

**“METAHEURISTICA HONEY-BEES MATTING OPTIMIZATION (HBMO)  
APLICADA A LA SOLUCIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA  
(FLP) DE UN SOLO NIVEL Y DEPARTAMENTOS DE ÁREAS IGUALES O  
DESIGUALES”**

**GINNA MARCELA COTE RAMÍREZ  
DIANA CAROLINA MENDOZA MORENO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES  
BUCARAMANGA**

**2012**

**“METAHEURISTICA HONEY-BEES MATTING OPTIMIZATION (HBMO)  
APLICADA A LA SOLUCIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA  
(FLP) DE UN SOLO NIVEL Y DEPARTAMENTOS DE ÁREAS IGUALES O  
DESIGUALES”**

**GINNA MARCELA COTE RAMÍREZ  
DIANA CAROLINA MENDOZA MORENO**

**TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE  
INGENIERA INDUSTRIAL**

**Director  
CARLOS EDUARDO DIAZ BOHORQUEZ  
Ingeniero Industrial**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES  
BUCARAMANGA**

**2012**

## AGRADECIMIENTOS

- ❖ A la Universidad Industrial de Santander y a la Escuela de Estudios Industriales y empresariales por la formación recibida.
- ❖ Al grupo de Investigación Ópalo porque bajo su nombre se desarrolló este proyecto de investigación.
- ❖ A nuestros profesores por el conocimiento compartido y por la formación impartida, especialmente a nuestro Director de Proyecto, Magister Carlos Eduardo Díaz Bohórquez, por confiar en nosotras, por su tiempo y por su apoyo de principio a fin.
- ❖ A John Alexander Díaz, estudiante de Ingeniería de Sistemas de la Universidad Industrial de Santander, porque el conocimiento que nos compartió fue indispensable en el desarrollo de este proyecto.
- ❖ A nuestros familiares y amigos por ser motivadores en esta importante etapa de nuestras vidas.

## DEDICATORIA

A Dios por ayudarme a superar los obstáculos y terminar con éxito esta etapa de mi vida. A mis padres, por su formación y ejemplo, por crear en mí una actitud de trabajo y persistencia, especialmente a mi madre por su incansable lucha y por estar siempre a mi lado. A Jaime Andrés, por su apoyo incondicional, a mis amigos, con quienes compartí estos años de formación, siempre estarán en mi mente y en mi corazón.

*Diana*

A Dios. A mi familia. A mis padres por su apoyo. A Omar y Cristina por ser más que mis hermanos, mis grandes amigos. A Juan Felipe y a mis amigos que me acompañaron en todo momento, gracias.

*Ginna*

## CONTENIDO

	Pág.
INTRODUCCIÓN.....	20
1. GENERALIDADES.....	23
1.1. TÍTULO.....	23
1.2. MODALIDAD.....	23
1.3. RESPONSABLES.....	23
1.3.1. Autores.....	23
1.3.2. Director del Proyecto.....	23
1.4. ALCANCE.....	24
1.5. OBJETIVOS.....	24
1.5.1. Objetivo General.....	24
1.5.2. Objetivos Específicos.....	24
2. PROBLEMAS DE OPTIMIZACIÓN COMBINATORIA.....	26
2.1. DEFINICIÓN.....	26
2.2. COMPLEJIDAD.....	27
2.3. CLASES DE COMPLEJIDAD.....	28
2.4. COMPLEJIDAD DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA.....	29

3.	PROBLEMA DE DISTRIBUCIÓN DE PLANTA.....	32
3.1.	ANTECEDENTES.....	32
4.	CARACTERÍSTICAS DE LA PLANTA.....	35
4.1.	TIPOS DE DISTRIBUCIÓN DE PLANTA.....	36
4.1.1.	Distribución por proceso o por funciones.....	36
4.1.2.	Distribución por producto.....	36
4.1.3.	Distribución por celdas.....	37
4.1.4.	Distribución por proyecto.....	37
4.2.	FLUJO DEL PROCESO.....	37
4.3.	CARACTERÍSTICAS ESPACIALES DE LOS DEPARTAMENTOS.....	40
4.3.1.	Características de área.....	40
4.3.2.	Características de forma.....	42
5.	FORMULACIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA.....	44
5.1.	FORMULACIÓN DISCRETA.....	44
5.2.	FORMULACIÓN CONTINUA.....	46
6.	MÉTODOS DE SOLUCIÓN HEURISTICOS.....	49
6.1.	HEURÍSTICA.....	49

6.2.	METAHEURÍSTICA.....	50
6.2.1.	Clasificación de las metaheurísticas.....	51
7.	METAHEURÍSTICA HBMO.....	53
7.1.	ESTRUCTURA DE LA COLONIA.....	54
7.2.	EL VUELO DE APAREAMIENTO.....	54
7.3.	MODELO ANÁLOGO.....	55
7.4.	ALGORITMO HBMO PROPUESTO PARA EL FLP.....	59
7.4.1.	Parámetros utilizados en el algoritmo.....	60
7.4.2.	Obtención de la población inicial.....	61
7.4.3.	Selección del material genético del zángano.....	61
7.4.4.	Genotipo de los zánganos.....	62
7.4.5.	Cruce de material genético y obtención de las crías.....	63
7.4.6.	Mejora de las crías utilizando las trabajadoras.....	63
7.4.7.	Actualización de la población.....	73
7.4.8.	Condiciones de parada.....	73
8.	HERRAMIENTA DE APLICACIÓN.....	75
8.1.	INSTALACIÓN DE MATLAB.....	75
8.2.	ABRIR LA HERRAMIENTA DE APLICACIÓN.....	75

8.3.	PROBLEMA DE ÁREAS IGUALES.....	78
8.3.1.	Ingresar datos de características de la planta.....	79
8.3.2.	Ingresar parámetros del algoritmo HBMO.....	84
8.3.3.	Obtención de la solución final.....	85
8.4.	PROBLEMA DE ÁREAS DESIGUALES.....	86
8.4.1.	Ingresar datos de características de la planta.....	86
8.4.2.	Ingresar parámetros del algoritmo HBMO.....	88
8.4.3.	Obtención de la solución final.....	88
8.5.	GUARDAR DATOS EN EL EQUIPO.....	89
9.	ANÁLISIS ESTADÍSTICO.....	91
9.1.	DESEMPEÑO DEL ALGORITMO.....	92
9.1.1.	Problemas de 12 Departamentos.....	92
9.1.2.	Problemas de 16 Departamentos.....	92
9.1.3.	Problemas de 20 Departamentos.....	93
9.1.4.	Problemas de 25 Departamentos.....	93
9.2.	ANÁLISIS DEL EFECTO DE LOS FACTORES SOBRE LA VARIABLE DE SALIDA.....	94
9.3.	ANÁLISIS FACTOR PROBLEMA.....	97
9.4.	TIEMPO COMPUTACIONAL.....	100

10. CONCLUSIONES.....	105
11. RECOMENDACIONES.....	107
12. LÍNEAS FUTURAS DE INVESTIGACIÓN.....	108
BIBLIOGRAFÍA.....	109
ANEXOS.....	114

## LISTA DE ILUSTRACIONES

	<b>Pág.</b>
Ilustración 1. Relación entre los problemas P, NP, NP – Completo y NP-duros....	29
Ilustración 2. Distribución lineal simple.....	38
Ilustración 3. Distribución multilineal.....	38
Ilustración 4. Distribución circular cerrada.....	38
Ilustración 5. Distribución de campo abierto.....	39
Ilustración 6. Distribución vertical.....	39
Ilustración 7. Retroceso y <i>bypass</i> .....	40
Ilustración 8. Ejemplo líneas de llenado.....	42
Ilustración 9. Figuras regulares e irregulares.....	42
Ilustración 10. Estructura de una colonia.....	55
Ilustración 11. Representación general del algoritmo HBMO.....	57
Ilustración 12. Ejemplo del genotipo de un zángano.....	60
Ilustración 13. Genotipo de los zánganos.....	62
Ilustración 14. Genotipo marcado.....	63
Ilustración 15. Genotipo de la reina.....	63
Ilustración 16. Cruce de material genético.....	63

Ilustración 17. Genotipo de la cría.....	63
Ilustración 18. Óptimo global y óptimos locales.....	65
Ilustración 19. Vista ventana “Interfaz de usuario MATLAB.....	76
Ilustración 20. Vista ventana “Carpeta ALGORITMO HMBO”.....	76
Ilustración 21. Vista ventana “Abrir el interfaz gráfico de la herramienta”.....	77
Ilustración 22. Vista “Entorno para abrir interfaz gráfico de la herramienta”.....	78
Ilustración 23. Vista “Entorno gráfico algoritmo HBMO”.....	78
Ilustración 24. Vista “Entorno gráfico características de la planta”.....	79
Ilustración 25. Vista “Entorno gráfico características de la planta”.....	80
Ilustración 26. Vista ventana “Cargar matriz de distancias”.....	81
Ilustración 27. Vista ventana “Matriz de distancias”.....	81
Ilustración 28. Vista ventana “Guardar matriz de distancias”.....	82
Ilustración 29. Vista ventana “Generar distribución actual de la planta”.....	83
Ilustración 30. Vista ventana “Guardar distribución actual de la planta”.....	83
Ilustración 31. Vista ventana “Confirmar datos de características de la planta”.....	84
Ilustración 32. Vista ventana “Ingresar parámetros del algoritmo HBMO”.....	85
Ilustración 33. Vista ventana “Solución final, problema de áreas iguales”.....	85
Ilustración 34. Vista “Entorno gráfico algoritmo HBMO”.....	86
Ilustración 35. Vista ventana “Ingresar datos de unidades de área por departamento”.....	87
Ilustración 36. Vista ventana “Línea de llenado”.....	88

Ilustración 37. Vista ventana “Solución final, problema de áreas desiguales”.....	89
Ilustración 38. Vista ventana “Guardar datos en el equipo”.....	90
Ilustración 39. Vista ventana “Asignar nombre a archivo para guardar en el equipo”.....	90
Ilustración 40. Gráfico de perfil –DesvOpt Vs. Número de vuelos de apareamiento (25 Departamentos).....	98
Ilustración 41. Gráfico de perfil – DesvOpt Vs. Número de crías (25 Departamentos).....	98
Ilustración 42. Tiempo computacional Vs. Número de vuelos de apareamiento (12 Departamentos).....	100
Ilustración 43. Tiempo computacional Vs. Número crías (12 Departamentos).....	101
Ilustración 44. Tiempo computacional Vs. Número de vuelos de apareamiento (16 Departamentos).....	101
Ilustración 45. Tiempo computacional Vs. Número crías (16 Departamentos).....	102
Ilustración 46. Tiempo computacional Vs. Número de vuelos de apareamiento (20 Departamentos).....	102
Ilustración 47. Tiempo computacional Vs. Número crías (20 Departamentos).....	103
Ilustración 48. Tiempo computacional Vs. Número de vuelos de apareamiento (25 Departamentos).....	103
Ilustración 49. Tiempo computacional Vs. Número crías (25 Departamentos).....	104

## LISTA DE TABLAS

	<b>Pág.</b>
Tabla 1. Tabla de cumplimiento de Objetivos.....	22
Tabla 2. Complejidad del problema.....	30
Tabla 3. Parámetros definidos para la implementación del algoritmo HBMO.....	61
Tabla 4. Ejemplo de vecindad de una solución.....	67
Tabla 5. Solución óptima de cada problema.....	91
Tabla 6. Desempeño del algoritmo – 12 Departamentos.....	92
Tabla 7. Desempeño del algoritmo – 16 Departamentos.....	92
Tabla 8. Desempeño del algoritmo – 20 Departamentos.....	93
Tabla 9. Desempeño del algoritmo – 20 Departamentos.....	93
Tabla 10. Parámetros utilizados para el experimento.....	95
Tabla 11. ANOVA – Problemas de 20 Departamentos.....	96
Tabla 12. ANOVA – Problemas de 25 Departamentos.....	97
Tabla 13. ANOVA – Factor Problema Vs. Número de vuelos de apareamiento (25 Departamentos).....	99
Tabla 14. ANOVA – Factor Problema Vs. Número de crías (25 Departamentos)..	99

## LISTA DE ANEXOS

ANEXO A. CÓDIGO DEL ALGORITMO HBMO.....	114
ANEXO B. PROBLEMAS DE ESTUDIO.....	118
ANEXO C. DISEÑO DE EXPERIMENTOS.....	132
ANEXO D. ARTÍCULO ACADÉMICO.....	141

## RESUMEN

**TITULO:** Metaheurísticas Honey-Bees Matting Optimization (HBMO) aplicada a la solución del Problema de Distribución de Planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales.<sup>a</sup>

**AUTORES:** Ginna Marcela Cote Ramírez y Diana Carolina Mendoza Moreno.<sup>b</sup>

**PALABRAS CLAVE:** Distribución de planta (FLP), metaheurísticas, Honey-Bees Matting Optimization (HBMO).

El problema de distribución de plantas, consiste en ubicar de forma óptima los departamentos, en un área disponible. Encontrar una buena distribución de planta ha sido un área de la investigación de operaciones que ha despertado un gran interés de estudio a través del tiempo, debido a que un diseño de planta bien hecho contribuye a reducir tiempos y minimizar costos dentro de la empresa. Sin embargo, la distribución óptima de una planta puede ser interpretada como un problema de optimización combinatoria, lo que hace difícil su tratamiento por métodos convencionales, debido a que a medida que va aumentando el número de departamentos y de restricciones el problema se hace imposible de solucionar por métodos exactos y se hace exponencialmente más complejo. Este trabajo trata el problema distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales.

El enfoque utilizado para dar solución al problema es mediante la metaheurística Honey Bees Matting Optimization (HBMO), basada en el comportamiento y las características que presenta una colonia de abejas, donde principalmente se modela el proceso de apareamiento de las mismas.

A partir de la investigación y documentación del problema, se estructura de acuerdo a las características de la planta, la formulación y modelación matemática, así como el diseño del algoritmo para la solución del mismo. Se desarrolla una herramienta en MATLAB, capaz de solucionar el problema usando el algoritmo HBMO. Se resuelve un conjunto de problemas y posteriormente se examinan los resultados obtenidos mediante un diseño de experimentos con el fin de concluir acerca del desempeño de la herramienta diseñada y de la metaheurística aplicada, respecto a otros métodos utilizados en la literatura.

---

<sup>a</sup> Proyecto de Grado

<sup>b</sup> Escuela de Estudios Industriales y Empresariales. Director: Carlos Eduardo Díaz Bohórquez

## ABSTRACT

**TITLE:** Metaheuristic Honey-Bees Matting Optimization (HBMO) applied to solving the facility layout problem (FLP) with one floor and departments with equal or unequal areas.<sup>a</sup>

**AUTHORS:** Ginna Marcela Cote Ramírez and Diana Carolina Mendoza Moreno.<sup>b</sup>

**KEY WORDS:** Facility layout problem (FLP), metaheuristic, Honey-Bees Matting Optimization (HBMO).

The facility layout problem is to optimally locate the departments in an area available. Finding a good facility layout has been an area of operations research that has aroused great interest in this study over time, because a well layout helps to reduce time and minimize costs within the company. However, the optimal facility layout can be interpreted as a problem of combinatorial optimization, which makes it difficult to treat by conventional methods, because as you increase the number of departments and constraints the problem becomes intractable by exact methods and makes it exponentially more complex. This paper addresses the facility layout problem (FLP) with one floor and departments with equal or unequal areas.

The approach used to solve the problem is by metaheuristic Honey Bees Matting Optimization (HBMO) based on the behavior and characteristics presented by a bee colony, where mainly models the process of mating them.

From the research and documentation of the problem, is structured according to the characteristics of the plant, the formulation and mathematical modelling as well as algorithm design for solving it. It develops a MATLAB tool, able to solving the problem using the algorithm HBMO. It solves a set of problems and then discusses the results by the experimental design in order to conclude about the performance of the tool designed and metaheuristic applied with respect to other methods used in the literature.

---

<sup>a</sup> Proyecto de Grado

<sup>b</sup> Escuela de Estudios Industriales y Empresariales. Director: Carlos Eduardo Díaz Bohórquez

## INTRODUCCIÓN

El Problema de Distribución de Plantas (FLP), consiste en ubicar de forma óptima los departamentos, ya sean oficinas, máquinas o centros de trabajo en un área disponible. Sin embargo, la distribución óptima de una planta puede ser interpretada como un problema de optimización combinatoria, cuyo espacio de soluciones está conformado por todas las distribuciones factibles a realizarse, por lo tanto a medida que aumenta el número de departamentos el problema se hace exponencialmente más complejo.

Encontrar una buena distribución de planta ha sido un área de la investigación de operaciones que ha despertado un gran interés de estudio a través del tiempo, debido a que un diseño de planta bien hecho contribuye a reducir tiempos y minimizar costos dentro de la empresa, es por eso que se debe conocer el sistema productivo, máquinas, capacidad de producción, procesos tanto administrativos como operativos, con el fin de determinar el flujo del proceso, el espacio que necesita cada departamento y la cercanía que debe existir entre estos.

Los métodos de solución exactos para el problema de distribución de planta (FLP), se hacen inadecuados a medida que aumenta la complejidad del problema, es así como muchos investigadores se han concentrado en el estudio de métodos de solución heurísticos y metaheurísticos, con el fin de encontrar buenas soluciones en tiempos computacionales razonables. Aunque muchas veces estos procedimientos no garantizan encontrar la solución óptima, con seguridad un método heurístico bien diseñado puede proporcionar una solución que al menos está cerca de ser óptima.

Este trabajo de investigación se concentra en el estudio de la metaheurística Honey-Bees Matting Optimization (HBMO), la definición de sus parámetros y su

aplicación al problema planteado, mediante el diseño de un algoritmo codificado en MATLAB versión 2009. El software suministra herramientas suficientes para dar solución al problema, cuenta con un lenguaje de programación sencillo y buen desempeño en el procesamiento de datos.

Para conocer el desempeño de la herramienta diseñada y la eficiencia del método de solución elegido, se resuelve un conjunto de problemas históricos y posteriormente se examinan los resultados obtenidos mediante un diseño de experimentos.

**Tabla 1. Tabla de Cumplimiento de Objetivos**

<b>CUMPLIMIENTO DE OBJETIVOS</b>	
<b>OBJETIVOS ESPECÍFICOS</b>	<b>NUMERALES RELACIONADOS</b>
Llevar a cabo la revisión bibliográfica del problema de distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales; los avances y enfoques que se le ha dado al problema.	3,4,5
Analizar la Metaheurística Honey-Bees Matting Optimization (HBMO), comprendiendo el algoritmo por medio del estudio de las aplicaciones que se le ha dado en otras áreas de la ingeniería.	7
Recopilar toda la información encontrada en la revisión bibliográfica, para condensarlo en un documento que defina claramente el problema y la metodología utilizada.	2,3,4,5,6
Definir el algoritmo Honey-Bees Matting Optimization (HBMO) para el problema de distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales.	7
Programar el algoritmo HBMO en MATLAB y realizar los ajustes correspondientes al diseño del programa.	8
Evaluar la efectividad del algoritmo, mediante la solución de problemas históricamente resueltos.	9

## **1. GENERALIDADES**

### **1.1. TÍTULO**

METAHEURISTICA HONEY-BEES MATTING OPTIMIZATION (HBMO) APLICADA A LA SOLUCIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA (FLP) DE UN SOLO NIVEL Y DEPARTAMENTOS DE ÁREAS IGUALES O DESIGUALES.

### **1.2. MODALIDAD**

TRABAJO DE INVESTIGACIÓN

### **1.3. RESPONSABLES**

#### **1.3.1. Autores**

GINNA MARCELA COTE R.  
Estudiante de Ingeniería Industrial  
Código: 2050523

DIANA CAROLINA MENDOZA M.  
Estudiante de Ingeniería Industrial  
Código: 2052018

#### **1.3.2. Director del Proyecto**

CARLOS EDUARDO DIAZ BOHORQUEZ  
Magister en Ingeniería

Docente Escuela de Estudios Industriales y Empresariales

## **1.4. ALCANCE**

Los productos a entregar como resultado de este proyecto son los siguientes:

- Documento en el cual se define el problema, la metaheurística como alternativa de solución, el algoritmo HBMO aplicado al problema, el procedimiento de desarrollo del programa y los resultados obtenidos en la ejecución de la herramienta.
- Programa que permite la solución del FLP mediante la metaheurística HBMO. El algoritmo está programado en lenguaje MATLAB.
- Manual del uso de la herramienta.
- Artículo académico que resume el desarrollo de la investigación y los resultados obtenidos.

## **1.5. OBJETIVOS**

### **1.5.1. Objetivo General**

Evaluar la aplicación de la Metaheurística Honey-Bees Matting Optimization (HBMO) en la solución del problema de distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales.

### **1.5.2. Objetivos Específicos**

- Llevar a cabo la revisión bibliográfica del problema de distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales; los avances y enfoques que se le ha dado al problema.

- Analizar la Metaheurística Honey-Bees Matting Optimization (HBMO), comprender el algoritmo por medio del estudio de las aplicaciones que se la ha dado en otras áreas de la ingeniería.
- Recopilar toda la información encontrada en la revisión bibliográfica, para condensarlo en un documento que defina claramente el problema y la metodología utilizada.
- Definir el algoritmo Honey-Bees Matting Optimization (HBMO) para el problema de distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales.
- Programar el algoritmo HBMO en MATLAB y realizar los ajustes correspondientes al diseño del programa.
- Evaluar la efectividad del algoritmo, mediante la solución de problemas históricamente resueltos.

## 2. PROBLEMAS DE OPTIMIZACIÓN COMBINATORIA

### 2.1. DEFINICIÓN

La optimización es una disciplina fundamental en campos de la ciencia tales como la informática, la inteligencia artificial o la investigación operativa. El concepto de optimización se concibe como el proceso de intentar encontrar la mejor solución posible a un problema de optimización, generalmente en un tiempo limitado.

Duarte, Pantrigo y Gallego [15] definen un problema de optimización como un problema en el que hay varias (en general muchas) posibles soluciones y alguna forma clara de comparación entre ellas, de manera que este existe si y sólo si se dispone un conjunto de soluciones candidatas diferentes que pueden ser comparadas. Desde un punto de vista matemático, un problema de optimización  $P$  se puede formular como una 3-tupla  $P = (f, SS, F)$ , definida como sigue [15]:

$$P = \left\{ \begin{array}{l} \text{Optimizar: } f(x), \text{ Función Objetivo} \\ \text{sujeto a,} \\ x \in F \subset SS \text{ Restricciones} \end{array} \right.$$

donde  $f$  es la función a optimizar (maximizar o minimizar),  $F$  es el conjunto de soluciones factibles y  $SS$  es el espacio de soluciones.

Este tipo de problemas se pueden dividir de forma natural en dos categorías: aquellos en los que la solución está codificada mediante valores reales y aquellos cuyas soluciones están codificadas por valores enteros. Dentro de la segunda categoría se encuentran un tipo particular de problemas denominados problemas de optimización combinatoria. Un problema de optimización combinatoria consiste en encontrar un objeto entre un conjunto finito o al menos contable de

posibilidades. Este objeto suele ser un número natural o conjunto de naturales, una permutación o una estructura de grafo o subgrafo.

Dentro de los problemas de optimización combinatoria más conocidos se encuentran el problema del agente viajero (*en inglés T.S.P., Travelling Salesman Problem*), los problemas de planificación de la producción, el problema de la asignación cuadrática (*en inglés Q.A.P., Quadratic Assignment Problem*) y problemas de secuenciación de tareas.

Los problemas combinatorios presentan como particularidad que siempre existe un algoritmo exacto que permite obtener la solución óptima. Este método consiste en la exploración de forma exhaustiva del conjunto de soluciones. Este algoritmo suele ser extremadamente ineficiente, ya que para la mayoría de los problemas de interés que se pueden encontrar en el ámbito de la optimización combinatoria el tiempo que emplearía en encontrar una solución crece de forma exponencial con el tamaño del problema. En general este tipo de problemas se caracterizan por tener espacios de soluciones con una cardinalidad muy elevada [15].

## **2.2. COMPLEJIDAD**

La complejidad computacional de un algoritmo está relacionada primordialmente con la dificultad del problema a resolver y su tiempo de ejecución que determina la eficiencia del algoritmo; sin embargo, recientemente se está empezando a considerar el tiempo en paralelo y el tamaño del hardware como importantes medidas de complejidad.

El objetivo general de la complejidad es clasificar los problemas de acuerdo a su tratabilidad tomando el o los algoritmos más eficientes para resolverlos.<sup>1</sup>Por

---

<sup>1</sup>RIOJAS CAÑARI, Alicia Cirila. Búsqueda Tabú: conceptos, algoritmo y aplicación al problema de las N-reinas. Monografía (Licenciatura en Investigación Operativa), UNIVERSIDAD MAYOR DE SAN MARCOS, Facultad ciencias Matemáticas. Lima, 2005

ejemplo, se puede afirmar que un problema es tratable si se puede encontrar una solución óptima aún para instancias grandes [28].

