125	112	90319,77	84,3359375
29	8363,7	81,24023438	71	185525,5	88,50585938	113	94460,15	81,2109375
30	9446,96	81,2890625	72	242834,74	80,81054688	114	107358,2	85,3515625
31	47077,08	78,90625	73	156830,82	86,92382813	115	92934,53	79,11132813
32	47924,68	79,765625	74	180179,2	82,60742188	116	144526,55	80,46875
33	33574,78	81,31835938	75	175809,69	80,20507813	117	145740,65	88,97460938
34	25000,57	81,92382813	76	17329,33	88,7109375	118	105326,3	82,39257813
35	39543,82	87,23632813	77	21587,09	82,97851563	119	187087,57	82,16796875
36	88817,22	80,546875	78	15145,36	76,25	120	162895,12	82,08984375
37	67991,26	80,2734375	79	27261,63	81,10351563	121	211318,44	85,72265625
38	77926,98	81,04492188	80	28229,09	80,53710938	122	181181,69	82,8515625
39	89930,49	79,90234375	81	42290,22	81,3671875	123	175868,27	85,2734375
40	77000,06	79,5703125	82	47611,72	84,03320313	124	167107,34	86,03515625
41	131186,41	81,33789063	83	55051,96	82,2265625	125	199879,88	84,73632813
42	190602,56	86,1328125	84	53770,82	79,09179688			

INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN	INSTANCIA	PROMEDIO TARDANZA TOTAL PONDERADA	PROMEDIO TIEMPO DE EJECUCIÓN
1	13793,76	0,105712891	43	160566,53	0,081806641	85	30438,39	0,078037109
2	10389,27	0,086894531	44	164333,13	0,081630859	86	109587,25	0,088193359
3	13911,68	0,083447266	45	128067,27	0,080224609	87	99273,92	0,082089844
4	14991,58	0,085351563	46	137147,36	0,089052734	88	81859,4	0,082167969
5	9786,38	0,083105469	47	214917,68	0,079394531	89	122299,26	0,083007813
6	38865,75	0,089951172	48	179972,09	0,087958984	90	69069,62	0,088222656
7	34692,33	0,092324219	49	180063,64	0,080976563	91	147055,71	0,086464844
8	37844,15	0,086152344	50	185288,35	0,088837891	92	135945,59	0,088701172
9	45843,57	0,084765625	51	31958,72	0,082324219	93	129315,26	0,085498047
10	41683,04	0,084550781	52	19957,77	0,084785156	94	129439,51	0,088154297
11	85424,77	0,087158203	53	20089,34	0,080693359	95	151737,34	0,083261719
12	63050,57	0,088671875	54	16945,34	0,084521484	96	205428,17	0,082958984
13	91249,44	0,083496094	55	21766,85	0,076865234	97	206188,88	0,083134766
14	69012,34	0,086728516	56	25124,74	0,081103516	98	215669,33	0,086347656
15	86315,34	0,082929688	57	51263,52	0,083212891	99	206227,13	0,078671875
16	180059,93	0,089355469	58	60455,13	0,081582031	100	269927,65	0,085625
17	155370,04	0,086152344	59	60761,73	0,076708984	101	17354,27	0,084560547
18	160038,21	0,084384766	60	37565,69	0,085048828	102	30260,8	0,083066406
19	169976,57	0,081240234	61	72569,8	0,089150391	103	25220,39	0,082236328
20	145369,9	0,081298828	62	65898,61	0,080097656	104	29962,67	0,0840625
21	162109,28	0,080673828	63	91548,63	0,087392578	105	20014,42	0,086503906
22	159477,54	0,086376953	64	91507,02	0,086933594	106	57312,15	0,082919922
23	239146,57	0,078720703	65	81060,28	0,079541016	107	72960,09	0,083974609
24	227794,85	0,079375	66	158211,48	0,07859375	108	61288,48	0,081425781
25	218568,57	0,082822266	67	167940,92	0,083330078	109	47807,8	0,081923828
26	16587,49	0,081113281	68	180726,45	0,084433594	110	41330,7	0,088291016
27	13673,51	0,080771484	69	161608,78	0,0865625	111	130938,48	0,083310547
28	10958	0,085439453	70	177587,58	0,084707031	112	90319,77	0,084335938
29	8363,7	0,081240234	71	185525,5	0,088505859	113	94460,15	0,081210938
30	9446,96	0,081289063	72	242834,74	0,080810547	114	107358,2	0,085351563
31	47077,08	0,07890625	73	156830,82	0,086923828	115	92934,53	0,079111328
32	47924,68	0,079765625	74	180179,2	0,082607422	116	144526,55	0,08046875
33	33574,78	0,081318359	75	175809,69	0,080205078	117	145740,65	0,088974609
34	25000,57	0,081923828	76	17329,33	0,088710938	118	105326,3	0,082392578
35	39543,82	0,087236328	77	21587,09	0,082978516	119	187087,57	0,082167969
36	88817,22	0,080546875	78	15145,36	0,07625	120	162895,12	0,082089844
37	67991,26	0,080273438	79	27261,63	0,081103516	121	211318,44	0,085722656
38	77926,98	0,081044922	80	28229,09	0,080537109	122	181181,69	0,082851563
39	89930,49	0,079902344	81	42290,22	0,081367188	123	175868,27	0,085273438
40	77000,06	0,079570313	82	47611,72	0,084033203	124	167107,34	0,086035156
41	131186,41	0,081337891	83	55051,96	0,082226563	125	199879,88	0,084736328
42	190602,56	0,086132813	84	53770,82	0,079091797			

Tabla 11: Promedio tardanza total ponderada y tiempo de ejecución para cada una de las instancias (luego de 100 corridas). Elaborado por Daniel Páez y Andrea Vargas



Datos Descriptivos		Grupo	Frecuencia
Cuenta	100	75,1953125	1
Promedio	105,712891	81,78710938	0
Desviación	10,0642617	88,37890625	4
Varianza	101,289364	94,97070313	12
Curtosis	-0,2718721	101,5625	16
Sesgo	-0,3991274	108,1542969	23
Mínimo	75,1953125	114,7460938	22
Máximo	127,929688	121,3378906	20
Rango	52,734375	127,9296875	2
Norm (p-valor)	0,08518861	134,5214844	0

Gráfico 20: Frecuencia del tiempo de ejecución. Elaborado por Daniel Páez y Andrea Vargas

Después de realizar la un test de normalidad, se puede observar en estos gráficos (Histograma y Gráfico de probabilidad Normal), que los datos tienden a formar una campana de Gauss, además podemos afirmar que con un 95% de confianza los datos tienden a una distribución normal. Esto significa que son más probables los valores cercanos a la media y conforme nos separamos de esta, la probabilidad va decreciendo de igual forma a derecha e izquierda.

Por otro lado, se realizan diagramas de cajas de todas las instancias (**Anexo 2**) tanto para la tardanza total ponderada como para los tiempos de ejecución. Para la tardanza total ponderada por ejemplo (**Gráfico 21**), se tiene que: En la instancia 51, los datos tienen una distribución asimétrica positiva (existe mayor concentración de los datos hacia los valores mayores a la media). En el caso de las instancias 1, 26 y 103, estas tienen distribución asimétrica negativa (existe mayor concentración de los datos hacia los valores menores a la media), y la instancia 52 tiene una distribución simétrica (las frecuencias en ambos lados son parecidas).

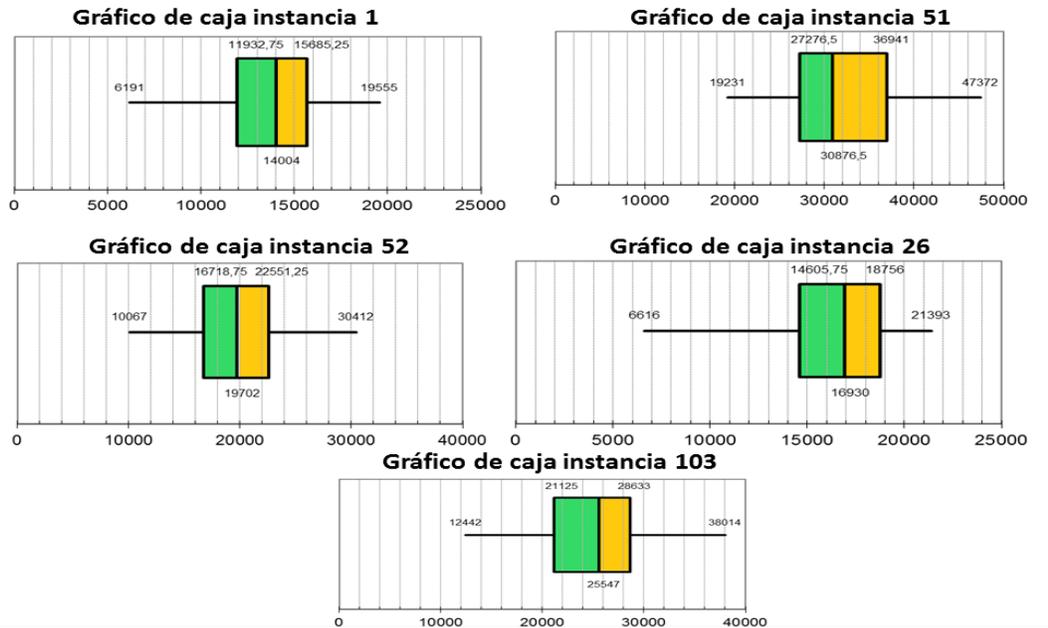


Gráfico 18: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas

En el caso del tiempo promedio de ejecución, el tiempo arrojado por el sistema Multiagente se multiplica por 1000 para facilitar la realización de la gráfica y se obtiene para las mismas instancias que (**Gráfico 22**): tienen una distribución simétrica las instancias 1, 52 y 103, mientras que las instancias 51 y 26 tienen una distribución

asimétrica

negativa.

