

SECUENCIACIÓN DE OPERACIONES DE MANUFACTURA EN LA LÍNEA DE
PRODUCCIÓN AUTOMATIZADA MEDIANTE LA APLICACIÓN DEL ALGORITMO DE
COLONIA DE HORMIGAS

MARÍA ELVIRA MARTÍNEZ VÁSQUEZ

176342

UNIVERSIDAD PONTIFICIA BOLIVARIANA

ESCUELA DE INGENIERÍA

INGENIERÍA INDUSTRIAL

BUCARAMANGA

2018

SECUENCIACIÓN DE OPERACIONES DE MANUFACTURA EN LA LÍNEA DE
PRODUCCIÓN AUTOMATIZADA MEDIANTE LA APLICACIÓN DEL
ALGORITMO DE COLONIA DE HORMIGAS

MARÍA ELVIRA MARTÍNEZ VÁSQUEZ

176342

DIRECTOR: M.Sc. ORLANDO FEDERICO GONZÁLEZ CASALLAS

UNIVERSIDAD PONTIFICIA BOLIVARIANA

ESCUELA DE INGENIERÍA

INGENIERÍA INDUSTRIAL

BUCARAMANGA

2018

Tabla de contenido

Introducción	10
1. Definición del problema	12
1.1. Procesos de la línea de producción del laboratorio.....	15
1.1.2 Proceso de selección de piezas.	15
1.1.2 Proceso de clasificación de piezas por altura.	15
1.1.3 Proceso de asignación de color al producto.	15
1.1.4 Proceso de transporte y empaado del producto.	16
2. Justificación	18
3. Objetivos	20
3.1. Objetivo general	20
3.2. Objetivos específicos	20
4. Antecedentes	21
5. Marco teórico	22
5.1 Flow Shop	22
5.2 Algoritmo de colonia de hormigas.....	23
5.3 Diseño de experimento.....	28
5.3.1 Experimento.	28
5.3.2 Diseño de experimento factorial.	29
6. Diseño metodológico	34

	4
7. Solución del problema.	36
7.1. Aplicación de la regla de Johnson en la solución inicial de secuenciación para la línea de producción automatizada	36
7.1.1 Calculo del tiempo mínimo de procesamiento de todos los trabajos.....	42
8. Análisis de resultados	54
8.1 Resultados del problema de cuatro máquinas y cuatro trabajos.	54
9. Evaluación de la calidad de la solución del algoritmo desarrollado en el programa Microsoft Visual Basic	57
9.1 Análisis de varianza para la instancia 10_10_1	60
9.2 Análisis de varianza para la instancia 10_10_2	61
9.3 Análisis de varianza para la instancia 10_15_1	63
9.4 Análisis de varianza para la instancia 10_15_2	64
9.5 Análisis de varianza para la instancia 60_20_1	66
10. Conclusiones.	69
Trabajos citados	72
11. Anexos	74

Lista de tablas

Tabla 1. Procesos de la línea de producción del laboratorio de procesos industriales	14
Tabla 2. Diseño factorial para tres factores	30
Tabla 3. Análisis de varianza del modelo de tres factores con efectos fijos.....	31
Tabla 4. Pasos de la metodología.....	35
Tabla 5. Análisis de varianza para los datos de Makespan para el problema de cuatro máquinas y cinco trabajos.	55
Tabla 6. Análisis de varianza para los datos del tiempo computacional para el problema de cuatro máquinas y cinco lotes.	56
Tabla 7. límites superior e inferior de las instancias y resultados promedio hallados con el programa propuesto.....	58
Tabla 8. Niveles de los factores.	59
Tabla 9. ANOVA para los resultados del makespan de la instancia 10_10_1	60
Tabla 10. ANOVA para el tiempo computacional de la instancia 10_10_1.....	61
Tabla 11. ANOVA para la instancia 10_10_2.....	62
Tabla 12. ANOVA para el tiempo computacional de la instancia 10_10_2.....	62
Tabla 13. ANOVA para el makespan de la instancia 10_15_1.	63
Tabla 14. ANOVA para el tiempo computacional de la instancia 10_15_1.....	64
Tabla 15. ANOVA para el análisis de varianza del makespan de la instancia 10_15_2	65
Tabla 16. ANOVA para el tiempo computacional de la instancia 10_15_2.....	66
Tabla 17. ANOVA para el makespan de la instancia 60_20_1.	67
Tabla 18. ANOVA para el tiempo computacional de la instancia 60_20_1.....	68

Lista de Figuras

Figura 1. Visualización de la planta industrial del laboratorio de control de procesos.	12
Figura 2: Procesos de la línea de producción del laboratorio de procesos industriales.	13
Figura 3. Proceso de Control de piezas.	15
Figura 4. Proceso de clasificación de piezas.	16
Figura 5. Colores asignados según tamaño y material.	16
Figura 6. Proceso de transporte y empaçado.	17
Figura 7. Proceso de empaçado.	17
Figura 8. Producción Flow shop.	22
Figura 9. Búsqueda del camino más corto realizada por las hormigas.	26
Figura 10. Modelo general de un sistema.	29
Figura 11. Pseudocódigo de la secuenciación de máquinas.	40
Figura 12. Programación de Operaciones de n-lotes en el sistema de producción automatizado.	42
Figura 13. Secuenciación cuando no hay ningún trabajo en las máquinas.	43
Figura 14. Asignación de un trabajo cuando existen máquinas previamente asignadas.	44
Figura 15. Asignación de un trabajo a una máquina que se encuentra ocupada.	45
Figura 16. Diagrama de flujo para el diagrama de Gantt.	47
Figura 17. Algoritmo de colonia de hormigas.	49
Figura 18. Inicio de la página web de las instancias.	57
Figura 19. Gráfica de los límites superior e inferior y los resultados hallados por el programa propuesto.	59

Lista de Anexos

Anexo A. Diagrama Gantt del problema trabajado (4 x 4 en el laboratorio de automatización de la UPB seccional Bucaramanga).....	74
Anexo B. Tiempos de procesamiento y diagrama de Gantt de la solución de la instancia 10_10_1.....	74
Anexo C. Tiempos de procesamiento y diagrama Gantt de la solución de las instancias 10_10_1 , 10_10_2 , 10_15_1 , 10_15_2 y 60_20_1. (Excel).....	83
Anexo D. Programa propuesto en Visual Basic. (Excel)	83

RESUMEN GENERAL DE TRABAJO DE GRADO

TITULO: Secuenciación de operaciones de manufactura en la línea de producción automatizada mediante la aplicación del algoritmo de colonia de hormigas

AUTOR(ES): Maria Elvira Martinez Vasquez

PROGRAMA: Facultad de Ingeniería Industrial

DIRECTOR(A): Orlando Federico Gonzalez Casallas

RESUMEN

En este proyecto se desarrolla un programa de secuenciación de operaciones de manufactura tipo Flow shop en la línea de producción automatizada del laboratorio de automatización de procesos en la Universidad Pontificia Bolivariana, para generar una propuesta de utilización de la planta, mediante la aplicación del algoritmo de colonia de hormigas. El desarrollo de este proyecto de investigación, considera tres momentos base; la primera fase de tipo cuasi-experimental debido a que se observa el funcionamiento estándar de los procesos que integran la línea de producción; para el caso del análisis de tiempo de procesamiento, y consideración del tipo de productos o lotes que se desean producir, se considera que el diseño experimental transeccional, debido a que se escoge de manera aleatoria el tipo de lote que se desea producir. Finalmente, el proceso de validación del algoritmo, se desarrolla mediante la aplicación de instancias de la literatura, de tal forma que se pueda inferir sobre la calidad de solución y tiempo computacional que emplea el algoritmo para encontrar la solución al problema de secuencia de operaciones bajo un sistema de línea de ensamble. Para analizar las fuentes de variación del problema (cantidad de hormigas, cantidad de iteraciones y disipación) se realizan 128 pruebas repartidas en grupos de 8 réplicas de un diseño experimental de tres factores, finalmente se obtiene un tiempo computacional de 0.22 segundos y un makespan de 41 minutos promedio. Después de realizar la evaluación del programa propuesto con respecto a 5 instancia de la literatura, e implementar un diseño de experimento factorial con su respectivo análisis de varianza, se concluye que la calidad del algoritmo propuesto es buena ya que los resultados al ejecutar el programa con los datos de las instancias, se encuentran dentro del rango aprobado por la literatura.

PALABRAS CLAVE:

Colonia de hormigas, algoritmo, linea de ensamble

V° B° DIRECTOR DE TRABAJO DE GRADO

GENERAL SUMMARY OF WORK OF GRADE

TITLE: Manufacturing shedulling in a flowshop through the ant colony algorithm application

AUTHOR(S): Maria Elvira Martinez Vasquez

FACULTY: Facultad de Ingeniería Industrial

DIRECTOR: Orlando Federico Gonzalez Casallas

ABSTRACT

In this Project, a shedulling program in Flow Shop on Universidad Pontificia Bolivariana's automation laboratory, to generate a proposal for using the plant, applying the ant colony algorithm. The development of this research project, considers three moments; the first phase is quasi-experimental type because because it is observing the standard processes operation that integrate the flow shop; in the case of analysis of processing time, and consideration of the type of products or lots that are to be produced, the transectional experimental design is considered, because the type of batch that one wishes to produce is chosen randomly. The validation process of the algorithm, is developed by the application of instances of the literature, in such a way that it can be inferred about the quality of solution and computational time that the algorithm uses to find the solution to the problem of operations shedulling. To analyze the sources of the problem variation (number of ants, number of iterations and dissipation), 128 tests are made distributed in groups of 8 réplicas of an experimental design of three factors, finally a computational time of 0.22 seconds is obtained and makespan of 41 minutes on average. After carrying out the evaluation of the proposed program with respect to 5 literature instances, and implementing a factorial experiment design with the variance analysis, it is concluded that the quality of the proposed algorithm is good since the results when executing the program with data of the instances are within the range approved by the literature.

KEYWORDS:

Ant colony algorithm, flowshop, makespan

V° B° DIRECTOR OF GRADUATE WORK

Introducción

El desarrollo de procesos de producción en sistemas empresariales actuales, requiere de la aplicación de técnicas cuantitativas para la planificación, control y supervisión de manufactura. Para este fin, se considera el estudio de los tiempos, capacidad de producción, variación de la demanda de productos y la eficiencia de los procesos que constituyen un sistema de producción.

Adicionalmente, se debe considerar que la evolución tecnológica en las plantas de producción, ha permitido que se implementen dispositivos que permite al gerente de operaciones de producción, realizar mediciones y control de variables de proceso; por tanto, se debe considerar el efecto de la automatización de procesos de producción como parte de los procesos de planificación y control de la producción en las organizaciones.

Una de las técnicas que se aplica en el proceso de control y supervisión de la producción, se basa en el control de los tiempos y asignación de operaciones de transformación en las diferentes estaciones de trabajo que constituyen al sistema de producción. Lo anterior es conocido como la teoría de la secuenciación de operaciones de manufactura.

La teoría de la secuenciación o Scheduling, permite al gerente de operaciones de manufactura hacer mediciones, control y seguimiento de la ejecución de producción, considerando diferentes criterios de eficiencia de operación como reducción de tiempos de producción, reducción de los niveles de tardanza, o la maximización de la eficiencia de utilización de los recursos de manufactura. Según la complejidad del sistema de manufactura se puede llegar a modelar el proceso mediante técnicas de optimización, simulación o técnicas heurísticas.

Para este trabajo de investigación, se abordará la utilización de técnicas heurísticas para solucionar problemas de secuenciación con ambientes de manufactura tipo *Flow shop*; donde se

estudiará e implementará un algoritmo basado en la heurísticas de Colonia de Hormigas, para evaluar la eficiencia computacional de solucionar el problema de secuenciación en una planta piloto de automatización, que se encuentra en las instalaciones del laboratorio de control de procesos industriales de la Universidad Pontificia Bolivariana, Seccional Bucaramanga.

Inicialmente, se explicará el funcionamiento de la planta de automatización que se encuentra en las instalaciones del laboratorio de control de procesos de la Universidad Pontificia Bolivariana, Seccional Bucaramanga; de tal forma que se pueda identificar los puntos clave de registro de información de los tiempos de procesamiento. Seguidamente, se realizarán muestras estadísticas de los tiempos, para establecer los valores promedio de procesamiento en cada estación de trabajo.

Una vez se obtenga la información de los tiempos de procesamiento de cada estación de trabajo, se procede con la programación del algoritmo de colonia de hormigas para el caso de un problema de secuenciación tipo *Flow shop*, a través del lenguaje de programación de Visual Basic para Microsoft Excel, se estudiará la calidad del algoritmo propuesto mediante la valoración de diferentes instancias de la literatura; y posteriormente, se procederá con la solución para el caso de estudio que se presenta en el laboratorio de control de procesos.

1. Definición del problema

El problema de secuenciación de operaciones es un problema relacionado con la programación de producción, en el cual la secuencia de trabajos es la misma para todas las máquinas. Dicho problema consiste en secuenciar un conjunto de n trabajos llevados a cabo unidireccionalmente en un sistema compuesto por m máquinas dispuestas en serie.

La programación de la producción tiene como propósito optimizar la utilización de los recursos de forma que se cumpla con los objetivos de producción; debido al aumento de la productividad y la competitividad en los últimos años ha ganado atención por parte de académicos y empresas. La planta del laboratorio de automatización de ingeniería industrial está diseñada para la fabricación de productos, los cuales pasan por procesos secuenciales; es decir, está diseñada para que todas las referencias pasen por el mismo flujo de transformación. (Ver Figura 1).



Figura 1. Visualización de la planta industrial del laboratorio de control de procesos. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014). Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.

Por otra parte, los criterios de funcionamiento son predeterminados e independientes con respecto a los demás procesos. Adicionalmente, se debe considerar que existe un proceso manual referente a la asignación de color al producto, y después se pasa a un proceso integrado de

transporte y empackado. (Ver Figura 2). A continuación, se presenta las características de funcionamiento de cada proceso que constituye a la planta industrial.

El sistema de producción cuenta con 68 piezas para desarrollar el proceso de selección, clasificación y empackado. Debido a que se desea estudiar la condición de los tiempos de procesamiento, y la forma como se deben asignar los trabajos a la planta industrial automatizada, se considera el uso de 28 piezas.

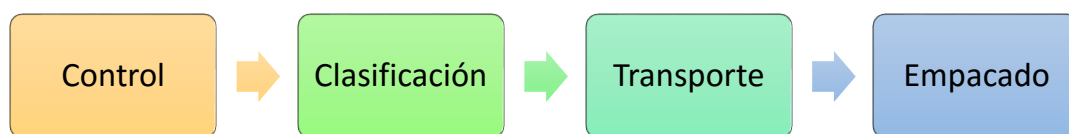


Figura 2: Procesos de la línea de producción del laboratorio de procesos industriales. *Fuente: Autor.*

Las 28 piezas fueron muestreadas para conocer los tiempos de ejecución de cada proceso, de acuerdo a la especificación de parámetros de funcionamiento del sistema. La banda transportadora del sistema trabaja con una velocidad de 0-60 RPM, el robot cartesiano del motor eléctrico que se encarga de los movimientos para el proceso de empaque del producto maneja una frecuencia de corriente de 30 Hz y 30 Hz; Los resultados de los tiempos por máquina y por trabajo se presentan en la Tabla 1.

Tabla 1:

Tiempos de procesamiento observados

	Lote 1	Lote 2	Lote 3	Lote 4
Lotes	5 amarillas	2 amarillas 3 verdes	5 verdes	3 verdes 2 amarillas
Máquinas	Tiempo en minutos			
Control	3	3	4	3
Clasificación	8	8	9	7
Transporte	5	6	7	7
Empacado	1	1	2	1

Fuente: Autor.

La línea de producción del laboratorio cuenta con cuatro máquinas en serie, para este caso, no se consideran restricciones en el sistema, se conocen los tiempos de procesamiento de cada lote en cada máquina y se desea conocer el tiempo de terminación de procesar todos los lotes; a continuación, se muestran de manera organizada los tiempos de procesamiento para el caso de los cuatro lotes en una matriz.

	L₁	L₂	L₃	L₄
M₁	3	3	4	3
M₂	8	8	9	7
M₃	5	6	7	7
M₄	1	1	2	1

1.1. Procesos de la línea de producción del laboratorio

1.1.2 Proceso de selección de piezas.

Inicialmente, en la fase de selección, se realiza el control de calidad de las piezas; el operario hace una prueba de selección según los requerimientos de calidad, también, revisa si hay imperfecciones en el producto y se continúa ejecutando el registro de cada clasificación mediante un sistema de código de barras.