### **2.3. CLASES DE COMPLEJIDAD**

Los problemas pueden caracterizarse y formularse matemáticamente, de acuerdo a la dificultad que implica su solución por un computador. Duarte, Patringo y Gallego [15] han definido varias clases de problemas, entre los que se destacan las clases P, NP, NP-completo y NP-complejo.

#### **- Clase P**

Se dice que un problema se puede resolver en un tiempo polinómico cuando el tiempo de ejecución de un algoritmo que lo resuelve se puede relacionar el tamaño de entrada con una fórmula polinómica. Los algoritmos para los que existe un algoritmo polinómico se denominan P. Se considera que los problemas P se pueden resolver en un tiempo de ejecución razonable para la informática actual.

#### **- Clase NP (Non-Deterministic Polynomial Time)**

Se llaman problemas de complejidad NP a los problemas para los que no se ha encontrado un algoritmo polinómico que los resuelva, aún así, dado un valor a ese problema, se puede comprobar en tiempo polinómico si es solución o no.

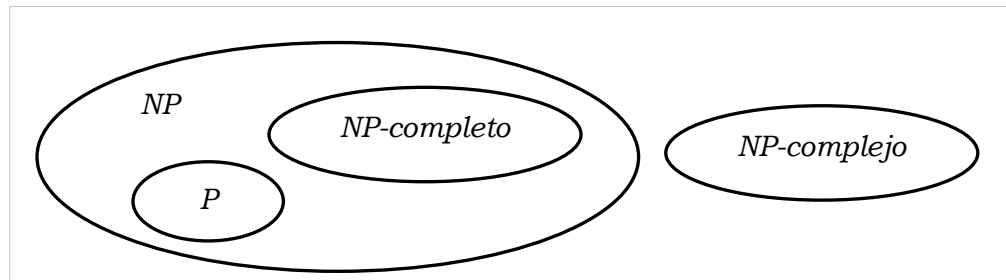
Los problemas P también son problemas NP, ya que siempre es posible comprobar que un valor es solución al problema en tiempo polinómico. Si no se encuentra una forma más sencilla de hacerlo, siempre se puede ejecutar el propio algoritmo polinómico que lo resuelve y comparar el resultado con el valor obtenido.

#### **- Clase NP-completo (Non-Deterministic Polynomial Time Complete)**

Son un tipo de problemas que no tienen un algoritmo en tiempo polinómico que los resuelva. No se ha podido demostrar formalmente que no exista, pero los matemáticos creen que realmente no existe. Este tipo de problemas es un subconjunto de los problemas NP, es decir, existe un algoritmo polinómico que

puede determinar si un valor es una solución a un problema. La relación de esta clase de problemas con las demás se muestra en la *Ilustración 1*.

**Ilustración 1. Relación entre los problemas P, NP, NP-Completo y NP-duros**



*GALLEGO CARRILLO, Micael; DUARTE MUÑOZ, Abraham; PANTRIGO FERNÁNDEZ, Juan José, Metaheurísticas. 1ª ed. Madrid: Editorial Dykinson S.L., 2008*

- **Clase NP-complejo (Non-Deterministic Polynomial Time Hard)**

Los problemas NP-duros no son un subconjunto de los problemas NP. Es decir, para los problemas NP-duros no existe un algoritmo polinómico que permita verificar una solución. Son al menos tan difíciles de resolver como los problemas NP-completos, aunque es posible que sean incluso más difíciles.

**2.4. COMPLEJIDAD DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA**

La disposición eficiente de los Departamentos dentro de una instalación, se define como el “Problema de Distribución de Planta” conocido como FLP por sus siglas en inglés (Facility Layout Problem). Las instalaciones pueden ser plantas manufactureras, edificios administrativos de oficinas o instalaciones de servicios.

Lograr la mejor ubicación para un departamento dentro de un espacio dado, de tal manera que los departamentos no se superpongan, se respeten sus respectivas

restricciones y se minimice el costo de manejo de los materiales, es básico en una distribución de planta. Para las instalaciones manufactureras, el costo de manejo de materiales es una medida muy importante ya que representa un gran porcentaje del costo de operación y del costo total de un producto.

El problema de distribución de planta se puede resolver por métodos exactos como Ramificación y Acotación (*Branch and Bound*). Este algoritmo ha obtenido muy buenos resultados en su aplicación a varios problemas de optimización combinatoria y ha sido considerado como el algoritmo exacto más eficiente al resolver el Problema de Asignación Cuadrática.

Además se encuentran diferentes programas, como *General Algebraic Modeling System* (GAMS), que mediante la aplicación de diferentes métodos dan solución al FLP. Una vez ingresados los datos, el software procede a realizar múltiples iteraciones hasta encontrar y garantizar una solución óptima.

**Tabla 2. Complejidad del problema**

Número de departamentos (n)	Número de soluciones (n!)
1	1! = 1
5	5! = 120
10	10! = 3628800
15	15! = 1.307.674.368.000
20	20! = 2,4329E+18
25	25! = 1,55112E+25
30	30! = 2,65253E+32
40	40! = 8,15915E+47
50	50! = 3,04140E+64

*Fuente: autores*

Sin embargo, estos métodos solo permiten encontrar soluciones óptimas para instancias pequeñas, debido a que el problema de distribución de planta es de

alta complejidad matemática. Por lo tanto si existen  $n$  departamentos y  $n$  lugares, la cantidad de posibles soluciones a evaluar para determinar la solución óptima (bajo algún criterio) es de  $n!$ . Considerando esto Garey y Johnson [15] definieron el FLP como un problema NP-complejo. Debido a que a medida que va aumentando el número de departamentos y de restricciones el problema se hace imposible de solucionar por métodos exactos y se hace exponencialmente más complejo. [Ver Tabla 2]

### **3. PROBLEMA DE DISTRIBUCIÓN DE PLANTA**

El análisis del diseño de planta, así como el diseño del manejo de materiales es muy importante en cualquier planta manufacturera. El correcto análisis del diseño existente, podría mejorar el rendimiento de la línea de producción, disminuyendo los cuellos de botella, minimizando los costos de manejo de materiales, reduciendo el tiempo de inactividad, reflejándose de esta manera en una disminución significativa en los costos de producción.

#### **3.1. ANTECEDENTES**

Un gran número de procedimientos han sido desarrollados para resolver el problema de distribución de planta. Estos procedimientos se pueden clasificar en dos categorías principales: los de construcción y los de mejora. Básicamente, los métodos de construcción implican el desarrollo de un nuevo diseño partiendo de cero y los procedimientos de mejora generan alternativas de trazado sobre la base de un diseño existente.

El problema de distribución de planta (FLP) ha sido estudiado por diferentes investigadores los cuales lo han enfocado acercándose a lo que es una planta real, con sus características y restricciones. A medida que se aumenta el número de restricciones a considerar, aumenta la complejidad del problema.

- En 1957, T. Koopmans y M. Beckmann [22] definieron por primera vez la distribución de planta como un problema de optimización, en donde el objetivo es ubicar las instalaciones dentro de un área dada y minimizar el costo de transporte de material entre las mismas.
- El problema de distribución de planta con áreas desiguales fue formulado originalmente por Armour y Buffa [3] en 1963.

- En 1996, Meller y Gau [25] en 1996 plantearon que las restricciones para la solución de la distribución de planta en áreas desiguales son: (1) todos los departamentos deben estar ubicados en una zona o instalación, (2) que los departamentos no se superpongan entre sí, y (3) el diseño debe cumplir con las restricciones de área de la instalación.
- En 1997, el objetivo del problema de distribución de planta para Heragu [13] se enfoca en encontrar un diseño en donde el hombre, el material y las máquinas puedan trabajar al mismo tiempo de forma adecuada para generar un producto tangible o proveer un servicio de calidad, a bajo costo y con el mínimo uso de material.
- En 1999, Meller, Narayanan y Vance [26] afirman que el problema consiste en encontrar un arreglo ortogonal plano para áreas iguales o desiguales, sin superposición, de  $n$  departamentos rectangulares dada un área igualmente rectangular.
- En 1998, Chwif, Pereira Barretto, y Moscato [10] le aumentan complejidad al problema afirmando que un departamento puede tener dimensiones de longitud y ancho fijos; para este caso, los departamentos se consideran bloques fijos o rígidos. También afirman que un departamento puede estar definido por su área; teniendo en cuenta límites para sus dimensiones, estableciendo así una relación de aspecto entre las mismas.
- En 1998, Kochhar y Heragu [20] introducen al problema la posibilidad de realizar una distribución no sólo de forma horizontal sino también vertical. Ellos argumentan su propuesta en una relación entre el costo de una distribución vertical y el costo de la horizontal, teniendo en cuenta en la segunda la disponibilidad del terreno.
- En 2004, Shayan y Chittilappilly [29] definen el FLP como un problema de optimización en el que no sólo se incluyen dentro del análisis el transporte del

producto, sino también las herramientas necesarias para llevar a cabo dicho transporte.

- En 2007, Meller, Cheny Sherali [24] presentan una nueva formulación para la solución del problema de distribución de planta para áreas desiguales, usando algoritmo genético y MIP (programación entera mixta). Este enfoque permitió solucionar problemas con hasta 11 departamentos.
- En 2009, Komarudin y Kuan Yew Wong [21] proponen el algoritmo de colonia de hormigas (ACO) como alternativa de solución al problema de distribución de planta para áreas desiguales, encontrando buenos resultados.
- En 2010, Duarte y Calderón [11], estudian las Metaheurísticas: Recocido simulado, búsqueda tabú y algoritmo genético, como alternativas de solución al problema de distribución de planta (FLP) de un solo nivel y departamentos iguales ó desiguales.

En la revisión bibliográfica se han encontrado diferentes heurísticas y metaheurísticas, que permiten encontrar soluciones de buena calidad en tiempos computacionales razonables.

La elección de la heurística a utilizar depende de las características del problema planteado, en los siguientes capítulos se exponen las características de la planta y la formulación del problema, para luego describir el método de solución elegido.

#### 4. CARACTERÍSTICAS DE LA PLANTA

El problema de distribución de planta, es la determinación de la disposición más eficiente de los departamentos dentro de una instalación. Departamento se refiere a todo centro de trabajo, oficina, máquina o grupo de máquinas que pertenezcan a la empresa, mientras instalación hace referencia al espacio físico disponible en la planta, en el que será ubicado uno de los departamentos definidos.

Existen diferentes ejemplos de distribución de planta, tales como plantas industriales, edificios administrativos, oficinas, instalaciones de servicios, etc. La disposición eficiente de los recursos (máquinas, departamentos o personas) dentro de una instalación permite que el flujo del proceso sea coordinado y no existan retrocesos en el proceso productivo.

Para plantas manufactureras, el costo de manejo de materiales es la medida más importante para determinar la eficiencia de un diseño. Según Tompkins [31], el costo de manejo de materiales representa del 20 al 50% del costo total de operación y del 15 al 70% del costo total de producción de un producto. El costo de manejo de materiales se determina con base en los flujos de los materiales a través del proceso y las distancias entre los distintos departamentos.

En la actualidad las empresas deben considerar que el diseño de una planta debe estar en capacidad de adaptarse fácilmente a los cambios que exige el entorno, como aumento de la demanda, mejoras en el diseño de un producto, etc. Por lo tanto, se debe prestar atención a la naturaleza dinámica que debe tener la distribución de planta, esto se refiere a que el flujo del proceso varía en diferentes periodos de tiempo, es decir, el diseño varía con el tiempo. Este problema se conoce como el problema de distribución de planta dinámico. Sin embargo cuando la empresa establece que no se van a presentar dichos cambios

o que estos no son significativos, se puede realizar una distribución estática en la cual el diseño se hace para un largo periodo de tiempo.

#### **4.1. TIPOS DE DISTRIBUCIÓN DE PLANTA**

Richard Chase, Robert Jacobs y Nicolás Aquilano (2009) [9] definen los siguientes tipos de distribución de planta que se pueden dar en una empresa: distribución por proceso, distribución por producto, distribución por celdas y distribución por proyecto.

**4.1.1. Distribución por proceso o por funciones.** El formato de centro de trabajo agrupa funciones o equipamientos similares, por ejemplo, todos los tornos en un área y todas las prensas en otra; a continuación, la pieza que se está trabajando avanza en una secuencia preestablecida de operaciones de un área a otra, donde se encuentran las máquinas necesarias para cada operación. Este tipo de distribución es común en los hospitales, donde las áreas están dedicada a tipos particulares de servicios médicos, como las salas de maternidad y las unidades de cuidados intensivos.

El enfoque más común para efectuar la distribución de un centro de trabajo consiste en ordenar centros de trabajo que tienen procesos similares de modo que optimicen su ubicación relativa. Por ejemplo<sup>2</sup>, en una fábrica de juguetes que maneja bajos volúmenes, los centros de trabajo incluirían uno para embarcar y recibir, uno para el moldeo y el troquelado del plástico, uno para las formas de metal, uno de costura y uno de pintura. Las piezas de los juguetes son fabricadas en estos centros de trabajo y, a continuación, se envían a los centros de trabajo de ensamble, donde son armados completamente. En muchas instalaciones, la ubicación óptima con frecuencia significa que los centros de trabajo que tienen un

---

<sup>2</sup> CHASE, R.; JACOBS, R.; y AQUILANO, N. Administración de Operaciones: Producción y Cadena de Suministro. 12 ed. México: McGraw Hill, 2009; p. 221 – 222

volumen grande de movimiento entre departamentos queden colocados unos junto a otros.

**4.1.2. Distribución por producto.** También se conoce como línea de ensamble, consiste en que los procesos están organizados siguiendo paso a paso la manufactura del producto. La ruta de cada parte del producto es, una línea recta. Un ejemplo de este tipo se encuentra en una línea de ensamble de calzado.

**4.1.3. Distribución por celdas.** Agrupa un conjunto de máquinas diferentes, para trabajar en productos distintos, pero que tienen formas y requieren procesos similares. Una celda de manufactura se parece a un centro de trabajo porque las celdas están diseñadas para desempeñar un conjunto específico de procesos y se parece a una línea de ensamble por que las celdas se dedican a una gama limitada de productos.<sup>2</sup>

**4.1.4. Distribución por proyecto.** El producto tiene un gran volumen o peso, razón por la cual está fijo en un área determinada; las máquinas, herramientas y el personal deben ir al producto. Las obras de construcción de edificios, puentes o la fabricación de un avión son ejemplo de este tipo.

## **4.2. FLUJO DEL PROCESO**

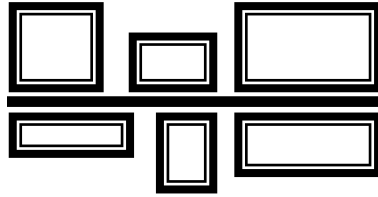
El flujo de proceso garantiza la entrada y salida adecuada del material y producto en proceso entre departamentos. El flujo de las partes del producto está documentado por Amine Drira et ál [2] según sus formas así:

**a. Distribución lineal simple:** el producto se mueve a través de una única línea por las diferentes instalaciones, esta línea puede ser recta (*Ilustración 2*), semicircular o en forma de U.

---

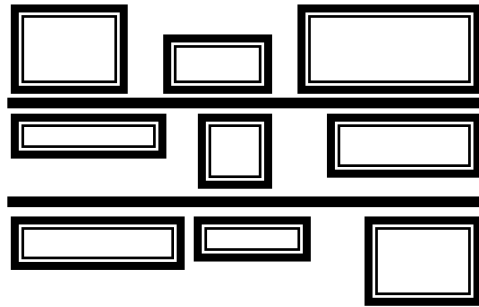
<sup>2</sup> CHASE, JACOBS & AQUILANO, Administración de Operaciones: Producción y Cadena de Suministro. 12 ed. México: McGraw Hill, 2009; p. 221 – 222

**Ilustración 2. Distribución lineal simple**



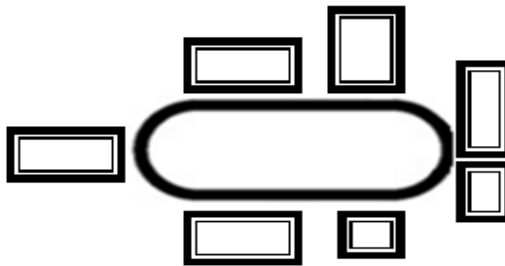
**b. Distribución multilineal:** el desplazamiento de las partes puede ser únicamente por una de las líneas o se pueden intercambiar si el proceso de fabricación así lo requiere (*Ilustración 3*).

**Ilustración 3. Distribución multilineal**



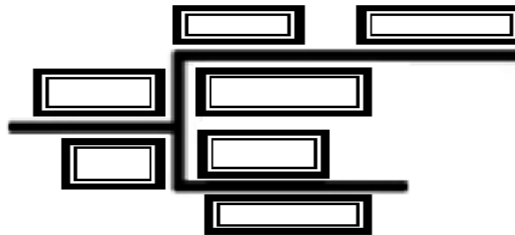
**c. Distribución circular cerrada:** se refiere a la asignación de las instalaciones en una red en anillo cerrado, alrededor del cual las piezas son transportadas en una dirección definida (*Ilustración 4*).

**Ilustración 4. Distribución circular cerrada**



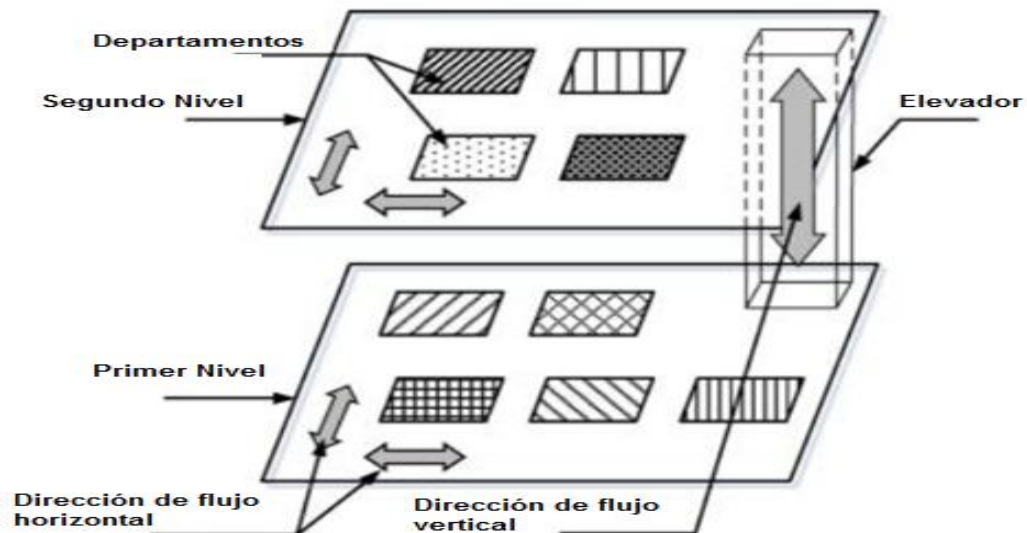
d. **Distribución de campo abierto:** esta distribución se caracteriza por tener un mismo punto de entrada de material, pero a medida que el producto avanza en su proceso de fabricación el producto toma diferentes direcciones creando ramificaciones en el diseño (*Ilustración 5*).

**Ilustración 5. Distribución de campo abierto**



e. **Distribución vertical:** se presenta cuando el área horizontal es un limitante y se hace necesario contar con  $n$  niveles de planta; en este tipo de distribución entran más variables como el número de niveles a construir, el número y ubicación de los elevadores y el tamaño de los mismos (*Ilustración 6*).

**Ilustración 6. Distribución vertical**

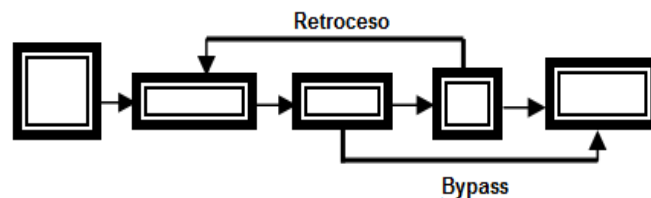


TOMPKINS, J.A., WHITE, J.A., BOZER, Y.A, FRAZELLE, E.H., TANCHOCO, J.M., & TREVINO, J. Facilities planning. En : New York: Wiley. (1996).

En cualquiera de las formas de flujo de material, anteriormente definidas, se pueden presentar dos movimientos muy comunes, conocidos como retroceso y saltos (*bypass*).

El retroceso, consiste en pasar el producto por un proceso ya realizado con anterioridad. Este movimiento puede darse porque la secuencia de fabricación lo requiera o por defecto en el proceso realizado anteriormente. A diferencia del retroceso, el salto (*bypass*) permite al producto evitar determinado proceso en la línea de producción, estos movimientos en lo posible, se deben minimizar ya que agilizaría el flujo del material, mejoraría tiempos y permitiría un transporte uniforme de un proceso o máquina a otro.

**Ilustración 7. Retroceso y Bypass**



### **4.3. CARACTERÍSTICAS ESPACIALES DE LOS DEPARTAMENTOS**

Un departamento está definido físicamente por su área o por su forma, dependiendo del proceso productivo que se llevará a cabo, las máquinas necesarias y el espacio disponible para el manejo y ubicación de materiales que garanticen el flujo normal del proceso.

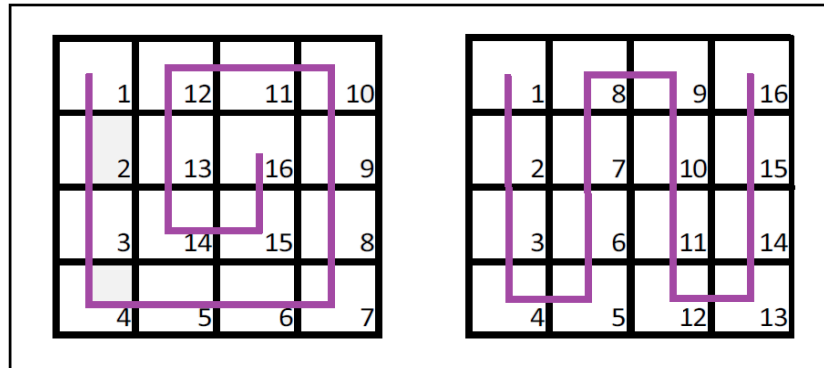
**4.3.1. Características de área.** Según su área los departamentos se pueden clasificar en departamentos de áreas iguales y departamentos de áreas desiguales. De acuerdo a esta clasificación, el procedimiento para solucionar el FLP se hace de manera diferente para cada una de estas. Para áreas iguales el

problema se considera de asignación cuadrática y para áreas desiguales, recibe el nombre de problema del conjunto de cobertura.

De acuerdo con Kyu-Yeul Lee et ál [23], el problema de áreas iguales consiste en ubicar dentro de un conjunto de instalaciones discretas, un conjunto discreto de departamentos, de manera que a cada instalación se le asigne un sólo departamento, también llamado problema de asignación uno a uno o asignación cuadrática. Mientras que el problema de áreas desiguales consiste en asignar todas las instalaciones en un plano o área disponible. En este problema, un departamento es representado por un polígono que puede tomar cualquier forma o número de instalaciones, dependiendo de las unidades de área que requiera para su operación.

Para solucionar el problema de áreas desiguales, se toma la planta a distribuir y se construye sobre ella una cuadrícula formada por cuadros o rectángulos (unidad de área). A partir de este momento, el problema se puede resolver de dos formas diferentes dependiendo del tipo de distribución que se vaya a realizar; un problema de distribución basado en bloques de una cuadrícula o un problema de distribución de bloques continuos. En el primero, a cada departamento se asigna el número de unidades de área que necesite, el objetivo es distribuir estas unidades de área en el espacio total disponible. Para garantizar que las unidades de área correspondientes a cada departamento queden juntas en la distribución, se penaliza la función objetivo. En el segundo, se manejan los mismos departamentos con sus unidades de área correspondientes pero para garantizar que dichas unidades de área queden juntas en la distribución, se maneja una curva de llenado, esta fue desarrollada por Bartholdi y Platzman [5] para el problema del agente viajero y problemas de rutas cortas. Esta curva, consiste en una línea continua que atraviesa cada una de las unidades de área y va definiendo la secuencia en la que van a ser ubicados los departamentos, garantizando que aquellos que estén constituidos por más de una unidad de área, no vayan a ser divididos (*Ilustración 8*).