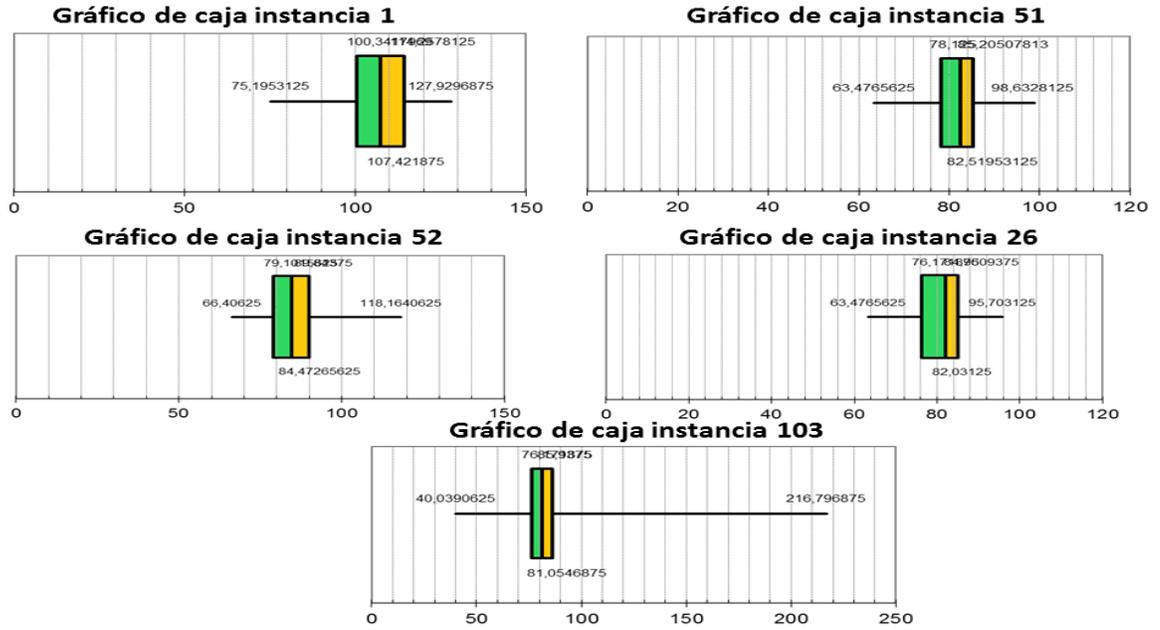


Gráfico 19: Gráficos de caja Instancias aleatorias. Elaborado por Daniel Páez y Andrea Vargas

CAPÍTULO 4

METODOLOGÍA DE COMPARACIÓN

I. DESCRIPCIÓN DE LA METODOLOGÍA

Un experimento puede definirse como una prueba o serie de pruebas en las que se hacen cambios deliberados en las variables de entrada de un proceso o sistema para observar e identificar las razones de los cambios que pudieran observarse en la respuesta de la salida (Montgomery, 2004).

En general, los experimentos se usan para estudiar el desempeño de procesos y sistemas. El proceso o sistema puede representarse con un modelo como el de la siguiente gráfica (**Gráfica 23**). El proceso puede por lo general visualizarse como una combinación de máquinas, métodos, personas y otros recursos que transforman cierta entrada en una salida que tienen una o más respuestas observables. Algunas variables del proceso son controlables, mientras que otras no lo son (Montgomery, 2004).

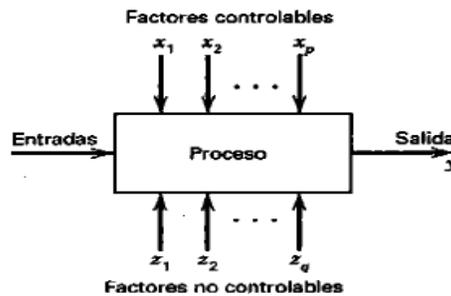


Gráfico 203: Modelo general de un proceso o sistema (Montgomery, 2004)

Teniendo claro esto, se puede definir un diseño de experimentos como un modelo estadístico clásico cuyo objetivo es averiguar si unos determinados factores influyen en una variable de interés y, si existe influencia de algún factor, cuantificar dicha influencia.

El objetivo del experimento es estudiar si cuando se utiliza un determinado tratamiento se produce una mejora en el proceso o no. Adicionalmente, la metodología del diseño de experimentos estudia cómo variar las condiciones habituales de realización de un proceso empírico para aumentar la probabilidad de detectar cambios significativos en la respuesta, de esta forma se obtiene un mayor conocimiento del comportamiento del proceso de interés.

En ingeniería, por ejemplo, la experimentación desempeña un papel importante en el diseño de productos nuevos, el desarrollo de procesos de manufactura y mejoramiento de procesos. En este ámbito, la aplicación de las técnicas del diseño experimental permiten (Montgomery, 2004):

- Realizar mejoras en el rendimiento del proceso
- Reducir la variabilidad para obtener conformidad más cercana con los requerimientos.

- Reducir tiempos
- Reducir costos
- Evaluar y comparar la configuración de diseños básicos.
- Evaluar materiales alternativos.
- Seleccionar parámetros de diseño para que el producto tenga un buen funcionamiento.
- Determinar parámetros claves en el diseño del producto

Dadas todas las aplicaciones y ventajas que tienen los diseños de experimentos, se decide realizar un diseño que permita comparar los resultados arrojados por el algoritmo genético y el sistema Multiagente, para determinar estadísticamente, cuál modelo es más eficiente en la programación de la producción en una máquina, cuando se desea reducir la tardanza total ponderada.

Como los criterios de evaluación de los modelos (tardanza total ponderada y tiempo de ejecución) son independientes, es necesario realizar un diseño de experimentos para cada uno de ellos. Adicionalmente, la comparación de los resultados no se puede realizar entre instancias, pues estas tienen condiciones diferentes, por lo tanto, se van a ejecutar 2 diseños de experimentos para cada una de las 125 instancias.

Los experimentos que se van a realizar son completamente aleatorizados, es decir, aquellos en los que el ambiente en el que se aplica los tratamientos es lo más uniforme posible, dado que las observaciones fueron tomadas aleatoriamente en momentos de tiempo diferentes (Montgomery, 2004). Con esto claro, se establece que:

- Los diseños van a tener un solo factor (tardanza total ponderada o tiempo de ejecución)
- Los diseños van a tener dos tratamientos o niveles de factor (algoritmo genético y sistema Multiagente).
- Los datos de entrada serán los arrojados por las 100 corridas realizadas para cada una de las instancias en estudio.
- El nivel de confianza seleccionado para el desarrollo del diseño de experimentos es de 99%
- El modelo de medias que se utiliza está dado por la siguiente ecuación:

$$x_{ij} = \mu + \tau_i + \varepsilon_{ij} \begin{cases} i = 1,2,3, \dots, 100 \\ j = 1,2 \end{cases}$$

donde μ es un parámetro común a todos los tratamientos conocido como media global, y τ_i es un parámetro único del tratamiento i -ésimo al que se le llama el efecto del tratamiento i -ésimo

Posteriormente, se lleva a cabo una prueba formal de la hipótesis en la que no hay diferencia en las medias de los tratamientos:

$$\begin{cases} H_0: \mu_1 = \mu_2 \\ H_1: \mu_1 \neq \mu_2 \end{cases}$$

Esta se realiza utilizando un análisis de varianza Anova, con el objetivo de comparar los diversos valores medios para determinar si alguno de ellos difiere significativamente del resto. De ser así, se puede seleccionar uno de los dos modelos como el más eficiente para reducir la tardanza total ponderada, cuando se programa la producción en una máquina admitiendo perturbaciones que tienen prioridad para ser realizadas.

En la siguiente tabla (**Tabla 12**), se muestran las operaciones para la realización del análisis Anova, donde k es el número de tratamientos, y N es el número de observaciones:

Fuente de variación	Suma de Cuadrados	Grados de libertad	Cuadrados medios	F calculado
Tratamiento	$\sum_{k=1}^K n_k (\bar{x}_k - \bar{x})^2$	K - 1	$\frac{\text{Suma de cuadrados del tratamiento}}{\text{Grados de libertad del tratamiento}}$	$\frac{\text{Cuadrados medios de tratamiento}}{\text{Cuadrados medios del error}}$
Error	$\sum_{k=1}^K \sum_{j=1}^{n_k} (x_{kj} - \bar{x}_k)^2$	N - K	$\frac{\text{Suma de cuadrados del error}}{\text{Grados de libertad del error}}$	
Total	$\sum_{k=1}^K \sum_{j=1}^{n_k} (x_{kj} - \bar{x})^2$	N - 1		

Tabla 12: Cuadro de Anova. (Montgomery, 2004)

Luego de realizar este análisis, H_0 deberá rechazarse y concluirse que hay diferencia en las medias de los tratamientos si: $F_0 > F_{\alpha, k-1, N-k}$ (Montgomery, 2004)

Una vez rechaza la hipótesis nula, se realiza la prueba de Tukey la cual utiliza la distribución del estadístico estudentizado:

$$q = \frac{\overline{X}_{max} - \overline{X}_{min}}{\sqrt{\frac{MS_E}{n}}}$$

donde \overline{X}_{max} y \overline{X}_{min} son las medias muestrales mayor y menor, respectivamente. Con una significancia de 0,01 para esta prueba, se establecerá que dos medias son significativamente diferentes si el valor absoluto de sus diferencias muestrales excede:

$$T_\alpha = q_\alpha(a, f) \sqrt{\frac{MS_E}{n}}$$

Una vez realizada la prueba de Tukey, se puede seleccionar una como mejor o peor estadísticamente. En este caso, dado que se busca minimizar la tardanza total ponderada, se elegirá el modelo con la menor media; de igual manera, para el tiempo de ejecución, se elegirá el modelo con la menor media.

II. DESCRIPCIÓN DE LA SOLUCIÓN Y ANÁLISIS DE RESULTADOS

Como se mencionó anteriormente, la metodología de comparación se realiza para cada una de las 125 instancias, y para la tardanza total ponderada y el tiempo de ejecución.

En el caso de la tardanza total ponderada, para la instancia 125, por ejemplo, se obtiene que:

Fuente de variación	Suma de Cuadrados	Grados de libertad	Cuadrados medios	F calculado	F teórico	Hipótesis
Tratamiento	23161911881	1	2,3162E+10	490,506379	3,88885293	Se rechaza Ho
Error	9349641001	198	47220409,1			
Total	32511552882	199				

Tabla 13: Tabla Anova para la tardanza total ponderada de la instancia 125. Realizado por Daniel Páez y Andrea Vargas

Dado que $F_{\text{calculado}} > F_{\text{teórico}}$ con un nivel de confianza de 99% la hipótesis nula se rechaza, y con esto se puede realizar la prueba de Tukey, en la que se establece que para esta instancia el mejor modelo es el algoritmo genético.