Figura 3. Proceso de Control de piezas. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014). Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.

1.1.2 Proceso de clasificación de piezas por altura.

Un conjunto de sensores incorporado mide la altura y el material de las piezas para que estas sean clasificadas entre 14mm, 16mm de altura y metal, aluminio o plástico.

1.1.3 Proceso de asignación de color al producto.

En este proceso las piezas son etiquetadas y ubicadas en los depósitos de almacenamiento por un operario y transportadas por un robot cartesiano hacia la estación de empaclado.

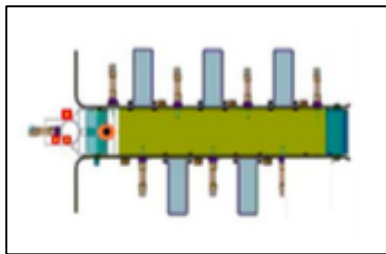


Figura 4. Proceso de clasificación de piezas. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014).

Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.



Figura 5. Colores asignados según tamaño y material. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014). Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.

1.1.4 Proceso de transporte y empaqueo del producto.

En el proceso de transporte y empaqueo del producto un operario realiza la orden de producción mediante la interfaz de usuario HMI (Human machine interface), el empaqueo se efectúa haciendo uso de un sistema de sellado y corte.



Figura 6. Proceso de transporte y empackado. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014).

Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.



Figura 7. Proceso de empackado. Fuente: Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014). Línea de producción manual de usuario. Laboratorio de automatización. Versión 1.0. Bucaramanga, Colombia.

2. Justificación

Este proyecto permitirá en primera instancia un estudio y caracterización del sistema de producción automatizado que se encuentra en el laboratorio de automatización de procesos en la Universidad Pontificia Bolivariana, Seccional Bucaramanga. Dentro del estudio y caracterización del sistema de producción, se aplicará una de las técnicas de metaheurística con el propósito de generar alternativas de solución adecuadas para el funcionamiento del sistema.

La línea de producción de la Universidad cuenta con cuatro estaciones de trabajo, control, clasificación, transporte y empaçado de piezas cilíndricas, al no haberse realizado estudios de secuenciación, es necesario conocer las condiciones tácticas y operacionales de la planta.

Adicionalmente, permitirá a las personas involucradas en el proyecto relacionarse con temas que se trabajan en el campo de la Ingeniería Industrial y desarrollar competencias en el campo del diseño de propuestas algorítmicas y de programación. El impacto al minimizar el *makespan* es lograr hacer los procesos de la planta más eficientes. Lo anterior, con el propósito de beneficiar a distintos agentes de la comunidad universitaria en general tales como grupos de investigación, docentes y estudiantes que desarrollan prácticas de laboratorio con la planta industrial automatizada.

Este proyecto de investigación desarrollará la fase preliminar de estudio del sistema de producción automatizado que se encuentra en el laboratorio de procesos de la Universidad Pontificia Bolivariana; se considera la aplicación de conceptos de secuenciación de operaciones de manufactura en un ambiente tipo línea de ensamble (*Flow shop*) sin restricciones, para minimizar el tiempo de terminación de todos los trabajos asignados a la línea.

Se desarrollará un programa base en Microsoft Visual Basic, de tal forma que se pueda obtener la propuesta de diseño de secuenciación de la línea, así como la validación del algoritmo mediante la comparación de instancias de la literatura en cuanto al ambiente de producción tipo *flow shop*; de tal forma que se dé respuesta a la pregunta de investigación: ¿Cómo se puede planificar la secuenciación de operaciones de manufactura en una línea de producción automatizada con ambiente de trabajo tipo *Flow shop*?

3. Objetivos

3.1. Objetivo general

Desarrollar un programa de secuenciación de operaciones de manufactura tipo *Flow shop* en la línea de producción automatizada del laboratorio de automatización de procesos en la Universidad Pontificia Bolivariana, para la propuesta de utilización de la planta, mediante la aplicación del algoritmo de colonia de hormigas.

3.2. Objetivos específicos

Caracterizar los procesos industriales que componen la línea de producción automatizada del laboratorio de procesos para la obtención de los tiempos de procesamiento en cada estación de trabajo.

Implementar el algoritmo de colonia de hormigas, en el programa Microsoft Visual Basic, de acuerdo a los criterios de la secuenciación de operaciones en ambientes de manufactura tipo *flow shop* sin restricciones.

Desarrollar un estudio de las instancias de la literatura del problema *flow shop*, de tal forma que se evalúe la calidad de solución del algoritmo desarrollado en el programa Microsoft Visual Basic.

Proponer la secuenciación de operaciones de la línea de producción automatizada, mediante la aplicación de algoritmo de colonias de hormigas, desarrollado en el programa Microsoft Visual Basic.

4. Antecedentes

En la literatura existen trabajos relacionados con el problema de secuenciación de operaciones referentes al ambiente de manufactura tipo *flow shop*. Algunos autores han aplicado el algoritmo de colonia de hormigas para solucionar el problema. Krishnaraj (2012), propone minimizar la varianza del tiempo de terminación del trabajo en ambiente de producción tipo *flow shop* mediante la aplicación del algoritmo de colonia de hormigas modificado, obteniendo mejores resultados con respecto a otras técnicas heurísticas.

Por otro lado, Gajpal (2005) aplica el método de colonia de hormigas para abordar el problema de secuenciación de operaciones con ambiente de manufactura tipo *flow shop*, considerando la restricción de tiempos de alistamiento de máquinas.

Además Hornig (2016) realizó el estudio de un problema de *flow shop* flexible con tiempos de alistamiento dependientes de la secuencia y minimización de la tardanza total. Se empleó un algoritmo de optimización de colonia de hormigas híbrido con una búsqueda en vecindad de intercambio de pares y se evaluó varias instancias del problema. Los resultados fueron comparados con otros métodos de solución, presentando mejores resultados con el algoritmo propuesto. Finalmente, se concluye que el algoritmo propuesto no entrega siempre los mejores resultados comparado con el algoritmo mejorado desarrollado en trabajos anteriores.

Así mismo Hornig (2011) plantea la secuenciación de trabajos en ambiente de manufactura tipo taller de asignación aleatoria con tiempos de alistamiento que dependen de la secuencia, de tal forma que se minimice la terminación de todos los trabajos. Para ello propuso un algoritmo de optimización mediante colonia de hormigas, adaptando la estructura del algoritmo al problema del agente viajero con distancias asimétricas, considerando la evaluación del algoritmo con instancias propuestas de la literatura.

5. Marco teórico

5.1 Flow Shop

Un sistema productivo tipo *Flow shop* según Pinedo (1995) es un tipo de trabajo en el cual el problema es secuenciar una cantidad de trabajos que siguen una misma ruta en una serie de máquinas ordenadas linealmente. En cuanto al almacenamiento entre máquinas, éste suele ser casi ilimitado cuando los productos en procesamiento tienen un tamaño pequeño, en consecuencia, cambiar la serie de los trabajos que se presentan en una línea de espera con dos máquinas puede generar una disminución en el tiempo de terminación.

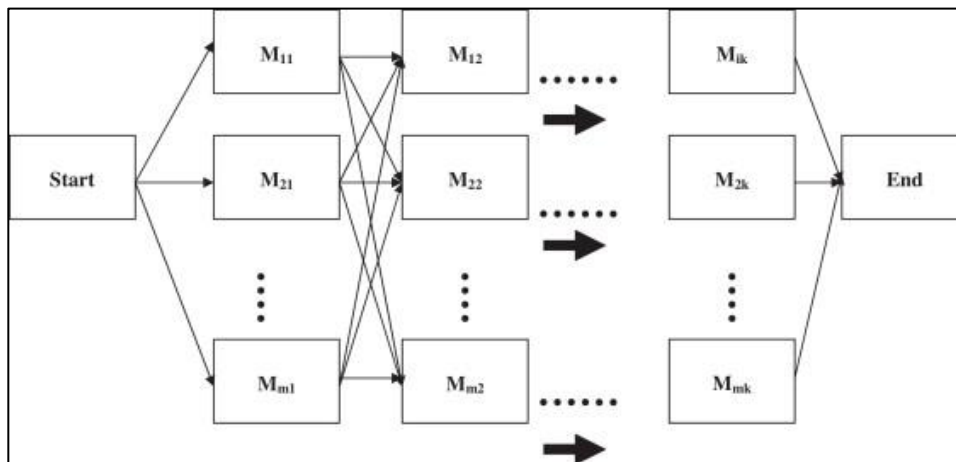


Figura 8. Producción Flow shop. Fuente: Marichelvam et ál. (June de 2014). Improved cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan. (Elsevier, Ed.).

Cuando la producción se realiza en grandes volúmenes y poca variedad, un sistema *Flow Shop* es apropiado. Generalmente este sistema de producción se usa cuando las condiciones son intermedias y la producción de taller de trabajo se aplica para configuraciones de alto y bajo volumen. Por otro parte, suele ser útil usar el taller de trabajo para la producción de componentes individuales y para el ensamble de componentes (Hartl & Preusser, 2009).

5.2 Algoritmo de colonia de hormigas

El algoritmo de colonia de hormigas nace del estudio de algoritmos computacionales orientados a la dinámica o comportamiento social; según Dorigo y Gambardella (1996) se considera que una entidad conocida como hormiga, tiene el propósito de buscar una solución al problema de alimento, conocido como criterio de función objetivo; para esto, debe considerar la búsqueda de fuentes de alimento y transmitir la información a otras entidades a través del mecanismo de la feromona. Los caminos de feromona suelen memorizar las características de las buenas soluciones, cambian de manera dinámica durante la búsqueda para de esa manera reflejar el conocimiento adquirido (Dréo, 2006).

La estructura algorítmica de Colonia de Hormigas, fue propuesta como una herramienta de solución a problemas de optimización combinatoria con un grado mayor de complejidad (Robles, 2010). Lo anterior permite estructurar problemas complejos en procedimientos computacionales que finalmente ayudan a obtener una solución factible en una cantidad razonable de tiempo computacional. La secuenciación de hormigas se realiza al inicio. Este tipo de búsqueda local al no basarse en ordenes, no genera una solución óptima.

En el sistema de colonia de hormigas, cada hormiga genera una solución al problema, dicho sistema cuenta con una serie de agentes computacionales que transitan por las soluciones parciales de un problema, el desplazamiento es realizado con base en los parámetros: rastro de feromona (τ) y el atractivo (η). Además, la evaluación y modificación del valor del rastro de feromona es realizada al completar la solución o durante la etapa de construcción (Mario A. Muñoz, 2007).

Cabe señalar que la feromona se actualiza con cada iteración, esto se hace con la solución

generada anteriormente y se aplica en dos fases; la fase de evaporación es cuando la feromona decrece de manera automática, la fase de refuerzo es cuando la feromona se actualiza según la solución generada. Por otra parte, cabe resaltar que la función de la evaporación presente en la actualización es evitar una convergencia prematura y favorecer la diversificación.

Esta búsqueda, basada en el uso de la feromona, permite al algoritmo establecer condiciones de diversificación o intensificación de la solución; permitiendo la flexibilidad de explorar la región de posibles soluciones a un problema tratado. Lo anterior es concordante con Alonso et al (2004) considerando que se han desarrollado varios estudios que permiten identificar diferentes alternativas de comportamiento, entre las entidades para la exploración apropiada de la región de factibilidad; esto es utilizado como estrategia para definir una solución según la calidad de tiempo computacional empleado en las instancias del problema. Por tanto, Dréo (2006) considera que el proceso de intensificación del algoritmo se traduce en la mayor probabilidad de que exista mayor concentración de feromonas en una ruta específica; mientras que el proceso de diversificación establece un nivel moderado de feromonas, lo que hace que las hormigas busquen nuevos caminos con probabilidad de alta concentración de feromonas.

Los insectos sociales como las hormigas se diferencian de los integrantes de la colonia de los intrusos reconociendo la feromona depositada por cada insecto, este químico varía dependiendo de la colonia; esto les permite reconocer a los demás miembros del nido (Nehring et al. 2016). Al ser una sustancia olfativa y volátil, la feromona puede ser reconocida por la colonia, dicha sustancia va formando un camino para guiar a las demás hormigas hasta el punto deseado (Dréo, 2006).

Lo dicho anteriormente supone que, cuando un individuo no coincide con las señales de la colonia propia, éste ataca al individuo encontrado, dichas señales son copiadas por medio de

síntesis de las sustancias receptivas. Al buscar alimento, estos animales exploran el área alrededor del nido de manera aleatoria, al encontrar el lugar dónde está situada la comida evalúan la cantidad y calidad y llevan una porción a su nido (Mathur et al, 2000).

Los algoritmos de colonias de hormigas corresponden a los métodos heurísticos, estos son algoritmos usados para obtener soluciones a problemas complejos mediante aproximaciones (Algarín, 2010). Estos algoritmos, se basan en metáforas naturales, buscando así simular la forma en que las hormigas construyen la ruta más corta desde el nido hasta el lugar dónde se encuentran los alimentos y viceversa. En cuanto a la cantidad de feromona depositada por una hormiga, esta depende de la porción de comida que se encuentra en el sitio destino y la longitud del camino, puesto que los caminos cortos son más atractivos al requerir un menor tiempo de viaje, dicho de otra manera, la cantidad de feromona varía dependiendo de la ración de comida y la magnitud del trayecto a seguir por la hormiga (Dréo, 2006). En consecuencia, algunos caminos contienen mayor cantidad de feromona, y por ende estos comienzan a ser transitados por más hormigas, por lo tanto, la probabilidad del rastro de ser seguido por otras hormigas se incrementa con la deposición de más feromona (Mathur et al, 2000).

Los caminos de la feromona describen la búsqueda de la solución por el sistema en cada iteración y los agentes modifican la forma en que el problema será representado y percibido por los otros agentes. Esta información es compartida por las hormigas mediante modificaciones del medio ambiente, en forma de una comunicación indirecta: señal de comunicación. La información se almacena así durante cierto tiempo en el sistema, lo que llevó a ciertos autores a considerar este proceso como una forma de memoria adaptativa, donde la dinámica de almacenamiento y división de información será crucial para el sistema (Dréo, 2006).

En la figura 9 se visualiza un ejemplo de la búsqueda que hacen las hormigas, los lugares donde más se ven líneas son en los que mayor cantidad de feromona han depositado previamente las hormigas antecesoras.

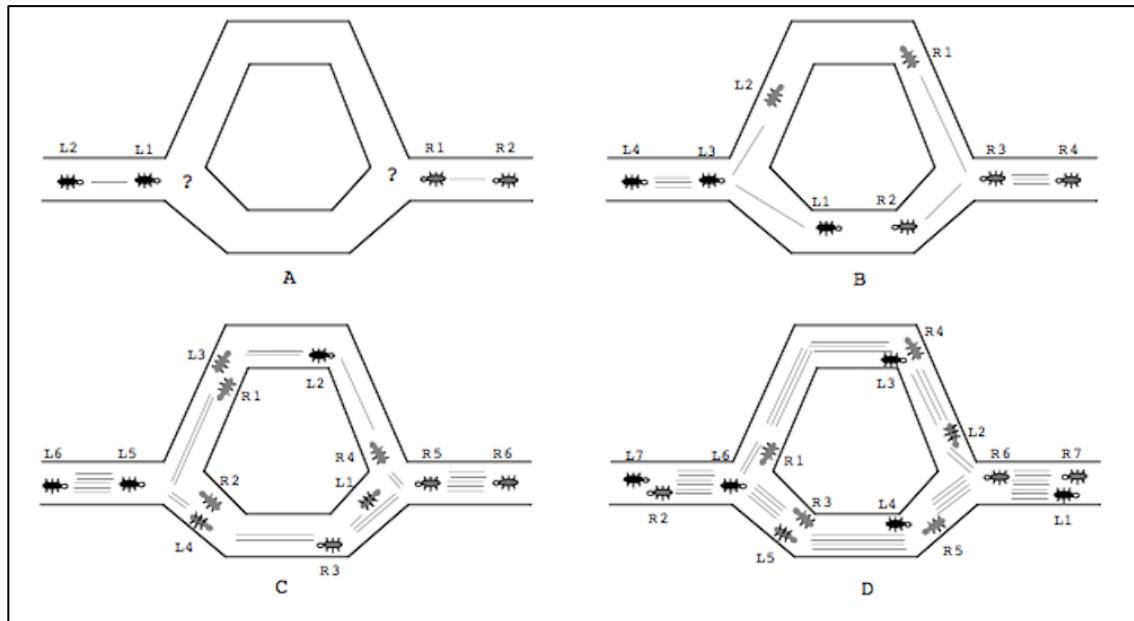


Figura 9. Búsqueda del camino más corto realizada por las hormigas. Fuente: Dorigo, M. G. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem, Bruselas, Université Libre de Bruxelles. 1(1), 1 - 26.