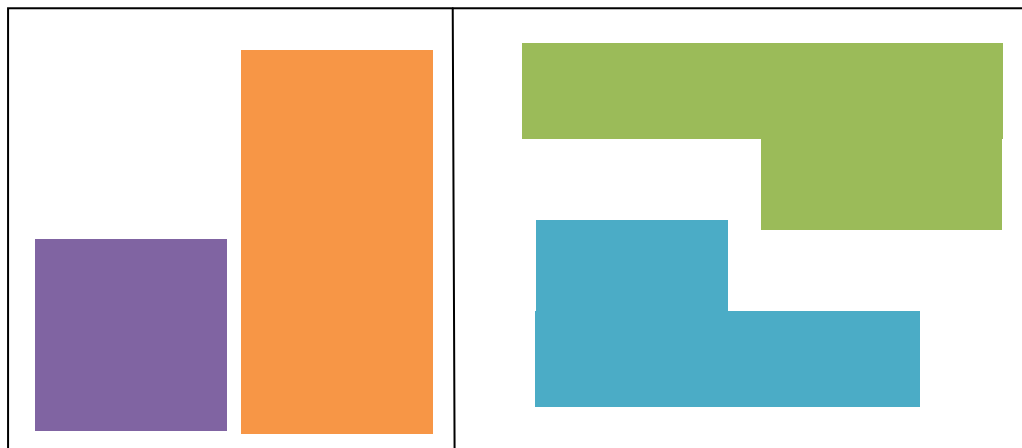
**Ilustración 8. Ejemplo líneas de llenado**



*DUARTE & CALDERON, Alternativas de Solución al Problema de Distribución de Planta (FLP) de un Solo Nivel y Departamentos Iguales ó Desiguales, Por Medio Del Análisis De Métodos Metaheurísticos. 2010*

**4.3.2. Características de forma.** Cuando un departamento se clasifica según su forma, se puede decir que puede ser regular o irregular. Regular, si es de forma cuadrada o rectangular; definida únicamente por dos dimensiones: ancho y largo, de lo contrario, se considerará un departamento de forma irregular (*Ilustración 9*).

**Ilustración 9. Figuras regulares e irregulares**



En los casos en los que se presentan departamentos irregulares Lee y Kim [19] dan a conocer algoritmos que permiten corregir dicha irregularidad construyendo una nueva distribución en bloque usando la información de la posición relativa de los departamentos dados en la distribución. Este método consiste en hacer una partición en bloques al área disponible para distribuir, esta partición se hace únicamente vertical y/o horizontalmente y garantiza que al finalizar el proceso, todos los departamentos tendrán una forma rectangular, inclusive los que al inicio no poseían una como tal. Sin embargo, se pueden presentar casos en los que la distribución final propondrá departamentos demasiado largos o en otro caso demasiado pequeños, en estas situaciones, estas distribuciones serán inaceptables. Estos autores proponen también algoritmos para modificar las formas de los departamentos dada una distribución inicial, de igual forma que el método anterior, el objetivo es pasar de formas irregulares a formas regulares. Algunos de los algoritmos propuestos fueron:

Shaping algorithm using distance between centroids (SA±DC)

Shaping algorithm using beam search (SA±BS)

Shaping algorithms using traversing lines (SA±TL1 and SA±TL2)

## 5. FORMULACIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA

Teniendo definidas las características de la planta, se hace necesario formular matemáticamente el problema de distribución de planta de tal manera que pueda ser resuelto. Esta formulación puede considerar varios tipos de modelos, permitiendo tener en cuenta las relaciones entre los distintos elementos que intervienen en un problema de diseño.

El criterio de decisión se define a través de la función objetivo que puede ser una única o se pueden definir también múltiples funciones para un mismo caso, dependiendo del alcance definido. A su vez el número de restricciones para un problema, estarán determinadas por los objetivos que se definan. Los objetivos más usados o planteados para la solución del FLP son:

- Minimización del costo de manejo y manipulación del material.
- Minimización de los costos de adaptación de las instalaciones.
- Minimización del número de retrocesos y saltos (*bypass*).
- Minimización del tráfico de material.
- Minimización de las formas irregulares de las instalaciones.

De igual forma se definen las restricciones para los objetivos planteados, estas restricciones pueden ser:

- Ubicación de ciertas instalaciones fijas.
- Distancia mínima entre las instalaciones.
- Orientación de los puestos de trabajo.

- Puntos de entrada y salida de material preestablecidos.
- No traslape entre las instalaciones.

### 5.1. FORMULACIÓN DISCRETA

Cuando el diseño es considerado como discreto, el problema de optimización es tratado como un problema de asignación cuadrática. El espacio disponible para la planta se divide en bloques rectangulares con la misma área y forma, y cada bloque se asigna a una instalación. El objetivo, es asignar cada departamento en una instalación de tal forma que la distancia recorrida del material, producto en proceso o producto terminado sea mínima.

Una formulación típica, al determinar la ubicación relativa de las instalaciones a fin de reducir al mínimo el costo de manejo de materiales es el siguiente: <sup>3</sup>

$$\sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \sum_{l=1}^N f_{ik} d_{jl} X_{ij} X_{kl} \quad (1)$$

*Sujeto a:*

$$\sum_{i=1}^N X_{ij} = 1, \quad j = 1, \dots, N \quad (2)$$

$$\sum_{j=1}^N X_{ij} = 1, \quad i = 1, \dots, N \quad (3)$$

Donde  $N$  es el número de departamentos a distribuir y siendo las matrices  $F$  y  $D$  de orden  $N \times N$ ;  $F = (f_{ik})$  y  $D = (d_{jl})$ ,  $f_{ik}$  es el flujo entre el departamento  $i$  al departamento  $k$ ,  $d_{jl}$  la distancia entre la instalación  $j$  y la instalación  $l$  y  $X_{ij}$  es una variable binaria  $\{0,1\}$  para localizar el departamento  $i$  en la instalación  $j$ . La función objetivo (1) representa la suma de las distancias totales recorridas dado que el

<sup>3</sup>BALAKRISHNAN J., CHENG C.H., CONWAY Y LAUD., A hybrid genetic algorithm for the dynamic plant layout problem. International journal of production economics. Vol. 2, No. 86, 2003; p. 107 -120

departamento  $i$  esté ubicado en la instalación  $j$ . En la ecuación (2), está la restricción que a cada instalación sólo se le asigne un departamento y en la ecuación (3) se asegura que cada departamento esté ubicado en una instalación.

La formulación discreta también es usada para minimizar los retrocesos de una distribución en línea continua, la congestión de tráfico del producto y en la formulación de problemas dinámicos. Dentro de las desventajas del uso de esta formulación se encuentra que no se pueden establecer restricciones específicas como la orientación de las instalaciones, puntos de entrada y salida del material y la distancia mínima entre las mismas. En dichos casos se tiende a preferir el uso de la formulación continua.

## 5.2. FORMULACIÓN CONTINUA

El objetivo de esta formulación, es ubicar las instalaciones de acuerdo a las coordenadas del centroide  $(x_i, y_i)$  y sus dimensiones (ancho y largo). A diferencia de la formulación discreta, esta permite establecer restricciones especiales para cada escenario que se presente. La formulación continua es usada para solucionar el problema de áreas desiguales llamado también problema del conjunto de cobertura, dentro de las restricciones que los investigadores han propuesto, se encuentran:

Según Chwif et ál [8] la distancia entre dos instalaciones, está definida como la distancia rectilínea entre los dos centroides de dichas instalaciones así:

$$d_{ij}((x_i, y_i), (x_j, y_j)) = |x_j - x_i| + |y_j - y_i|$$

Kim y Kim [19] definen la distancia entre dos instalaciones como la distancia rectilínea entre los puntos de entrada y salida de las instalaciones así:

$$d_{ij} = |x_i^0 - x_j^1| + |y_i^0 - y_j^1|,$$

en donde,  $(x_i^0, y_i^0)$  define las coordenadas del punto de salida del material de la instalación  $i$ ,  $(x_j^1, y_j^1)$  define las coordenadas del punto de entrada del material de la instalación  $j$ .

Otra restricción muy importante, es que las instalaciones no deben superponerse. Welgama y Gibson [33] proponen un modelo para garantizar que no se presente traslape entre las instalaciones a distribuir. Se determinan dentro del modelo dos condiciones que aseguren la no presencia de superposición, cada condición va dirigida hacia cada dimensión (ancho y largo), finalmente se asegura la ausencia de traslape por medio del área del mismo, siendo que esta sea menor o igual a cero y se cumpla solamente una de las siguientes restricciones:

$$(x_{jt}, x_{ib}), (x_{jb}, x_{it}) \geq 0$$

$$(y_{jt}, y_{ib}), (y_{jb}, y_{it}) \geq 0 ,$$

donde  $(x_{it}, y_{it})$  y  $(x_{ib}, y_{ib})$  son las coordenadas de las esquinas superior izquierda e inferior derecha de la instalación  $i$ ,  $(x_{jt}, y_{jt})$  y  $(x_{jb}, y_{jb})$  son las coordenadas de las esquinas superior izquierda e inferior derecha de la instalación  $j$ .

Mir e Imam [28] consideran que el área de traslape entre dos instalaciones, también debe ser tomada en cuenta como restricciones dentro del modelo, de esta manera, el problema puede ser expresado de la siguiente forma:

Función objetivo sujeta a:

$$A_{ij} \leq 0 ,$$

en donde:

$$A_{ij} = \lambda_{ij} (\Delta X_{ij})(\Delta Y_{ij}),$$

$$\Delta X_{ij} = \lambda_{ij} \left( \frac{L_i + L_j}{2} \right) - |x_i - x_j|,$$

$$\Delta Y_{ij} = \lambda_{ij} \left( \frac{W_i + W_j}{2} \right) - |y_i - y_j|,$$

$$\lambda_{ij} = \begin{cases} -1, & \text{para } \Delta X_{ij} \leq 0 \text{ y } \Delta Y_{ij} \leq 0 \\ +1, & \text{caso contrario} \end{cases},$$

$(L_i, W_i)$  son el largo y el ancho de la instalación  $i$ , y  $(x_i, y_i)$  son las coordenadas de la instalación  $i$ .

Otras restricciones pueden ser consideradas en la formulación para el diseño de una planta. Una formulación apropiada para el problema de distribución de planta con departamentos de áreas desiguales e irregulares puede ser [12]:

$$\text{Minimizar } C = \sum_{i=1}^N \sum_{j=1}^N f_{ij} (|x_j - x_i| + |y_j - y_i|),$$

donde  $N$  es el número de instalaciones,  $f_{ij}$  el costo del flujo de material desde el centroide de la instalación  $i$  hasta el centroide de la instalación  $j$ ,  $(x_i, y_i)$  las coordenadas del centroide de la instalación  $i$  y  $(x_j, y_j)$  las coordenadas del centroide de la instalación  $j$ .

También es necesario al establecer el diseño de una planta, tener en cuenta que el área total disponible no debe ser inferior a la suma del área de todas las instalaciones.

## 6. MÉTODOS DE SOLUCIÓN HEURÍSTICOS

El término heurística proviene del vocablo griego *heurikein* que traduce descubrir o hallar. Se vincula a la supuesta exclamación *eureka* de Arquímedes al descubrir su famoso principio [7].

### 6.1. HEURÍSTICA

Un método heurístico es un procedimiento que trata de descubrir una solución factible muy buena, pero no necesariamente una solución óptima para el problema considerado. No puede darse garantía acerca de la calidad de la solución que se obtiene, pero un método heurístico bien diseñado puede proporcionar una solución que al menos está cerca de ser óptima. El procedimiento también debe ser suficientemente eficiente como para manejar problemas muy grandes. Con frecuencia el procedimiento es un algoritmo iterativo novedoso, donde cada iteración implica la realización de una búsqueda de una nueva solución que puede ser mejor que la encontrada con anterioridad [14].

Para problemas de programación no lineal que parecen ser muy difíciles, un método heurístico común es ejecutar un procedimiento de mejora local. Tal procedimiento comienza con una solución de prueba inicial y después, en cada iteración, busca en la vecindad de la solución de prueba para tratar de encontrar una mejor solución que la actual. Este proceso continúa hasta que no se pueda encontrar una solución mejorada en la vecindad de la solución de prueba actual.<sup>4</sup>

La elección de la heurística a utilizar depende de las características del problema planteado.

---

<sup>4</sup> HILLER & LIEBERMAN, Introducción a la investigación de Operaciones. 8 ed. México: McGraw Hill, 2006; p. 570

## 6.2. METAHEURÍSTICA

El término metaheurística se obtiene de anteponer a heurística el sufijo *meta* que significa “más allá” o “a un nivel superior”. Los conceptos existentes de lo que es una metaheurística se basan en las diferentes interpretaciones de lo que es una forma inteligente de resolver un problema.

Las metaheurísticas son estrategias inteligentes para diseñar o mejorar procedimientos heurísticos muy generales con un alto rendimiento. El término metaheurística se conoció por primera vez por F. Glover [17] en 1986.

Una metaheurística es un tipo de método de solución que organiza de forma inteligente la interacción entre los procedimientos de mejora local y las estrategias de más alto nivel para generar un proceso que esté en capacidad de escapar de un óptimo local y realizar una búsqueda efectiva en un espacio de soluciones factibles.

Para la resolución de un problema mediante la aplicación de una técnica metaheurística, resulta fundamental un balance entre la diversificación y la intensificación, entendiéndose por diversificación la exploración del espacio de búsqueda y por intensificación la explotación de la experiencia acumulada a través del proceso de búsqueda [6].

### **Ventajas**

- Tiende a moverse relativamente rápido hacia soluciones muy buenas, por lo que proporciona una forma muy eficiente de abordar problemas grandes y complicados.
- Gran éxito en la práctica.

### **Desventajas**

- No existe garantía de optimalidad.
- Son aproximados y usualmente no determinísticos (probabilísticos).

**6.2.1. Clasificación de las metaheurísticas.** Las metaheurísticas son estrategias para diseñar procedimientos heurísticos. Por tanto, se puede establecer los tipos de metaheurísticas en función del tipo de procedimientos a los que hace referencia [7]. Algunos de los tipos fundamentales son las metaheurísticas para los métodos de relajación, las metaheurísticas para los procesos constructivos, las metaheurísticas para las búsquedas por entornos y las metaheurísticas para los procedimientos evolutivos.

- **Metaheurísticas de relajación**

Se refieren a procedimientos de resolución de problemas que utilizan relajaciones del modelo original, es decir, modificar o eliminar restricciones del problema real, cuya solución facilita la solución del problema original [7].

- **Metaheurísticas constructivas**

Las heurísticas constructivas aportan soluciones del problema por medio de un procedimiento que incorpora iterativamente elementos a una estructura, inicialmente vacía, que representa a la solución. Las metaheurísticas constructivas establecen estrategias para seleccionar las componentes con las que se construye una buena solución del problema. Dentro de este tipo de metaheurística, se destaca GRASP (*Greedy Randomized Adaptive Search Procedures*), que en la primera de sus dos fases, incorpora a la estrategia “*greedy*” pasos aleatorios con criterios adaptativos para la selección de los elementos a incluir en la solución [7].

- **Metaheurísticas de búsqueda**

Establecen estrategias para recorrer el espacio de soluciones del problema transformando de forma iterativa la solución de partida. Un ejemplo de este tipo es *escalada de montaña* (*en ingles hillclimbing*) que es un procedimiento de búsqueda local. Esta denominación obedece a que basa su estrategia en el estudio de soluciones del vecindario o entorno de la solución que realiza el recorrido. El principal inconveniente de estas búsquedas locales es que se quedan

atrapadas en óptimos locales, sin permitir explorar otro espacio de soluciones. Por ello, el propósito fundamental de las primeras metaheurísticas era extender la búsqueda local para continuarla más allá de los óptimos locales, denominándose búsqueda global. Las metaheurísticas de búsqueda global incorporan pautas para escapar de los óptimos locales de baja calidad. *El recocido simulado* (será definido en la sección 7) es el ejemplo más importante de este tipo. Las metaheurísticas de búsqueda con memoria utilizan información sobre el recorrido realizado para evitar que la búsqueda se concentre en una misma zona del espacio. El ejemplo más representativo es la *búsqueda tabú* que prohíbe temporalmente soluciones muy parecidas a las últimas soluciones del recorrido [7].

#### - **Metaheurísticas evolutivas**

Establecen estrategias para conducir la evolución en el espacio de búsqueda de conjuntos de soluciones (usualmente llamados poblaciones) con la intención de acercarse a la solución óptima con sus elementos. Solo se distinguen de las metaheurísticas de búsqueda en que es un conjunto de soluciones y no una solución el que evoluciona sobre el espacio de búsqueda. El aspecto fundamental de las heurísticas evolutivas consiste en la interacción entre los miembros de la población frente a la búsqueda que se guía por la información de soluciones individuales. Las diferentes metaheurísticas evolutivas se distinguen por la forma en que combinan la información proporcionada por los elementos de la población para hacerla evolucionar mediante la obtención de nuevas soluciones [7]. Las principales metaheurísticas con enfoques evolutivos encontradas son el *algoritmo genético*, *algoritmo de colonia de hormigas* y el *algoritmo HBMO*.

## 7. METAHEURÍSTICA HBMO

La metaheurística HBMO está basada en el comportamiento y las características que presenta una colonia de abejas, donde principalmente se modela el proceso de apareamiento de las mismas. Las abejas son insectos eusociales, las principales características de los insectos eusociales son que estos cooperan en el cuidado de la cría y el nido, que existe solapamiento de generaciones y se da la división del trabajo reproductivo.

Hay dos métodos diferentes que permiten formar una colonia. El primero es la fundación independiente, donde se inicia con una o más abejas reproductoras que se encargan desde la formación del nido, la producción de los huevos y la alimentación de las crías. La primera camada se cría sola hasta que cada cría pueda hacerse cargo de alguna labor en la colonia, posteriormente cada abeja se especializa en una labor, la reina en poner huevos y las trabajadoras en cuidar las crías. El segundo es el enjambre donde se inicia con una o más abejas reproductoras y con un grupo de trabajadoras y cada una se especializa en su labor desde el inicio, la reina se encarga de poner los huevos y las trabajadoras en cuidar las crías.

Las colonias fundadas por una sola reina son colonias llamadas haplometrosis de lo contrario son pleometrosis. Además la colonia puede ser monógama, es decir una colonia conformada por una reina durante su ciclo de vida o de lo contrario es una colonia polígama.

Para el desarrollo de este proyecto, se trabajará inicialmente con un enjambre, es decir se tendrá una sola reina sin crías y un grupo de trabajadoras, dicha colonia evolucionará basándose en el apareamiento de las abejas pasando por diferentes generaciones para finalizar con una colonia eusocial, además esta será haplometrosis y polígama.

## **7.1. ESTRUCTURA DE LA COLONIA**

Una colonia de abejas puede llegar a tener hasta 80.000 individuos, en una colmena hay miles de trabajadoras, cientos de zánganos y una reina, cada uno de estos tiene una labor asignada dentro de la colonia.

La reina es la madre de la colmena, única con capacidad reproductiva y es la encargada de poner los huevos, la reina se alimenta con “jalea real”, una sustancia de color blanco lechoso, que la hace más grande que cualquier otra abeja en la colmena.

Los zánganos son los machos reproductores, es decir, los padres de la colonia, tienen ojos notablemente más grandes que los demás miembros de la colmena, con el fin de identificar a la reina cuando ella está en su vuelo de apareamiento.

Las trabajadoras son exclusivamente abejas hembras, las más numerosas de la colonia, son pequeñas, se encargan del cuidado de la colmena, de recolectar néctar, polen, agua y ciertas resinas vegetales utilizadas en la construcción de la colmena, trabajan sin tregua y son las encargadas del cuidado de las crías.

Las crías provienen de huevos que pueden ser fecundados o no fecundados, aquellas que provienen de huevos fecundados serán futuras reinas o trabajadoras y los que provienen de huevos no fecundados serán zánganos [32].

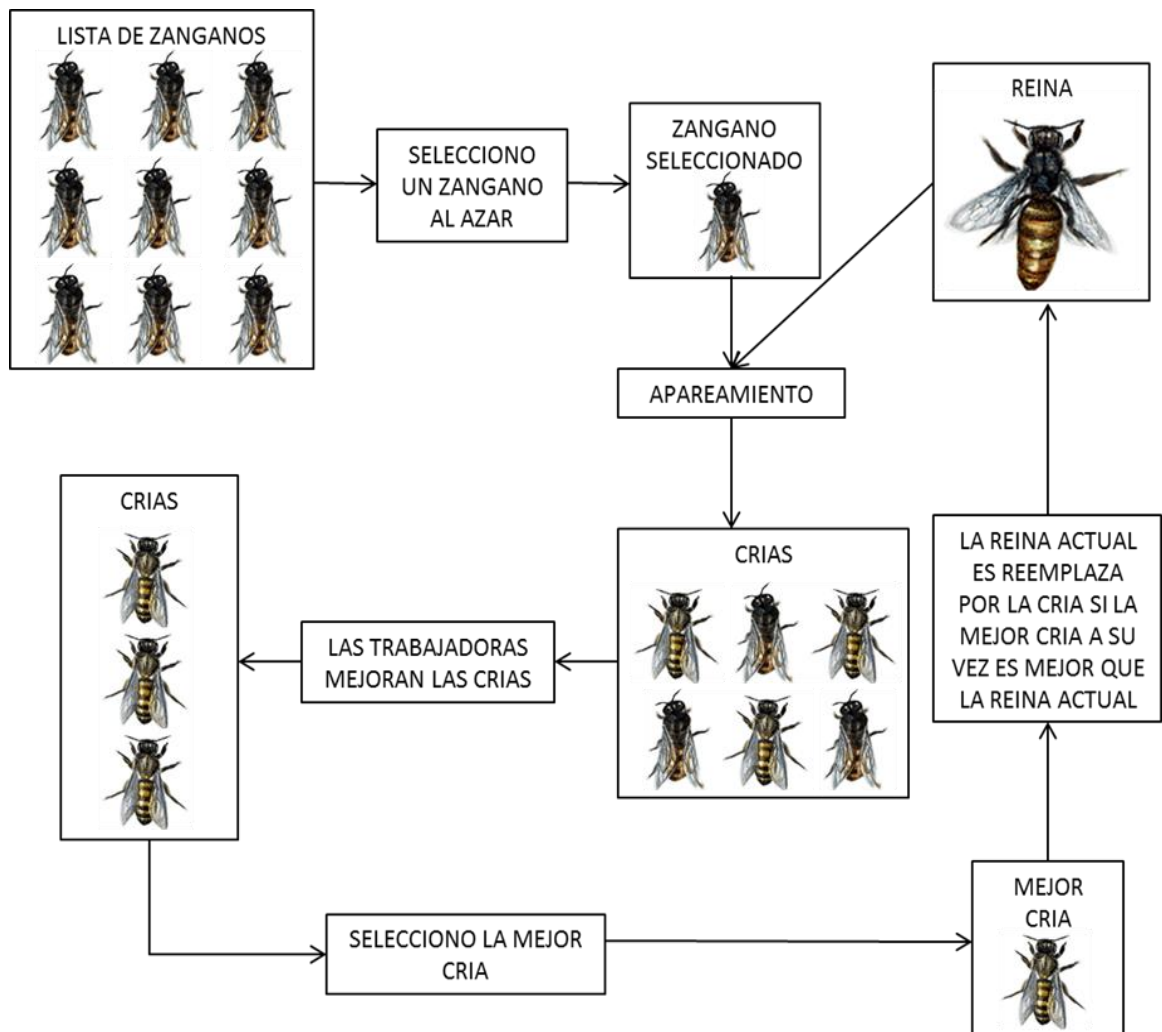
## **7.2. EL VUELO DE APAREAMIENTO**

El proceso de apareamiento comienza con una danza interpretada por la reina, la cual se hace lejos de la colmena, luego durante el vuelo de apareamiento los zánganos siguen a la reina en el aire para lograr aparearse con ella, la reina se apareará con entre 7 a 20 zánganos y almacena de cada uno de ellos su material genético en una bolsa llamada espermateca para así formar la reserva genética de la colonia, en seguida del apareamiento se da la muerte del zángano. La reina

puede poner más de 1.500 huevos por día y esta obtiene una mezcla al azar de material acumulado en la espermateca para fertilizar cada uno, generándose las crías [32].

### 7.3. MODELO ANÁLOGO

**Ilustración 10. Estructura de una colonia**



Los procesos principales del algoritmo HBMO según J. Teo y H.A. Abbas [30] son:

- El vuelo de apareamiento de las reinas con los zánganos para seleccionar a los padres de las futuras crías.
- El cruce de material genético de la(s) abeja(s) reina con los zánganos seleccionados, para generar las crías.
- Mejora de la aptitud (características) de las crías bajo el cuidado de las trabajadoras.
- Adaptación de la aptitud de las trabajadoras, basado en la cantidad de mejoras logradas en crías.
- Sustitución de las reinas débiles por las mejores crías encontradas.

Todos estos procesos, son igualmente relevantes en el algoritmo, por lo cual cada uno va generando la estructura de la solución final que entregará el algoritmo.

Antes de comenzar el proceso de apareamiento, el usuario debe definir el número que corresponde al tamaño de la espermateca de la reina. Este número corresponde al número máximo de apareamientos de la reina en un único vuelo de apareamiento. Cada vez que la reina se aparea exitosamente con un zángano, el genotipo del zángano es almacenado y esta variable se incrementa en 1, hasta que la espermateca está llena. Otros dos parámetros deben ser definidos, el número de reinas y el número de crías que nacerán de las reinas. En la implementación del algoritmo HBMO, el número de reinas es uno, porque en la vida real solo una reina sobrevivirá en una colmena.

Una representación general del algoritmo HBMO original propuesto por Abbas<sup>5</sup>, se presenta en la *ilustración 11*.