 Yas - Ymas 		DSH	Mejor	Algoritmo genético
21522,97	>	1903,46389		

Este procedimiento se realizó para las 125 instancias con el mismo nivel de confianza del análisis de varianza (**Anexo 4**) y se obtiene que: para el 99.20% de las instancias (124 instancias), el mejor modelo es el algoritmo genético y para el 0,8% de las instancias (1 instancia) el mejor modelo es sistema Multiagente:

	Tardanza Ponderada
Algoritmo Genético	124
Sistema Multiagente	1

Por otro lado, en el caso del tiempo promedio de ejecución, para la instancia 125, por ejemplo, se obtiene que:

Fuente de variación	Suma de Cuadrados	Grados de libertad	Cuadrados medios	F calculado	F teórico	Hipótesis
Tratamiento	0,299329419	1	0,29932942	5565,80725	3,88885293	Se rechaza Ho
Error	0,010648451	198	5,378E-05			
Total	0,30997787	199				

Tabla 14: Tabla Anova para la tardanza total ponderada de la instancia 125. Realizado por Daniel Páez y Andrea Vargas

Dado que $F_{\text{calculado}} > F_{\text{teórico}}$ con un nivel de confianza de 99%, la hipótesis nula se rechaza, y con esto se puede realizar la prueba de Tukey, en la que se establece que para esta instancia el mejor modelo es sistema Multiagente.

 Yas - Ymas 		DSH		
0,07737305	>	0,00203138	Mejor	Sistema Multiagente

Este procedimiento se realizó para las 125 instancias con un nivel de significancia de 0,01 (**Anexo 5**), y se obtiene que: para el 99.20% de las instancias (124 instancias), el mejor modelo es el sistema Multiagente y para el 0,8% de las instancias (1 instancia) el mejor modelo es el algoritmo genético:

	Tiempo de Ejecución
Algoritmo Genético	1
Sistema Multiagente	124

Teniendo estos resultados, y dado que con este estudio se pretende minimizar la tardanza total ponderada cuando se programa la producción en una máquina admitiendo perturbaciones que tienen prioridad para ser realizadas, el modelo que se elegirá para el caso de estudio es el que haya tenido el mejor desempeño en la evaluación de la función objetivo. Por lo tanto, se concluye que el algoritmo genético es el modelo más eficiente.

CAPÍTULO 5

**EVALUACIÓN Y ANÁLISIS DEL
SISTEMA COMPLEJO PARA EL
PROBLEMA DE PROGRAMACIÓN DE
LA PRODUCCIÓN EN UNA MÁQUINA
*(1/ prmp / $\sum w_j T_j$)”***

I. CONCLUSIONES

Dada la necesidad que tienen las compañías manufactureras de sobresalir en la industria y adquirir una ventaja competitiva en el mercado, la planeación y programación de la producción juegan un papel determinante, pues hasta en los mejores planes se pueden presentar eventos inesperados o perturbaciones. Para controlarlos, es preciso establecer un sistema de inspección sobre los progresos de las operaciones que pueda revelar desviaciones de los planes y poner en marcha medidas correctivas, que permitan mejorar la eficiencia, eficacia y efectividad de los sistemas productivos de dichas industrias.

Dentro de las variables críticas que tienen efectos relevantes en la operación de una industria, está el tiempo. Factores como el tiempo de entrega y el tiempo de procesamiento, juegan un papel determinante y pueden ser criterios diferenciadores entre empresas. Sin embargo, dadas las condiciones de un mercado globalizado y cambiante, es casi inevitable que las industrias incumplan en su totalidad con los tiempos de entrega establecidos, incurriendo de esta forma en tardanzas que obligan a reprogramar la producción.

Con el fin de minimizar dichas tardanzas, se encuentra necesario realizar la simulación de la programación de la producción en una máquina en la que se consideren perturbaciones reales. Dichas perturbaciones, fueron establecidas como tareas que llegan de improviso y deben realizarse de inmediato. Sin embargo, para evitar tiempos innecesarios en la preparación de la máquina para realizar dicha perturbación, se establece que la tarea imprevista se va a poder ejecutar únicamente cuando la tarea que se esté llevando a cabo finalice.

Una vez establecido el sistema complejo que se quiso evaluar, se desarrolló un algoritmo genético con criterio de selección por ruleta, sistema de reproducción uniforme y criterio de mutación por intercambio. Con este, se busca minimizar la tardanza total ponderada cada vez que se realice una tarea, con el fin de hallar la mejor combinación posible de trabajos en tiempo real. En el instante en que una tarea es realizada y se ha incurrido en un tiempo de procesamiento, este algoritmo reorganiza las tareas faltantes para seleccionar la siguiente, de esta forma la producción se va reprogramando, y por lo tanto la tardanza se va minimizando.

Por otro lado, se realizó un sistema Multiagente definido con una arquitectura Heterárquica, en el cual las tareas y las perturbaciones representan agentes independientes que negocian entre sí, e interactúan con la máquina cada vez que se realiza un trabajo. De esta manera, a medida que el tiempo de entrega de un trabajo se va acercando, éste cobra mayor importancia, de tal modo que dentro de la negociación será más relevante y apto para ser ejecutado.

Una vez ejecutados estos dos modelos, se realizó un diseño de experimentos tanto para la tardanza total ponderada, como para el tiempo promedio de ejecución de cada uno. En este se obtuvo, que el algoritmo genético, al ser un modelo más robusto por tener una característica evolutiva que mejora generación tras generación, arrojó mejores resultados para la tardanza total ponderada que el sistema Multiagente. Por su parte, el sistema Multiagente, al ser un modelo más liviano, no requiere tanto tiempo computacional para ser ejecutado, por lo tanto arrojó mejores resultados para el tiempo promedio de ejecución.

En efecto, en el 99,2% de las instancias evaluadas, el algoritmo genético fue 2,16 veces mejor que el sistema Multiagente, es decir la tardanza total ponderada arrojada por el algoritmo genético, aproximadamente fue la mitad de la tardanza total ponderada del sistema Multiagente. Por otro lado, en cuanto al tiempo de ejecución de los modelos, en el 99,2% de las instancias valoradas, el sistema Multiagente fue 2,00 veces mejor que el algoritmo genético, lo que indica que el sistema Multiagente tiene el doble de rapidez computacional que el algoritmo genético.

Ahora bien, como se ha establecido, el objetivo principal de la simulación de la programación de la producción para una máquina es la minimización de la tardanza total ponderada, y aunque el tiempo de ejecución se tomó como un indicador, no es lo que va a determinar el modelo más eficiente a la hora de evaluar la función objetivo; por esta razón se concluye que el algoritmo genético es el método más adecuado para modelar la programación de la producción en una máquina teniendo en cuenta perturbaciones con prioridad.

Las metaheurística son métodos de solución que adquieren una mayor flexibilidad por su gran número de iteraciones. En este caso particular, el algoritmo genético como método de búsqueda, está diseñado para localizar un óptimo dentro de una estructura de datos, mediante un proceso iterativo que permite eliminar soluciones a través de las generaciones hasta llegar a una solución óptima. Este factor hace que el tiempo computacional de ejecución del modelo sea mayor, pero logre alcanzar óptimos locales mejores que el sistema Multiagente.

Teniendo en cuenta lo mencionado anteriormente, se puede concluir que los algoritmos genéticos son modelos eficientes para realizar la programación de la producción en una máquina, con la inclusión de perturbaciones, gracias a que añade flexibilidad y adaptabilidad en la resolución, combinando la robustez y las ventajas de la búsqueda global.

II. RECOMENDACIONES

Dadas las nuevas condiciones del mercado global, las industrias manufactureras están evolucionando hacia la automatización de todos sus procesos, buscando metodologías que les permitan realizar sus tareas de forma eficaz y eficiente. Para este cambio, se deberían tener en cuenta no solo los métodos tradicionales (heurísticas y metaheurística), sino también los Sistemas Multiagentes, ya que los agentes actúan en representación de los usuarios humanos para la realización de tareas complejas, con el fin de colaborar en la solución de un conjunto de problemas o en la consecución de una serie de objetivos individuales o colectivos.

Esta teoría viene cambiando la forma en que se conciben y diseñan los sistemas productivos de las industrias, ya que ha permitido mejorar el desempeño de éstos en la ejecución de tareas en tiempo real y de los imprevistos que puedan suceder a diario. Gracias a las propiedades de los sistemas Multiagentes (autonomía, habilidad social, reactividad, proactividad, movilidad, continuidad temporal, adaptabilidad y aprendizaje) estos obtienen una mayor capacidad productiva, dando origen de esta forma a un trabajo cooperativo más eficiente y adaptable para la solución de problemas más complejos.

Por otro lado, en futuros estudios, se recomienda analizar el caso en el que las perturbaciones no tienen prioridades para ser realizadas, y por lo tanto, en el caso del algoritmo genético, estas hacen parte de las tareas que se deben reorganizar y no entran directamente en el vector solución; mientras que en el caso del sistema Multiagente, estas perturbaciones entran en negociaciones con los agentes para seleccionar qué tarea se debe realizar (si la perturbación o la tarea con $r_j=0$), y luego de esta negociación sí es incluida directamente en el vector solución.

BIBLIOGRAFÍA

- A, H., & Narasimhan, L. (2005). A New Classification Scheme for Software Agents. *Proceedings of the Third International Conference on Information Technology and Applications*, 191-196.
- Agnētis, A., Pacciarelli, D., & Pacifici, A. (2007). Multi-agent single machine scheduling. *Annals of operations research*, 3-15.
- Alkhateeb, F., Maghayreh, E. A., & Doush, I. A. (2011). *Multi-agent systems - Modeling, control, programming, simulations and applications*. Rijeka: Intech Open.
- Amaya Amaya, J. (2007). *Gerencia: Planeación y estrategia*. Bucaramanga: Publiarte.
- Amir Hamidinia, S. K. (2012). A genetic algorithm for minimizing total tardiness/earliness of weighted jobs. *Computers & Industrial Engineering*, 29-38.
- Arranz de la Peña, J., & Parra Tuyol, A. (07 de Junio de 2005). *Algoritmos genéticos*. Recuperado el 20 de Enero de 2013, de <http://www.it.uc3m.es/jvillena/irc/practicas/06-07/05.pdf>
- Arroyo Sánchez, J. (2000). *Modelos y algoritmos para la explotación óptima de la generación en sistemas electrónicos centralizados y competitivos mediante algoritmos genéticos y programación lineal entera mixta*. Castilla, La mancha.
- Baker, K. R., & Trietsch, D. (2009). *Principles of Sequencing and Scheduling*. New Jersey: John Wiley & Sons, Inc.
- Ballard, H. G. (Mayo de 2000). The last planner system of production control. *A thesis submitted to the Faculty of Engineering of The University of Birmingham for the degree of Doctor of Philosophy*. Birmingham, Inglaterra.
- Bar-yam, Y. (2003). *Dynamics of Complex Systems*. Massachusetts: Perseus Books.
- Ben-Daya, M. (2002). Failure Mode and Effect Analysis. *Handbook Maintenance Management Engineering*, 65-80.
- Blum, C. (2005). Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews*, 353–373.
- Caridia, M., & Cavalierib, S. (2007). Multi-agent systems in production planning and control: an overview. *Production Planning & Control: The Management of Operations*, 106-118.
- Castillo Hernández, M. (2006). *Toma de decisiones en las empresas: Entre el arte y la técnica*. Bogotá, D.C.: Ediciones Uniandes.
- Castillo Hernández, M. (2006). *Toma de decisiones en las empresas: Entre el arte y la técnica*. Bogotá, D.C.: Ediciones Uniandes.