La concepción anterior, según Dréo (2006), está inspirada en la imitación del comportamiento de la evaporación del nivel de feromona de las hormigas en varias rutas o caminos y propone una solución basada en la experiencia colectiva, al evaporarse, la cantidad feromona decrece. Por otro lado, señala que este tipo de metaheurística suele ser más eficaz cuando se mezcla con algoritmos de búsqueda local; lo anterior, mejora sustancialmente las soluciones debido a que se puede hacer un proceso de diversificación para actualizar el nivel de feromonas y la definición de nuevos caminos o rutas de solución.

Los métodos metaheurística para ser más específicos, son algoritmos usados para obtener soluciones a problemas complejos mediante aproximaciones (Algarín, 2010). En esta técnica una

serie de hormigas artificiales generadas computacionalmente seleccionan un conjunto de soluciones al problema; las hormigas usan información heurística (Información específica del problema que da indicios para construir la solución) e información suministrada por las hormigas precursoras (feromona) para llevar a cabo este proceso sobre la calidad de las soluciones construidas (Ortega & Bedia).

Con respecto a la implementación del algoritmo de colonia de hormigas, básicamente se debe inicializar la intensidad de la feromona y los parámetros, luego se construye una solución y se mejora haciendo uso de la búsqueda local; se actualiza la intensidad de la feromona, por último, se encuentra una solución. Teniendo una ciudad aleatoria i , se escoge la ciudad siguiente j con la probabilidad (P_{ij})

$$P_{ij} = \frac{\varnothing_{ij}(k).n_{ij}^{\beta}}{\sum_{l \in j} \varnothing_{il}(k).n_{il}^{\beta}} \quad (1)$$

Dónde $\varnothing_{ij}(k)$ es la intensidad de la feromona y n_{ij} es el atractivo de la feromona siendo esta última el valor inverso del tiempo de procesamiento en el total de las máquinas de cada trabajo, es decir, $n_{ij} = 1/\text{tiempo}$.

Por otro lado, la actualización de la feromona realizada al final de cada iteración está dada por la siguiente expresión:

$$\varnothing_{ij}(k + 1) = (1 - \rho) * \varnothing_{ij}(k) \quad (2)$$

Dónde ρ es la evaporación, es decir, el decrecimiento de la feromona, y $\varnothing_{ij}(k)$ es la intensidad.

Por último, la inclusión de feromona está dada por la expresión

$$\emptyset_{ij}(k + 1) = (1 - \rho) * \emptyset_{ij}(k) + \rho * \Delta\emptyset_{ij}(k) \quad (3)$$

$$\text{Dónde } \Delta\emptyset_{ij}(k) = \frac{1}{c_{max}} \quad (4)$$

5.3 Diseño de experimento

Los diseños de experimentos son modelos estadísticos cuyo principal objetivo es analizar si unos determinados factores influyen en una variable de interés y si existe influencia de algún factor, por otro lado, se desea cuantificar la influencia que ejercen dichos factores en los efectos de los tratamientos, es decir, los resultados del experimento.

5.3.1 Experimento.

Un experimento según Montgomery (2004) es un conjunto de pruebas en las que se hacen cambios en las variables de entrada del sistema para identificar los cambios en las variables de respuesta, es decir, los efectos de los tratamientos. Los experimentos son usados para evaluar el desempeño del sistema, el principal objetivo de esto es determinar la influencia que tienen los factores sobre los efectos de los tratamientos en las salidas del sistema. En la Figura 10 se muestra el modelo general de un sistema.

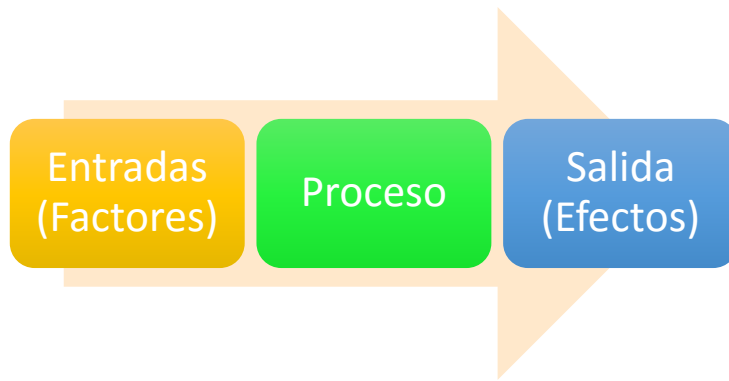


Figura 10. Modelo general de un sistema. Fuente: Autor.

Las siete fases para diseñar experimentos según Montgomery (2004) son:

1. Identificación y evaluación del problema,
2. elección de los factores, niveles y rangos,
3. selección de la variable de respuesta,
4. elección del diseño experimental,
5. realización del experimento,
6. análisis estadístico de los datos,
7. conclusiones.

5.3.2 Diseño de experimento factorial.

Los diseños de experimento factoriales suelen ser más eficientes en los casos en los cuales se desea analizar los efectos de dos o más factores, según Montgomery (2004) en un diseño factorial se evalúan todas las combinaciones posibles de los niveles de los factores. Por otro lado, afirma que, al existir el error experimental, es conveniente llevar a cabo varias observaciones, es decir, dos o más réplicas del experimento para cada tratamiento y finalmente, aplicar promedio al

conjunto de datos, obteniendo así un único valor por tratamiento para posteriormente realizar el análisis de varianza con dichos datos. En la Tabla 2 se muestra el aspecto del diseño factorial para un experimento de 3 factores, cada uno con 4 niveles.

Tabla 2

Diseño factorial para tres factores

		Factor A															
Niveles		1				2				3				4			
		Factor B				Factor B				Factor B				Factor B			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Factor C		1															
		2															
		3															
		4															

Fuente: Autor.

El modelo estadístico para este diseño está dado por

$$y_{ijkl} = \mu + \tau_i + \beta_j + \gamma_k + (\tau\beta)_{ij} + (\tau\gamma)_{ik} + (\beta\gamma)_{jk} + (\tau\beta\gamma)_{ijk} + \varepsilon_{ijkl}; \quad (5)$$

$$\left\{ \begin{array}{l} i = 1, 2, \dots, a \\ j = 1, 2, \dots, b \\ k = 1, 2, \dots, c \\ l = 1, 2, \dots, n \end{array} \right.$$

5.3.2.1 Análisis de varianza.

Luego de la identificación del problema, elección de los factores, niveles y rangos, selección de las variables de respuesta y la elección del diseño experimental, se da paso a la realización del experimento para después aplicar el análisis de varianza (ANOVA). En la Tabla 3 se muestra el ANOVA para el diseño factorial general.

Tabla 3

Análisis de varianza del modelo de tres factores con efectos fijos

Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	Cuadrado medio esperado	F_0
A	SS_A	$a - 1$	MS_A	$\sigma^2 + \frac{bcn \sum \tau_i^2}{a-1}$	$F_0 = \frac{MS_A}{MS_E}$
B	SS_B	$b - 1$	MS_B	$\sigma^2 + \frac{acn \sum \beta_j^2}{b-1}$	$F_0 = \frac{MS_B}{MS_E}$
C	SS_C	$c - 1$	MS_C	$\sigma^2 + \frac{abn \sum \gamma_k^2}{c-1}$	$F_0 = \frac{MS_C}{MS_E}$
AB	SS_{AB}	$(a-1)(b-1)$	MS_{AB}	$\sigma^2 + \frac{cn \sum \sum (\tau\beta)_{jk}^2}{(a-1)(b-1)}$	$F_0 = \frac{MS_{AB}}{MS_E}$
AC	SS_{AC}	$(a-1)(c-1)$	MS_{AC}	$\sigma^2 + \frac{bn \sum \sum (\tau\gamma)_{jk}^2}{(a-1)(c-1)}$	$F_0 = \frac{MS_{AC}}{MS_E}$
BC	SS_{BC}	$(b-1)(c-1)$	MS_{BC}	$\sigma^2 + \frac{an \sum \sum (\beta\gamma)_{jk}^2}{(b-1)(c-1)}$	$F_0 = \frac{MS_{BC}}{MS_E}$
ABC	SS_{ABC}	$(a-1)(b-1)(c-1)$	MS_{ABC}	$\sigma^2 + \frac{n \sum \sum \sum (\tau\beta\gamma)_{jkl}^2}{(a-1)(b-1)(c-1)}$	$F_0 = \frac{MS_{ABC}}{MS_E}$
Error	SS_E	$abc(n-1)$	MS_E	σ^2	
Total	SS_T	$abcn - 1$			

Fuente: Montgomery, D. C. (2004). *Diseño y análisis de experimentos* (Segunda ed.). México, D.F, México: Limusa Wiley.

La suma de cuadrados totales está dada por

$$SS_T = \sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^c \sum_{l=1}^n y_{ijkl}^2 - \frac{y_{\dots}^2}{abcn} \quad (6)$$

por otra parte, la suma de cuadrado de los efectos principales se construye a partir de los factores A, B y C así:

$$SS_A = \frac{1}{bcn} \sum_{i=1}^a y_{i\dots}^2 - \frac{y_{\dots}^2}{abcn} \quad (7)$$

$$SS_B = \frac{1}{acn} \sum_{j=1}^b y_{j\dots}^2 - \frac{y_{\dots}^2}{abcn} \quad (8)$$

$$SS_C = \frac{1}{abn} \sum_{k=1}^c y_{\dots k}^2 - \frac{y_{\dots}^2}{abcn} \quad (9)$$

para calcular la suma de cuadrados de las interacciones de dos factores, se necesitan calcular los totales de las celdas A X B, A X C, B X C. Y posteriormente aplicar las siguientes formulas

$$\begin{aligned} SS_{AB} &= \frac{1}{cn} \sum_{i=1}^a \sum_{j=1}^b y_{ij\dots}^2 - \frac{y_{\dots}^2}{abcn} - SS_A - SS_B \\ &= SS_{subtotales(AB)} - SS_A - SS_B \end{aligned} \quad (10)$$

$$\begin{aligned} SS_{AC} &= \frac{1}{bn} \sum_{i=1}^a \sum_{k=1}^c y_{i\dots k}^2 - \frac{y_{\dots}^2}{abcn} - SS_A - SS_C \\ &= SS_{subtotales(AC)} - SS_A - SS_C \end{aligned} \quad (11)$$

$$\begin{aligned} SS_{BC} &= \frac{1}{an} \sum_{j=1}^b \sum_{k=1}^c y_{j\dots k}^2 - \frac{y_{\dots}^2}{abcn} - SS_B - SS_C \\ &= SS_{subtotales(BC)} - SS_B - SS_C \end{aligned} \quad (12)$$

la suma de cuadrados de la interacción de los tres factores se calcula así

$$SS_{ABC} = \sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^c y_{ijk}^2 - \frac{y_{\dots}^2}{abcn} - SS_A - SS_B - SS_C - SS_{AB} - SS_{AC} - SS_{BC} \quad (13)$$

finalmente, la suma de cuadrados del error puede encontrarse de la siguiente manera

$$SS_E = SS_T - SS_{subtotales(ABC)} \quad (14)$$

6. Diseño metodológico

El desarrollo de este proyecto de investigación, considera tres momentos base. La primera fase de investigación es de tipo cuasi-experimental debido a que se observa el funcionamiento estándar de los procesos que integran la línea de producción; por tanto, no se realizará ajustes de velocidad de bandas transportadoras o cambios en los controladores industriales que hacen que el sistema integral funcione.

Para el caso del análisis de tiempo de procesamiento, y consideración del tipo de productos o lotes que se desean producir, se considera que la diseño experimental transeccional, debido a que se escogerá de manera aleatoria el tipo de lote que se desea producir, y se tomarán muestras de los tiempos de procesamiento para obtener un tiempo promedio de producción en cada proceso del sistema, durante el tiempo de disponibilidad para trabajar en el laboratorio de automatización; lo anterior permitirá que se construya una instancia del problema de secuenciación en la línea de producción automatizada, de tal forma que se proceda con la aplicación del algoritmo de Colonia de Hormigas.

Finalmente, el proceso de validación del algoritmo, se desarrollará mediante la aplicación de instancias de la literatura, de tal forma que se pueda inferir sobre la calidad de solución y tiempo computacional que emplea el algoritmo para encontrar la solución al problema de secuencia de operaciones bajo un sistema de línea de ensamble *-flow shop-*.

Debido a que es un sistema componente de automatización, se establece un registro de cinco ciclos de registro de tiempos de procesamiento; lo anterior se debe a que los datos no presentan demasiada incertidumbre de medición, ya que la sincronización de los dispositivos con controladores industriales permite un mayor grado de precisión en los tiempos de ejecución de

manufactura. La tabla 4, presenta los pasos lógicos de la metodología que se plantea en este proyecto de grado.

Tabla 4:

Pasos de la metodología

Metodología cuasi-experimental		Metodología experimental	
Caracterización del sistema de producción	Registros de tiempo de cada proceso del sistema	Desarrollo del algoritmo de Colonia de hormigas	Validación del algoritmo
Determinar el tipo de proceso de fabricación de la planta del laboratorio de automatización de la Universidad Pontificia Bolivariana seccional Bucaramanga	Se toman como referencia cinco ciclos de tiempo de procesamiento para cada proceso	Implementación del algoritmo de colonia de hormigas.	Se estudia la calidad del algoritmo propuesto mediante la valoración de diferentes instancias de la literatura.

Fuente: Autor.

7. Solución del problema.

Una vez considerada la formulación del problema de secuencia de operaciones de manufactura para la línea de producción automatizada, se procede con la implementación de la estructura algorítmica en el lenguaje computacional Visual Basic para Microsoft Excel. Posteriormente se continuará con la declaración de las variables del modelo, considerando el uso de variables escalares o arreglos para el manejo de información del algoritmo.

En cuanto son identificadas las variables clave para el desarrollo de la solución inicial del problema *Flow shop*, se continúa con la declaración del procedimiento general que permita la obtención del vector inicial de solución para el algoritmo de colonia de hormigas. Por tanto, se considera el proceso de identificar cuantos lotes de productos y máquinas se disponen en la línea de producción automatizada, así como la lectura de la información obtenida con la simulación de producción desarrollada en el laboratorio.

7.1. Aplicación de la regla de Johnson en la solución inicial de secuenciación para la línea de producción automatizada

Para la construcción de la solución inicial del problema, se aplica la regla de Johnson adaptada para cuatro máquinas en serie teniendo en cuenta que este algoritmo es usado únicamente para el caso de dos máquinas en serie. Dicho lo anterior, se deduce que:

$$F_4 // C_{max} \rightarrow F_2 // C_{max}$$

Así mismo, se considera la información de los tiempos como una matriz de procesamiento para cada máquina:

	L₁	L₂	L₃	...	L_n
M₁	p ₁₁	p ₁₂	p ₁₃	...	p _{1n}
M₂	p ₂₁	p ₂₂	p ₂₃	...	p _{2n}
M₃	p ₃₁	p ₃₂	p ₃₃	...	p _{3n}
M₄	p ₄₁	p ₄₂	p ₄₃	...	p _{4n}

Posteriormente, se reduce la matriz a una de dos máquinas, concentrando en M₁ y M₂ la información de las máquinas M₁ y M₂, y las máquinas M₃ y M₄ respectivamente. Para obtener la nueva matriz se aplican las siguientes expresiones (14) y (15):

$$p_{iM1'} = \sum_{j=M1}^{M2} p_{ij} \quad \forall i \in \text{lotes} \quad (14)$$

$$p_{iM2'} = \sum_{j=M3}^{M4} p_{ij} \quad \forall i \in \text{lotes} \quad (15)$$

Como resultado, se obtiene la siguiente matriz:

	L₁	L₂	L₃	...	L_n
M_{1'}	p _{11'}	p _{12'}	p _{13'}	...	p _{1n'}
M_{2'}	p _{21'}	p _{22'}	p _{23'}	...	p _{2n'}

Teniendo en cuenta la información anterior, se aplica el algoritmo de Johnson así:

$$\begin{array}{l}
 \min \\
 \text{If } \forall i \{p_{ij}\} \rightarrow p_{iM1'} \text{ Then} \\
 \{M1', M2'\} \\
 \text{Schedule } \begin{array}{cccccc} 1 & 2 & 3 & \dots & n \\ \{i, & 0, & 0, & \dots & ,0\} \end{array} \quad \forall i \in \text{producto}
 \end{array}$$

$$\begin{array}{l}
 \text{min} \\
 \text{ElseIf } \forall i \{p_{ij}\} \rightarrow p_{iM2'} \text{ Then} \\
 \forall j \in \{M1', M2'\} \\
 \\
 \text{Schedule } \begin{array}{cccccc} 1 & 2 & 3 & \dots & n \\ \{i, & 0, & 0, & \dots & ,0\} \end{array} \quad \forall i \in \text{producto}
 \end{array}$$

Para el caso particular que se trata en este proyecto, se cuenta con los tiempos de procesamiento de cuatro lotes de producción en cuatro máquinas, esta información es ordenada en una matriz para luego reducirlas de acuerdo a las expresiones (14) y (15) expuestas anteriormente.