---

<sup>5</sup>Abbass H. A, 2001, MBO: Marriage in Honey Bees Optimization A HaplometrosisPolygynous Swarming Approach. Proc. of the IEEE Congress on Evolutionary Computation 2001; Seoul, pp. 207-214.

## Ilustración 11. Representación general del algoritmo HBMO

---

Define  $W$  y  $B$ , número de trabajadoras y crías, respectivamente

Define  $M$ , será el tamaño de la espermateca

Define  $E(t)$  y  $S(t)$  energía y velocidad de la abeja reina, respectivamente

Inicializa cada trabajadora como una única heurística

Genera aleatoriamente el genotipo de la reina

Selecciona aleatoriamente una trabajadora para mejorar el genotipo de la reina

**Mientras** el criterio de parada no se satisfaga

$t = 0$

inicie  $E(t)$  y  $S(t)$

inicie  $g = \frac{0.5 \times E(t)}{M}$

genere un zángano  $D$  aleatoriamente

**mientras**  $E(t) > 0$

evalúe el genotipo del zángano

**si** el zángano pasa la condición de probabilidad, **entonces**

**si** el espermateca de la reina no está llena, **entonces**

adicione su esperma al espermateca de la reina

**fin si**

**fin si**

$E(t + 1) = E(t) - g$

$S(t + 1) = S(t) \times 0.9$

Desmarque la mitad del genotipo del zángano

**fin mientras**

**para**  $cría = 1$  hasta  $B$

Seleccione aleatoriamente un esperma del espermateca de la reina

Genere una cría, cruzando el genotipo de la reina con el esperma seleccionado

Seleccione una trabajadora para mejorar la cría

**fin para**

**mientras** la mejor cría sea mejor que la reina

Reemplace la reina por la mejor cría

Retire la mejor cría de la lista de crías

**fin mientras**

Mate todas las crías

**fin mientras**

Al comienzo del vuelo de apareamiento, la reina inicia con un contenido de energía. Inicialmente tanto la energía como la velocidad de la reina son generadas aleatoriamente. La reina regresa a su nido cuando la energía se acerca a cero o la espermateca está llena.

La probabilidad de que un zángano se aparee con la reina está dada por la siguiente función de recocido [1]:

$$Prob(D) = e^{\left[\frac{-\Delta(f)}{S(t)}\right]},$$

donde  $Prob(D)$  es la probabilidad de colocar el esperma del zángano en la espermateca de la reina, es decir, la probabilidad de un apareamiento exitoso,  $\Delta(f)$  es la diferencia absoluta entre la aptitud del zángano  $D$  y la aptitud de la reina y  $S(t)$  es la velocidad de la reina en un tiempo  $t$ . La probabilidad de apareamiento es alta cuando la reina se encuentra en el comienzo de su vuelo de apareamiento, por lo tanto su velocidad es alta, o cuando la aptitud del zángano es tan buena como la de la reina. Después de cada transición en el espacio, la velocidad y la energía de la reina disminuyen de acuerdo a las siguientes ecuaciones [1]:

$$S(t + 1) = \alpha * S(t) ,$$

$$E(t + 1) = E(t) - g ,$$

donde  $\alpha$  y  $g$ , corresponden a los factores de reducción de la velocidad y la energía de la reina en el instante  $t$  respectivamente,  $\alpha$  es un factor entre 0 y 1. Una serie de vuelos de apareamiento se realizan. Si el apareamiento tiene éxito, el esperma del zángano es almacenado en la espermateca de la reina. Cruzando los genotipos del zángano con los de la reina, se forma una nueva cría (solución de prueba), la cual después puede ser mejorada utilizando trabajadoras, que en la implementación del algoritmo son procedimientos heurísticos que conducen a una búsqueda. Una de las mayores diferencias del HBMO con los algoritmos evolutivos clásicos es que desde que la reina almacena un número de espermatozoides en su espermateca ella puede usar partes del genotipo de

diferentes zánganos para crear una nueva solución lo cual da la posibilidad de tener las crías más aptas.

En la vida real el papel de las trabajadoras es restringido al cuidado de la cría y por esta razón las trabajadoras no son miembros separados de la población, pero ellas son usadas como procedimientos de búsqueda local a fin de mejorar las crías producidas por el vuelo de apareamiento de la reina. Cada una de las trabajadoras, tiene diferentes capacidades y la selección de dos trabajadoras diferentes puede producir diferentes soluciones. Cada una de las trabajadoras tiene la posibilidad de activar o desactivar una serie de diferentes características. Cada una de las crías se elige al azar, una trabajadora alimenta esta con jalea real, esto corresponde a la fase de búsqueda local y finalmente se tiene como resultado la posibilidad de reemplazar la reina si la solución de la cría es mejor que la solución de la reina actual. Si la cría no reemplaza la reina, en el siguiente vuelo de apareamiento de la reina esta cría será uno de los zánganos.<sup>6</sup>

#### **7.4. ALGORITMO HBMO PROPUESTO PARA EL FLP**

El genotipo de la reina, los zánganos y las crías son soluciones al problema que se está desarrollando, el cual está representado por una cadena de genes, donde el número de genes que conforma dicho genotipo es igual al número de departamentos (N) del problema.

Un ejemplo de un genotipo, de un zángano, una cría o la reina para un problema de 12 departamentos se muestra en la *ilustración 12*.

---

<sup>6</sup> Yannis Marinakis, Magdalene Marinaki and Nikolaos Matsatsinis. A Hybrid Clustering Algorithm based on Honey Bees Mating Optimization and Greedy Randomized Adaptive Search Procedure. Lecture Notes in Computer Science. Vol. 5313, (2008); p. 138-152.

### Ilustración 12. Ejemplo, genotipo de un zángano

2	6	3	8	5	10	7	1	12	4	11	9
---	---	---	---	---	----	---	---	----	---	----	---

**7.4.1. Parámetros utilizados en el algoritmo.** Los parámetros generales con los que se trabaja en el algoritmo propuesto, son:

- La cantidad de abejas reina (Q): representa el número de abejas reina que se tendrán al iniciar cada vuelo de apareamiento. En esta implementación del algoritmo se trabajará con una sola reina.
- La cantidad de zánganos (D): representa el número máximo de zánganos con que la reina podrá aparearse en un vuelo. El valor que se ha establecido es 100 zánganos en cada vuelo.
- El tamaño de la espermoteca (M): representa la cantidad de material genético de zánganos, que puede almacenar la abeja reina en cada vuelo. Este valor se fijó en 100, para el experimento realizado.

El parámetro “factor de reducción de energía de la abeja reina”, de determina utilizando la expresión introducida por Teo y Abbas [30], expresada como:

$$g = \frac{0,5 * E(t)}{M},$$

donde  $E(t)$ , corresponde a la energía de la abeja reina en un tiempo  $t$  y  $M$ , al tamaño de la espermoteca respectivamente. La energía inicial y la velocidad inicial de la abeja reina, se inician aleatoriamente tomando valores entre 0,5 y 1. En la *Tabla 3*, se muestra un resumen de los parámetros utilizados en el algoritmo HBMO propuesto en este estudio.

**Tabla 3. Parámetros definidos para la implementación del algoritmo HBMO**

PARAMETRO	VALOR
Cantidad de abejas reina	1
Cantidad de zánganos	100
Velocidad inicial de la abeja reina	Aleatorio entre 0,5 y 1
Energía inicial de la abeja reina	Aleatorio entre 0,5 y 1
Factor de reducción de energía de la reina	$g$
Factor de reducción de velocidad de la reina	0,9
Tamaño de la espermoteca	100
Número de vuelos	3, 6, 9 y 12
Numero de crías	5, 10, 15 y 20

*Fuente: Autores*

**7.4.2. Obtención de la población inicial.** Como ya se definió se tendrá una abeja reina, y se estableció que la solución inicial o abeja reina inicial, se obtendrá aleatoriamente. Luego la abeja reina inicial, es mejorada por medio de la utilización de las trabajadoras, es decir, por medio de procedimientos heurísticos, para así asegurar que la reina represente una buena solución. Para el desarrollo del proyecto se eligieron tres metaheurísticas de búsqueda local: escalada de montaña, búsqueda tabú y recocido simulado.

Siguiendo el modelo inicial propuesto por Abbas [1], la población inicial de zánganos también se obtiene de manera aleatoria.

**7.4.3. Selección del material genético del zángano.** Anteriormente se presentó la función de recocido, para representar la probabilidad de que el material genético de un zángano sea agregado a la espermoteca de abeja reina. Esta probabilidad

debe ser comparada con un número aleatorio generado entre 0 y 1, si la probabilidad de apareamiento es mayor que el número generado, la abeja reina acepta el material genético de dicho zángano, añadiendo su genotipo a la espermoteca, dando como resultado un apareamiento exitoso.

Cuando la aptitud de algún zángano es mejor que la aptitud de la reina, este reemplazará a la reina.

**7.4.4. Genotipo de los zánganos.** Cada zángano posee un genotipo y un operador llamado “marcador de genotipo”, el cual determina el “genotipo marcado” del zángano, cuando el número de genes N (número de departamentos) es par, se marca aleatoriamente la mitad de los genes en el genotipo del zángano, es decir  $N/2$  genes, mientras que cuando el número de genes es impar, se marca  $(N-1)/2$  genes. El “genotipo marcado” del zángano es el utilizado en el cruce con el genotipo de la reina para crear una nueva cría, esto garantiza que solo la mitad de los genes del zángano sean usados, ya que se considera que los zánganos son haploides.

A continuación se muestra el genotipo de un zángano, el operador “marcador de genotipo” y el “genotipo marcado”, este último será el aportado a la cría en el momento del cruce con el genotipo de la reina. La demarcación m, indica que el gen correspondiente ha sido marcado, la demarcación nm indica que el gen no ha sido marcado y el asterisco (\*), representa un gen inexistente.

**Ilustración 13. Genotipo de los zánganos**

<b>Genotipo del zángano</b>	<b>2</b>	<b>6</b>	<b>3</b>	<b>8</b>	<b>5</b>	<b>10</b>	<b>7</b>	<b>1</b>	<b>12</b>	<b>4</b>	<b>11</b>	<b>9</b>
<b>Marcador de genotipo</b>	m	nm	nm	m	nm	m	m	nm	m	nm	nm	m
<b>Genotipo marcado</b>	2	*	*	8	*	10	7	*	12	*	*	9

**7.4.5. Cruce de material genético y obtención de las crías.** Para el cruce entre el “genotipo marcado” del zángano y el genotipo de la abeja reina, se toma el “genotipo marcado” y en las posiciones donde no hay genes existentes, se completa con el gen correspondiente en dicha posición del genotipo de la abeja reina. Luego se revisan cuales genes se repiten a lo largo del genotipo y cuales genes hacen falta en el mismo, para así llevar a cabo la corrección del genotipo de la cría, donde un genotipo que se repite es reemplazado por un genotipo faltante. A continuación se ilustra con un ejemplo:

**Ilustración 14. Genotipo marcado**

<b>“Genotipo marcado”</b>	2	*	*	8	*	10	7	*	12	*	*	9
---------------------------	---	---	---	---	---	----	---	---	----	---	---	---

**Ilustración 15. Genotipo de la reina**

<b>Genotipo de la reina</b>	3	5	1	8	10	4	12	6	2	11	9	7
-----------------------------	---	---	---	---	----	---	----	---	---	----	---	---

**Ilustración 16. Cruce de material genético**

<b>Cruce</b>	2	5	1	8	10	10	7	6	12	11	9	9
--------------	---	---	---	---	----	----	---	---	----	----	---	---

Se repiten los genes 10 y 9, y faltan los genes 3 y 4. Se realiza la corrección y de esta forma se obtiene el genotipo de la cría.

**Ilustración 17. Genotipo de la cría**

<b>Genotipo de la cría</b>	2	5	1	8	10	3	7	6	12	11	9	4
----------------------------	---	---	---	---	----	---	---	---	----	----	---	---

**7.4.6. Mejora de las crías utilizando las trabajadoras.** Se consideraron 3 trabajadoras, las cuales son las encargadas de realizar el mejoramiento tanto de la

abeja reina como de las crías obtenidas después de realizar el cruce de material genético del zángano y de la reina. Las trabajadoras están representadas por las metaheurísticas búsqueda tabú, recocido simulado y escalada de montaña. Las cuales son descritas a continuación:

#### - **BÚSQUEDA TABÚ**

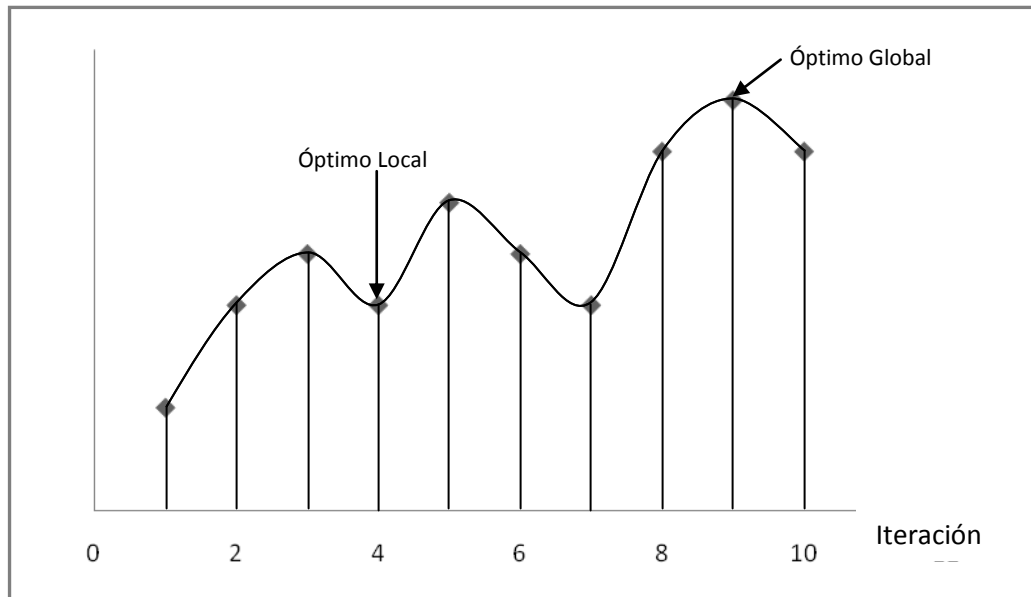
La búsqueda tabú es una metaheurística muy usada que utiliza algunas ideas de sentido común para permitir que el proceso de búsqueda escape de un óptimo local. El procedimiento para búsqueda tabú fue introducido por primera vez por F. Glover en 1986 [17]. Esta metaheurística se caracteriza por el uso de una memoria adaptativa y múltiples estrategias de solución.

Hiller y Lieberman [14] definen que cualquier aplicación de la búsqueda tabú incluye como una subrutina algún procedimiento de búsqueda local que parezca apropiado para el problema bajo consideración. Un procedimiento de búsqueda local opera como un procedimiento de mejora local excepto que no requiere que cada nueva solución de prueba sea mejor que la solución de prueba anterior. El proceso comienza con este método como un procedimiento de mejora local de la manera usual, es decir, al aceptar sólo una solución mejorada en cada iteración para encontrar un óptimo local. Una estrategia de la búsqueda tabú es permitir movimientos sin mejora en la vecindad del óptimo local. Una vez que se alcanza este punto, se aplica de nuevo el procedimiento de mejora local para encontrar un nuevo óptimo local.

Debido a la analogía con la escalada de una montaña, algunas veces este proceso se conoce como enfoque del ascenso más empinado (descenso más suave) porque cada iteración selecciona el movimiento disponible que sube más en la pendiente o, cuando no hay disponible algún movimiento hacia arriba, selecciona el movimiento que baja menos en la pendiente si todo sale bien, el

proceso seguirá un patrón como el que se muestra en la *Ilustración 18*, donde se deja atrás un óptimo local con la intención de escalar hacia el óptimo global.

### Ilustración 18. Óptimo global y óptimos locales



HILLER & LIEBERMAN, Introducción a la investigación de Operaciones. 8 ed. México: McGraw Hill, 2006; p. 566

El peligro de este enfoque es que después de dejar un óptimo local, el proceso se puede entrar en un ciclo repetitivo y regresar al mismo óptimo local. Para evitar este círculo vicioso, la búsqueda tabú prohíbe en forma temporal los movimientos que pudieran regresar el proceso a una solución visitada recientemente. Una lista tabú registra estos movimientos prohibidos, los cuales se conocen como movimientos tabú. La única excepción a la prohibición de un movimiento de este tipo se presenta cuando un movimiento tabú en realidad es mejor que la mejor solución factible que se haya encontrado hasta ese momento [14].

Este uso de memoria para dirigir la búsqueda al utilizar listas que registran cierta parte de la historia reciente es una característica distintiva de la búsqueda tabú. Esta condición tiene raíces en el campo de la inteligencia artificial.

La búsqueda tabú también puede incorporar algunos conceptos avanzados. Uno de ellos es la intensificación, que implica la exploración de una parte de la región factible con más intensidad de la usual después de que se haya identificado como una parte particularmente promisoría para encontrar en ella muy buenas soluciones. Otro concepto es la diversificación, que implica forzar la búsqueda por medio de la entrada en áreas de la región factible que no se han explorado con anterioridad. Para implementar ambos conceptos se utiliza una memoria de largo plazo.

A continuación se presentan algunos detalles importantes que según Hiller y Lieberman, deben trabajarse para satisfacer el tipo específico de problema que se aborde

- **Esquema de un algoritmo de búsqueda tabú básico:**

***Paso inicial.*** Comience con una solución de prueba inicial factible.

***Iteración.*** Utilice un procedimiento de búsqueda local apropiado para definir los movimientos factibles en la vecindad local de la solución de prueba actual. No considere la realización de ningún movimiento incluido en la lista tabú actual a menos que ese movimiento genere una mejor solución que la mejor solución de prueba que se haya encontrado hasta ahora. Determine cuál de los movimientos restantes proporciona la mejor solución. Adopte esta solución como la próxima solución de prueba, sin importar si esta es mejor o peor que la solución de prueba actual. Actualice la lista tabú para evitar el regreso a la última solución de prueba actual. Si la lista tabú ya está llena, elimine el elemento más antiguo de la lista para proporcionar más flexibilidad a los movimientos futuros.

***Regla de parada.*** Utilice algún criterio de parada, como un número fijo de iteraciones, una cantidad fija de tiempo del CPU o un número de iteraciones consecutivas que no produzca ninguna mejoría al mejor valor de la función objetivo, este es un criterio particularmente popular. El proceso también puede

detenerse en cualquier iteración donde no existan movimientos factibles en la vecindad local de la solución de prueba actual. Se acepta la mejor solución de prueba que se haya encontrado en cualquier iteración como la solución final.

- **Procedimiento de búsqueda local.** En cada iteración se elige el mejor vecino inmediato de la solución de prueba actual que no esté descartado por la lista tabú.
- **Estructura de vecindad.** Existen diferentes estrategias para la definición de la vecindad, en este caso se empleará el intercambio en la ubicación de dos departamentos. En la *tabla 3*, se detallan los vecinos de la solución dada. Para este caso, es un problema de cinco departamentos, dicha solución de prueba tiene 10 vecinos.

**Tabla 4. Ejemplo de vecindad de una solución**

1-2-3-4-5	
MOVIMIENTO	VECINO
1-2	2-1-3-4-5
1-3	3-2-1-4-5
1-4	4-2-3-1-5
1-5	5-2-3-4-1
2-3	1-3-2-4-5
2-4	1-4-3-2-5
2-5	1-5-3-4-2
3-4	1-2-4-3-5
3-5	1-2-5-4-3
4-5	1-2-3-5-4

*Fuente: Autores*

- **Forma de los movimientos tabú.** Se enumeran los pares de departamentos que hicieron el intercambio.
- **Adición de un movimiento tabú.** En cada iteración, el par de departamentos que al intercambiarse hace el menor Z entre todos los vecinos de la solución de prueba actual, se agrega como movimiento tabú.
- **Tamaño máximo de la lista tabú.** Siempre que se agregue un movimiento tabú a una lista llena, se elimina el más antiguo de los movimientos tabú que ya estaban en la lista.
- **Criterio de aspiración.** Permite aceptar una solución obtenida por un movimiento clasificado como tabú, siempre que esta sea una mejor solución.
- **Regla de detención.** El proceso se detiene después de realizar un número de iteraciones dado, otra condición puede ser el límite de tiempo del equipo.

## - **RECOCIDO SIMULADO**

El recocido simulado es otra metaheurística muy utilizada, permite al proceso de búsqueda escapar de un óptimo local. Según Hiller y Lieberman, la localización del óptimo global de un problema complicado de maximización es análogo a determinar cuál de todas las montañas existentes es la más alta y después escalar hasta la cumbre de esa montaña. Desafortunadamente, un proceso de búsqueda matemática no tiene el beneficio de una visión aguda que pudiera ayudar a rastrear una cumbre a la distancia. En su lugar, el proceso consiste en algo parecido a escalar en medio de una niebla densa donde la única pista de la dirección que debe tomarse es qué tan pronunciada es la pendiente ascendente o descendente del siguiente paso.

Un enfoque, adoptado por la búsqueda tabú es escalar la pendiente actual en la dirección más empinada hasta alcanzar ese pico particular y después descender lentamente para buscar otra cuesta que escalar. La desventaja es que se emplea

demasiado tiempo (iteraciones) en escalar cada cuesta que se encuentra en lugar de buscar la más alta.

Por otro lado, el enfoque que se utiliza en el recocido simulado es concentrarse de manera principal en la búsqueda de la cuesta más alta. Como el pico más alto puede estar en cualquier sitio dentro de la región factible, se le otorga la mayor importancia a dar pasos en direcciones aleatorias (excepto por el rechazo de algunos, pero no de todos, los pasos que descienden en lugar de ascender) con la intención de explorar tanta región factible como sea posible. Como la mayoría de los pasos aceptados son ascendentes, en forma gradual la búsqueda gravitará hacia aquellas partes de la región factible que contienen las cumbres más altas.

En recocido simulado, la fase de búsqueda se mueve de la solución de prueba actual a un vecino inmediato en la vecindad local de esta solución, similar a búsqueda tabú. Sin embargo, la diferencia con esta radica en cómo se selecciona un vecino candidato a ser la próxima solución de prueba.

Sea:

$Z_c$  = *valor de la función objetivo de la solución de prueba actual.*

$Z_n$  = *valor de la función objetivo del candidato actual a ser la siguiente solución de prueba.*

$T$  = *un parámetro que mide la tendencia a aceptar el candidato actual para ser la próxima solución de prueba si este candidato no es una mejora sobre la solución de prueba actual.*

La regla para seleccionar cuál vecino inmediato será la próxima solución de prueba es la siguiente:

- **Regla de selección del movimiento:** entre todos los vecinos inmediatos de la solución de prueba actual, seleccione uno de manera aleatoria para convertirse en el candidato actual a ser la próxima solución de prueba. Si se supone que el objetivo es la maximización de la función objetivo, acepte o rechace este candidato para ser la próxima solución de prueba como sigue:

*Si  $Z_n \geq Z_c$ , siempre acepte este candidato*

*Si  $Z_n < Z_c$ , acepte el candidato con la siguiente probabilidad:*

$$Prob(\text{aceptación}) = e^x, \text{ donde } x = \frac{Z_n - Z_c}{T}$$

Si el objetivo es la minimización, se deben invertir  $Z_n$  y  $Z_c$  en las fórmulas anteriores. Si el candidato es rechazado repita el proceso con un vecino inmediato de la solución de prueba actual seleccionado de manera aleatoria. Si ya no existen vecinos inmediatos restantes, el algoritmo termina.

Por tanto, si el candidato actual bajo consideración es mejor que la solución de prueba actual, este siempre se acepta como la próxima solución de prueba. Si es peor, la probabilidad de aceptación depende de que tan peor es y del tamaño de  $T$ . La regla de selección del movimiento por lo general aceptará un paso que sólo es un poco descendente, pero será muy difícil que acepte un paso descendente muy pronunciado. Si se inicia con un valor relativamente grande de  $T$ , como lo hace el recocido simulado, la probabilidad de aceptación correspondiente será alta, lo que permite a la búsqueda proceder en direcciones casi aleatorias. Al disminuir de manera gradual el valor de  $T$  a medida que la búsqueda continua, la probabilidad de aceptación disminuye de un modo paulatino, lo cual aumenta la importancia de ascender la mayor parte del tiempo. Así, la elección de los valores de  $T$  a través del tiempo controla el grado de aleatoriedad del proceso para permitir pasos descendentes. Este componente aleatorio, que no está presente en

la búsqueda tabú básica, proporciona más flexibilidad de movimiento hacia otra parte de la región factible con la esperanza de encontrar una cumbre más alta.

El método usual de implantar la regla de selección del movimiento para determinar si se aceptará un paso descendente particular es comparar un número aleatorio entre 0 y 1 con la probabilidad de aceptación. Ese número aleatorio puede considerarse como una observación aleatoria en una distribución uniforme entre 0 y 1.

*Si número aleatorio  $<$  Prob(aceptación), se acepta un paso descendente.*

En otro caso, se rechaza el paso.