- Congram, R., & Potts, C. (2004). Weighted tardiness. *OR Library*, <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/wtinfo.html>.
- Crauwels, H. A., & Wassenhovet, C. L. (Junio de 1996). *Local search heuristics for single-machine scheduling with batching to minimize the number of late jobs*. Recuperado el 2 de Agosto de 2012, de http://flora.insead.edu/fichiersti_wp/inseadwp1996/96-46.pdf
- Darmoula, S., Pierrevalb, H., & Hajri-Gaboujc, S. (2012). Handling disruptions in manufacturing systems: An immune perspective. *Engineering Applications of Artificial Intelligence*, 110–121.
- Dilts, D., Boyd, N., & Whorms, H. (1991). The evolution of control architectures for automated manufacturing systems. *Journal of Manufacturing Systems*, 79-93.
- Diwekar, U. (2008). *Introduction To Applied Optimization*. Nueva York: Springer Science .
- Ericsson, J., & Dahlén, P. (1997). A conceptual model for disruption causes: A personnel and organization perspective. *International Journal of Production Economics*, Pages 47–53.
- Ernani Vieira, G., & Pereira Veiga, C. (2009). Hierarchical and centralized architectures for distributed production planning, scheduling and control activities. *Service Operations, Logistics and Informatics*,, 341-346.
- Falkenberg, E. (1998). A framework of information system concepts. *The FRISCO Report. Task Group FRISCO*.
- Fowler, J. W., & Rose, O. (2004). Grand Challenges in Modeling and Simulation of Complex Manufacturing Systems. *Simulation*, 469-476.
- Gaither, N., & Frazier, G. (2003). *Administración de producción y operaciones*. México: Thomson editores.
- Gigerenzer, G., & Gaissmaier, W. (2011). Heuristic Decision Making. *Review of Psychology*, 451-482.
- Glover, F. E. (2003). *Handbook of metaheuristics*. Kluwer Academic Publishers.
- González, E. (16 de Enero de 2013). Selección de criterios del algoritmo genético. (D. Páez , & A. Vargas, Entrevistadores)
- Gunasekaran, A., & Ngai, E. W. (2012). The future of operations management: An Outlook and analysis. *International Journal of Production Economics*, 687-701.
- H. Khorshidian a, N. J. (2011). A genetic algorithm for JIT single machine scheduling with preemption. *Expert Systems with Applications*, 7911–7918.

- Hartmut, S., & Christoph, K. (2008). Production Planning and Scheduling. En H. Stadtler, *Supply Chain Management and Advanced Planning* (págs. 199-216). Berlin: Springer-Verlag Berlin Heidelberg.
- Hishamuddina, H., Sarkera, R., & Essama, D. (2012). A disruption recovery model for a single stage production-inventory system. *European Journal of Operational Research*, 464–473.
- Houa, D.-l., & Li, T.-k. (2012). Analysis of Random Disturbances on Shop Floor in Modern Steel Production Dynamic Environment. *Procedia Engineering*, 663–667.
- Hu, X., Gurnani, H., & Wang, L. (2013). Managing Risk of Supply Disruptions: Incentives for Capacity Restoration. *Production and Operations Management*, 137–150.
- Huang, G., Shi, J., & Mak, K. (2000). Failure Mode and Effect Analysis (FMEA) Over the WWW. *The International Journal of Advanced Manufacturing Technology*, 603-608.
- Hurtado, F. A. (2005). *Gestión y auditoría de la calidad para organizaciones públicas*. Medellín: Universidad de Antioquia.
- Inza, A. U. (2006). *Manual Básico de Logística integral*. Madrid: Ediciones Díaz de Santos.
- Islam, A., & Tedford, D. (2012). Risk determinants of small and medium-sized manufacturing enterprises (SMEs) - an exploratory study in New Zealand. *Journal of Industrial Engineering International*, 1-13.
- Islam, A., Bagum, N., & Abul, C. (2012). Operational Disturbances and Their Impact on the Manufacturing Business- An Empirical Study in the RMG Sector of Bangladesh. *International Journal of Research in Management & Technology*, 184-191.
- Kanawaty. (1996). *Introducción al estudio del trabajo*. Ginebra, Suiza: Organización internacional del trabajo.
- Larrosa, J., & Meseguer, P. (2003). Algoritmos para satisfacción de restricciones. *Revista Iberoamericana de Inteligencia Artificial*.
- Lee, J., & Kimb, C. (2008). Multi-agent systems applications in manufacturing systems and supply chain management: a review paper. *International Journal of Production Research*, 233-265.
- Leitão, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Application of Artificial Intelligence*, 979-991.
- Lima, M., Zhang, Z., & Goh, W. (2009). An iterative agent bidding mechanism for responsive manufacturing. *Engineering Applications of Artificial Intelligence*, 1068–1079.

- Maříka, V., & Lařanskýa, J. (2007). Industrial applications of agent technologies. *Control Engineering Practice*, 1364–1380.
- Miramontes, O. (1999). Los sistemas complejos como instrumentos de conocimiento y transformación del mundo. En S. Ramirez, *Perspectivas sobre la teoría de sistemas*. Centro de Investigaciones Interdisciplinarias en Ciencias y Humanidades. México D.F.: Siglo XXI y el CICH-UNAM.
- Montgomery, D. (2004). *Diseño y análisis de experimentos*. México: Limusa S.A.
- Navadon Sortrakul, C. R. (2007). Genetic algorithms for total weighted expected tardiness. *Journal of Quality in Maintenance Engineering*, 49-61.
- Nwana, H. S., & Ndumu, D. T. (1997). An introduction to agent technology. En *Software agents and soft computing towards enhancing machine intelligence. Lecture notes in computer science* (págs. 1-26). Springer.
- Padhia, S., Wagnera, S., Niranjana, T., & Aggarwalc, V. (2012). A simulation-based methodology to analyse production line disruptions. *International Journal of Production Research*, 1885-1897.
- Páez, D. (2011). *Solución del problema de knapsack con algoritmos genéticos*. Bogotá.
- Park, H.-S., & Tran, N.-H. (2012). An autonomous manufacturing system based on swarm of cognitive agents. *Journal of Manufacturing Systems*, 337–348.
- Pinedo, M. L. (2008). *Scheduling. Theory, Algorithms and Systems*. New York: Prentice Hall.
- Prasad, R. A. (1998). A conceptual and Operational Definition of Personal innovativeness in the Domain of Information Technology. *Inform: Information System Research*, 204-215.
- Prawda, J. (2004). *Métodos y modelos de investigación de operaciones*. México, D.F.: Limusa.
- Rao, S. S. (2009). *Engineering Optimization: Theory and Practice*. New Jersey: John Wiley & Sons, Inc.
- Rao, S. S. (2009). *Engineering Optimization: Theory and Practice*. New Jersey: John Wiley & Sons, Inc.
- Real Academia de la Lengua. (2011).
- Render, B., & Heizer, J. (2004). *Principios de administración de operaciones*. México: Pearson Educación.

- Rial Boubeta, A. (2008). *Estadística práctica para la investigación en ciencias de la salud*. España: Netbiblo.
- Rodriguez, L., Insfran, E., & Cernuzzi, L. (2011). Requirements Modeling for Multiagent Systems. *Multiagent systems modeling, control, programming, simulations and applications.*, 3-22.
- Sabarc, M., Montreuil, B., & Frayret, J.-M. (2009). A multi-agent-based approach for personnel scheduling in assembly centers. *Engineering Applications of Artificial Intelligence*, 1080–1088.
- Shi, Y., & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. *Evolutionary Programming VII*, 591-600.
- Shoham, Y., & Leyton-Brown, K. (2010). *Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations*.
- Sortrakul, N., & Cassady, C. R. (2007). Genetic algorithms for total weighted expected tardiness integrated preventive maintenance planning and production scheduling for a single machine. *Journal of Quality in Maintenance Engineering*, 49-61.
- Steiner, G. (2004). *Planeación estratégica*. México: CECSA.
- Talbi, E.-G. (2009). *Metaheuristics from design to implementation*. John Wiley & Sons.
- Tang, J. L. (1999). A modified genetic algorithm for single machine scheduling. *Computer & Industrial Engineering*, 43-46.
- Trentesaux, D. (2009). Distributed control of production systems. *Engineering Applications of Artificial Intelligence*, 971–978.
- Valencia, J. R. (2003). *Introducción a la Administración Con Enfoque de Sistemas*. México: Thomson.
- Vega, C. (Septiembre de 2011). Presentación en clase: DEA. Bogotá.
- Vollmann, T. E., Berry, W. L., & Whybark, D. C. (1997). *Manufacturing Planning and Control Systems*. Nueva York: McGraw-Hill.
- Wanga, M., Wang, H., Vogel, D., Kumarc, K., & Chiud, D. K. (2009). Agent-based negotiation and decision making for dynamic supply chain formation. *Engineering Applications of Artificial Intelligence*, 1046–1055.
- Weea, H., & Widyadana, G. (2012). A production model for deteriorating items with stochastic preventive maintenance time and rework process with FIFO rule. *Omega*, 941–954.