	L ₁	L ₂	L ₃	L ₄
M ₁	3	3	4	3
M ₂	8	8	9	7
M ₃	5	6	7	7
M ₄	1	1	2	1

	L ₁	L ₂	L ₃	L ₄
M ₁	11	11	13	10
M ₂	6	7	9	8

En segunda instancia, se hace uso del algoritmo de Johnson, se encuentra como menor tiempo el lote uno en la máquina dos, seguido de este, el lote dos en la máquina dos, luego lote cuatro en la máquina dos y por último el lote cuatro en la máquina dos. Como resultado se obtiene el resultado de la secuencia.

Secuencia { L₃, L₄, L₂, L₁ }

En otras palabras, para llevar a cabo lo anteriormente dicho, el procedimiento se escribe en lenguaje de programación Visual Basic asociado al programa de Microsoft Excel. Inicialmente se realiza la lectura de los lotes mediante un ciclo *while*, luego se encarga de guardar los tiempos mediante un ciclo *for* en las respectivas variables declaradas previamente tales como control, clasificación, transporte y empaçado.

A continuación, antes de aplicar el algoritmo de Johnson, se suman las máquinas: control, clasificación, transporte y empaçado. Después se hace la búsqueda de cada tiempo menor en cada máquina usando *Do until* para hacer las iteraciones respectivas entre la máquina seleccionada y el lote seleccionado. Luego se selecciona el vector trabajo con el tiempo mínimo y se imprime el vector solución con los trabajos en el orden en que se deben llevar a cabo. En la figura 11 se presenta el diagrama de flujo referente a la dinámica del algoritmo de Johnson.

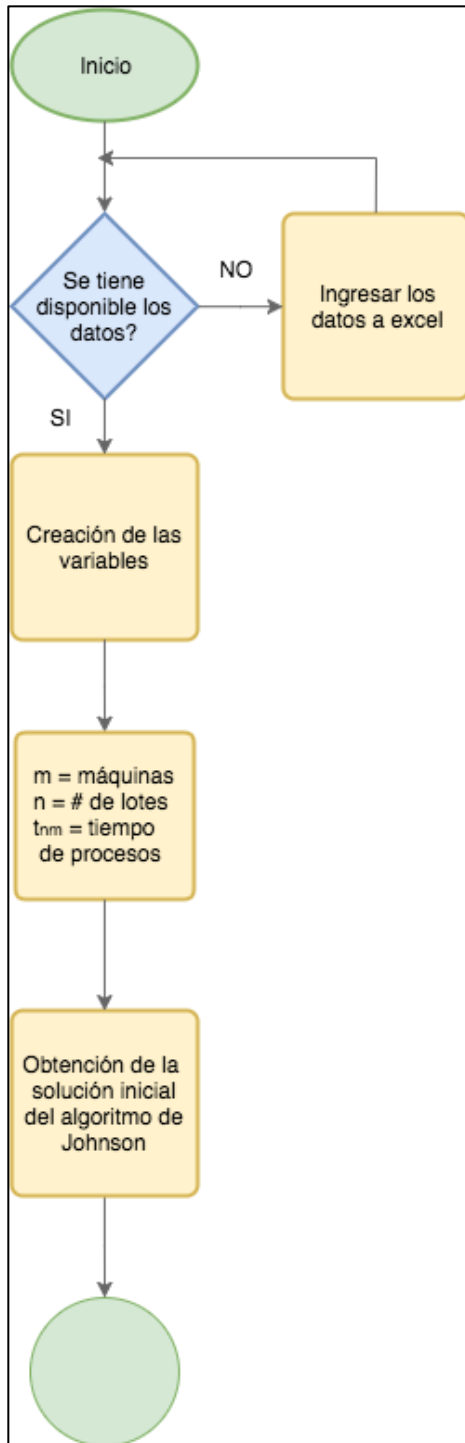


Figura 11. Pseudocódigo de la secuenciación de máquinas. Fuente: Autor

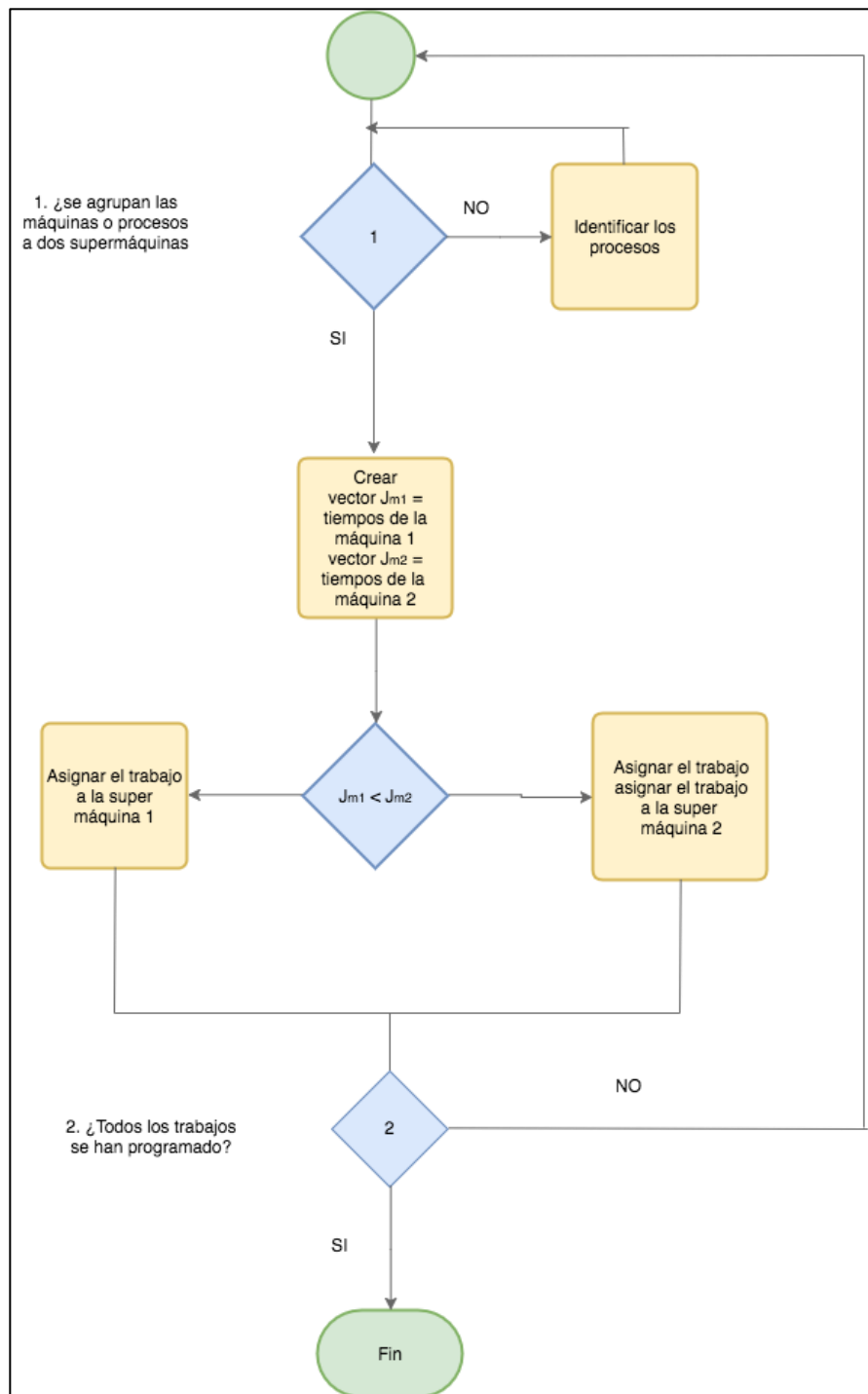


Figura 11. Pseudocódigo de la secuenciación de máquinas (continuación). Fuente: Autor

Finalmente se obtiene el valor del tiempo de terminación de los trabajos, el cual es de 42 minutos, el diagrama de Gantt – el cuál fue desarrollado en el código de programación Visual Basic - presenta la secuenciación y tiempo de terminación de los trabajos como se muestra en la

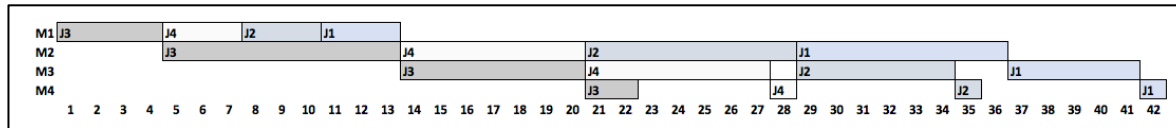


Figura 12. Programación de Operaciones de n-lotes en el sistema de producción automatizado. Fuente: Autor.

7.1.1 Calculo del tiempo mínimo de procesamiento de todos los trabajos.

Para la construcción del diagrama de Gantt, donde se presenta el tiempo de terminación de todos los trabajos (C_{max}), se consideran los siguientes casos para la secuenciación:

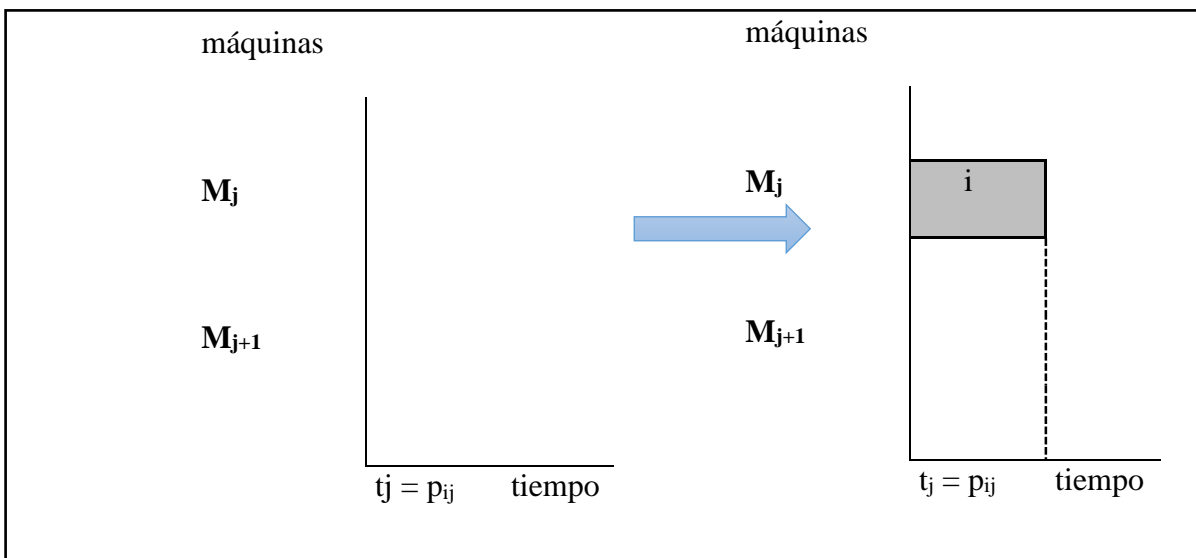
1. Ningún trabajo procesado en las máquinas
2. Asignación del primer trabajo cuando existe máquinas previamente asignadas
3. Asignación de un trabajo a una máquina que se encuentra ocupada

7.1.1.1 Ningún trabajo procesado en las máquinas.

En el caso de no haber ningún trabajo procesado en las máquinas, se considera que el tiempo de ocupación de la máquina (t_j) es cero; por tanto, al asignar un lote de producción a esa máquina, se actualiza el tiempo de ocupación con la expresión (16), ver figura 13.

$$t_j = p_{ij} \quad \forall j \in \text{máquina}$$

$$\forall i \in \text{secuenciación} \quad (16)$$



7.1.1.2 Asignación del primer trabajo cuando existen máquinas previamente asignadas.

Para este caso es necesario tener en cuenta el tiempo acumulado en la máquina previa. La actualización del tiempo de ocupación se realiza según la expresión (17). Ver figura 14.

$$T_{j+1} = t_j + p_{ij+1} \quad \forall j \in \text{máquina} \quad (17)$$

$$\forall i \in \text{secuenciación}$$

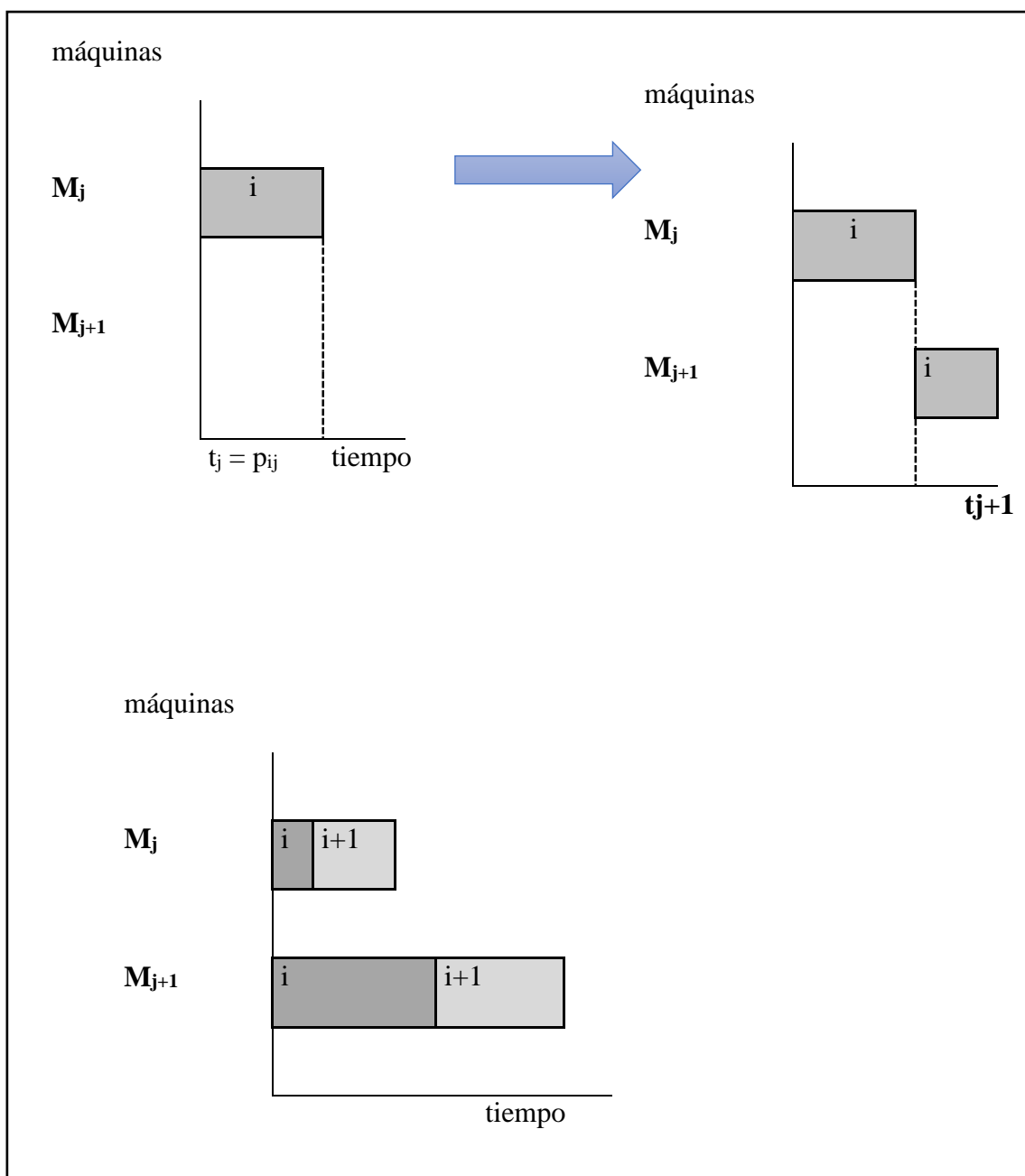


Figura 14. Asignación de un trabajo cuando existen máquinas previamente asignadas. *Fuente: Autor.*

7.1.1.3 Asignación de un trabajo a una máquina que se encuentra ocupada

Para este caso, se debe comparar y obtener el máximo tiempo entre procesar el lote en la máquina previa, y el tiempo de ocupación de la máquina actual. Esto se logra mediante la expresión (18).

$$t_{j+1} = \max\{t_j + \rho_{ji+1}, t_{j+1}\} + \rho_{j+1,i+1} \quad (18)$$

$$\forall \in \text{duling}$$

$$\forall \in \text{máquinas}$$

Si el máximo valor es t_{j+1} , la asignación es $t_{j+1} = t_{j+1} + \rho_{j+1, i+1}$, en caso contrario, la asignación es $t_{j+1} = t_j + \rho_{j+1} + \rho_{j+1, i+1}$ (ver figura 15).