El recocido simulado usa esta fórmula particular para la probabilidad de aceptación, porque esta metaheurística se basa en la analogía con un proceso de templado físico. En principio, este proceso implica fundir un metal o cristal a una temperatura alta para después enfriar lentamente la sustancia hasta que alcance un estado estable de energía con propiedades físicas deseables. A una temperatura dada  $T$  durante este proceso, el nivel de energía de los átomos de la sustancia fluctúa pero tiende a disminuir. Un modelo matemático de cómo fluctúa el nivel de energía supone que los cambios ocurren de manera aleatoria excepto que solo se aceptan algunos de los incrementos. En particular, la probabilidad de aceptar un incremento cuando la temperatura es  $T$  tiene la misma forma que para *Prob(aceptación)* en la regla de selección de movimiento de recocido simulado.

La analogía con un problema de optimización en la forma de minimización es que el nivel de la energía de la sustancia en el estado actual del sistema corresponde al valor de la función objetivo con la solución factible actual del problema. El objetivo de que la sustancia alcance un nivel estable con un nivel de energía tan pequeño como sea posible corresponde al problema de llegar a una solución factible con un valor de la función objetivo tan pequeño como sea posible.

Igual que en un proceso físico de templado, una cuestión clave cuando se diseña un algoritmo de recocido simulado de un problema de optimización es seleccionar un programa de temperatura apropiado para ser usado. (Debido a la analogía con el templado físico, en el algoritmo de recocido simulado se hará referencia a  $T$  como la temperatura.) Este programa debe especificar el valor inicial de  $T$ , relativamente grande, así como los valores subsecuentes que se reducen en forma progresiva. También debe precisar cuántos movimientos, es decir, iteraciones, se deben hacer para cada valor de  $T$ . La selección de estos parámetros para ajustarse al problema bajo consideración es un factor clave de la eficacia del algoritmo.

Con esta base, Hiller y Lieberman proponen un esquema del algoritmo de recocido simulado.

- **Esquema de un algoritmo de recocido simulado básico:**

***Paso inicial.*** Comience con una solución de prueba inicial.

***Iteración.*** Utilice la regla de selección del movimiento para elegir la próxima solución de prueba. Si ninguno de los vecinos inmediatos de la solución de prueba actual se acepta, el algoritmo termina.

***Verificación del programa de temperatura.*** Después de realizar el número deseado de iteraciones en el valor actual de  $T$ , disminuye  $T$  hasta el siguiente valor en el programa de temperatura y reinicie las iteraciones en dicho valor.

***Regla de parada.*** El algoritmo termina cuando ya se ha realizado el número de iteraciones deseado en el valor más pequeño de  $T$  en el programa de temperatura, o cuando no se acepta ninguno de los vecinos inmediatos de la solución de prueba actual. Acepte la mejor solución de prueba que encontró en cualquier iteración (incluso de los valores grandes de  $T$ ) como la solución final.

Para recocido simulado, la estructura de vecindad utilizada es la misma de búsqueda tabú.

#### - **ESCALADA DE MONTAÑA**

Es un procedimiento de mejora local, el cual usa una técnica de mejoramiento iterativo donde comienza a partir de un punto en el espacio de soluciones y se dirige siempre a un estado mejor que el actual es decir, se mueve de la solución de prueba actual a un vecino inmediato en la vecindad local de esta solución, que no solamente debe ser mejor que la solución de prueba actual, si no el mejor de todos los vecinos. El criterio de parada es el número de iteraciones definida por el usuario o cuando no se alcanza ninguna mejora.

Encuentra buenas soluciones, pero es más difícil que encuentre un óptimo global, teniendo en cuenta que puede converger a un óptimo local y estancar su búsqueda allí. El óptimo local en el que se detenga dependerá del punto en el que el procedimiento inicie su búsqueda, es decir de la solución inicial. De ahí que el procedimiento convergirá hacia el óptimo global sólo si inicia la búsqueda en la vecindad de este.

**7.4.7. Actualización de la población.** Después de realizado cada vuelo y una vez las crías generadas mediante el cruce de material genético han sido mejoradas a través de la aplicación de las trabajadoras (metaheurísticas), se evalúa la aptitud de cada una de las crías mejoradas. Si la aptitud de alguna de las crías mejorada es mejor que la aptitud de la abeja reina actual ( $Z_{cría} < Z_{reina}$ ), entonces se reemplaza la abeja reina por la cría de mejor aptitud, siendo esta última la nueva reina.

**7.4.8. Condiciones de parada.** El criterio de parada que se estableció, es alcanzar cierto número de iteraciones, el cual es representado por el número de vuelos, que dicho anteriormente variará entre 3, 6, 9 y 12 vuelos. La regla de detención para las trabajadoras utilizadas en el experimento es 10 iteraciones en

búsqueda tabú y 7 el tamaño de la lista tabú; 20 iteraciones en escalada de montaña y en recocido simulado 10 iteraciones, 10 temperaturas, temperatura inicial igual a 100 y factor de reducción de la temperatura igual a 0,9.

## **8. HERRAMIENTA DE APLICACIÓN**

Para el diseño de la herramienta se utilizó el software MATLAB versión 2009. Se codificó el algoritmo correspondiente a la metaheurística HBMO, siguiendo los parámetros estudiados y definidos para el desarrollo del proyecto. La herramienta computacional permite dar solución al problema de distribución de planta de un solo nivel y departamentos de áreas iguales o desiguales. Este capítulo pretende darla a conocer y explicar las condiciones para su uso.

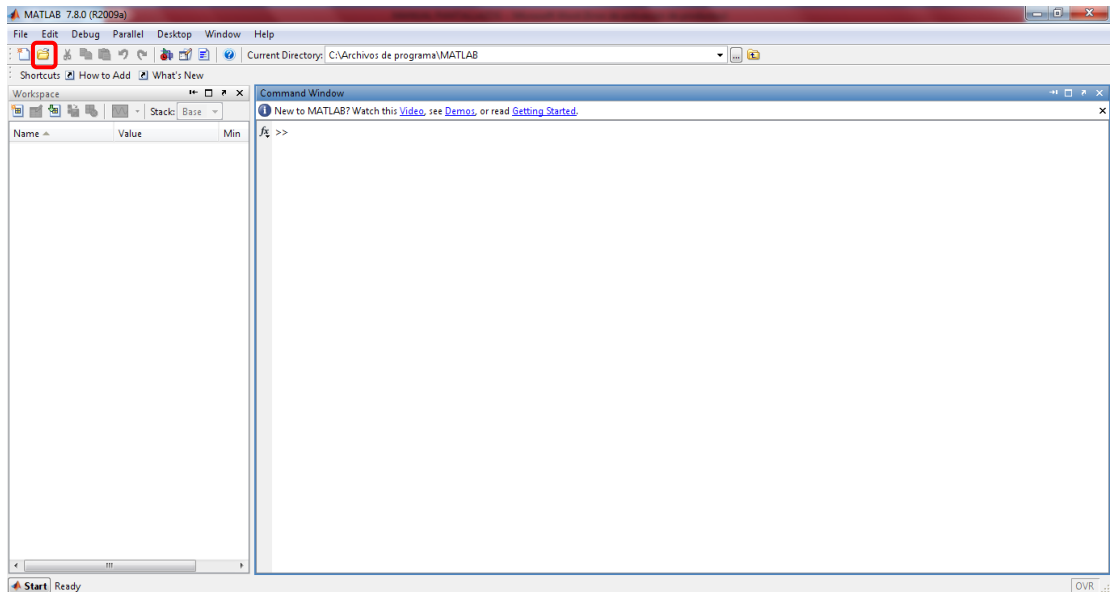
### **8.1. INSTALACIÓN DE MATLAB**

Para iniciar, es necesario realizar la instalación de software MATLAB, versión 2009 o posterior en el equipo que se va a trabajar. Después de realizar la instalación se debe guardar la carpeta “ALGORITMO HBMO” en el equipo, la carpeta contiene todas las funciones que se codificaron para la ejecución de la aplicación.

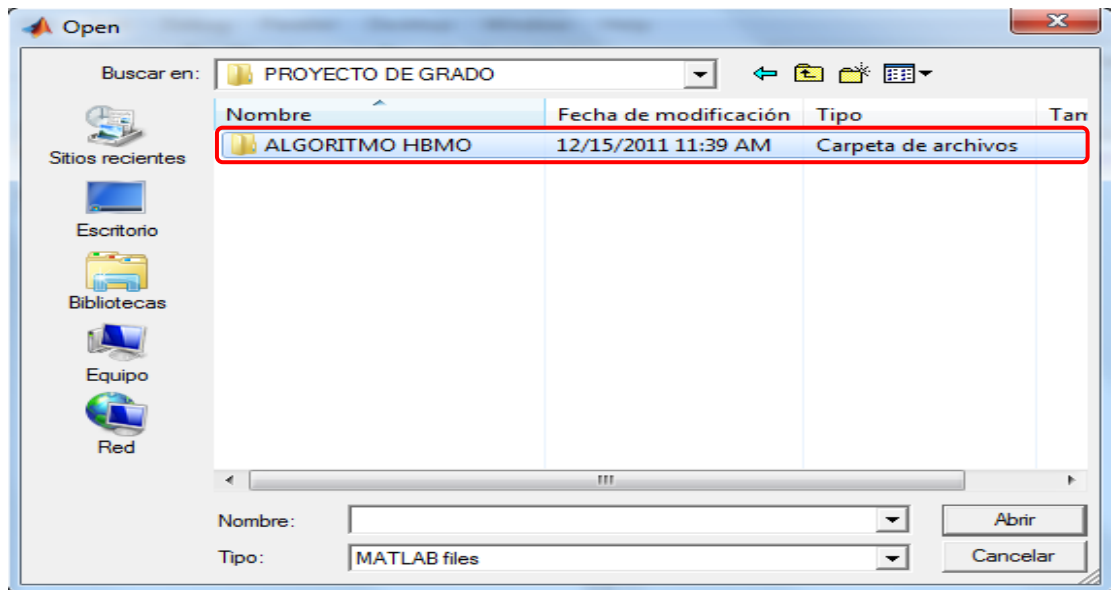
### **8.2. ABRIR LA HERRAMIENTA DE APLICACIÓN**

Al abrir MATLAB en el equipo, se despliega el interfaz de usuario, como se observa en la ilustración 19, en la cual se encuentra el icono “ABRIR”, que corresponde a la carpeta señalada en color rojo, al hacer clic en ella permite al usuario ir a la carpeta “ALGORITMO HBMO” que ha sido guardada con anterioridad, como se evidencia en la *ilustración 20*.

**Ilustración 19: Vista ventana “Interfaz de usuario MATLAB”**

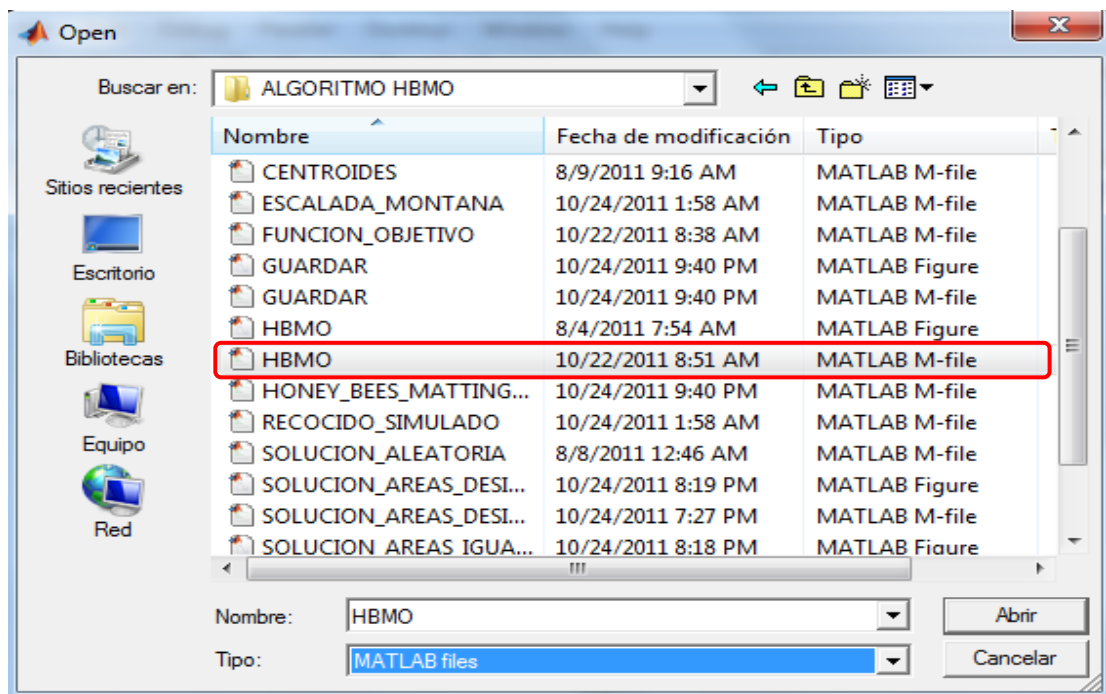


**Ilustración 20: Vista ventana “Carpeta ALGORITMO HMBO”**



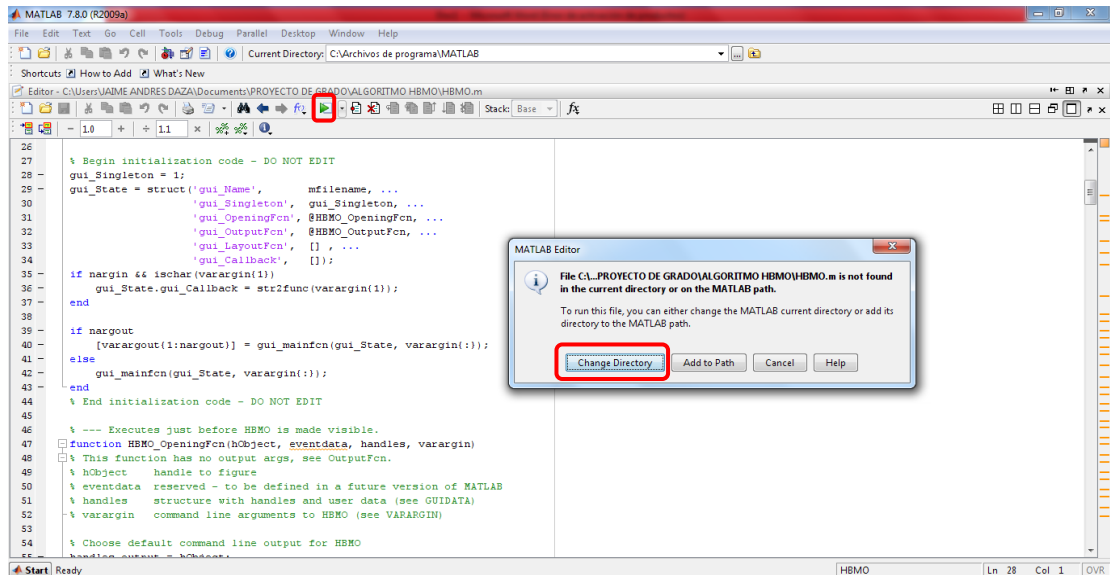
Al hacer clic en la carpeta “ALGORITMO HBMO” se observan las funciones que han sido codificadas para la metaheurística HBMO, así como una carpeta con el nombre “Datos”, en la cual están guardados los problemas de aplicación que se utilizaron en el desarrollo de este trabajo, de la misma manera el usuario podrá guardar en dicha carpeta los problemas de distribución de planta que desee abordar utilizando la herramienta. Para abrir el interfaz gráfico de usuario de la herramienta diseñada, se hace clic en la carpeta “HBMO”, del tipo “MATLAB files” (Ilustración 21).

**Ilustración 21: Vista ventana “Abrir el interfaz gráfico de la herramienta”**



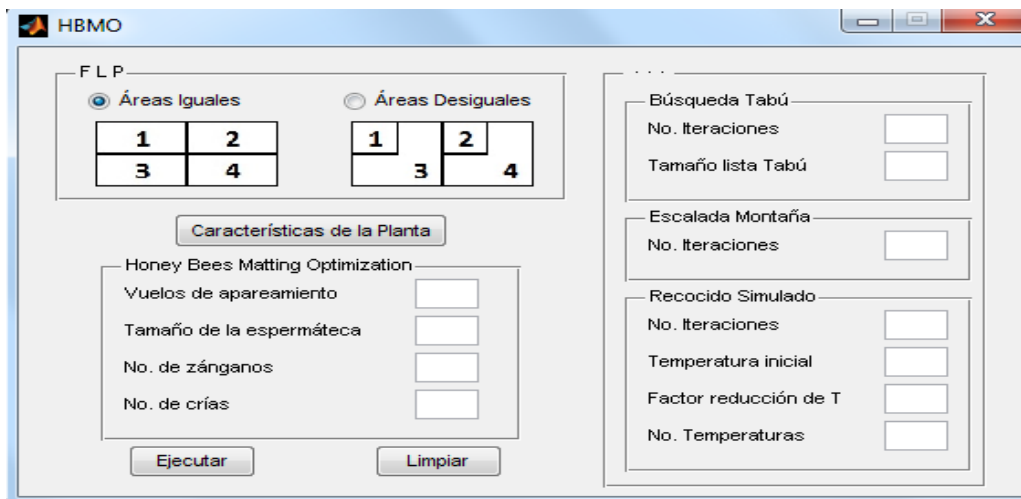
Al hacer clic en la carpeta se despliega el código general del algoritmo HBMO, se hace clic en el icono “Ejecutar” y en seguida se despliega una ventana más pequeña como se evidencia en la ilustración 22, al hacer clic en el icono “Change Directory”, se abre el entorno gráfico de la herramienta diseñada (Ilustración 23).

**Ilustración 22. Vista “Entorno para abrir interfaz gráfica de la herramienta”**



### 8.3. PROBLEMA DE ÁREAS IGUALES

**Ilustración 23. Vista “Entorno gráfico algoritmo HBMO”**

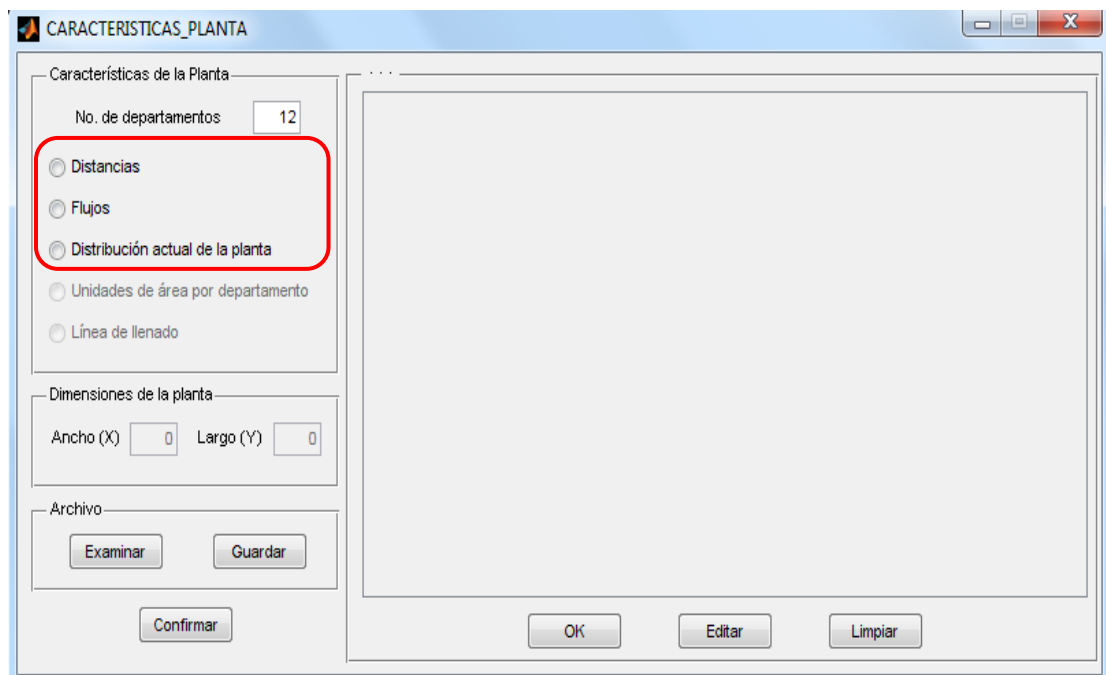


Para iniciar es necesario elegir si la planta se va a distribuir en departamentos de áreas iguales o desiguales, luego se hace clic en el icono “Características de la Planta”, para ingresar los datos de características de la planta (*Ilustración 23*).

**8.3.1. Ingresar datos de características de la planta.** Para el problema de áreas iguales se requiere el número de departamentos, la matriz de distancias, la matriz de flujos y la distribución actual de la planta.

- **Número de departamentos.** Se refiere al número de departamentos, ya sean oficinas, grupo de máquinas o centros de trabajo, entre los cuales se va a distribuir el área total disponible para la planta. Para el ejemplo se abordará un problema de 12 departamentos. Al digitar el número de departamentos se activan los comandos que aplican para un problema de áreas iguales (*Ilustración 24*).

**Ilustración 24. Vista “Entorno gráfico características de la planta”**

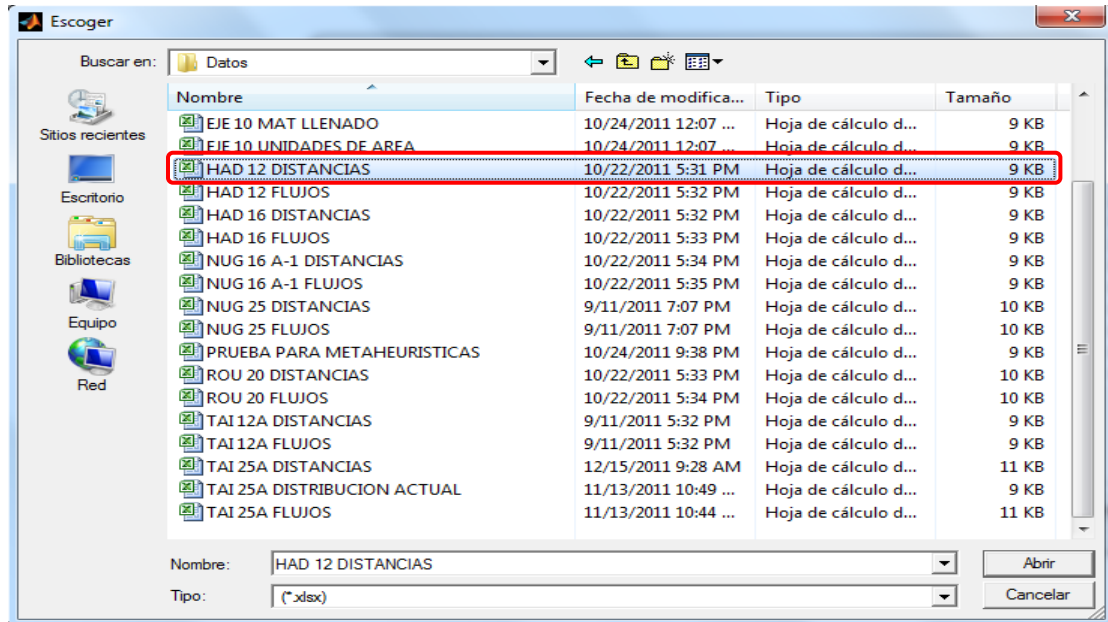


- **Matriz de distancias.** Corresponde a las distancias que se requieren entre los departamentos a distribuir. Al seleccionar este comando se despliega la cuadrícula con una matriz cuadrada de tamaño 12x12. Para ingresar la matriz de distancias se puede digitar directamente en la cuadrícula o cargarse desde un archivo de Excel, donde ya ha sido guardada, haciendo clic en el comando “Examinar” (*Ilustración 25*), de esta manera se despliega una ventana en donde se visualiza la carpeta “Datos”, y se hace clic en el archivo que se requiere (*Ilustración 26*).