- Wei , Q. (2001). Data Envelopment analysis. *Chinese Science Bulletin*, 1321-1332.
- Wiers, V. C. (1997). *Human computer interaction in production scheduling: Analysis and design of decision support systems for production scheduling*. Eindhoven: Tesis Doctoral.
- Will, A. (2010). Algoritmos genéticos y optimización heurística. *Aplicaciones de inteligencia artificial Universidad Nacional de Tucumán*, 1-38.
- Wooldridge, M. (2002). *An Introduction to Multiagent Systems*. England: John Wiley & Sons, Ltd.
- Wooldridge, M. (2002). *An Introduction to Multiagent Systems*. Liverpool, England: John Wiley & Sons Ltd.
- Zambonelli, F., Jennings, N., & Wooldridge, M. (2003). Developing Multiagent Systems: The Gaia Methodology. *ACM Transactions on Software Engineering and Methodology*, 317 – 370.
- Zbib, N., Pach, C., Sallez , Y., & Trentesaux, D. (2012). Heterarchical production control in manufacturing systems using the potential fields concept. *Journal of Intelligent Manufacturing*, 1649-1670.
- Zhonga, R., Daib,, Q., Quc, T., Hud, G., & Huang, G. (2013). RFID-enabled real-time manufacturing execution system for mass-customization production. *Robotics and Computer-Integrated Manufacturing*, 283–292.

ANEXOS

1. **RANQUEO DE ALGORITMOS:** En la tabla se muestra las instancias, las posiciones y los datos de llenado representan los algoritmos. Así por ejemplo para la instancia 1, el mejor algoritmo es el 1, el segundo mejor es el 6, el tercero es el 4 y así sucesivamente.

INSTANCIA	RANKING DE ALGORITMOS							
	1	2	3	4	5	6	7	8
1	1	6	2	4	3	7	8	5
2	2	4	5	3	6	8	7	1
3	1	4	2	3	6	8	7	5
4	3	1	6	4	8	5	2	7
5	2	1	5	4	3	7	6	8
6	3	4	1	7	8	5	2	6
7	8	5	4	1	7	3	6	2
8	1	2	5	8	7	6	4	3
9	4	6	8	3	5	1	7	2
10	2	8	1	4	6	3	7	5
11	6	1	4	2	5	8	3	7
12	1	6	3	2	7	8	4	5
13	1	5	6	3	7	4	8	2
14	1	6	2	8	3	7	5	4
15	5	7	4	1	2	8	3	6
16	5	1	3	2	4	6	7	8
17	1	4	8	3	5	2	6	7
18	1	8	7	5	2	4	3	6
19	2	1	4	5	6	8	7	3
20	1	5	2	6	3	8	7	4
21	1	7	5	8	6	3	4	2
22	1	4	8	2	6	7	3	5
23	7	3	1	2	6	5	4	8
24	1	2	8	6	7	5	4	3
25	1	5	6	3	8	7	4	2
26	1	2	5	6	8	3	4	7
27	7	1	6	4	5	3	2	8
28	1	5	6	2	3	4	7	8
29	1	3	2	5	6	4	8	7

30	1	5	8	6	2	7	3	4
31	5	2	8	1	7	6	4	3
32	6	1	4	7	5	2	8	3
33	1	7	2	6	8	5	3	4
34	1	4	8	6	2	5	7	3
35	1	2	5	3	8	6	4	7
36	4	5	2	1	7	6	8	3
37	1	5	6	3	7	4	2	8
38	2	1	5	6	8	3	4	7
39	4	1	6	5	7	2	8	3
40	1	3	2	5	7	4	8	6
41	2	6	4	8	7	3	1	5
42	1	5	4	2	3	7	8	6
43	1	7	2	4	6	5	3	8
44	4	1	3	7	2	6	5	8
45	3	7	5	4	1	2	8	6
46	1	2	5	6	7	4	3	8
47	1	6	7	8	3	2	5	4
48	1	2	8	6	7	5	3	4
49	7	1	6	3	8	2	4	5
50	5	1	2	8	6	7	3	4
51	6	1	5	7	8	4	3	2
52	5	1	8	4	3	7	2	6
53	2	1	5	4	8	6	7	3
54	2	6	8	1	4	7	5	3
55	2	1	5	6	8	3	7	4
56	1	5	6	2	4	8	7	3
57	3	6	1	2	4	7	8	5
58	1	5	7	6	2	8	3	4
59	8	1	7	6	4	2	5	3
60	8	4	1	2	5	3	7	6
61	1	6	4	8	7	2	5	3
62	1	7	6	5	2	4	8	3
63	1	6	5	2	8	4	3	7
64	1	3	5	7	8	4	6	2
65	5	4	6	3	8	2	1	7
66	1	8	4	7	5	6	2	3
67	5	6	4	2	7	1	8	3
68	5	1	6	2	8	4	7	3
69	7	1	3	2	6	4	5	8
70	1	2	5	6	8	3	4	7

71	5	8	1	4	6	2	3	7
72	4	5	6	2	1	7	8	3
73	4	1	6	3	8	5	2	7
74	1	8	4	7	3	2	5	6
75	5	6	8	2	4	1	3	7
76	1	2	3	4	8	7	5	6
77	8	1	7	2	3	4	5	6
78	4	1	5	3	7	6	2	8
79	1	6	4	8	5	3	2	7
80	2	1	6	7	4	5	3	8
81	1	7	6	2	4	5	8	3
82	3	2	6	4	7	5	8	1
83	1	6	7	3	5	8	4	2
84	5	1	7	4	2	3	8	6
85	1	7	5	3	4	8	2	6
86	1	7	6	2	5	4	3	8
87	1	6	8	2	5	3	7	4
88	8	1	5	4	2	6	3	7
89	1	3	7	4	2	6	5	8
90	1	6	2	3	4	7	8	5
91	1	2	4	8	5	6	3	7
92	3	1	5	2	6	4	8	7
93	1	6	5	8	2	3	4	7
94	1	8	7	6	3	5	4	2
95	1	2	6	8	7	4	5	3
96	1	6	7	2	5	3	4	8
97	2	6	1	8	3	5	7	4
98	6	1	3	7	8	4	2	5
99	1	2	5	6	3	7	8	4
100	7	2	3	6	5	1	8	4
101	1	6	4	2	3	5	7	8
102	2	1	8	3	6	7	4	5
103	3	1	4	8	7	6	2	5
104	1	6	2	5	8	7	3	4
105	1	4	7	8	5	6	2	3
106	5	7	2	1	6	4	3	8
107	5	1	6	2	3	4	7	8
108	1	5	6	2	4	7	8	3
109	1	8	4	6	2	5	7	3
110	2	4	8	5	1	7	6	3
111	1	2	6	5	8	3	7	4

112	1	2	6	8	4	7	3	5
113	6	4	3	2	8	5	1	7
114	2	4	8	1	6	5	7	3
115	1	7	2	6	4	3	8	5
116	6	5	2	8	1	7	4	3
117	2	1	6	5	3	7	4	8
118	4	6	1	5	2	7	3	8
119	7	8	3	2	4	1	5	6
120	1	7	3	4	8	5	6	2
121	1	8	2	5	4	3	7	6
122	7	2	5	6	4	8	1	3
123	1	8	6	5	4	7	3	2
124	8	1	7	3	5	2	6	4
125	1	2	6	8	5	3	4	7

Tabla 15: Ranking de algoritmos. Elaborado por Daniel Paez y Andrea Vargas

2. RESUMEN DIAGRAMAS DE CAJA PROMEDIO TARDANZA TOTAL PONDERADA: En la siguiente tabla se muestra:

- a) El valor de la mejor función objetivo para cada una de las instancias
- b) El algoritmo al que pertenece cada uno de los valores de la función objetivo. Esta columna se crea para observar cual fue el modelo que arrojó el resultado.
- c) Diagramas de caja del promedio de la tardanza total ponderada tanto para el algoritmo genético como para el sistema Multiagente.
 - **Diagrama rojo y azul:** Representa los valores del algoritmo genético para la función objetivo. El rojo representa los valores que están entre el primer cuartil y la mediana y los valores azules son los que están entre la mediana y el tercer cuartil
 - **Diagrama verde y naranja:** Representa los valores del sistema Multiagente para la función objetivo. El verde representa los valores que están entre el primer cuartil y la mediana y los valores naranja son los que están entre la mediana y el tercer cuartil

INS	MEJOR TARDANZA PONDERADA		DIAGRAMA DE CAJA ALGORITMO GENÉTICO Y SISTEMA MULTIAGENTE
	VALOR	MODELO	
1	6974,33	ALGORITMO GENÉTICO	
2	6974,33	ALGORITMO GENÉTICO	
3	7489,41	ALGORITMO GENÉTICO	
4	8071,39	ALGORITMO GENÉTICO	
5	12901,89	ALGORITMO GENÉTICO	

6	26542,94	ALGORITMO GENÉTICO	
7	21329,38	ALGORITMO GENÉTICO	
8	31302,26	ALGORITMO GENÉTICO	
9	30172,82	ALGORITMO GENÉTICO	
10	27772,73	ALGORITMO GENÉTICO	

11	68001,19	ALGORITMO GENÉTICO	
12	56465,54	ALGORITMO GENÉTICO	
13	58197,25	ALGORITMO GENÉTICO	
14	45715,54	ALGORITMO GENÉTICO	
15	69187,01	ALGORITMO GENÉTICO	

16	146039,28	ALGORITMO GENÉTICO	
17	128319,02	ALGORITMO GENÉTICO	
18	144976,04	ALGORITMO GENÉTICO	
19	141563,37	ALGORITMO GENÉTICO	
20	112462,67	ALGORITMO GENÉTICO	

21	138203,62	ALGORITMO GENÉTICO	
22	135731,44	ALGORITMO GENÉTICO	
23	199567,36	ALGORITMO GENÉTICO	
24	191073,73	ALGORITMO GENÉTICO	
25	186948,01	ALGORITMO GENÉTICO	

26	10410,98	ALGORITMO GENÉTICO	
27	5970,62	ALGORITMO GENÉTICO	
28	8252,45	ALGORITMO GENÉTICO	
29	5167,3	ALGORITMO GENÉTICO	
30	4469,99	ALGORITMO GENÉTICO	

31	32998,32	ALGORITMO GENÉTICO	
32	36368,26	ALGORITMO GENÉTICO	
33	20339,74	ALGORITMO GENÉTICO	
34	18735,43	ALGORITMO GENÉTICO	
35	22765,87	ALGORITMO GENÉTICO	