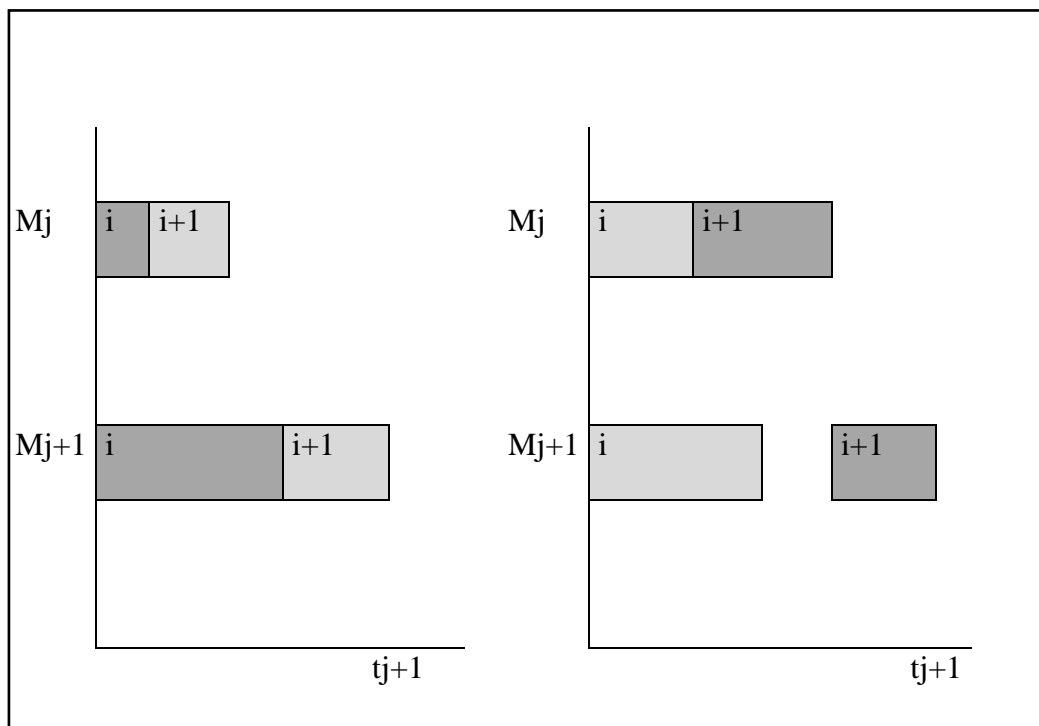


Figura 15. Asignación de un trabajo a una máquina que se encuentra ocupada. Fuente: Autor.

Una vez que se consideran los escenarios de asignación, se considera que la máquina tiempo de terminación de los trabajos está dado por la expresión (19).

$$Cmax = \underset{\forall i \in}{Max} \{\rho_i\} \quad (19)$$

Donde:

$$\rho_i = \sum_{j=1}^{M-1} t_j + \rho_{iM} \quad (20)$$

$\forall \in scheduling$

El diagrama de flujo para la generación de la solución inicial se muestra en la figura 16.



Figura 16. Diagrama de flujo para el diagrama de Gantt. Fuente: Autor

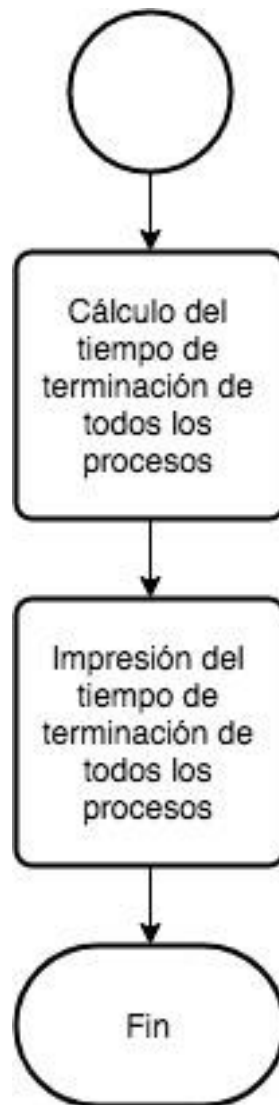


Figura 16. Diagrama de flujo para el diagrama de Gantt (continuación). Fuente: Autor

Para aplicar el algoritmo de colonia de hormigas, inicialmente se ha realizado la creación de hormigas con un bucle que inicializa la cantidad de hormigas requeridas por el usuario, luego se inicializan los parámetros tales como cantidad e iteraciones y disipación. Posteriormente se emplea un ciclo *Do until* para hacer la lectura de la información del problema, es decir, cantidad de trabajos, valor inicial del *makespan* (obtenido de la fase en la cual se utilizó el algoritmo de Jhonson). Seguido de esta lectura, se realiza el cálculo de los pesos para obtener la feromona mediante un ciclo *For*. Ver figura 17.

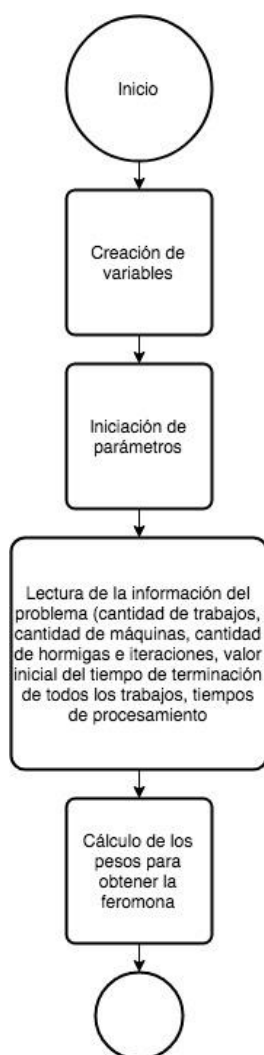


Figura 17. Algoritmo de colonia de hormigas. Fuente: Autor.

En la siguiente fase del algoritmo, se construye la solución generada por cada hormiga de la colonia, para esto, se hace uso de una serie de ciclos *Do until* para calcular inicialmente la intensidad del atractivo de la hormiga; luego, se define la feromona inicial con un *Do until* y varios *If* para realizar distintos procedimientos dependiendo de la iteración ya que debe ser diferente cuando se hace la primera iteración y cuando no. Continuamente, se obtiene el numerador y el denominador para el cálculo de la probabilidad, finalmente, se realiza el cálculo de la probabilidad de cada hormiga haciendo un bucle con la función *Do until* para que se generen todas las probabilidades por hormiga. Luego, se escoge el trabajo a secuenciar en el sistema de manufactura, nuevamente se hace uso del ciclo *Do until* para que se calculen y guarden los respectivos valores para cada hormiga y un condicional *If* para que en caso de que la variable aleatoria generada en pasos anteriores esté entre alguno de los valores de probabilidad, se secuencie el trabajo correspondiente a dicho intervalo. Ver figura 17.

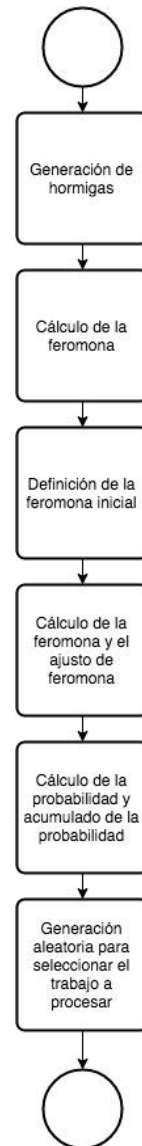


Figura 17. Algoritmo de colonia de hormigas (continuación). Fuente: Autor.

Se continúa empleando un par de *Do until* de tal manera que, según la solución obtenida por la hormiga, sea calculada la función objetivo, es decir, el *makespan*, posteriormente se hace uso del condicional *If para* realizar un diagrama de Gantt sin imprimir poder guardar el valor obtenido de C_{max} . Luego se calcula el *makespan* de la solución final. Posteriormente se calcula la contribución de feromona por hormiga. Ver figura 17.

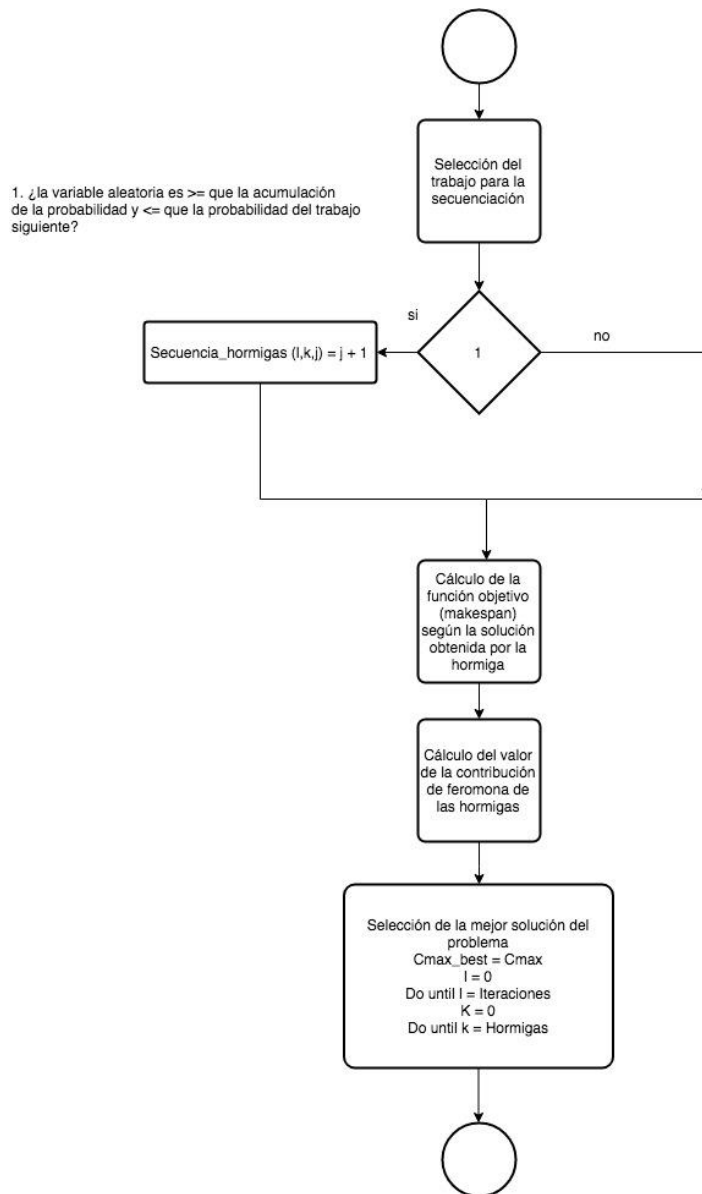


Figura 17. Algoritmo de colonia de hormigas (continuación). Fuente: Autor.

Al final, se selecciona la mejor solución del problema de acuerdo al algoritmo de colonia de hormigas seleccionando el *makespan* menor y su respectiva iteración y hormiga. Procediendo por último con la impresión de un diagrama de Gantt con la solución final. Ver figura 17.

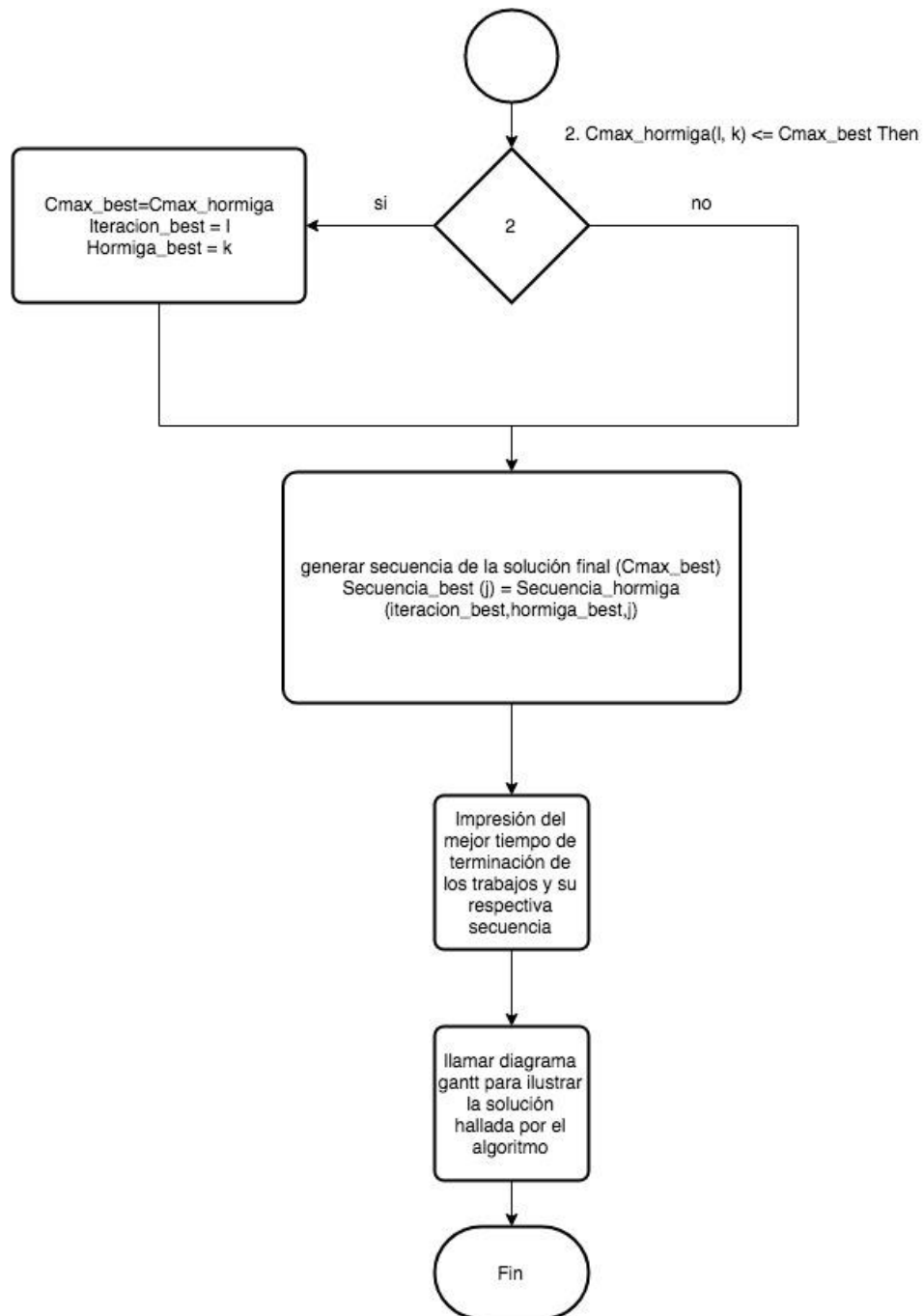


Figura 17. Algoritmo de colonia de hormigas (continuación). Fuente: Autor.

8. Análisis de resultados

Al analizar el comportamiento de las soluciones brindadas por la ejecución del algoritmo de colonia de hormigas, se han compilado una serie de resultados para el tiempo computacional y *makespan* realizando ocho réplicas por cada conjunto de efectos, en total se han realizado 128 ejecuciones del programa por cada instancia. Ver los mejores resultados en el anexo A y B.