**Ilustración 25. Vista “Entorno gráfico características de la planta”**

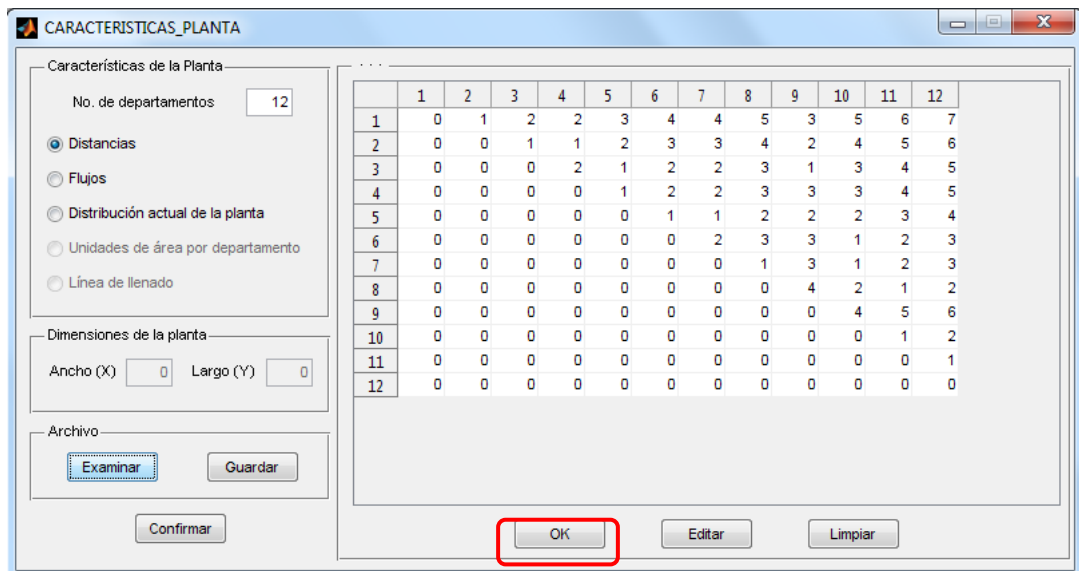
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0

**Ilustración 26. Vista ventana “Cargar matriz de distancias”**

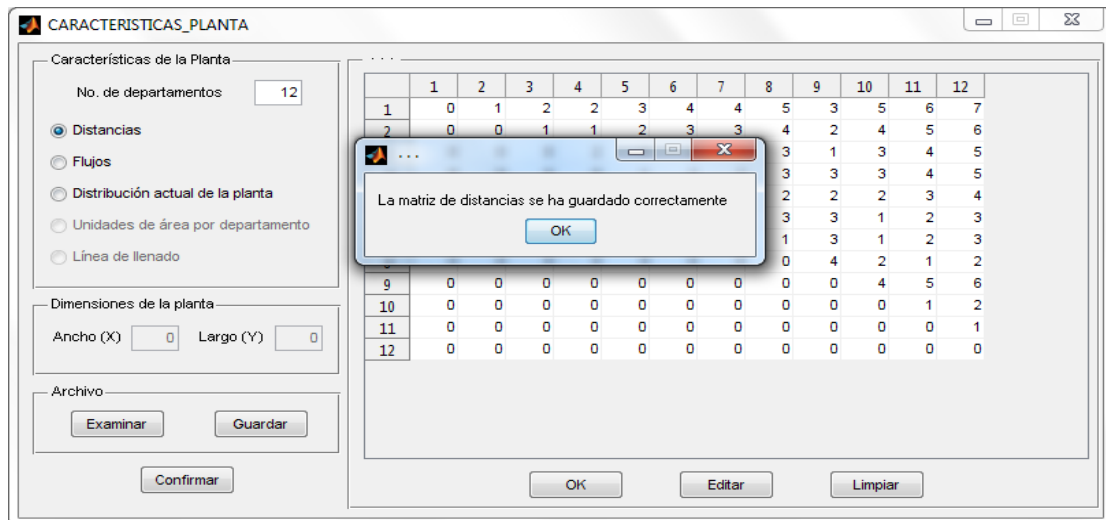


En la *ilustración 27*, se observa una matriz diagonal superior, que corresponde a la distancia entre departamentos. Para validar esta matriz se hace clic en el icono “OK” y se muestra que la matriz de distancias se ha guardado correctamente (*ilustración 28*).

**Ilustración 27. Vista ventana “Matriz de distancias”**



## Ilustración 28. Vista ventana “Guardar matriz de distancias”

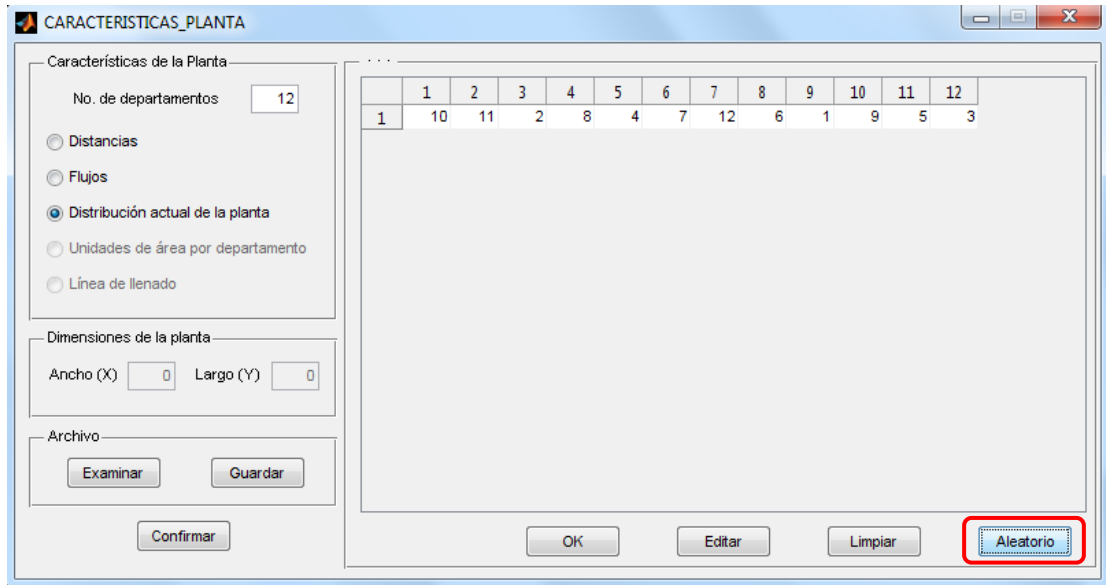


- **Matriz de flujos.** Corresponde a la cantidad de material y producto en proceso que se transporta entre departamentos. Para ingresar la matriz de flujos, se sigue el mismo procedimiento descrito para la matriz de distancias, en el ítem anterior.

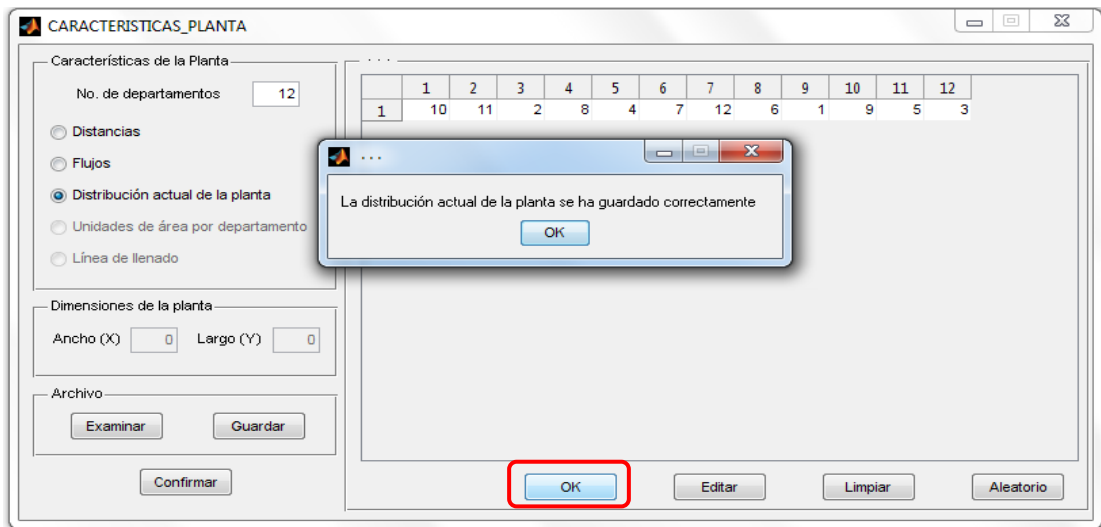
- **Distribución actual de la planta.** Es la asignación de departamentos, a partir de la cual se va a realizar la búsqueda y mejora. Para la metaheurística HBMO la solución inicial es la abeja reina con la cual se inicia el algoritmo. Esta solución puede ser digitada por el usuario, dependiendo del conocimiento que tenga de su proceso productivo, si es la primera vez que organiza su planta; o de la distribución que actualmente tiene, si ya viene operando de alguna forma. También existe la opción de generar una solución aleatoria, seleccionando el icono “Aleatorio” (*Ilustración 29*). Para el caso se genera un vector de tamaño 12. Para el ejemplo se ha generado una solución aleatoria, indicando por ejemplo, que el departamento 10 está ubicado en la instalación 1. Para validar esta solución se hace clic en el icono “OK” y se observa que la distribución actual de la planta se ha guardado correctamente (*Ilustración 30*).

-

**Ilustración 29. Vista ventana “Generar distribución actual de la planta”**



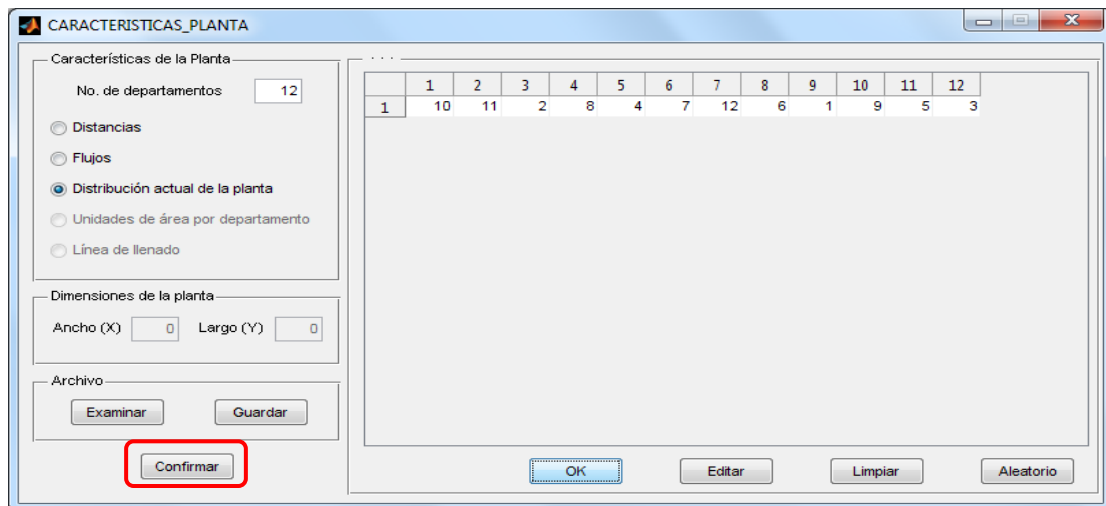
**Ilustración 30. Vista ventana “Guardar distribución actual de la planta”**



Teniendo todos los datos de las características de la planta para el problema de áreas iguales, se hace clic en el botón “Confirmar” ubicado en la parte inferior izquierda de la pantalla (Ilustración 31). De esta forma se despliega nuevamente el

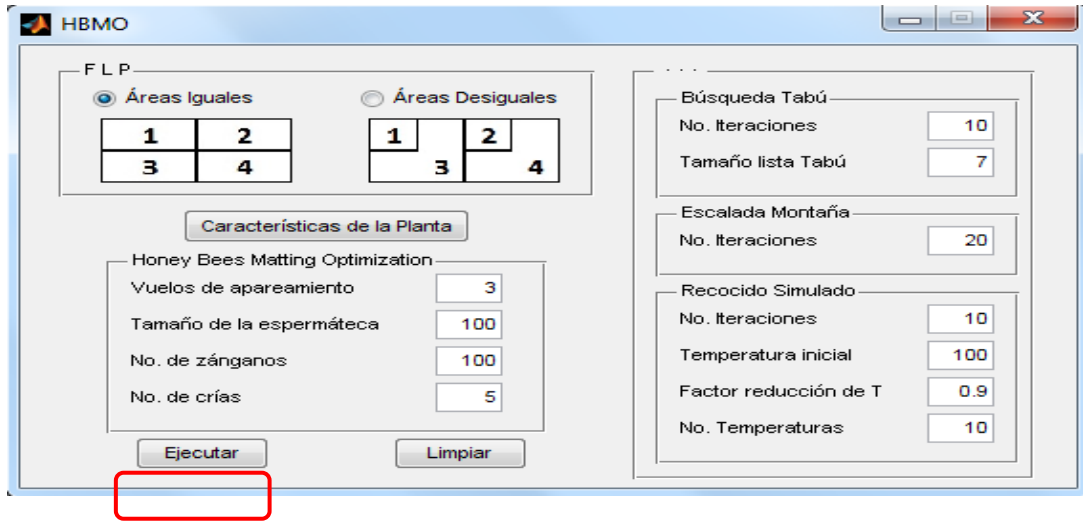
entorno gráfico del algoritmo HBMO, en el cual se procede a digitar los parámetros, de acuerdo al problema planteado (*Ilustración 32*).

**Ilustración 31. Vista ventana “Confirmar datos de características de la planta”**



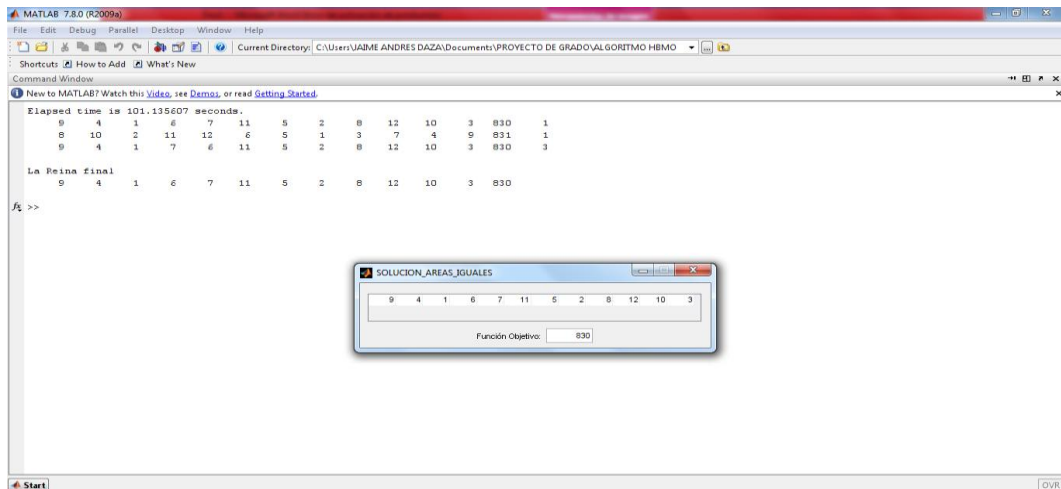
**8.3.2. Ingresar parámetros del algoritmo HBMO.** El usuario deberá digitar los parámetros correspondientes a vuelos de apareamiento, tamaño de la espermoteca, número de zánganos, número de crías y a su vez los parámetros de las trabajadoras (metaheurísticas búsqueda tabú, escalada de montaña y recocido simulado). Por ejemplo para el problema se estableció realizar 3 vuelos de apareamiento, que indica el número de iteraciones que se llevarán a cabo. Una vez se han ingresado los parámetros del algoritmo HBMO, se hace clic en el icono “Ejecutar”, de esta forma se inicia el proceso de mejora de acuerdo a la metaheurística HBMO, explicada en el *capítulo 7*.

**Ilustración 32. Vista ventana “Ingresar parámetros del algoritmo HBMO”**



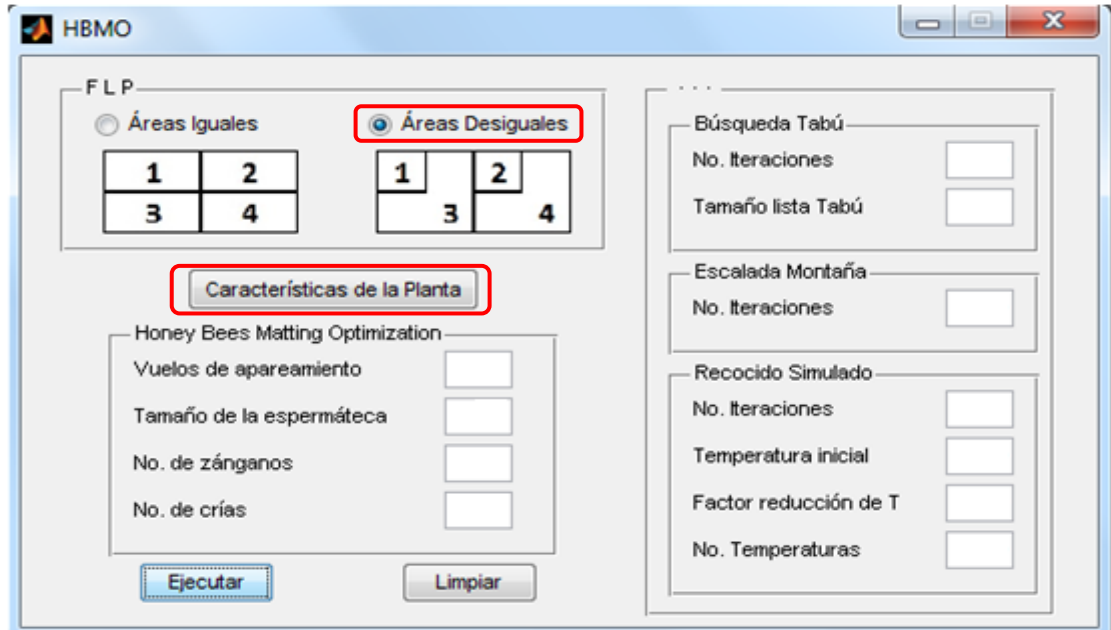
**8.3.3. Obtención de la solución final.** Al terminar la ejecución, se despliega en la pantalla la nueva abeja reina, que corresponde a la mejor solución encontrada resultado de la ejecución del algoritmo HBMO, junto a su valor de función objetivo. Como información adicional, se presenta la mejor solución encontrada en cada vuelo de apareamiento, con su valor de función objetivo y la metaheurística que logró la mejora. El número 1, corresponde a Búsqueda Tabú, el número 2 a Recocido Simulado y el número 3 a Escalada de Montaña (*Ilustración 33*).

**Ilustración 33. Vista ventana “Solución final, problema de áreas iguales”**



## 8.4. PROBLEMA DE ÁREAS DESIGUALES

Ilustración 34. Vista “Entorno gráfico algoritmo HBMO”

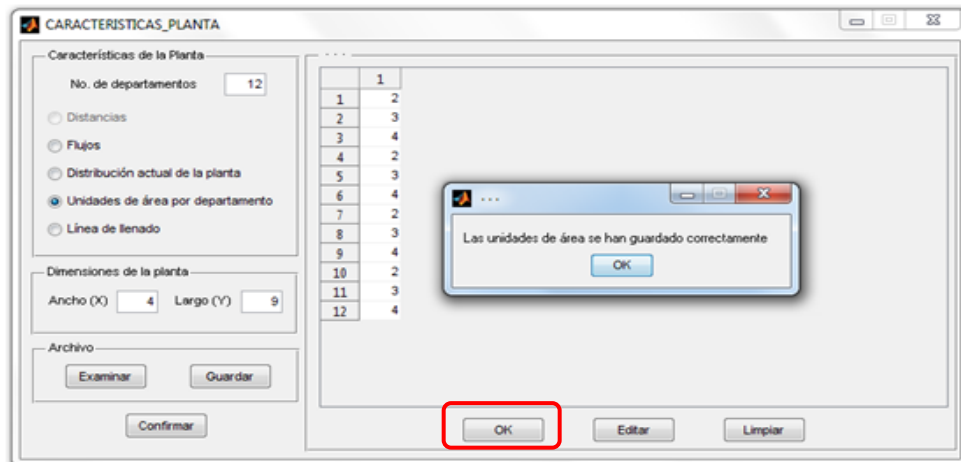


**8.4.1. Ingresar datos de características de la planta.** De la misma forma como se explicó en el numeral 8.3.1. se ingresan los datos del número de departamentos, matriz de flujos y distribución actual de la planta, para el problema de áreas desiguales. Para este caso no aplica matriz de distancias, teniendo en cuenta que las distancias entre departamentos se harán de centroide a centroide de cada departamento, para tal fin se requieren las dimensiones de la planta las unidades de área requeridas por cada departamento y la línea de llenado; todos conceptos explicados en los capítulos 4 y 5.

- **Dimensiones de la planta.** Se ingresa el largo y ancho del área total disponible para la planta (Ilustración 35).

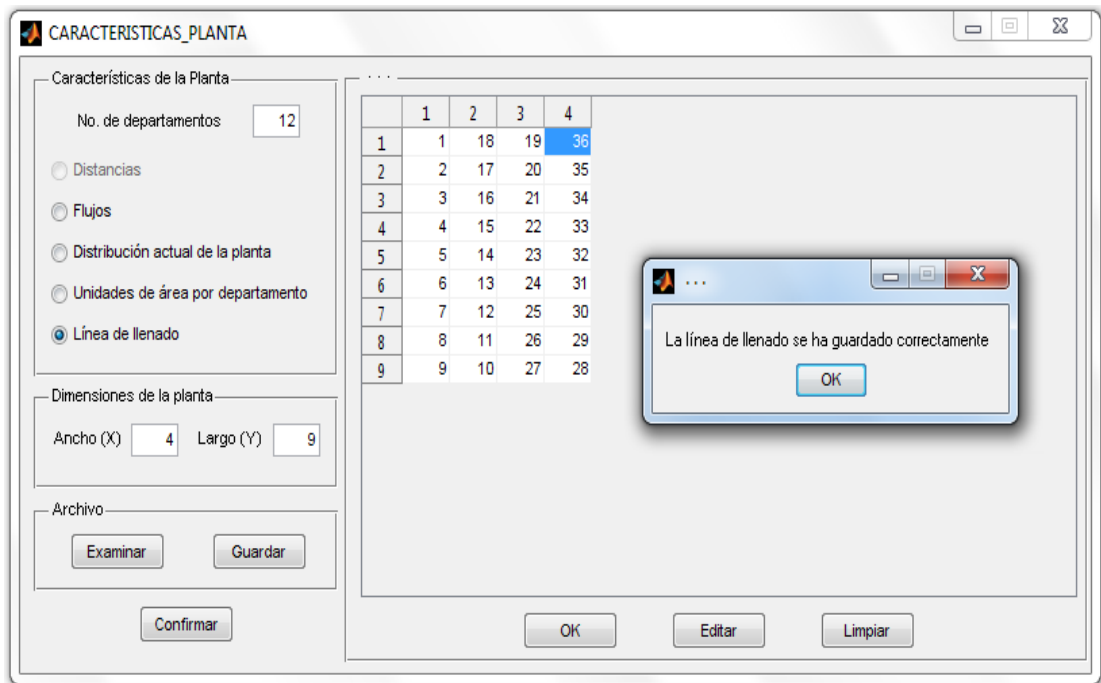
- **Unidades de área por departamento.** Cada departamento requiere un área precisa para su operación, para el ejemplo el departamento 1 requiere 2 unidades de área, el departamento 2 necesita 3 unidades de área y así sucesivamente según el usuario va llenando el vector; se debe tener claro que la suma de las unidades de área requeridas no debe superar las dimensiones de la planta. Para validar estos datos, se hace clic en el icono “OK” y de esta forma se guardan los datos en una memoria temporal.

**Ilustración 35. Vista ventana “Ingresar datos de unidades de área por departamento”**



- **Línea de llenado.** Para solucionar el problema de áreas desiguales, se toma la planta a distribuir y se construye sobre ella una cuadrícula formada por cuadros o rectángulos (unidad de área). Teniendo el número de unidades de área por cada departamento, se debe garantizar que éstas queden juntas en la distribución, para esto se maneja una línea de llenado, consiste en una línea continua que atraviesa cada una de las unidades de área y va definiendo la secuencia en la que van a ser ubicados los departamentos. En la herramienta diseñada esta secuencia está dada por números de 1 a 36, teniendo en cuenta que la planta ha sido dividida en 36 unidades de área (*Ilustración 36*).

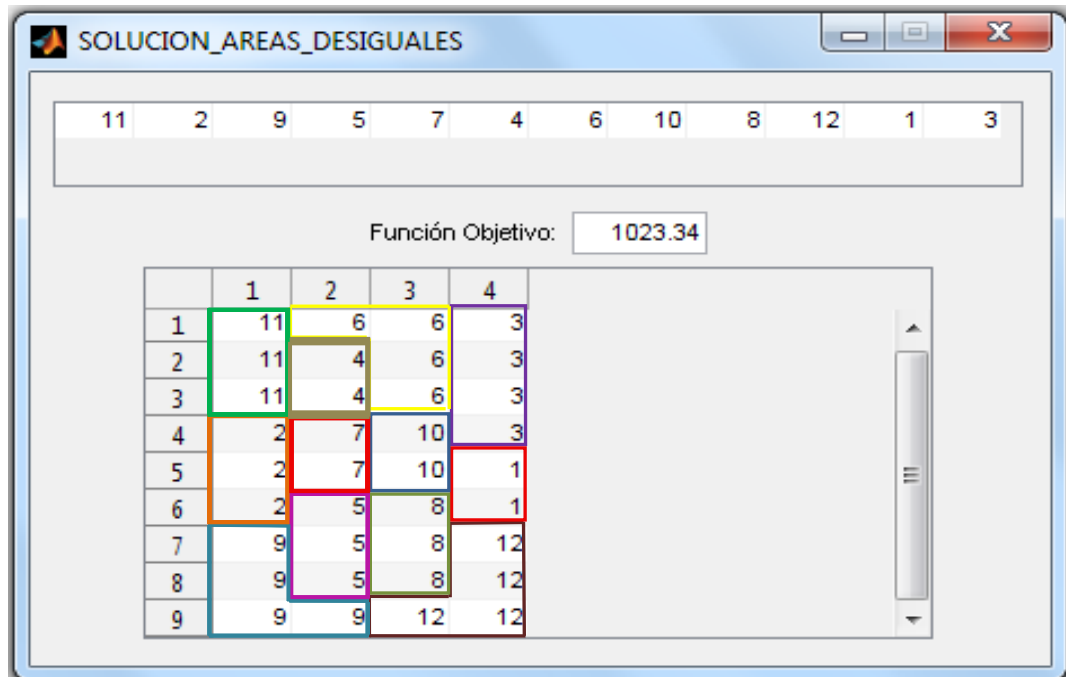
**Ilustración 36. Vista ventana “Línea de llenado”**



**8.4.2. Ingresar parámetros del algoritmo HBMO.** De la misma manera como está descrito en el numeral 8.3.2. se ingresan los datos correspondientes al algoritmo HBMO, para el problema de áreas desiguales.