36	62373,27	ALGORITMO GENÉTICO	<p>Box plot for row 36. The x-axis ranges from 0 to 120,000. The first distribution (red and blue) has a median of 59,083.27 and a mean of 63,272.3. The second distribution (green and yellow) has a median of 83,475.5 and a mean of 95,164.0. Whiskers extend from 54,280.0 to 74,584.0 and 72,272.0 to 110,141.0 respectively.</p>
37	50464,01	ALGORITMO GENÉTICO	<p>Box plot for row 37. The x-axis ranges from 0 to 100,000. The first distribution (red and blue) has a median of 46,923.63 and a mean of 52,363.5. The second distribution (green and yellow) has a median of 63,249.5 and a mean of 72,764.3. Whiskers extend from 42,395.0 to 63,011.0 and 53,820.0 to 84,438.0 respectively.</p>
38	61553,17	ALGORITMO GENÉTICO	<p>Box plot for row 38. The x-axis ranges from 0 to 100,000. The first distribution (red and blue) has a median of 58,476.48 and a mean of 64,898.8. The second distribution (green and yellow) has a median of 73,413.08 and a mean of 81,655.3. Whiskers extend from 52,071.0 to 70,927.0 and 66,103.0 to 92,776.0 respectively.</p>
39	64141,9	ALGORITMO GENÉTICO	<p>Box plot for row 39. The x-axis ranges from 0 to 120,000. The first distribution (red and blue) has a median of 59,876.52 and a mean of 67,522.3. The second distribution (green and yellow) has a median of 85,121.09 and a mean of 94,058.0. Whiskers extend from 53,996.0 to 78,663.0 and 75,641.0 to 107,374.0 respectively.</p>
40	56031,47	ALGORITMO GENÉTICO	<p>Box plot for row 40. The x-axis ranges from 0 to 120,000. The first distribution (red and blue) has a median of 49,222.60 and a mean of 5,355.5. The second distribution (green and yellow) has a median of 69,780.0 and a mean of 83,870.3. Whiskers extend from 42,352.0 to 77,994.0 and 52,905.0 to 99,039.0 respectively.</p>

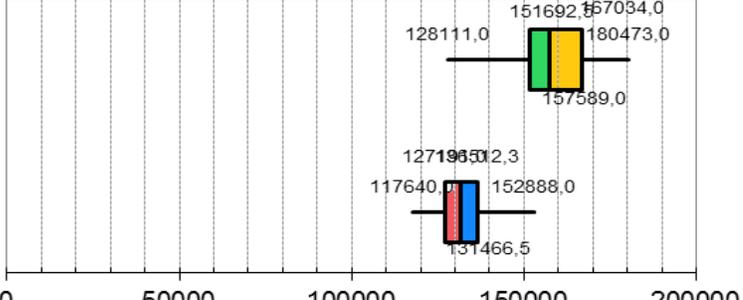
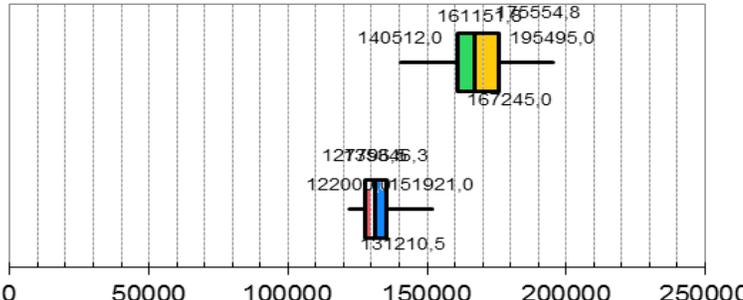
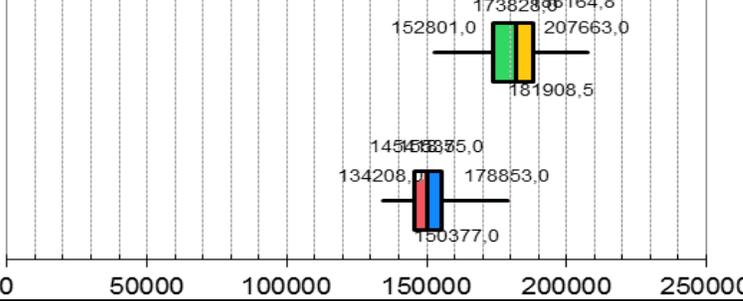
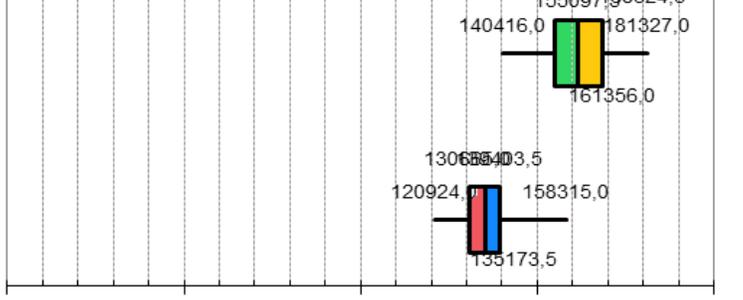
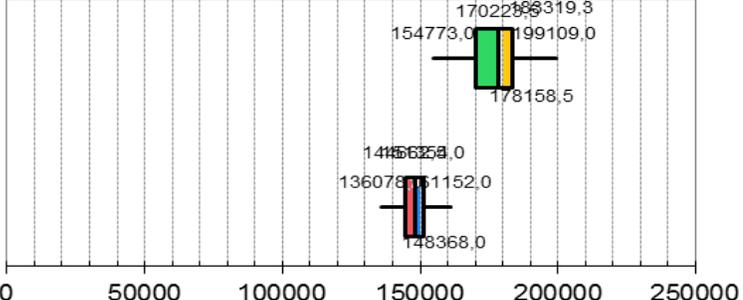
41	108082,54	ALGORITMO GENÉTICO	
42	144799,7	ALGORITMO GENÉTICO	
43	133894,7	ALGORITMO GENÉTICO	
44	131766,69	ALGORITMO GENÉTICO	
45	111893,7	ALGORITMO GENÉTICO	

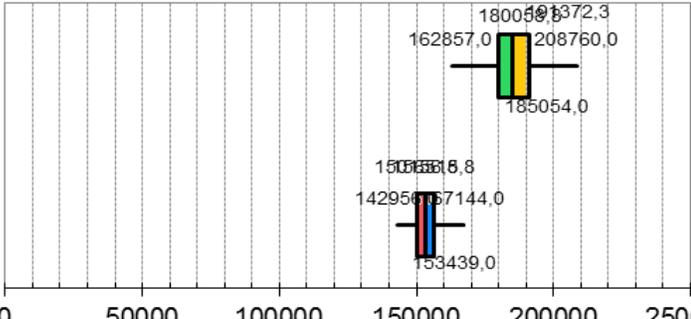
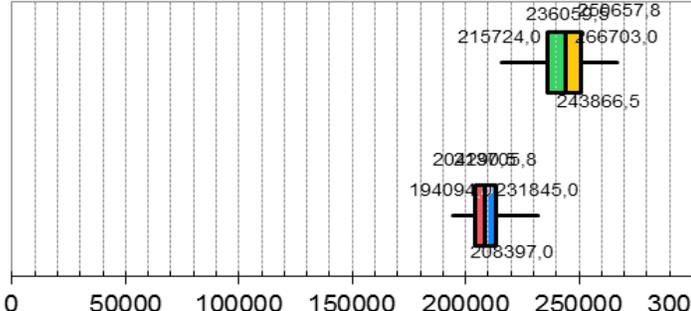
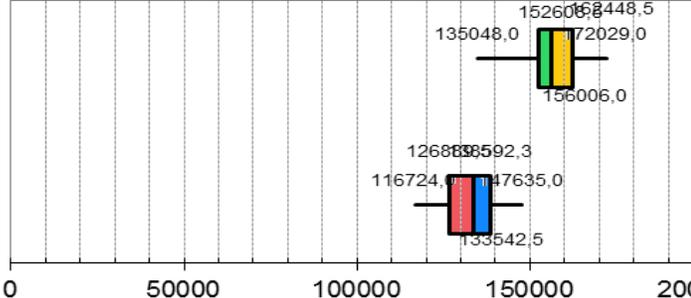
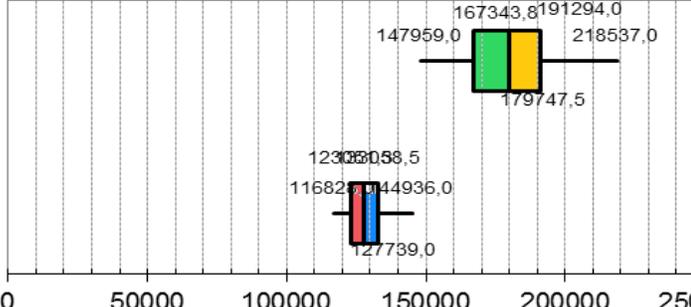
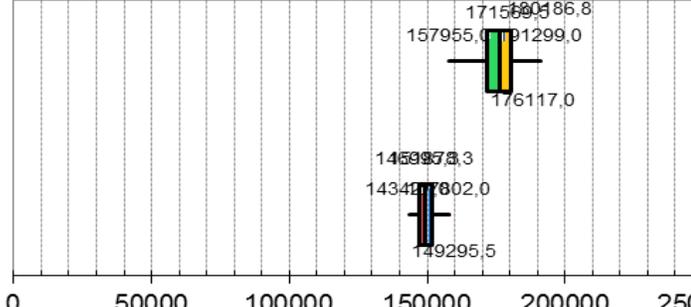
46	104896,69	ALGORITMO GENÉTICO	
47	181523,95	ALGORITMO GENÉTICO	
48	155524,6	ALGORITMO GENÉTICO	
49	150712,1	ALGORITMO GENÉTICO	
50	158633,07	ALGORITMO GENÉTICO	

51	27321,3	ALGORITMO GENÉTICO	
52	12432,06	ALGORITMO GENÉTICO	
53	10184,33	ALGORITMO GENÉTICO	
54	7186,46	ALGORITMO GENÉTICO	
55	8490,56	ALGORITMO GENÉTICO	

56	19000,58	ALGORITMO GENÉTICO	
57	36569,21	ALGORITMO GENÉTICO	
58	37017,34	ALGORITMO GENÉTICO	
59	43138,7	ALGORITMO GENÉTICO	
60	32276,73	ALGORITMO GENÉTICO	