8.1 Resultados del problema de cuatro máquinas y cuatro trabajos.

Para poder analizar el comportamiento de las fuentes de variación al dar solución al problema de la línea de producción del laboratorio de automatización de la Universidad Pontificia Bolivariana seccional Bucaramanga, se realizan 128 pruebas repartidas en grupos de 8 réplicas de un diseño experimental de tres factores, finalmente se obtiene un tiempo computacional de 0.22 segundos y un tiempo mínimo de terminación de todos los procesos (*Makespan*) de 451,02 minutos en promedio.

Las variables que se pueden controlar en el experimento son: cantidad de hormigas el factor A, cantidad de iteraciones el factor B y disipación el factor C, a continuación se resumen los análisis de varianza para los resultados obtenidos de *makespan* y tiempo computacional en la tabla 5 y 6 respectivamente. Las pruebas se realizaron con cuatro niveles para cada factor, así: 12, 13, 14 y 15, para el factor número de iteraciones; 9, 12, 20 y 25 y finalmente para el factor disipación se usaron los niveles; 0.2, 0.7, 0.8 y 0.9.

Tabla 5.

Análisis de varianza para los datos de Makespan para el problema de cuatro máquinas y cinco trabajos.

Makespan							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	0,0029296875	a-1	3,00	MSA	0,00	0,00
B	SSB	38158,85	b-1	3,00	MSB	12719,62	1085407,33
C	SSC	-2860530,93	c-1	3,00	MSC	-953510,31	-57824365,91
AB	SSAB	-2896998,44	(a-1)(b-1)	9,00	MSAB	-321888,72	-19520513,52
AC	SSAC	1719,53	(a-1)(c-1)	9,00	MSAC	191,06	11586,50
BC	SSBC	3216482,72	(b-1)(c-1)	9,00	MSBC	357386,97	21673257,92
ABC	SSABC	2895279,02	(a-1)(b-1)(c-1)	27,00	MSABC	107232,56	6502975,91
Subtotales(ABC)	SSSubtotales(ABC)	394110,76					
ERROR	SSE	9764263,31	abc(n-1)	833217072,00	MSE	0,01	
TOTAL	SST	10158374,07	abcn-1	833217135,00			

Fuente: Autor.

Teniendo en cuenta que el *makespan* es el tiempo mínimo de terminación de todos los procesos de una línea de producción, se busca que los resultados sean cada vez más pequeños y que el coeficiente F sea negativo ya que esto significa que al colocar los niveles de los factores con valores más altos, se espera los tiempos disminuyan continuamente.

Se puede observar en la tabla 5 que la suma de cuadrados para el factor cantidad de hormigas es positiva, por lo tanto al incrementar la cantidad de hormigas, aumenta el tiempo mínimo de producción, lo que significa que al usar en este problema una cantidad menor de hormigas conlleva un hallazgo de resultados más optimos. Igualmente para el factor cantidad de iteraciones resutaría mejor colocar disipaciones de valores pequeños. Por ultimo para la disipación se observa que al aumentar el valor, los tiempos del *makespan* resultan menores, es decir, se espera que disipaciones de 0.9 y mayores arrojen un resultado más deseable para este tipo de problema; mientras que 9 iteraciones e inferiores y una cantidad de hormigas igual o menor a 12 resultarían mejor.

Posteriormente en la interacción de las fuentes de variación AB, AC, BC y ABC, se observa en la tabla 5 que de estas combinaciones el unico caso que disminuye el resultado, es cuando se

aumenta el valor de los niveles de ambos factores al tiempo es AB, el resto de combinaciones daría por respuesta un resultado menor con valores pequeños de AC, BC y ABC.

Tabla 6.

Análisis de varianza para los datos del tiempo computacional para el problema de cuatro máquinas y cinco lotes.

Tiempo computacional							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	0,55	a-1	3,00	MSA	0,18	11,15
B	SSB	0,64	b-1	3,00	MSB	0,21	12,98
C	SSC	-0,15	c-1	3,00	MSC	-0,05	-3,10
AB	SSAB	-0,54	(a-1)(b-1)	9,00	MSAB	-0,06	-3,62
AC	SSAC	-0,37	(a-1)(c-1)	9,00	MSAC	-0,04	-2,49
BC	SSBC	0,15	(b-1)(c-1)	9,00	MSBC	0,02	1,03
ABC	SSABC	0,36	(a-1)(b-1)(c-1)	27,00	MSABC	0,01	0,81
Subtotales(ABC)	SSSubtotales(ABC)	0,65					
ERROR	SSE	2,59	abc(n-1)	157,20	MSE	0,02	
TOTAL	SST	3,24	abcn-1	220,20			

Fuente: Autor.

Para el caso del tiempo computacional, los factores individuales de: cantidad de hormigas y numero de iteraciones, es decir, A y B, suele aumentar el resultado al disminuir los valores de los niveles de A y B, para el factor C, los resultados disminuyen al aumentar los valores de los cuatro niveles; por otro lado, para las variaciones de AB y AC, si se aumentan las cantidades de ambas combinaciones, se hace menor el valor de los resultados, mientras que para el caso de ABC aumentan los resultados al aumentar los valores de las variables. Ver tabla 6.

9. Evaluación de la calidad de la solución del algoritmo desarrollado en el programa Microsoft Visual Basic.

Para evaluar la calidad de la solución del algoritmo desarrollado en el programa Microsoft Visual Basic, se ha hecho uso de instancias de la literatura halladas en la página de Sistemas de Optimización Aplicada SOA <http://soa.iti.es/problem-instances>. En la figura 18 se muestra la página de inicio de la página web en la cual se encontraron las instancias.

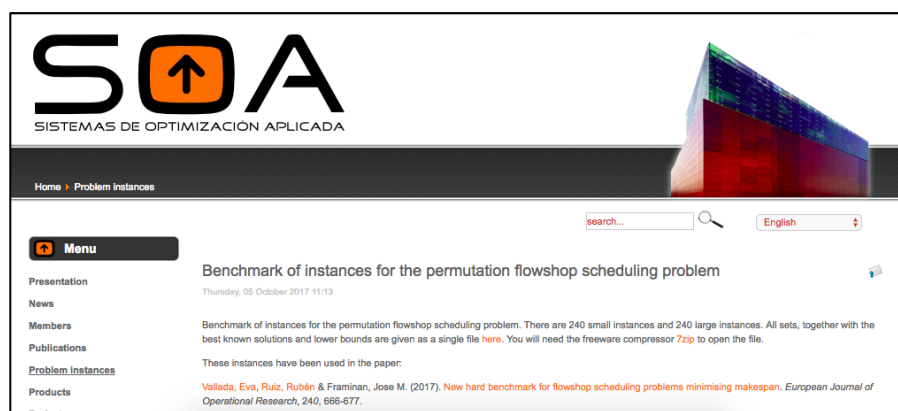


Figura 18. Inicio de la página web de las instancias. Fuente: (Sistema de Optimización Aplicada, s.f.)

Al llevar a cabo la evaluación del programa propuesto se ejecutan varios conjuntos de datos los cuales tienen rangos de respuesta encontradas en la literatura que permiten determinar qué tan óptimos son los resultados hallados por el algoritmo desarrollado.

Para evaluar el programa haciendo uso las instancias de la literatura, se tienen en cuenta los datos de tiempo de procesamiento por máquina y lote de:

- Dos instancias compuestas por diez máquinas y diez trabajos
- dos instancias de diez máquinas y quince lotes

- por último se tomó una instancia de sesenta máquinas y veinte lotes para un total de cinco instancias.

Tabla 7.

límites superior e inferior de las instancias y resultados promedio hallados con el programa propuesto.

Instancia	UB	LB	Resultado Promedio
10x01_1	1097	797	1133,582031
10x01_2	1146	845	1186,69375
10x15_1	1307	921	1380,435547
10x15_2	1399	988	1465,373047
60x20_1	4163	3689	4989,716797

Fuente: Autor.

Luego de realizar las respectivas ejecuciones del programa para encontrar el makespan promedio arrojado como solución por el programa propuesto, se compara este con los límites superior e inferior (UB y LB respectivamente) de las instancias de la literatura. En la tabla 7 se muestran dichos datos y se comparan, finalmente se puede observar que todos los resultados están dentro del rango; en la figura 19 se presenta una gráfica de los límites superiores e inferiores para todas las instancias y se evidencia visualmente que los resultados hallados se encuentran dentro del rango propuesto por las instancias de la literatura.

Teniendo en cuenta lo anteriormente dicho, se puede señalar que el programa propuesto da una buena respuesta para todos los casos, aunque no sean los resultados más óptimos que se pueden encontrar se evidencia que los hallazgos tienden al límite inferior.

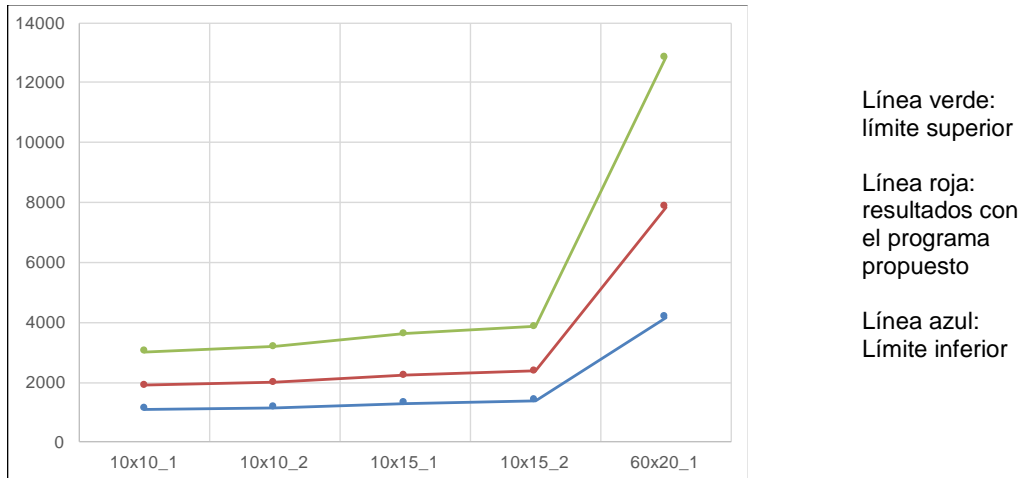


Figura 19. Gráfica de los límites superior e inferior y los resultados hallados por el programa propuesto. Fuente: Autor.

A continuación, se explica el comportamiento de las fuentes de variación, para poder comprender dicho comportamiento, se ha realizado un análisis de varianza (ANOVA) para el cual se han tenido en cuenta cuatro niveles en cada uno de los factores; seguidamente, se muestra en la tabla los niveles utilizados para los experimentos de cada instancia. Ver tabla 8.

Tabla 8.

Niveles de los factores.

Instancia	Niveles de los factores		
	Cantidad de hormigas	Número de iteraciones	Disipación
10_10_1	15,25,30,45	9,12,20,25	0.2, 0.7, 0.8, 0.9
10_10_2	2,4,6,8	19,29,39,49	0.1, 0.2, 0.3, 0.4
10_15_1	3,5,7,9	10,12,14,16	0.6, 0.7, 0.8, 0.9
10_15_2	9,12,15,18	19,22,25,28	0.1, 0.3, 0.5, 0.7
60_20_1	6,8,10,12	15,25,35,45	0.1, 0.4, 0.5, 0.8

Fuente: Autor.

9.1 Análisis de varianza para la instancia 10_10_1

La instancia 10_10_1 está compuesta por diez máquinas y diez lotes; luego de realizar 8 réplicas para cada conjunto de factores, se procede con un análisis de varianza tanto para el *makespan* como para el tiempo computacional. Con respecto al factor A y B, se evidencia en la tabla 9 que en ambos factores aumenta el resultado al aumentar el valor de los niveles, por lo tanto, sería útil emplear niveles menores a 25 y 15 respectivamente; en el factor C la solución disminuye, por lo cual en este último factor es mejor utilizar niveles mayores a 0.8.

Tabla 9.

ANOVA para los resultados del *makespan* de la instancia 10_10_1

Makespan							
Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	F°			
A	SSA	305,66	a-1	3,00	MSA	101,89	8694,42
B	SSB	268028,77	b-1	3,00	MSB	89342,92	7623929,39
C	SSC	-18058366,70	c-1	3,00	MSC	-6019455,57	-420093168,05
AB	SSAB	-18322448,75	(a-1)(b-1)	9,00	MSAB	-2035827,64	-142078843,01
AC	SSAC	4017,90	(a-1)(c-1)	9,00	MSAC	446,43	31156,23
BC	SSBC	20292204,48	(b-1)(c-1)	9,00	MSBC	2254689,39	157353035,82
ABC	SSABC	18318692,53	(a-1)(b-1)(c-1)	27,00	MSABC	678470,09	47349905,32
Subtotales(ABC)	SSSubtotales(ABC)	2502433,90					
ERROR	SSE	61683247,62	abc(n-1)	5263637066,00	MSE	0,01	
TOTAL	SST	64185681,51	abcn-1	5263637129,00			

Fuente: Autor.

En consideración al tiempo computacional se expone en la tabla 10 que al aumentar tanto el factor A como el B sube el valor del resultado de las pruebas, mientras que al aumentar el valor de los niveles del factor C, la cifra del resultado obtenido es menor.

Tabla 10.

ANOVA para el tiempo computacional de la instancia 10_10_1

Tiempo computacional							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	0,92	a-1	3,00	MSA	0,31	21,46
B	SSB	0,81	b-1	3,00	MSB	0,27	18,96
C	SSC	-0,16	c-1	3,00	MSC	-0,05	-3,76
AB	SSAB	-1,03	(a-1)(b-1)	9,00	MSAB	-0,11	-8,03
AC	SSAC	-0,59	(a-1)(c-1)	9,00	MSAC	-0,07	-4,59
BC	SSBC	0,16	(b-1)(c-1)	9,00	MSBC	0,02	1,25
ABC	SSABC	0,99	·1)(b-1)(c-1)	27,00	MSABC	0,04	2,56
Subtotales(ABC)	SSSubtotales(ABC)	1,10					
ERROR	SSE	4,12	abc(n-1)	287,34	MSE	0,01	
TOTAL	SST	5,22	abcn-1	350,34			

Fuente: Autor.

9.2 Análisis de varianza para la instancia 10_10_2

La instancia 10_10_2, es una instancia de diez máquinas y diez trabajos, esta instancia contiene tiempos de procesamiento diferentes a la instancia 10_10_1, a continuación, se realiza un análisis de varianza para los resultados obtenidos luego de 128 pruebas divididas en grupos de ocho réplicas por cada conjunto de factores variables.

En el ANOVA de la instancia 10_10_2 resumido en la tabla 11 se observa que, para el factor de cantidad de hormigas y número de iteraciones, es decir, A y B, hay una relación directamente proporcional, dicho de otra manera, al aumentar los valores de los niveles, aumenta el resultado del makespan y para el factor C, disipación, hay una relación inversamente proporcional, en otras palabras, al aumentar el valor de los niveles, el resultado disminuye. Por ende, al buscar valores mínimos, se puede decir que lo ideal es usar niveles pequeños para A y B y niveles altos para el factor disipación representado por la letra C.

Tabla 11.

ANOVA para la instancia 10_10_2

Makespan							
Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	F°			
A	SSA	581,51	a-1	3,00	MSA	193,84	0,00
B	SSB	295040,20	b-1	3,00	MSB	98346,73	0,28
C	SSC	-19777320,95	c-1	3,00	MSC	-6592440,32	-5575321642,84
AB	SSAB	-20068490,91	(a-1)(b-1)	9,00	MSAB	-2229832,32	-1885801283,48
AC	SSAC	3944,20	(a-1)(c-1)	9,00	MSAC	438,24	370629,90
BC	SSBC	22233810,32	(b-1)(c-1)	9,00	MSBC	2470423,37	2089272592,82
ABC	SSABC	20064929,00	(a-1)(b-1)(c-1)	27,00	MSABC	743145,52	628488858,99
Subtotales(ABC)	SSSubtotales(ABC)	2752493,37					
ERROR	SSE	67590327,02	abc(n-1)	192	MSE	352032,95	
TOTAL	SST	70342820,40	abcn-1	255			

Fuente: Autor.

En el caso del tiempo computacional, en todos los casos, es decir tanto para el factor cantidad de hormigas, número de iteraciones y disipación, es preferible hacer uso de variables o niveles pequeños ya que arrojarán un resultado más pequeño. En la tabla 10 se muestra el análisis de varianza para el tiempo computacional de la instancia 10_10_2. Ver tabla 12.

Tabla 12.