**8.4.3. Obtención de la solución final.** Al terminar la ejecución, se despliega en la pantalla, la mejor solución encontrada resultado de la ejecución del algoritmo HBMO, junto a su valor de función objetivo. En el ejemplo se observa como, primero queda ubicado el departamento 11 con 3 unidades de área. En la *ilustración 37*, se resaltan los doce departamentos.

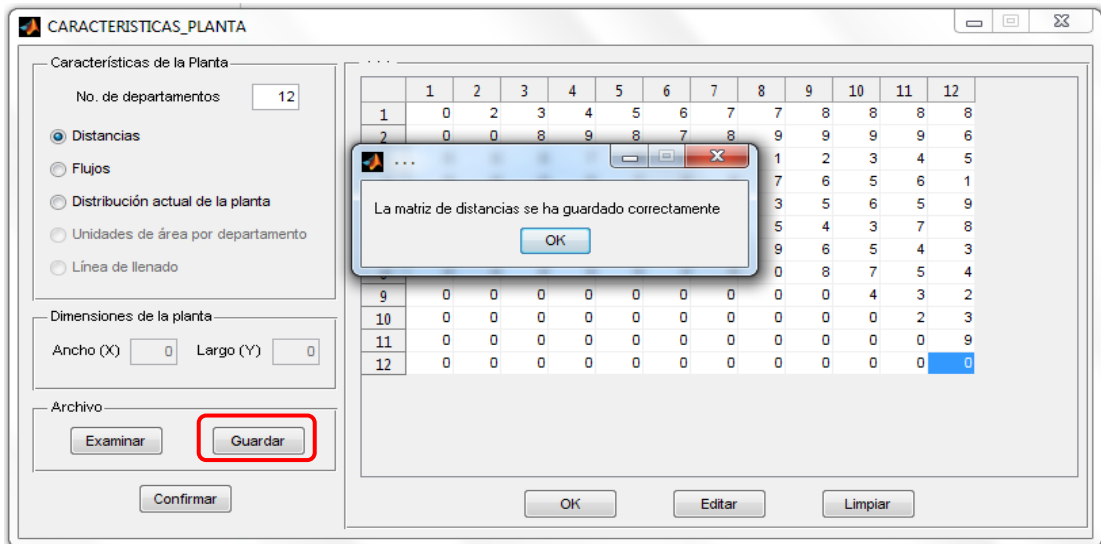
**Ilustración 37. Vista ventana “Solución final, problema de áreas desiguales”**



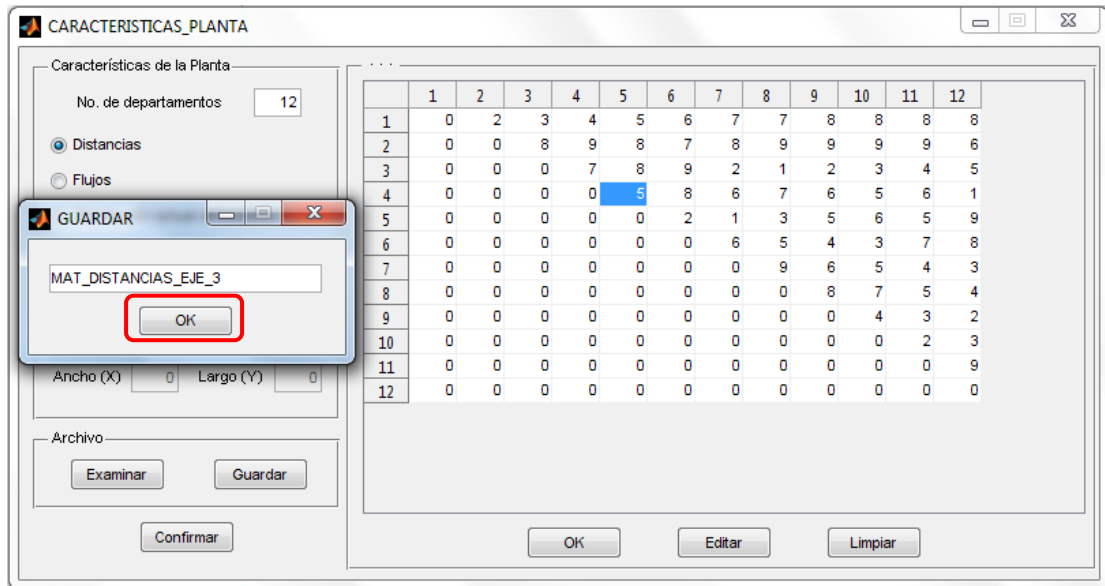
### 8.5. GUARDAR DATOS EN EL EQUIPO

Cuando los datos que se utilizan no están guardados en el equipo, por el contrario han sido digitados por el usuario, existe la opción de guardarlos en la carpeta “Datos” del equipo. Después de guardar en la memoria temporal de MATLAB, se elige la opción “Guardar” en la parte inferior izquierda de la pantalla (*Ilustración 38*), se escribe el nombre de la matriz o vector a guardar (*Ilustración 39*) y se hace un clic en el icono “OK”. Para el ejemplo, el archivo “MAT\_DISTANCIA\_EJE\_3” quedará guardado como un archivo de Excel, en la carpeta “Datos”, así el usuario podrá recurrir a este, cuando lo requiera.

**Ilustración 38. Vista ventana “Guardar datos en el equipo”**



**Ilustración 39. Vista ventana “Asignar nombre a archivo para guardar en el equipo”**



## 9. ANÁLISIS ESTADÍSTICO

De los problemas de prueba proporcionados por Burkard, Karisch y Rendl en 1994, disponibles en la librería electrónica QAPLIB [8], fueron seleccionados ocho problemas para probar la eficiencia del algoritmo HBMO propuesto en este trabajo para resolver el problema de distribución de planta.

Se analizaron 4 tamaños de planta diferente; 12, 16, 20 y 25 departamentos. De los ocho problemas se corrieron dos para cada escenario. Cada problema fue resuelto con 3, 6, 9 y 12 vuelos de apareamiento y 5, 10, 15 y 20 crías para cada vuelo.

La solución óptima dada por la librería QAPLIB para cada problema resuelto, se relaciona a continuación:

**Tabla 5. Solución óptima de cada problema**

No. Departamentos	Problema	Solución Óptima
12	tai12a	112208
12	had12	826
16	had16	1860
16	nug16	805
20	rou20	362761
20	nug20	1285
25	tai25a	583628
25	nug25	1872

Con las soluciones obtenidas mediante el algoritmo HBMO y la solución óptima de cada problema dado en la tabla anterior, se calculó el porcentaje de diferencia entre dichas soluciones (*Ver Anexo C*).

## 9.1. DESEMPEÑO DEL ALGORITMO

### 9.1.1. Problemas de 12 Departamentos

Tabla 6. Desempeño algoritmo – 12 Departamentos

PROBLEMA	ÓPTIMO	MEJOR Z ENCONTRADO	$\Delta Z\%$	PROMEDIO (%)
TAI 12 A	112208	112208	0,000%	0%
HAD 12	826	826	0,000%	

Se encontró el óptimo para los dos casos, por lo tanto el error promedio es del 0%. Para el problema tai12a siempre se encontró el óptimo para todos los parámetros definidos del modelo y para el problema had12 se encontró 14 de las 16 veces y se halló por primera vez para la combinación de 3 vuelos de apareamiento y 10 crías. Se observa un excelente desempeño del algoritmo para dar solución a este tipo de problemas. Una combinación recomendada sería 9 vuelos y 10 crías, teniendo en cuenta que a partir de ahí siempre se halló el óptimo, por lo que no se justifica aumentar más los parámetros.

### 9.1.2. Problemas de 16 Departamentos

Tabla 7. Desempeño algoritmo – 16 Departamentos

PROBLEMA	ÓPTIMO	MEJOR Z ENCONTRADO	$\Delta Z\%$	PROMEDIO (%)
had16	1860	1860	0,000%	0%
nug16	805	805	0,000%	

Para este tipo de problemas también se encontró el óptimo, se observa un error promedio de 0%. Para los dos casos se encontró el óptimo 8 veces de las 16 corridas hechas, para cada uno. Analizando los datos obtenidos, se recomienda

para este tipo de problemas hacer corridas con 9 vuelos y 15 crías o con 6 vuelos y 20 crías, teniendo en cuenta que con estos parametros o más, siempre se encuentra el óptimo y sería pérdida de tiempo computacional aumentarlos.

### 9.1.3. Problemas de 20 Departamentos

**Tabla 8. Desempeño algoritmo – 20 Departamentos**

PROBLEMA	ÓPTIMO	MEJOR Z ENCONTRADO	$\Delta Z\%$	PROMEDIO (%)
rou20	362761	363406	0,178%	0,089%
nug20	1285	1285	0,000%	

En el escenario de 20 Departamentos el error promedio es de 0,089%. Para el problema de rou20 no se encontró el óptimo, aunque se encontró una buena solución. Para el caso de nug20 se encontró el óptimo 3 veces. Para este tipo de problema se hace necesario hacer 12 vuelos de apareamiento y 20 crías, de tal forma que se garantice una buena solución, tal como se observó en el experiento con estos valores en los parámetros se obtuvo 0,178% de error para el problema rou20 y 0% de error para el caso de nug20.

### 9.1.4. Problemas de 25 Departamentos

**Tabla 9. Desempeño algoritmo – 25 Departamentos**

PROBLEMA	ÓPTIMO	MEJOR Z ENCONTRADO	$\Delta Z\%$	PROMEDIO (%)
tai25a	583628	593629	1,714%	0,857%
nug25	1872	1872	0,000%	

En el escenario de 25 Departamentos el error promedio es un poco mayor a los escenarios anteriores, esto debido a que la complejidad del problema empieza a ser mayor. El error promedio para este caso es 0,857%. En este caso se

recomienda hacer 12 vuelos de apareamiento y generar 20 crías, de tal forma que se garantice una solución muy buena, así como se observó en el experimento donde en el caso del problema nug25 se obtuvo el óptimo para esta combinación y para el tai25a aunque no se halló el óptimo, se encontró una solución con desviación del óptimo de 1,714%, que se considera una buena solución. Se pueden hacer pruebas con números mayores en los parámetros definidos, sin embargo este porcentaje de error es un buen índice del rendimiento del algoritmo.

## **9.2. ANÁLISIS DEL EFECTO DE LOS FACTORES SOBRE LA VARIABLE DE SALIDA**

Dado que el desempeño del algoritmo para los problemas de 12 y 16 departamentos fue muy bueno, con desviaciones del óptimo casi nulas para todos los problemas, no se justifica considerar estos tipos de problemas en el diseño de experimentos.

Con el fin de determinar la influencia de los parámetros utilizados en el algoritmo HBMO sobre el resultado obtenido en la ejecución del mismo, se realizó el siguiente estudio. Se identificaron dichos parámetros para hacer el diseño de experimentos sobre los problemas escogidos. Se analizó el comportamiento de la metaheurística para los problemas de 20 y 25 departamentos.

La variable de salida está dada por el índice de error porcentual (desviación del óptimo) entre las soluciones obtenidas mediante el algoritmo HBMO propuesto, respecto a la solución óptima dada en la librería QAPLIB. Así:

$$\text{DesvOpt} = \frac{Z_{HBMO} - Z_{OPT}}{Z_{OPT}}$$

Los factores que se consideraron posiblemente determinantes en la variable de salida fueron:

- NÚMERO DE VUELOS
- NÚMERO DE CRIAS

**Tabla 10. Parámetros utilizados para el experimento**

Número de Vuelos	Número de Crías
3	5
6	10
9	15
12	20

El modelo estadístico lineal planteado para describir las observaciones obtenidas, es el siguiente:

$$Y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \varepsilon_{ijk}, \quad \begin{cases} i = 3, 6, 9 \text{ y } 12 \\ j = 5, 10, 15 \text{ y } 20 \\ k = 1 \text{ y } 2 \end{cases}$$

donde:

$Y_{ijk}$  = respuesta observada

$\mu$  = es el efecto promedio global

$\tau_i$  = es el efecto del nivel  $i$ -ésimo del factor número de vuelos

$\beta_j$  = es el efecto del nivel  $j$ -ésimo del factor número de crías

$(\tau\beta)_{ij}$  = es el efecto de la interacción entre los factores número de vuelos y número de crías

$\varepsilon_{ijk}$  = es el componente del error aleatorio

El fin del experimento es probar las hipótesis de que no hay efecto principal para el factor número de vuelos, que no hay efecto principal para el factor número de crías, y que no hay efecto de la interacción número de vuelos y número de crías.

Las hipótesis que se probarán son las siguientes:

1.  $H_0: \tau_3 = \tau_6 = \tau_9 = \tau_{12} = 0$  (ningún efecto principal del factor número de vuelos)  
 $H_a$ : al menos una  $\tau_i \neq 0$
2.  $H_0: \beta_5 = \beta_{10} = \beta_{15} = \beta_{20} = 0$  (ningún efecto principal del factor número de crías)  
 $H_a$ : al menos una  $\beta_j \neq 0$
3.  $H_0: (\tau\beta)_{ij} = 0 ; \forall i, j$  (ninguna interacción)  
 $H_a$ : al menos una  $(\tau\beta)_{ij} \neq 0$

Se usó el análisis de varianza (ANOVA) para probar estas hipótesis para los dos escenarios, 20 y 25 departamentos. A continuación se presenta el ANOVA para cada escenario:

**Tabla 11. ANOVA – Problemas de 20 Departamentos**

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,002 <sup>a</sup>	15	,000	5,489	,001
Intersección	,004	1	,004	218,956	,000
NumVuelo	,001	3	,000	18,831	,000
NumCrias	,000	3	8,33E-005	4,163	,023
NumVuelo * NumCrias	,000	9	2,97E-005	1,484	,235
Error	,000	16	2,00E-005		
Total	,006	32			
Total corregida	,002	31			

a. R cuadrado = ,837 (R cuadrado corregida = ,685)

Del ANOVA para los problemas de 20 departamentos (*tabla 11*), se concluye que no existe una interacción significativa entre los dos factores principales: número de vuelos y número de crías, además se concluye que para este caso, el efecto producido por el factor número de vuelos es significativo sobre la variable respuesta, así como lo es también el factor número de crías. Es decir que el rendimiento del algoritmo depende de dichos parámetros.

**Tabla 12. ANOVA – Problemas de 25 Departamentos**

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	15	,000	1,339	,284
Intersección	,013	1	,013	59,533	,000
NumVuelo	,002	3	,001	2,551	,092
NumCrias	,002	3	,001	2,304	,116
NumVuelo * NumCrias	,001	9	,000	,613	,770
Error	,003	16	,000		
Total	,021	32			
Total corregida	,008	31			

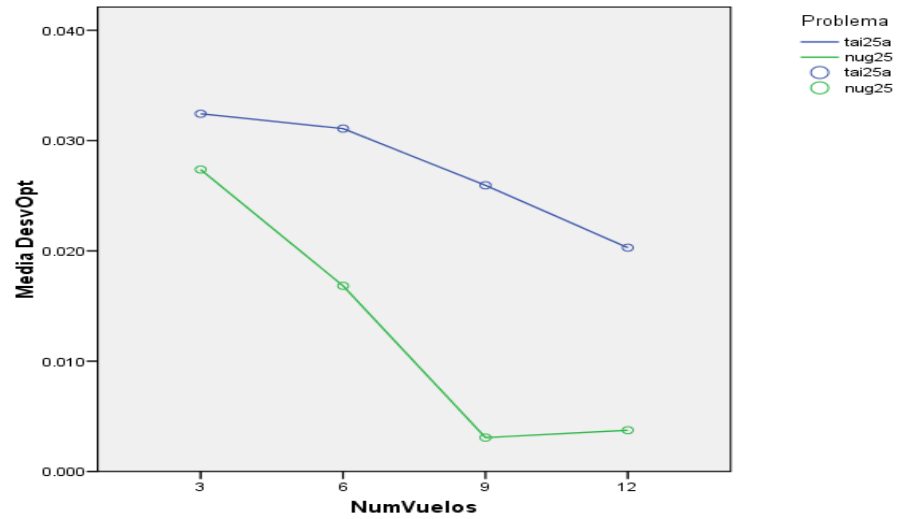
a. R cuadrado = ,557 (R cuadrado corregida = ,141)

Del análisis de varianza para los problemas de 25 departamentos, se observa que no existe interacción significativa entre los dos factores principales, sin embargo, el efecto producido por el número de vuelos sobre la variable de salida si se considera significativo y el nivel de significancia del factor número de crías podría considerarse, con un nivel de significancia del 0,116.

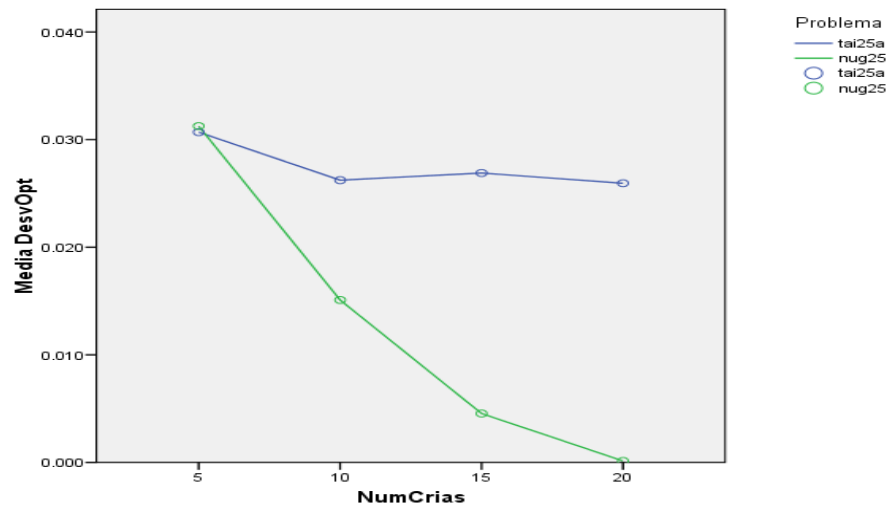
### 9.3. ANÁLISIS DEL FACTOR PROBLEMA

Se decidió hacer el análisis de un tercer factor para instancias de 25 departamentos, teniendo en cuenta los resultados obtenidos y dada la complejidad de este tipo de problemas. Al tener en cuenta el factor problema, se analiza si la variable de salida depende de las características propias del problema planteado.

**Ilustración 40. Grafico de perfil – DesvOpt Vs. Número de vuelos de apareamiento(25 Departamentos)**



**Ilustración 41. Grafico de perfil – DesvOpt Vs. Número de crías (25 Departamentos)**



En las ilustraciones 40 y 41 se observa como el comportamiento de los dos problemas es diferente. En el diseño de experimento se observó que el problema nug25 obtuvo mejores resultados, con errores porcentuales menores al problema de tai25a.

**Tabla 13. ANOVA – Factor Problema Vs. Número de vuelos de apareamiento (25 Departamentos)**

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	7	,001	3,091	,018
Intersección	,013	1	,013	75,297	,000
NumVuelos	,002	3	,001	3,227	,040
Problema	,002	1	,002	10,050	,004
NumVuelos * Problema	,000	3	,000	,635	,600
Error	,004	24	,000		
Total	,021	32			
Total corregida	,008	31			

a. R cuadrado = ,474 (R cuadrado corregida = ,321)

**Tabla 14. ANOVA – Factor Problema Vs. Número de crías (25 Departamentos)**

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	7	,001	3,729	,007
Intersección	,013	1	,013	82,668	,000
NumCrias	,002	3	,001	3,200	,041
Problema	,002	1	,002	11,034	,003
NumCrias * Problema	,001	3	,000	1,823	,170
Error	,004	24	,000		
Total	,021	32			
Total corregida	,008	31			

a. R cuadrado = ,521 (R cuadrado corregida = ,381)

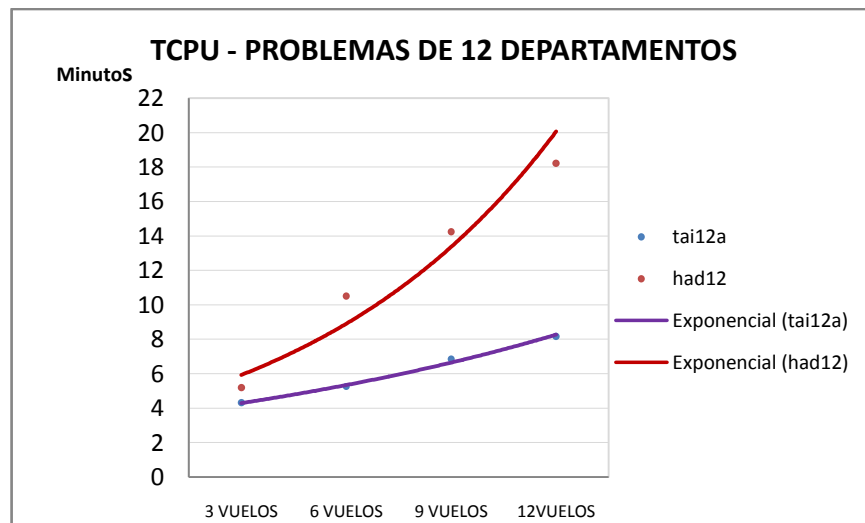
De las tablas 13 y 14, se concluye que para instancias de 25 departamentos el factor problema si tiene un efecto sobre la variable de salida, lo que corrobora que los parámetros deben ser definidos independientemente para cada problema, a su vez se consideran significativos los efectos principales de los factores número de

vuelos y número de crías sobre la variable de salida. Aunque no hay interacción significativa entre el factor problema y los otros dos factores.

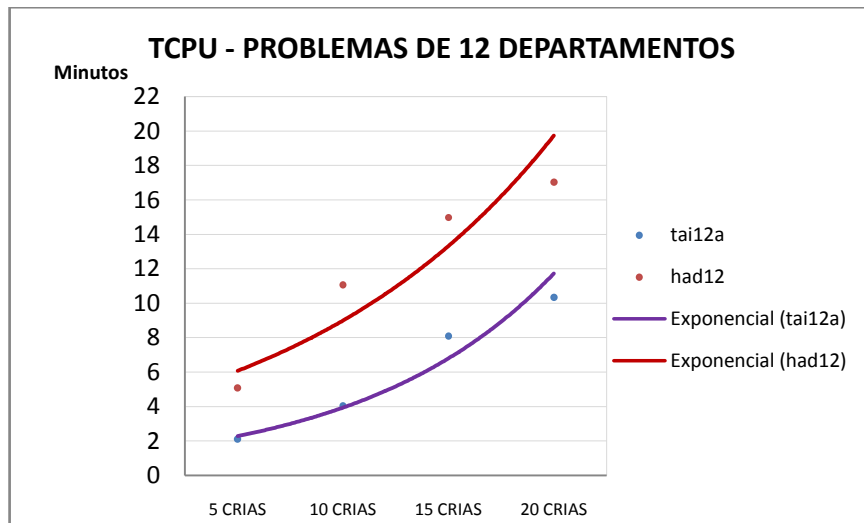
#### 9.4. TIEMPO COMPUTACIONAL

El experimento fue realizado en un computador Hewlett-Packard HG G62, con un procesador Intel® Core™ i3 CPU, 2.4 GHz y 4.00 GB de memoria. Sistema operativo de 64 bits. Se han graficado los tiempos computacionales promedio contra el número de vuelos de apareamiento y contra el número de crías.