61	57673,57	ALGORITMO GENÉTICO	
62	57795,47	ALGORITMO GENÉTICO	
63	80861,73	ALGORITMO GENÉTICO	
64	66178,34	ALGORITMO GENÉTICO	
65	56077	ALGORITMO GENÉTICO	

66	132006,98	ALGORITMO GENÉTICO	
67	131920,85	ALGORITMO GENÉTICO	
68	150748,13	ALGORITMO GENÉTICO	
69	135720,74	ALGORITMO GENÉTICO	
70	148580,21	ALGORITMO GENÉTICO	

71	153571,77	ALGORITMO GENÉTICO	
72	209688,83	ALGORITMO GENÉTICO	
73	132699,76	ALGORITMO GENÉTICO	
74	128135,91	ALGORITMO GENÉTICO	
75	149713,09	ALGORITMO GENÉTICO	

76	10266,8	ALGORITMO GENÉTICO	
77	11509,98	ALGORITMO GENÉTICO	
78	2912,6	ALGORITMO GENÉTICO	
79	12338,65	ALGORITMO GENÉTICO	
80	17168,83	ALGORITMO GENÉTICO	

81	37378,74	ALGORITMO GENÉTICO	
82	26479,15	ALGORITMO GENÉTICO	
83	32777,81	ALGORITMO GENÉTICO	
84	34658,71	ALGORITMO GENÉTICO	
85	18209,53	ALGORITMO GENÉTICO	

86	87830,09	ALGORITMO GENÉTICO	
87	84439,78	ALGORITMO GENÉTICO	
88	61981,36	ALGORITMO GENÉTICO	
89	93756,37	ALGORITMO GENÉTICO	
90	56037,86	ALGORITMO GENÉTICO	

91	118241,15	ALGORITMO GENÉTICO	
92	105684,78	ALGORITMO GENÉTICO	
93	107188,19	ALGORITMO GENÉTICO	
94	112968,29	ALGORITMO GENÉTICO	
95	119569,15	ALGORITMO GENÉTICO	

96	166956,58	ALGORITMO GENÉTICO	
97	179610,74	ALGORITMO GENÉTICO	
98	191702,15	ALGORITMO GENÉTICO	
99	176392,47	ALGORITMO GENÉTICO	
100	234323,78	ALGORITMO GENÉTICO	

101	5342,64	ALGORITMO GENÉTICO	
102	23547,41	ALGORITMO GENÉTICO	
103	10730,04	ALGORITMO GENÉTICO	
104	19054,35	ALGORITMO GENÉTICO	
105	20014,42	SISTEMA MULTIAGENTE	

106	50940,83	ALGORITMO GENÉTICO	
107	39317,14	ALGORITMO GENÉTICO	
108	53057,77	ALGORITMO GENÉTICO	
109	24124,97	ALGORITMO GENÉTICO	
110	20801,47	ALGORITMO GENÉTICO	

111	108839,4	ALGORITMO GENÉTICO	
112	62949,5	ALGORITMO GENÉTICO	
113	69183,2	ALGORITMO GENÉTICO	
114	78123,35	ALGORITMO GENÉTICO	
115	64415,12	ALGORITMO GENÉTICO	

116	104483,83	ALGORITMO GENÉTICO	
117	124265,9	ALGORITMO GENÉTICO	
118	78522,89	ALGORITMO GENÉTICO	
119	161520,12	ALGORITMO GENÉTICO	
120	129334,25	ALGORITMO GENÉTICO	

121	178924,38	ALGORITMO GENÉTICO	<p>Bar chart for item 121. The x-axis ranges from 0 to 30000. The chart shows two groups of bars. The first group (red and blue) has values: 178273,0, 164257,0, 173506,5, 173506,5. The second group (green and yellow) has values: 212810,0, 203261,0, 219108,3, 244225,0.</p>
122	149652,07	ALGORITMO GENÉTICO	<p>Bar chart for item 122. The x-axis ranges from 0 to 25000. The chart shows two groups of bars. The first group (red and blue) has values: 149289,5, 140960,0, 145378,8, 145378,8. The second group (green and yellow) has values: 180846,5, 157240,0, 175048,0, 202449,0.</p>
123	147567,98	ALGORITMO GENÉTICO	<p>Bar chart for item 123. The x-axis ranges from 0 to 25000. The chart shows two groups of bars. The first group (red and blue) has values: 147176,0, 126318,0, 140044,0, 14737,3. The second group (green and yellow) has values: 176244,5, 144855,0, 171039,0, 193077,0.</p>
124	139077,54	ALGORITMO GENÉTICO	<p>Bar chart for item 124. The x-axis ranges from 0 to 20000. The chart shows two groups of bars. The first group (red and blue) has values: 137656,0, 124940,0, 133150,0, 14896,3. The second group (green and yellow) has values: 166147,5, 143403,0, 159769,0, 174631,3.</p>
125	178356,91	ALGORITMO GENÉTICO	<p>Bar chart for item 125. The x-axis ranges from 0 to 25000. The chart shows two groups of bars. The first group (red and blue) has values: 179231,0, 160213,0, 172534,8, 172534,8. The second group (green and yellow) has values: 199657,0, 182324,0, 195920,0, 221069,0.</p>

3. RESUMEN DIAGRAMAS DE CAJA PROMEDIO TIEMPO TOTAL DE EJECUCIÓN: En la siguiente tabla se muestra:

- a) El valor del mejor tiempo promedio de ejecución para cada una de las instancias.
- b) El modelo al que pertenece cada uno de los valores del mejor tiempo promedio de ejecución. Esta columna se crea para observar cual fue el algoritmo que arrojó el resultado.
- c) Diagramas de caja del promedio del tiempo total de ejecución tanto para el algoritmo genético como para el sistema Multiagente.
 - **Diagrama rojo y azul:** Representa los valores del algoritmo genético para el tiempo promedio de ejecución. El rojo representa los valores que están entre el primer cuartil y la mediana y los valores azules son los que están entre la mediana y el tercer cuartil
 - **Diagrama verde y naranja:** Representa los valores del sistema Multiagente para el tiempo promedio de ejecución. El verde representa los valores que están entre el primer cuartil y la mediana y los valores naranja son los que están entre la mediana y el tercer cuartil.

INS	TIEMPO EJECUCIÓN		DIAGRAMA DE CAJA ALGORITMOS GENÉTICO Y SISTEMA MULTIAGENTE
	VALOR	MODELO	
1	0,10571289 1	SISTEMA MULTIAGENTE	<p>Box plot for INS 1 comparing Genetic Algorithms (green) and Multi-Agent System (red). Genetic Algorithms has a median of 107.4, while the Multi-Agent System has a median of 207.0. Whiskers extend from 75.2 to 127.9 for Genetic Algorithms and 183.6 to 246.1 for the Multi-Agent System.</p>
2	0,08689453 1	SISTEMA MULTIAGENTE	<p>Box plot for INS 2 comparing Genetic Algorithms (green) and Multi-Agent System (red). Genetic Algorithms has a median of 87.9, while the Multi-Agent System has a median of 150.4. Whiskers extend from 72.3 to 104.5 for Genetic Algorithms and 128.9 to 171.9 for the Multi-Agent System.</p>
3	0,08344726 6	SISTEMA MULTIAGENTE	<p>Box plot for INS 3 comparing Genetic Algorithms (green) and Multi-Agent System (red). Genetic Algorithms has a median of 83.0, while the Multi-Agent System has a median of 175.8. Whiskers extend from 64.5 to 101.6 for Genetic Algorithms and 156.3 to 195.3 for the Multi-Agent System.</p>
4	0,08535156 3	SISTEMA MULTIAGENTE	<p>Box plot for INS 4 comparing Genetic Algorithms (green) and Multi-Agent System (red). Genetic Algorithms has a median of 85.0, while the Multi-Agent System has a median of 160.2. Whiskers extend from 69.3 to 102.5 for Genetic Algorithms and 132.8 to 179.7 for the Multi-Agent System.</p>
5	0,08310546 9	SISTEMA MULTIAGENTE	<p>Box plot for INS 5 comparing Genetic Algorithms (green) and Multi-Agent System (red). Genetic Algorithms has a median of 83.0, while the Multi-Agent System has a median of 187.5. Whiskers extend from 60.5 to 100.6 for Genetic Algorithms and 152.3 to 207.0 for the Multi-Agent System.</p>

6	0,08995117 2	SISTEMA MULTIAGENT E	
7	0,09232421 9	SISTEMA MULTIAGENT E	
8	0,08615234 4	SISTEMA MULTIAGENT E	
9	0,08476562 5	SISTEMA MULTIAGENT E	
10	0,08455078 1	SISTEMA MULTIAGENT E	

11	0,08715820 3	SISTEMA MULTIAGENT E	
12	0,08867187 5	SISTEMA MULTIAGENT E	
13	0,08349609 4	SISTEMA MULTIAGENT E	
14	0,08672851 6	SISTEMA MULTIAGENT E	
15	0,08292968 8	SISTEMA MULTIAGENT E	

16	0,08935546 9	SISTEMA MULTIAGENT E	
17	0,08615234 4	SISTEMA MULTIAGENT E	
18	0,08438476 6	SISTEMA MULTIAGENT E	
19	0,08124023 4	SISTEMA MULTIAGENT E	
20	0,08129882 8	SISTEMA MULTIAGENT E	

21	0,08067382 8	SISTEMA MULTIAGENT E	
22	0,08637695 3	SISTEMA MULTIAGENT E	
23	0,07872070 3	SISTEMA MULTIAGENT E	
24	0,079375	SISTEMA MULTIAGENT E	
25	0,08282226 6	SISTEMA MULTIAGENT E	

26	0,08111328 1	SISTEMA MULTIAGENT E	
27	0,08077148 4	SISTEMA MULTIAGENT E	
28	0,08543945 3	SISTEMA MULTIAGENT E	
29	0,08124023 4	SISTEMA MULTIAGENT E	

30	0,08128906 3	SISTEMA MULTIAGENT E	
31	0,07890625	SISTEMA MULTIAGENT E	
32	0,07976562 5	SISTEMA MULTIAGENT E	
33	0,08131835 9	SISTEMA MULTIAGENT E	