ANOVA para el tiempo computacional de la instancia 10_10_2

Tiempo computacional							
Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	F°			
A	SSA	0,05	a-1	3,00	MSA	0,02	15,07
B	SSB	0,05	b-1	3,00	MSB	0,02	13,39
C	SSC	0,01	c-1	3,00	MSC	0,00	2,30
AB	SSAB	0,12	(a-1)(b-1)	9,00	MSAB	0,01	11,71
AC	SSAC	0,02	(a-1)(c-1)	9,00	MSAC	0,00	1,74
BC	SSBC	-0,01	(b-1)(c-1)	9,00	MSBC	0,00	-0,76
ABC	SSABC	-0,18	(a-1)(b-1)(c-1)	27,00	MSABC	-0,01	-5,77
Subtotales(ABC)	SSSubtotales(ABC)	0,06					
ERROR	SSE	0,23	abc(n-1)	192	MSE	0,00	
TOTAL	SST	0,29	abcn-1	255			

Fuente: Autor.

9.3 Análisis de varianza para la instancia 10_15_1.

Los resultados del análisis de varianza ANOVA para el caso de la instancia 10_15_1 (compuesta por quince máquinas y diez trabajos) se evidencian resumidos en la tabla 13.

Tabla 13.

ANOVA para el makespan de la instancia 10_15_1.

Makespan							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	990,52	a-1	3	MSA	330	0
B	SSB	356062,98	b-1	3	MSB	118688	0
C	SSC	-26794191,78	c-1	3	MSC	-8931397	-29418802495
AB	SSAB	-27146066,37	(a-1)(b-1)	9	MSAB	-3016230	-9935048258
AC	SSAC	4275,54	(a-1)(c-1)	9	MSAC	475	1564783
BC	SSBC	30133627,00	(b-1)(c-1)	9	MSBC	3348181	11028450101
ABC	SSABC	27142595,43	(a-1)(b-1)(c-1)	27	MSABC	1005281	3311259316
Subtotales(ABC)	SSSubtotales(ABC)	3697293,32					
ERROR	SSE	91475483,80	abc(n-1)	192	MSE	476435	
TOTAL	SST	95172777,12	abcn-1	255			

Fuente: Autor.

En la tabla 13 se observa que los niveles del factor A y el factor B se encuentran directamente relacionados con el resultado de las pruebas, por lo tanto, es conveniente hacer uso de niveles pequeños, mientras que los niveles del factor C tiene una relación inversamente proporcional con los resultados de las pruebas.

Tabla 14.

ANOVA para el tiempo computacional de la instancia 10_15_1.

Tiempo computacional							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	0,01	a-1	3,00	MSA	0,00489	16,11
B	SSB	0,01	b-1	3,00	MSB	0,00436	14,35
C	SSC	0,01	c-1	3,00	MSC	0,00250	8,24
AB	SSAB	0,10	(a-1)(b-1)	9,00	MSAB	0,01099	36,20
AC	SSAC	0,02	(a-1)(c-1)	9,00	MSAC	0,00232	7,65
BC	SSBC	-0,01	(b-1)(c-1)	9,00	MSBC	-0,00083	-2,74
ABC	SSABC	-0,13	(a-1)(b-1)(c-1)	27,00	MSABC	-0,00491	-16,17
Subtotales(ABC)	SSSubtotales(ABC)	0,02					
ERROR	SSE	0,06	abc(n-1)	192	MSE	0,00030	
TOTAL	SST	0,07	abcn-1	255			

Fuente: Autor.

Se puede observar en la tabla 14 el ANOVA del tiempo computacional de la instancia 10_15_1, en dicho análisis se evidencia que siempre que aumentan los niveles de los factores el tiempo computacional aumenta, dado que, al haber más iteraciones y hormigas, tomará más tiempo el cálculo del makespan.

9.4 Análisis de varianza para la instancia 10_15_2.

La instancia 10_15_2 está compuesta por quince máquinas y diez trabajos, los tiempos de procesamiento por máquina y trabajo proporcionados por esta instancia de la literatura, arrojan tiempos mínimos de procesamiento (*makespan*) que finalmente permite realizar un análisis de varianza para finalmente conocer la variación de los resultados con respecto a la variación de los niveles de los factores A, B y C, los cuales representan los factores cantidad de hormiga, número de iteraciones y disipación respectivamente. En la tabla 15 se muestra dicho análisis.

Tabla 15.

ANOVA para el análisis de varianza del makespan de la instancia 10_15_2

Makespan							
Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	F°			
A	SSA	265,04	a-1	3	MSA	88	0,000165
B	SSB	408663,87	b-1	3	MSB	136221	0,253744
C	SSC	-30195003,01	c-1	3	MSC	-10065001	-141985444
AB	SSAB	-30598435,85	(a-1)(b-1)	9	MSAB	-3399826	-47960833
AC	SSAC	5323,12	(a-1)(c-1)	9	MSAC	591	8344
BC	SSBC	33948225,10	(b-1)(c-1)	9	MSBC	3772025	53211385
ABC	SSABC	30593655,68	(a-1)(b-1)(c-1)	27	MSABC	1133098	15984447
Subtotales(ABC)	SSSubtotales(ABC)	4162693,94					
ERROR	SSE	103074187,14	abc(n-1)	192	MSE	536845	
TOTAL	SST	107236881,08	abcn-1	255			

Fuente: Autor.

Se puede observar en el ANOVA de makespan que existe relación entre la variación del makespan y la variación del de los niveles de la cantidad de hormigas y el número de iteraciones mientras que el factor disipación y la combinación de factores cantidad de hormigas – número de iteraciones tienen una relación inversamente proporcional entre la variación de los factores y la del makespan. Por otro lado, se puede decir que la combinación de factores, cantidad de hormigas – disipación y número de iteraciones – disipación tienen una relación directamente proporcional. Cabe resaltar que los factores cuya relación sea directamente proporcional a la variación del makespan deben tener niveles con valores más bajos y los factores con relación inversamente proporcional pueden tener niveles con valores más altos para obtener resultados de *makespan* pequeños. Ver tabla 16.

Tabla 16.

ANOVA para el tiempo computacional de la instancia 10_15_2.

Tiempo computacional							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	0,23	a-1	3,00	MSA	0,07519	16,07
B	SSB	0,21	b-1	3,00	MSB	0,06881	14,70
C	SSC	-0,02	c-1	3,00	MSC	-0,00753	-1,61
AB	SSAB	0,02	(a-1)(b-1)	9,00	MSAB	0,00222	0,47
AC	SSAC	-0,09	(a-1)(c-1)	9,00	MSAC	-0,00965	-2,06
BC	SSBC	0,02	(b-1)(c-1)	9,00	MSBC	0,00252	0,54
ABC	SSABC	-0,13	(a-1)(b-1)(c-1)	27,00	MSABC	-0,00498	-1,06
Subtotales(ABC)	SSSubtotales(ABC)	0,23					
ERROR	SSE	0,90	abc(n-1)	192	MSE	0,00468	
TOTAL	SST	1,13	abcn-1	255			

Fuente: Autor.

Una vez más para el tiempo computacional se puede decir que entre mayor sean la cantidad tanto de hormigas como de iteraciones, mayor será el tiempo computacional, dado el tiempo que representa para el software calcular una mayor cantidad de escenarios de secuenciación de máquina para escoger el orden más conveniente en el que se deben producir los lotes.

9.5 Análisis de varianza para la instancia 60_20_1

A continuación, se analizan los datos arrojados por el programa al ejecutar los datos ingresados, estos últimos, proporcionados por la instancia 60_20_1, una instancia de sesenta trabajos y veinte máquinas. Inicialmente, el análisis de varianza del *makespan* nos deja ver que los factores A y B, es decir, cantidad de hormigas y de iteraciones, se comportan de una manera directamente proporcional a la variación de los niveles de los factores, dicho de otra manera, al aumenta los valores de los niveles, aumenta el resultado final del *makespan*, por lo tanto, se deben usar valores pequeños en los niveles de dichos factores.

Por el contrario, para el factor número de iteraciones, la relación entre la variación las fuentes de variación y el makespan es inversamente proporcional. Igualmente, para la combinación de factores cantidad de hormiga – numero de iteraciones la relación se encuentra inversamente proporcional, lo que permite el uso de valores más altos para los niveles de dicha combinación. Ver tabla 17.

Tabla 17.

ANOVA para el makespan de la instancia 60_20_1.

Makespan							
Fuente de variación	Suma de cuadrados	Grados de libertad	Cuadrado medio	F°			
A	SSA	19,69	a-1	3	MSA	7	0,0000011
B	SSB	4686154,12	b-1	3	MSB	1562051	0,2509592
C	SSC	-350099689,30	c-1	3	MSC	-116699896	-11692549
AB	SSAB	-354767151,22	(a-1)(b-1)	9	MSAB	-39418572	-3949477
AC	SSAC	19003,96	(a-1)(c-1)	9	MSAC	2112	212
BC	SSBC	393669045,08	(b-1)(c-1)	9	MSBC	43741005	4382556
ABC	SSABC	354748370,56	(a-1)(b-1)(c-1)	27	MSABC	13138829	1316423
Subtotales(ABC)	SSSubtotales(ABC)	48255752,89					
ERROR	SSE	1195070067,53	abc(n-1)	192	MSE	6224323	
TOTAL	SST	1243325820,42	abcn-1	255			

Fuente: Autor.

Con respecto al tiempo computacional, se puede observar en la tabla 18 que tanto los niveles del factor A tanto como los del factor B, tienen una relación directamente proporcional con el tiempo computacional, por el contrario, el factor C tiene una relación inversamente proporcional. Por lo tanto, se deben usar cantidades de hormigas y de iteraciones pequeñas mientras que se pueden usar disipaciones altas para que el tiempo computacional no sea tan alto.

Tabla 18.

ANOVA para el tiempo computacional de la instancia 60_20_1

Tiempo computacional							
Fuente de variación	Suma de cuadrados		Grados de libertad		Cuadrado medio		F°
A	SSA	413,08	a-1	3	MSA	137,69	13,80
B	SSB	445,10	b-1	3	MSB	148,37	14,87
C	SSC	-146,41	c-1	3	MSC	-48,80	-4,89
AB	SSAB	-983,71	(a-1)(b-1)	9	MSAB	-109,30	-10,95
AC	SSAC	-407,23	(a-1)(c-1)	9	MSAC	-45,25	-4,53
BC	SSBC	146,45	(b-1)(c-1)	9	MSBC	16,27	1,63
ABC	SSABC	1023,12	(a-1)(b-1)(c-1)	27	MSABC	37,89	3,80
Subtotales(ABC)	SSSubtotales(ABC)	490,40					
ERROR	SSE	1916,30	abc(n-1)	192	MSE	9,98	
TOTAL	SST	2406,70	abcn-1	255			

Fuente: Autor.

Por ultimo cabe resaltar que los parámetros adecuados para el algoritmo según los experimentos estadísticos desarrollados en la investigación son: 9 hormigas, 19 iteraciones y 0,7 de disipación, ya que han sido los valores que menor varianza generan entre las respuestas.

10. Conclusiones.

El algoritmo de colonia de hormigas tiene un buen rendimiento y es aplicable en varias áreas dentro de los problemas de optimización complejos como, por ejemplo: El problema de equilibrado de líneas de montaje, balance de fases multiobjetivo en el sistema de distribución, problema del agente viajero, problema de la mochila, problema de enrutamiento y el problema de asignación de horarios.

En este trabajo se ha cumplido con el objetivo de implementar el algoritmo de colonia de hormigas en el programa Microsoft Visual Basic, caracterizar los procesos industriales que componen la línea de producción automatizada del laboratorio de procesos y de desarrollar un estudio de las instancias de la literatura del problema para evaluar la calidad de solución del programa desarrollado y finalmente proponer la secuenciación para el problema.

Después de realizar la evaluación del programa propuesto con respecto a 5 instancias de la literatura, e implementar un diseño de experimento factorial con su respectivo análisis de varianza, se concluye que la calidad del algoritmo propuesto es buena ya que los resultados al ejecutar el programa con los datos de las instancias, se encuentran dentro del rango aprobado por la literatura, por otro lado, se ha logrado comprobar la relación entre los factores y la variabilidad de los resultados. Para el caso particular estudiado, se concluye que tanto la cantidad de hormigas e iteraciones como el valor de la disipación, no altera considerablemente los resultados, por lo que no se sugiere una cantidad específica de estos factores para hacer uso del algoritmo. Por el contrario, en casos con diez o más trabajos, se puede decir que el programa dará mejores resultados al hacer uso de cantidades de hormigas e iteraciones pequeñas, mientras que, para el caso de la disipación, es recomendable hacer uso de valores más cercanos a uno.

Igualmente, la solución del caso de estudio que se presenta en el laboratorio de control de procesos tras 128 pruebas realizadas con diferentes cantidades de hormigas, número de iteraciones y disipación, es casi homogénea, del mismo modo, al hacer pruebas con dos instancias de la literatura de cinco máquinas y cinco trabajos no mencionada anteriormente, se ha observado igualmente que las soluciones finales son casi homogéneas, incluso en algunos casos homogéneas, por lo que se concluye que, al ser un caso con una cantidad muy pequeña de trabajos, se puede llegar a una solución fácilmente con una baja cantidad de hormigas.

BIBLIOGRAFÍA

Alonso, S. et ál. La metaheurística de optimización basada en colonias de hormigas: modelos y nuevos enfoques. Departamento de Ciencias de la Computación e Inteligencia Artificial. E.T.S. Universidad de Granada.

Dorigo, M. & Gambardella, M. (1996). The Ant System: Optimization by a Colony Cooperating Agents. IEEE Transactions on Systems, Man, and Cybernetics, vol. 26 (1).

Dréo, J. (2006). *Metaheuristics for hard optimization: Methods and case studies*. (Springer ed.). Berlin.

Hornig, E. S. (2011). Aplicación de un algoritmo ACO al problema de taller de flujo de permutación con tiempos de preparación dependientes de la secuencia y minimización de makespan. *Ingeniare. Revista chilena de Ingeniería*.

Horning, E. S. (2016). Aplicación de un algoritmo ACO al problema de flow shop flexible con tiempos de preparación dependientes de la secuencia y minimización de la tardanza total. *Ingeniare. Revista Chilena de ingeniería*.

Pinedo, M. L. (1995). *Scheduling. Theory, Algorithms, and Systems* (4^a edition ed.). Springer.

Robles, C.A. (2010). Optimización por colonia de hormigas: aplicaciones y tendencias. *Revista Ingeniería Solidaria*, vol. 6 (10).

Trabajos citados

Horning, E. S. (2016). Aplicación de un algoritmo ACO al problema de flowshop flexible con tiempos de preparación dependientes de la secuencia y minimización de la tardanza total. *Ingeniare. Revista Chilena de ingeniería*.

Hornig, E. S. (2011). Aplicación de un algoritmo ACO al problema de taller de flujo de permutación con tiempos de preparación dependientes de la secuencia y minimización de makespan. *Ingeniare. Revista chilena de Ingeniería*.

Pinedo, M. L. (1995). *Scheduling. Theory, Algorithms, and Systems* (4^a edition ed.). Springer.

Dréo, J. (2006). *Metaheuristics for hard optimization: Methods and case studies*. (Springer ed.). Berlin.

Universidad Pontificia Bolivariana. Bucaramanga. (Noviembre de 2014). Línea de producción manual de usuario. Laboratorio de automatización. 1. Bucaramanga, Colombia.

Reeves, C. R. (1998). Genetic Algorithms, Path Relinking and the Flowshop Sequencing Problem . (N. C. Laboratories, Ed.) *Evolutionary Computation journal*, 6(1), 230-234.

Sahraeian, A. (s.f.). Minimizing Makespan in Flow Shop Scheduling Using a Network Approach. *Department of Industrial Engineering, Payame Noor University, Asaluyeh, Iran*, 49-50.

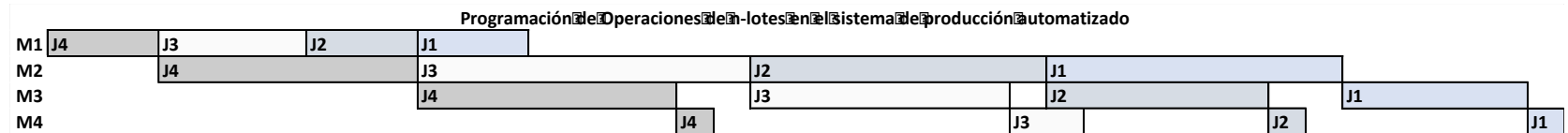
Hartl, R., & Preusser, M. (2009). *Layout and Design* . Curso Layout and Design / Cátedra Production and Operations Management / Department of Business Administration / Faculty of Business, Economics and St .

Mario A. Muñoz, J. A. (Mayo de 2007). Optimización por Colonia de Hormigas para la Asignación Dinámica de Recursos en una Plataforma de Experimentación de Temperatura Multizona . *IEEE LATIN AMERICA TRANSACTIONS*, 5(2).

- al, M. e. (2000). Ant Colony Approach to Continuous Function Optimization. *Ind. Eng. Chem*, 39(10).
- al., N. e. (2016). Chemical disguise of myrmecophilous cockroaches and its implications for understanding nestmate recognition mechanisms in leaf-cutting ants. *BMC Ecology*, 10 - 11.
- Ortega, A. C., & Bedia, A. S. (s.f.). UN ALGORITMO HÍBRIDO BASADO EN COLONIAS DE HORMIGAS PARA LA RESOLUCIÓN DE PROBLEMAS DE DISTRIBUCIÓN EN PLANTA ORIENTADOS A PROCESOS. *XIII Jornadas de ASEPUMA*.
- Tibaduiza, E. P., Hernandez, E. G., Figueredo, L. P., & Chimenty, E. M. (2017). Revisión de la Literatura Sobre el Problema de Programación de “Flow Shop” Híbrido con Máquinas Paralelas no Relacionadas. *Ingeniería*, 22(1), 10.
- ál., M. e. (June de 2014). Improved cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan. (Elsevier, Ed.)
- Algarín, C. A. (2010). Optimización por colonia de hormigas: Aplicaciones y tendencias. *Ingeniería solidaria*, 6(10), 83 - 89.
- Dorigo, M. G. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem, Bruselas, Université Libre de Bruxelles. *1*(1), 1 - 26.
- Pulido, H. G., & Salazar, R. d. (2008). *Análisis y diseño de experimentos* (2ª edición ed.). Guanajuato, México: Mc Graw Hill.
- Montgomery, D. C. (2004). *Diseño y análisis de experimentos* (Segunda edición ed.). México, D.F, México: Limusa Wiley.
- Sistema de Optimización Aplicada. (s.f.). *Problem instances*. Recuperado el 15 de Marzo de 2018, de Sistemas de Optimización Aplicada: <http://soa.iti.es/problem-instances>

11. Anexos

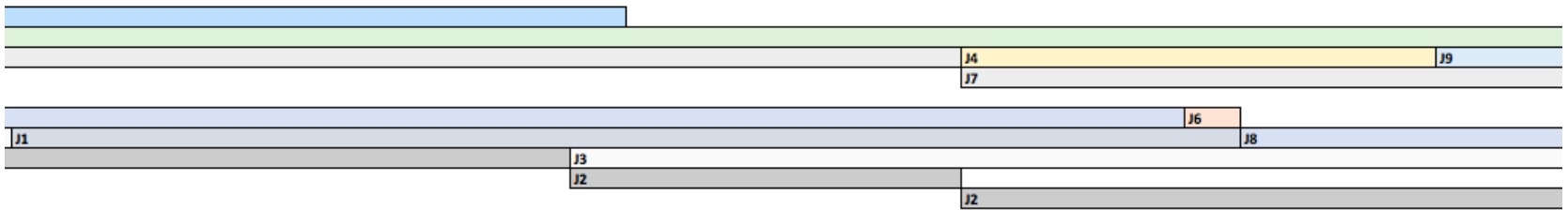
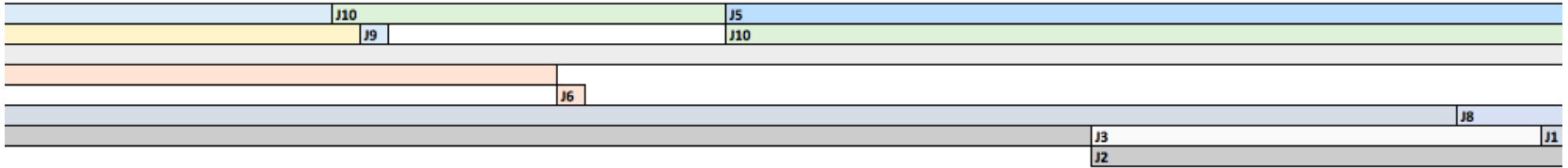
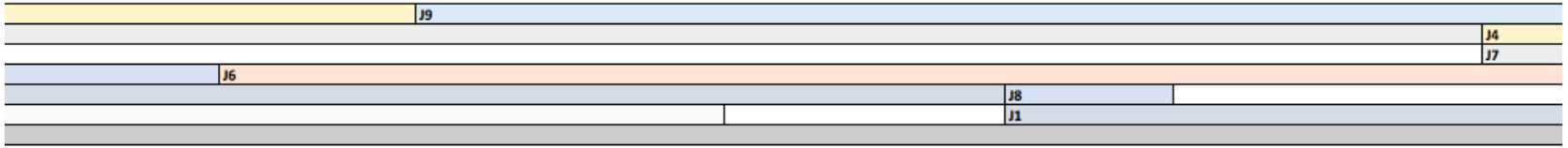
Anexo A. Diagrama Gantt del problema trabajado (4 x 4 en el laboratorio de automatización de la UPB seccional Bucaramanga)

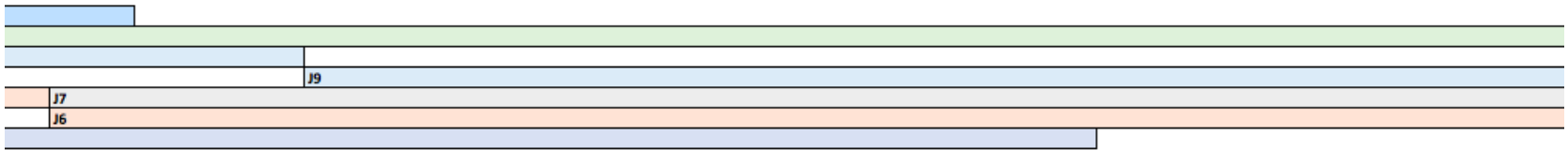
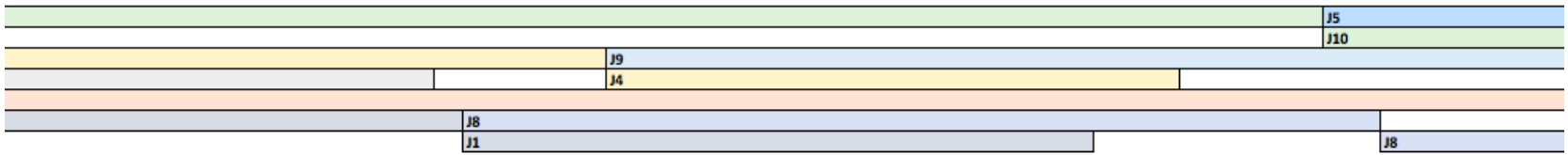
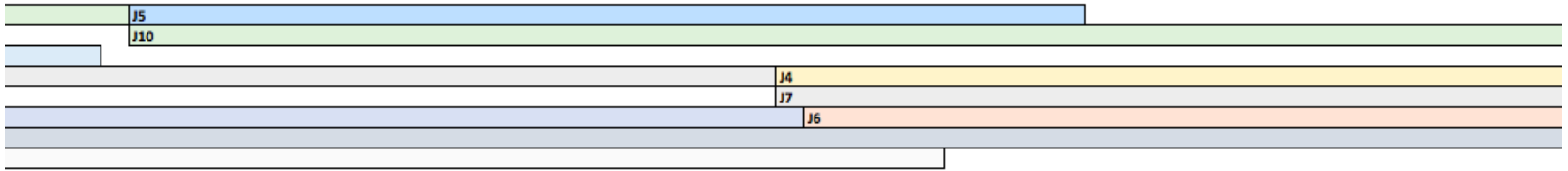
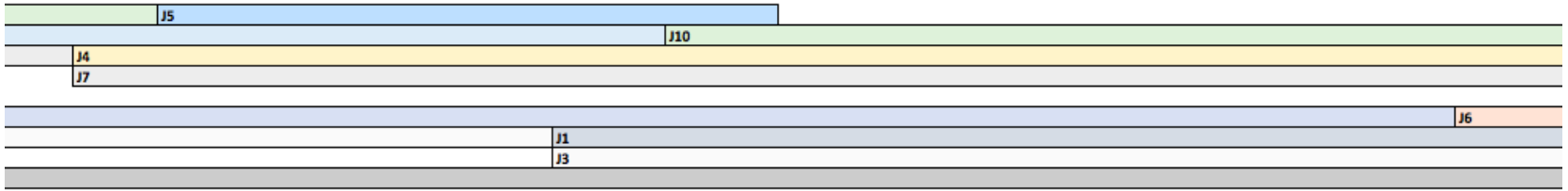


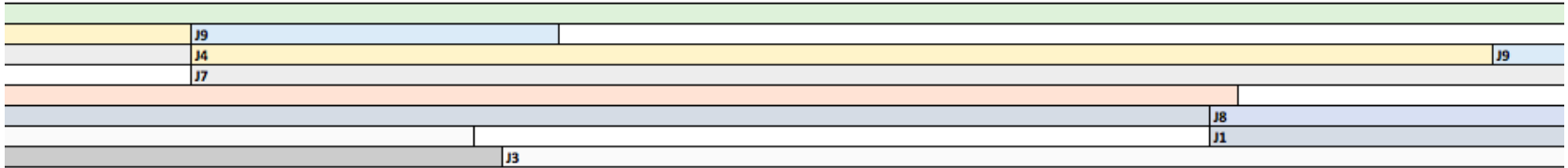
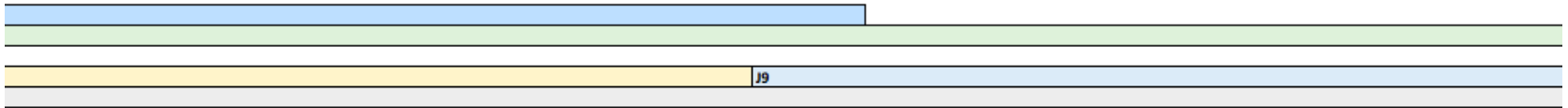
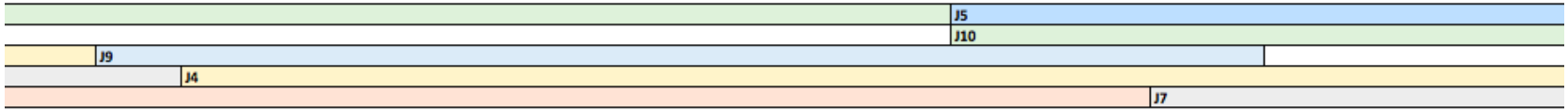
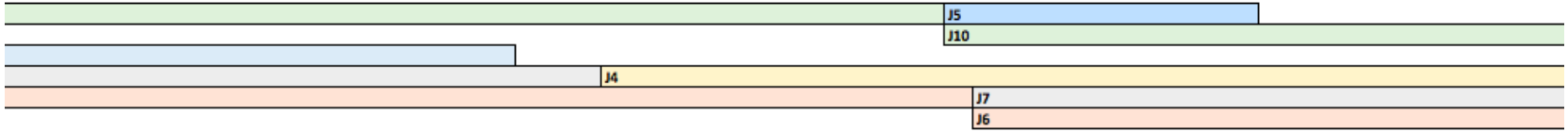
Anexo B. Tiempos de procesamiento y diagrama de Gantt de la solución de la instancia 10_10_1.

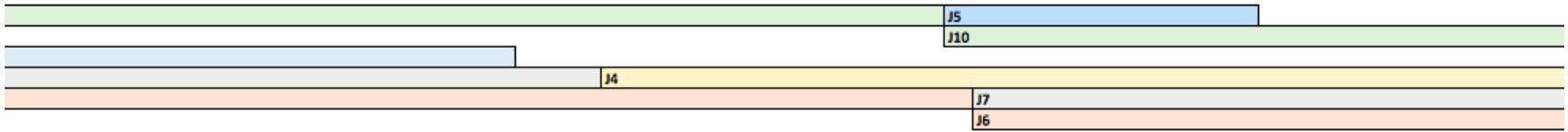
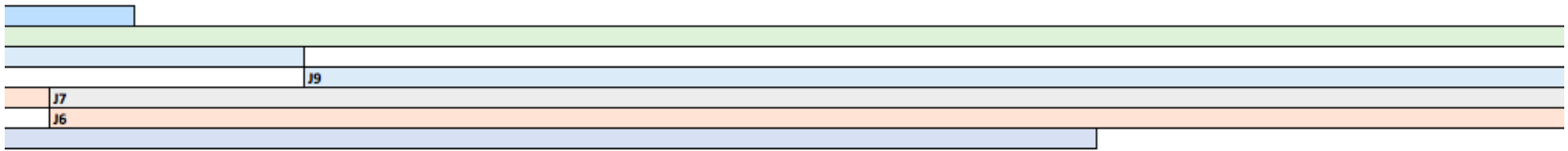
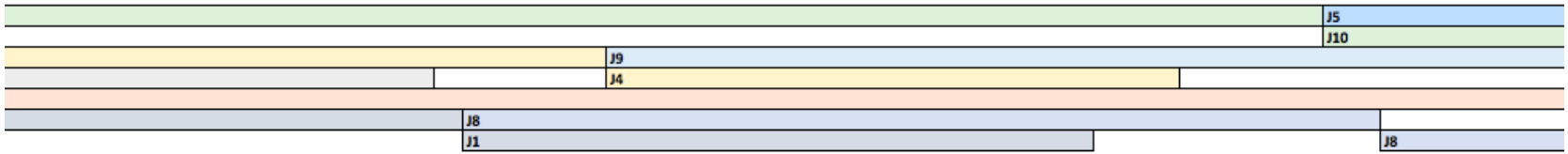
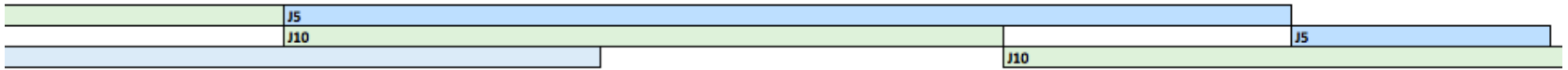
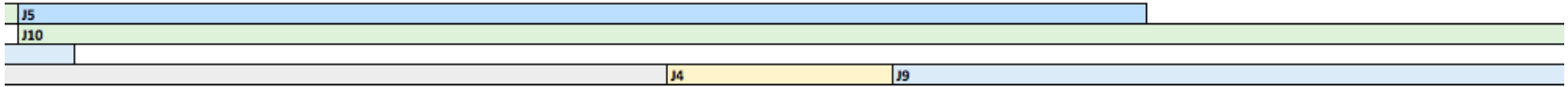
J1

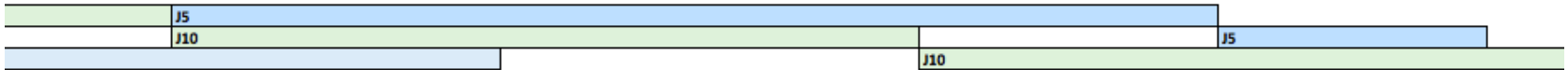
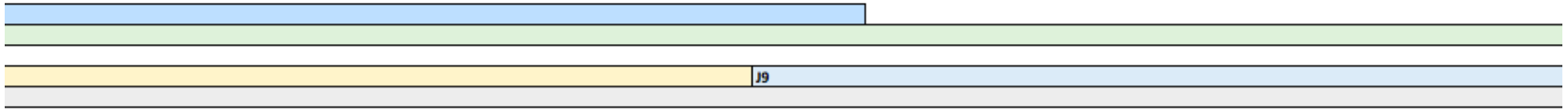
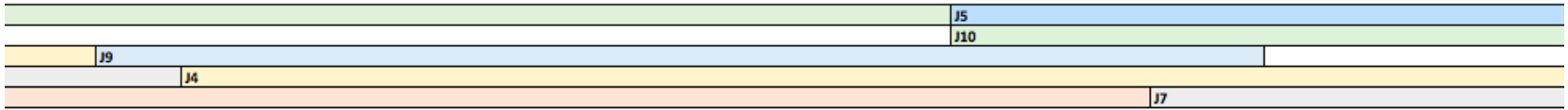
J8	J6
J3	
J2	











Anexo C. Tiempos de procesamiento y diagrama Gantt de la solución de las instancias 10_10_1 , 10_10_2 , 10_15_1 , 10_15_2 y 60_20_1. (Excel).

Anexo D. Programa propuesto en Visual Basic. (Excel)