34	0,08192382 8	SISTEMA MULTIAGENT E	<p>Box plot for row 34. The x-axis ranges from 0 to 250. The first distribution (green) has a minimum of 66.4, Q1 of 78.1, Median of 83.0, Q3 of 85.0, and Maximum of 99.6. The second distribution (blue) has a minimum of 148.4, Q1 of 191.9, Median of 193.4, Q3 of 199.2, and Maximum of 214.8.</p>
35	0,08723632 8	SISTEMA MULTIAGENT E	<p>Box plot for row 35. The x-axis ranges from 0 to 200. The first distribution (green) has a minimum of 70.3, Q1 of 82.8, Median of 86.9, Q3 of 92.8, and Maximum of 108.4. The second distribution (blue) has a minimum of 144.5, Q1 of 171.1, Median of 175.8, Q3 of 179.7, and Maximum of 187.5.</p>
36	0,08054687 5	SISTEMA MULTIAGENT E	<p>Box plot for row 36. The x-axis ranges from 0 to 200. The first distribution (green) has a minimum of 66.4, Q1 of 76.2, Median of 79.6, Q3 of 85.0, and Maximum of 116.2. The second distribution (blue) has a minimum of 121.1, Q1 of 150.6, Median of 150.2, Q3 of 164.1, and Maximum of 171.9.</p>
37	0,08027343 8	SISTEMA MULTIAGENT E	<p>Box plot for row 37. The x-axis ranges from 0 to 200. The first distribution (green) has a minimum of 62.5, Q1 of 76.2, Median of 80.1, Q3 of 84.2, and Maximum of 93.8. The second distribution (blue) has a minimum of 117.2, Q1 of 152.1, Median of 156.3, Q3 of 160.2, and Maximum of 168.0.</p>

38	0,08104492 2	SISTEMA MULTIAGENT E	
39	0,07990234 4	SISTEMA MULTIAGENT E	
40	0,07957031 3	SISTEMA MULTIAGENT E	
41	0,08133789 1	SISTEMA MULTIAGENT E	

42	0,08613281 3	SISTEMA MULTIAGENT E	
43	0,08180664 1	SISTEMA MULTIAGENT E	
44	0,08163085 9	SISTEMA MULTIAGENT E	
45	0,08022460 9	SISTEMA MULTIAGENT E	

46	0,08905273 4	SISTEMA MULTIAGENT E	
47	0,07939453 1	SISTEMA MULTIAGENT E	
48	0,08795898 4	SISTEMA MULTIAGENT E	
49	0,08097656 3	SISTEMA MULTIAGENT E	
50	0,08883789 1	SISTEMA MULTIAGENT E	

51	0,08232421 9	SISTEMA MULTIAGENT E	
52	0,08478515 6	SISTEMA MULTIAGENT E	
53	0,08069335 9	SISTEMA MULTIAGENT E	
54	0,08452148 4	SISTEMA MULTIAGENT E	
55	0,07686523 4	SISTEMA MULTIAGENT E	

56	0,08110351 6	SISTEMA MULTIAGENT E	
57	0,08321289 1	SISTEMA MULTIAGENT E	
58	0,08158203 1	SISTEMA MULTIAGENT E	
59	0,07670898 4	SISTEMA MULTIAGENT E	
60	0,08504882 8	SISTEMA MULTIAGENT E	

61	0,08915039 1	SISTEMA MULTIAGENT E	
62	0,08009765 6	SISTEMA MULTIAGENT E	
63	0,08739257 8	SISTEMA MULTIAGENT E	
64	0,08693359 4	SISTEMA MULTIAGENT E	
65	0,07954101 6	SISTEMA MULTIAGENT E	

66	0,07859375	SISTEMA MULTIAGENT E	
67	0,083330078	SISTEMA MULTIAGENT E	
68	0,084433594	SISTEMA MULTIAGENT E	
69	0,0865625	SISTEMA MULTIAGENT E	
70	0,084707031	SISTEMA MULTIAGENT E	

71	0,08850585 9	SISTEMA MULTIAGENT E	
72	0,08081054 7	SISTEMA MULTIAGENT E	
73	0,08692382 8	SISTEMA MULTIAGENT E	
74	0,08260742 2	SISTEMA MULTIAGENT E	
75	0,08020507 8	SISTEMA MULTIAGENT E	

76	0,08871093 8	SISTEMA MULTIAGENT E	
77	0,08297851 6	SISTEMA MULTIAGENT E	
78	0,07625	SISTEMA MULTIAGENT E	
79	0,08110351 6	SISTEMA MULTIAGENT E	
80	0,08053710 9	SISTEMA MULTIAGENT E	

81	0,08136718 8	SISTEMA MULTIAGENT E	<p>Box plot for row 81. The x-axis ranges from 0 to 200. The first distribution (green/yellow) has a minimum of 46.9, Q1 of 76.2, Median of 81.1, Q3 of 86.9, and Maximum of 95.7. The second distribution (red/blue) has a minimum of 113.3, Q1 of 132.8, Median of 140.6, Q3 of 148.4, and Maximum of 168.0.</p>
82	0,08403320 3	SISTEMA MULTIAGENT E	<p>Box plot for row 82. The x-axis ranges from 0 to 200. The first distribution (green/yellow) has a minimum of 41.0, Q1 of 80.1, Median of 84.0, Q3 of 87.9, and Maximum of 105.5. The second distribution (red/blue) has a minimum of 125.8, Q1 of 128.8, Median of 132.8, Q3 of 136.7, and Maximum of 144.5.</p>
83	0,08222656 3	SISTEMA MULTIAGENT E	<p>Box plot for row 83. The x-axis ranges from 0 to 200. The first distribution (green/yellow) has a minimum of 47.9, Q1 of 78.1, Median of 82.5, Q3 of 86.9, and Maximum of 96.7. The second distribution (red/blue) has a minimum of 132.8, Q1 of 147.1, Median of 152.3, Q3 of 156.3, and Maximum of 164.1.</p>
84	0,07909179 7	SISTEMA MULTIAGENT E	<p>Box plot for row 84. The x-axis ranges from 0 to 200. The first distribution (green/yellow) has a minimum of 38.1, Q1 of 74.2, Median of 79.1, Q3 of 83.3, and Maximum of 96.7. The second distribution (red/blue) has a minimum of 144.5, Q1 of 150.1, Median of 160.2, Q3 of 164.1, and Maximum of 168.0.</p>
85	0,07803710 9	SISTEMA MULTIAGENT E	<p>Box plot for row 85. The x-axis ranges from 0 to 200. The first distribution (green/yellow) has a minimum of 37.1, Q1 of 73.2, Median of 78.1, Q3 of 84.0, and Maximum of 95.7. The second distribution (red/blue) has a minimum of 128.9, Q1 of 139.1, Median of 144.5, Q3 of 148.4, and Maximum of 160.2.</p>

86	0,08819335 9	SISTEMA MULTIAGENT E	
87	0,08208984 4	SISTEMA MULTIAGENT E	
88	0,08216796 9	SISTEMA MULTIAGENT E	
89	0,08300781 3	SISTEMA MULTIAGENT E	
90	0,08822265 6	SISTEMA MULTIAGENT E	

91	0,08646484 4	SISTEMA MULTIAGENT E	
92	0,08870117 2	SISTEMA MULTIAGENT E	
93	0,08549804 7	SISTEMA MULTIAGENT E	
94	0,08815429 7	SISTEMA MULTIAGENT E	
95	0,08326171 9	SISTEMA MULTIAGENT E	

96	0,08295898 4	SISTEMA MULTIAGENT E	
97	0,08313476 6	SISTEMA MULTIAGENT E	
98	0,08634765 6	SISTEMA MULTIAGENT E	
99	0,07867187 5	SISTEMA MULTIAGENT E	
100	0,085625	SISTEMA MULTIAGENT E	

101	0,08456054 7	SISTEMA MULTIAGENT E	<p>Box plot showing two distributions. The first distribution (green/yellow) has a median of 84.0, with values ranging from 46.9 to 127.9. The second distribution (red) has a median of 152.3, with values ranging from 140.6 to 164.1.</p>
102	0,08306640 6	SISTEMA MULTIAGENT E	<p>Box plot showing two distributions. The first distribution (green/yellow) has a median of 83.0, with values ranging from 47.9 to 122.1. The second distribution (red) has a median of 136.7, with values ranging from 109.4 to 168.0.</p>
103	0,08223632 8	SISTEMA MULTIAGENT E	<p>Box plot showing two distributions. The first distribution (green/yellow) has a median of 81.1, with values ranging from 40.0 to 216.8. The second distribution (red) has a median of 109.4, with values ranging from 93.8 to 128.9.</p>
104	0,0840625	SISTEMA MULTIAGENT E	<p>Box plot showing two distributions. The first distribution (green/yellow) has a median of 84.0, with values ranging from 43.0 to 140.6. The second distribution (red) has a median of 156.3, with values ranging from 132.8 to 191.4.</p>
105	0,08488281 3	ALGORITMO GENÉTICO	<p>Box plot showing two distributions. The first distribution (green/yellow) has a median of 85.4, with values ranging from 50.8 to 136.7. The second distribution (red) has a median of 85.9, with values ranging from 74.2 to 93.8.</p>

106	0,08291992 2	SISTEMA MULTIAGENT E	
107	0,08397460 9	SISTEMA MULTIAGENT E	
108	0,08142578 1	SISTEMA MULTIAGENT E	
109	0,08192382 8	SISTEMA MULTIAGENT E	
110	0,08829101 6	SISTEMA MULTIAGENT E	

111	0,08331054 7	SISTEMA MULTIAGENT E	
112	0,08433593 8	SISTEMA MULTIAGENT E	
113	0,08121093 8	SISTEMA MULTIAGENT E	
114	0,08535156 3	SISTEMA MULTIAGENT E	
115	0,07911132 8	SISTEMA MULTIAGENT E	

116	0,08046875	SISTEMA MULTIAGENT E	
117	0,088974609	SISTEMA MULTIAGENT E	
118	0,082392578	SISTEMA MULTIAGENT E	
119	0,082167969	SISTEMA MULTIAGENT E	
120	0,082089844	SISTEMA MULTIAGENT E	

121	0,08572265 6	SISTEMA MULTIAGENT E	
122	0,08285156 3	SISTEMA MULTIAGENT E	
123	0,08527343 8	SISTEMA MULTIAGENT E	
124	0,08603515 6	SISTEMA MULTIAGENT E	
125	0,08473632 8	SISTEMA MULTIAGENT E	