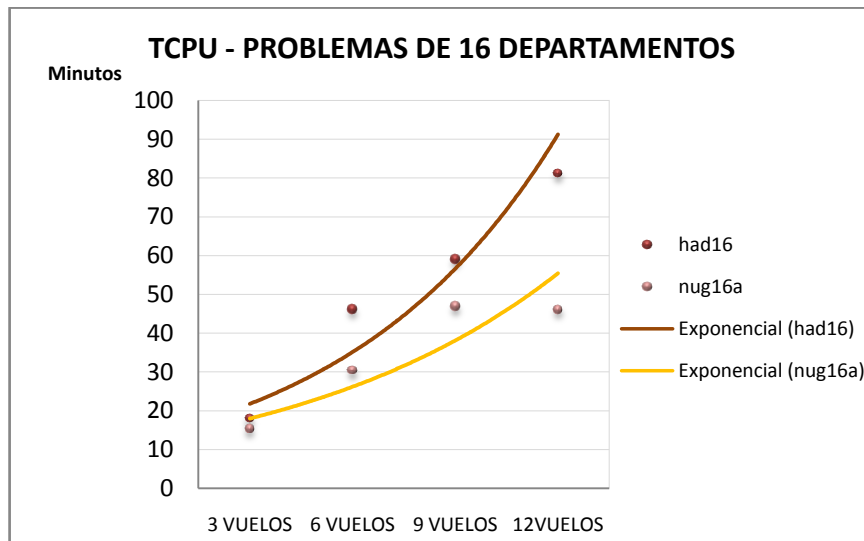
**Ilustración 41. Tiempo computacional Vs. Número de vuelos de apareamiento - 12 Departamentos**



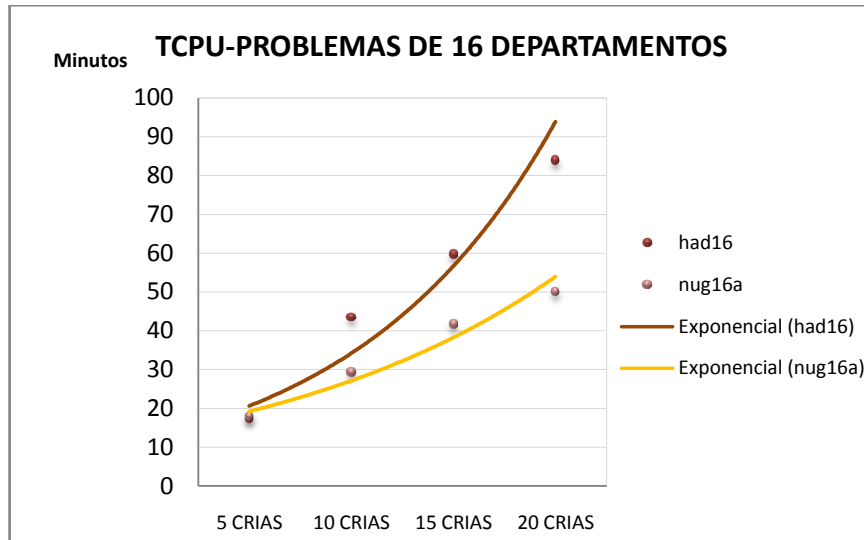
**Ilustración 42. Tiempo computacional Vs. Número crías - 12 Departamentos**



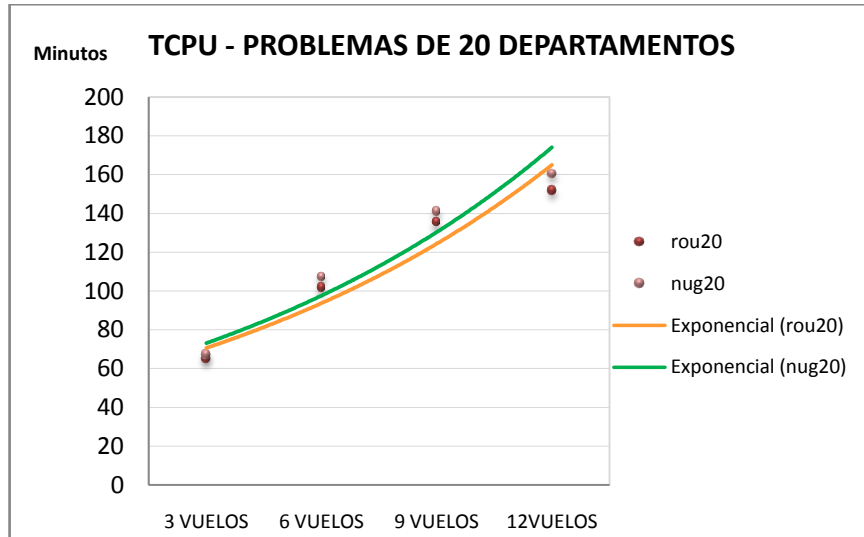
**Ilustración 43. Tiempo computacional Vs. Número de vuelos de apareamiento - 16 Departamentos**



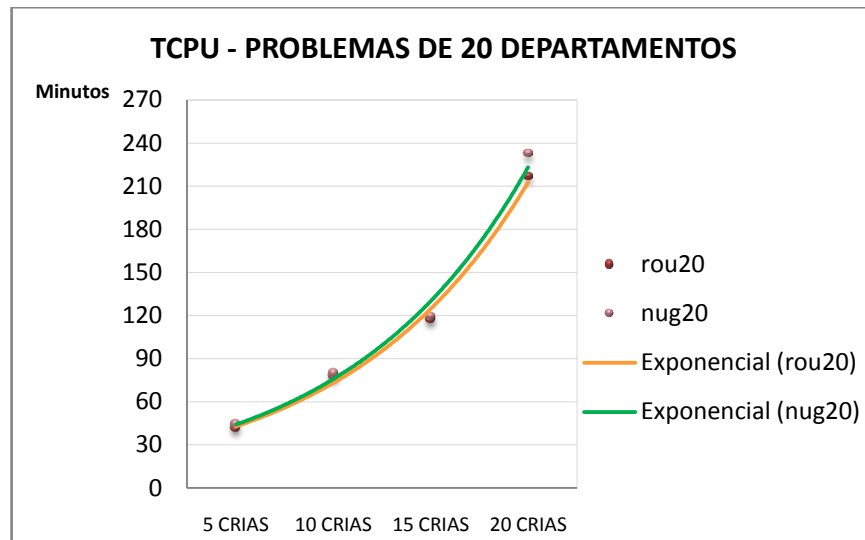
**Ilustración 44. Tiempo computacional Vs. Número crías - 16 Departamentos**



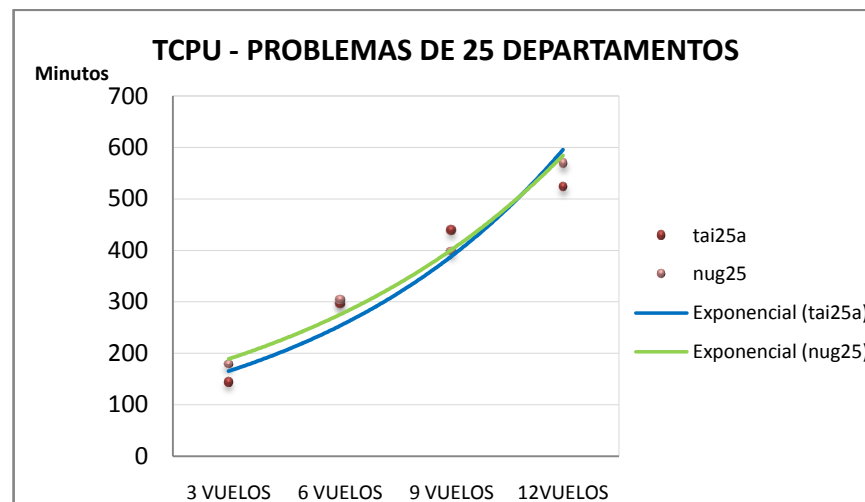
**Ilustración 45. Tiempo computacional Vs. Número de vuelos de apareamiento – 20 Departamentos**



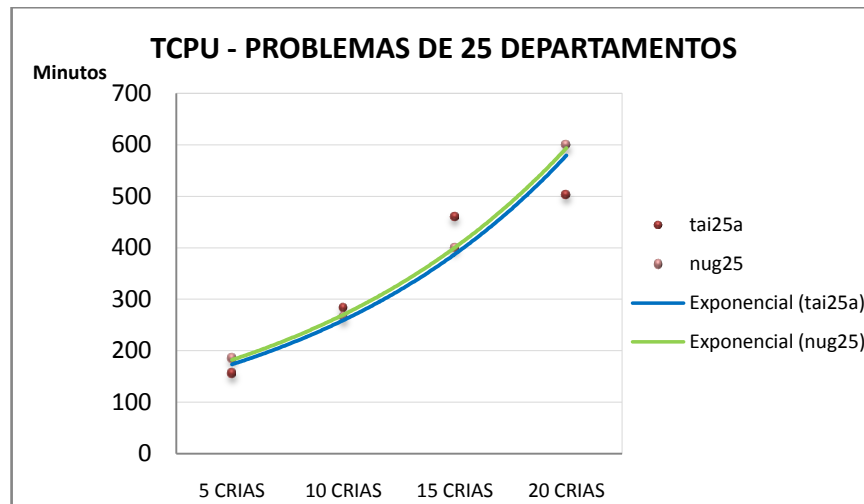
**Ilustración 46. Tiempo computacional Vs. Número crías - 20 Departamentos**



**Ilustración 47. Tiempo computacional Vs. Número de vuelos de apareamiento – 25 Departamentos**



**Ilustración 48. Tiempo computacional Vs. Número crías - 25 Departamentos**



Se puede observar como al aumentar cualquiera de los dos parámetros, el tiempo computacional empieza a aumentar exponencialmente. Para instancias hasta 16 departamentos como se ha explicado en el *numeral 9.1*, se pueden disminuir iteraciones, a fin de reducir el tiempo computacional, sin sacrificar la obtención de una buena solución.

## 10. CONCLUSIONES

- Se alcanzaron todos los objetivos propuestos en la investigación; el algoritmo codificado para la implementación de la metaheurística HBMO al problema arroja muy buenos resultados en tiempo computacional aceptable.
- En la programación del algoritmo se consideró dejar los códigos a disposición del usuario, a fin de que sean útiles como base para la solución de este mismo problema aumentando restricciones o en la solución de otros tipos de problemas de optimización.
- Los algoritmos metaheurísticos producen muy buenas soluciones y en tiempo computacional aceptable, ya que recurren a múltiples estrategias para escapar de óptimos locales y permite la exploración de muchas regiones en el espacio de soluciones factibles.
- La metaheurística HBMO ofrece un buen desempeño ya que retoma estrategias de otras metaheurísticas y las incorpora a su procedimiento de búsqueda y mejora.
- Teniendo en cuenta los resultados arrojados en el experimento se concluye que:
  - Las desviaciones del óptimo son muy pequeñas para todos los casos e incluso del 0% para los casos de 12 y 16 departamentos, lo que corrobora el buen desempeño del algoritmo HBMO.

- Una combinación recomendada para los problemas de 12 departamentos sería 9 vuelos y 10 crías, teniendo en cuenta que a partir de ahí siempre se encontró el óptimo para los problemas resueltos.
- Para los problemas de 16 departamentos se recomienda hacer corridas con 9 vuelos y 15 crías o con 6 vuelos y 20 crías, dado que con números mayores en estos parámetros siempre se encuentra el óptimo y no se justifica su incremento.
- Para los escenarios de 20 y 25 departamentos se hace necesario realizar 12 vuelos de apareamiento y generar 20 crías, de tal forma que se garantice una solución muy buena, así como se evidenció en el experimento realizado. Se pueden hacer pruebas con números mayores en los parámetros definidos, sin embargo los porcentajes de error que se obtuvieron son un buen índice del rendimiento del algoritmo.
- Para instancias de 20 y 25 departamentos, la definición de los parámetros es independiente para cada problema, es decir, el rendimiento del algoritmo depende del problema particular que se esté resolviendo.
- No fue posible realizar una comparación de resultados para problemas de áreas desiguales, debido a la falta de acceso a problemas de referencia que utilicen el procedimiento desarrollado en este trabajo, sin embargo en el manual de uso de la herramienta se han plasmado los pasos a seguir para la solución de este tipo de problemas y se evidenció su eficiencia y buen funcionamiento.

## 11. RECOMENDACIONES

- Que los profesores investigadores, muestren a sus estudiantes los trabajos hechos de tal forma que los códigos programados sean reutilizados y mejorados y no se pierda el avance hecho.
- Divulgar los proyectos de grado de investigación dentro de los estudiantes, ya que es una experiencia enriquecedora y de gran aporte a la formación profesional.
- Introducir a la aplicación, matrices con datos reales. Desde la academia es posible brindar asistencia a las empresas de la región, a fin de que las herramientas desarrolladas aporten a mejorar la eficiencia y productividad de las mismas.
- Aplicar la metaheurística HBMO a otros problemas de optimización, teniendo en cuenta, que se han obtenido buenos resultados.

## 12. LÍNEAS FUTURAS DE INVESTIGACIÓN

- Ampliar la investigación, considerando mayor número de restricciones, acercándose de esta manera a lo que es una planta real.
- Se recomienda aplicar la metaheurística al Problema de Distribución de Planta Dinámico, de tal manera que se consideren los cambios en la demanda a través del tiempo, lo cual tiene incidencia en el Diseño de la Planta.
- Incorporar otro tipo de trabajadoras al algoritmo HBMO, a fin de mejorar el desempeño de la metaheurística.
- Desarrollar proyectos de este tipo en empresas, de tal forma que se caracterice la planta siguiendo sus condiciones específicas y se haga la formulación y la implementación del algoritmo para el caso particular.

### 13. BIBLIOGRAFIA

1. Abbass H. A. MBO: Marriage in Honey Bees Optimization A Haplometrosis Polygynous Swarming Approach. IEEE Congress on Evolutionary Computation 2001; Seoul, pp. 207-214 vol. 1
2. AMINE, Drira.; HENRI, Pierreval.; y SONIA, Hajri-Gabouj. Facility layout problems: A survey. No. 31 (2007); p. 255 – 267
3. ARMOUR, G.C. y BUFFA, E.S., A heuristic algorithm and simulation approach to relative allocation of facilities. Management Science. Vol. 9, No. 2, 1963; p. 294–309
4. BALAKRISHNAN, J.; CHENG, C.H.; CONWAY D.G.; y LAU C.M. A hybrid genetic algorithm for the dynamic plant layout problem. International journal of production economics. Vol. 2, No. 86 (2003); p. 107 -120
5. BARTHOLDI, J.J. y PLATZMAN, L.K., An Onlong planar traveling salesman heuristic based on Space filling curves. En: Operation research letters. Vol. 1, No. 4 (1982); p. 121 – 125
6. BLUM, Christian y ROLI, Andrea. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. En: ACM Computing Survey. Vol. 35, No. 3, (2003); p. 268-308
7. BRITO, Julio; CAMPOS, Clara; GARCIA, Félix; BATISTA, Belén y MORENO, José A. Metaheurísticas: una revisión actualizada. Grupo de Computación Inteligente, Universidad de La Laguna, La Laguna, 2004

8. BURKARD R.E.; ÇELA E.; S.E. KARISCH y F. RENDL, QAPLIB - A Quadratic Assignment Problem Library. Disponible en: <http://www.opt.math.tu-graz.ac.at/qaplib/>
9. CHASE, R.; JACOBS, R.; y AQUILANO, N. Administración de Operaciones: Producción y Cadena de Suministro. 12 ed. México: McGraw Hill, 2009; p. 221 – 222
10. CHWIF, L.; PEREIRA BARRETTO, M.R.; y MOSCATO, L.A. A solution to the facility layout problem using simulated annealing. En: Computers in Industry. Vol. 2, No. 36 (1998); p. 125 – 132
11. DUARTE, M.; y CALDERON E. Alternativas de Solución al Problema de Distribución de Planta (FLP) de un Solo Nivel y Departamentos Iguales ó Desiguales, Por Medio Del Análisis De Métodos Metaheurísticos. Bucaramanga, 2010, 258 h. Trabajo de grado (Ingeniería Industrial). Universidad Industrial de Santander
12. Dunker, T.; Radonsb, G.; y Westkaïmpera, E. (2005). Combining evolutionary computation and dynamic programming for solving a dynamic facility layout problem. European Journal of Operational Research, 165(1), p. 55–69.
13. HERAGU, S.S. Facilities design. En: Boston: BWS. (1997); p. 3.
14. HILLIER, F., y LIEBERMAN, G. Introducción a la investigación de Operaciones. México: McGraw Hill, 2006; 8 ed. p. 617, 619
15. DUARTE, A; PANTRIGO, J.J; GALLEGO M. Metaheurísticas. 1ª ed. Madrid: Editorial Dykinson S.L., 2008; p. 9-12

16. GAREY, M. R.; y JOHNSON, D. S., Computers and intractability: A guide to the theory of NP-completeness. New York: WH Freeman, 1979
17. GLOVER, F. Future paths for integer programming and links to artificial intelligence. En: Computers & operations Research. Vol. 3, No. 1 (1986); p. 533 – 49
18. Guignard, M. (2002), Lagrangian Relaxation, Handbook of Applied Optimization, Oxford University Press, pag. 465-474
19. Kim, J.G.; y Y.D. Kim. Layout planning for facilities with fixed shapes and input and output points. En: International journal of Production Research. Vol. 18, No. 38 (2000); p. 4635 – 4653
20. KOCHHAR, Jasmit Singh.; FOSTER, Bryce. T.; y HERAGU, Sunderesh S. Hope: A genetic algorithm for the unequal area facility layout problem. En: Computers operation research. Vol. 25, No. 7/8 (1998); p. 583 – 594
21. WONG, K.Y. y KOMARUDIN. Applying Ant System for solving Unequal Area Facility Layout Problems. European Journal of Operational Research. Vol. 202, (2010); p. 730–746
22. KOOPMANS, T.C.; y BECKMANN, M. Assignment problems and the location of economic activities. Econometrica. Vol. 1, No. 25, 1957; p. 53-76
23. LEE, Kyu–Yeul.; HAN, Seong–Nam y ROH, Myung–LL. An improved genetic algorithm for facility layout problems having inner structure walls and passages. En: Computers and operations research. No. 30 (2003); p. 117 – 38

24. MELLER, R.D.; CHEN, W.; y SHERALI, H.D. Applying the sequence-pair representation to optimal facility layout designs. *Operations Research Letters*. Vol. 35, No. 5 (2007); p. 651–659
25. MELLER, R.D.; y GAU, K.Y. The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems*. Vol. 15, No. 5, 1996; p. 351–366
26. MELLER, R.D.; NARAYANAN, V.; y VANCE, P.H. Optimal facility layout design. *Operations Research Letters* Vol. 23 No. 3–5, 1999; p. 117–127
27. MIR, M.; e IMAM, M.H. A hybrid optimization approach for layout design of unequal – area facilities. En: *Computers and industrial engineering*. Vol. 1 - 2, No. 39 (2001); p. 49 – 63
28. RIOJAS CAÑARI, Alicia Cirila. Búsqueda Tabú: conceptos, algoritmo y aplicación al problema de las N-reinas. Monografía (Licenciatura en Investigación Operativa), Universidad Mayor De San Marcos, Facultad ciencias Matemáticas. Lima, 2005
29. SHAYAN, E.; y CHIHITTILAPPILLY, A. Genetic algorithm for facilities layout problems based on slicing tree structure. En: *International Journal of Production Research*. Vol. 19, No. 42 (2004); p. 4055-4067
30. TEO J.; y ABBASS H.A. An Annealing Approach to the Mating-Flight Trajectories in the Marriage in Honey Bees Optimization Algorithm, Technical Report CS04/01, School of Computer Science, University of New South Wales at ADFA, 2001

31. TOMPKINS, J.A.; WHITE, J.A.; BOZER, Y.A.; FRAZELLE, E.H.; TANCHOCO, J.M.; y TREVINO, J. Facilities planning. En: New York: Wiley. (1996).
  
32. Yannis Marinakis, Magdalene Marinaki y Nikolaos Matsatsinis. A Hybrid Clustering Algorithm based on Honey Bees Mating Optimization and Greedy Randomized Adaptive Search Procedure. Lecture Notes in Computer Science. Vol. 5313, (2008); p. 138-152
  
33. WELGAMA P.S.; y GIBSON P.R. A construction algorithm for the machine layout problem with fixed pick-up and drop-off points. En: International journal of production research. Vol. 11, No. 31 (1993); p. 2575 – 2590.

## ANEXOS

### ⊗ ANEXO A

#### CODIGO DEL ALGORITMO HBMO

#### FUNCION HONEY\_BEES\_MATTING\_OPTIMIZATION

- Inicializa variables

```
1 function vct_slnReina = HONEY_BEES_MATTING_OPTIMIZATION(vct_Valores0,mat_Fluj,mat_Dist,vct_DistAct1,vct_UnidArea,mat_LnLlenado)
2 % % % % % % % %
3 % -----
4 tic;
5 alpha = 0.9;
6 TipoArea = vct_Valores0(1);
7 VuelosApareamiento = vct_Valores0(2);
8 TamanoEspermateca = vct_Valores0(3);
9 NoZanganos = vct_Valores0(4);
10 NoCrias = vct_Valores0(5);
11 NoIteracionesTabu = vct_Valores0(6);
12 ListaTabu = vct_Valores0(7);
13 NoIteracionesEscalada = vct_Valores0(8);
14 NoIteracionesRecocido = vct_Valores0(9);
15 TemperaturaInicial = vct_Valores0(10);
16 ReduccionT = vct_Valores0(11);
17 NoTemperaturas = vct_Valores0(12);
18
19 NoDepartamentos = size(mat_Fluj);
20 NoDepartamentos = NoDepartamentos(1);
21 vct_Reina = zeros(1,NoDepartamentos + 1);
22 vct_Reina(1:NoDepartamentos) = vct_DistAct1;
23 if(TipoArea == 2)
24     mat_DistQ = CENTROIDES(vct_Reina(1:NoDepartamentos),vct_UnidArea,mat_LnLlenado);
25 else
26     mat_DistQ = mat_Dist;
27 end
```

- Selecciona aleatoriamente una trabajadora para mejorar el genotipo de la reina

```
28 vct_Reina(NoDepartamentos + 1) = FUNCION_OBJETIVO(mat_Fluj,mat_DistQ,vct_Reina(1:NoDepartamentos));
29 trnd = randi(3);
30 switch(trnd)
31     case 1
32         vct_ReinaMejorada = BUSQUEDA_TABU(vct_Reina,mat_Fluj,mat_Dist,NoIteracionesTabu,ListaTabu,TipoArea,vct_UnidArea,mat_LnLlenado);
33     case 2
34         vct_ReinaMejorada = RECOCIDO_SIMULADO(vct_Reina,mat_Fluj,mat_Dist,NoIteracionesRecocido,TemperaturaInicial,ReduccionT,NoTemperaturas,
35         TipoArea,vct_UnidArea,mat_LnLlenado);
36     case 3
37         vct_ReinaMejorada = ESCALADA_MONTANA(vct_Reina,mat_Fluj,mat_Dist,NoIteracionesEscalada,TipoArea,vct_UnidArea,mat_LnLlenado);
38 end
39 vct_Reina = vct_ReinaMejorada;
```

- Mientras no se satisfagan los criterios de parada, se generan zánganos aleatoriamente y se evalúa el genotipo del zángano

```

39 - vct_Reina = vct_ReinaMejorada;
40 - mat_SlmsParciales = zeros(VuelosApareamiento,NoDepartamentos + 2);
41 - for i = 1 : VuelosApareamiento
42 -     Energ = 0.5 + 0.5 * rand();
43 -     Veloc = 0.5 + 0.5 * rand();
44 -     mat_Zanganos = zeros(NoZanganos,NoDepartamentos + 1);
45 -     aux_Zanganos = zeros(NoZanganos,1);
46 -     for j = 1 : NoZanganos
47 -         mat_Zanganos(j,1:NoDepartamentos) = SOLUCION_ALEATORIA(NoDepartamentos);
48 -         if(TipoArea == 2)
49 -             mat_DistZ = CENTROIDES(mat_Zanganos(j,1:NoDepartamentos),vct_UnidArea,mat_LnLlenado);
50 -         else
51 -             mat_DistZ = mat_Dist;
52 -         end
53 -         mat_Zanganos(j,NoDepartamentos + 1) = FUNCION_OBJETIVO(mat_Flujo,mat_DistZ,mat_Zanganos(j,1:NoDepartamentos));
54 -     end
55 -     mat_Espermateca = zeros(TamanoEspermateca,NoDepartamentos + 1);
56 -     mEspermateca = TamanoEspermateca;
57 -     cnt = 0;
58 -     while(Energ > 0.001 && mEspermateca > 0)
59 -         Energ = Energ - (0.5 * Energ) / mEspermateca;
60 -         Veloc = Veloc * alpha;
61 -         rnd = randi(NoZanganos);
62 -         if(aux_Zanganos(rnd) == 0)
63 -             if(mat_Zanganos(rnd,NoDepartamentos + 1) < vct_Reina(NoDepartamentos + 1))
64 -                 aux_Zanga = vct_Reina;
65 -                 vct_Reina = mat_Zanganos(rnd,:);
66 -                 mat_Zanganos(rnd,:) = aux_Zanga;
67 -             end

```

- Si el zángano pasa la condición de probabilidad, entonces, si el espermateca de la reina no está llena, se adiciona su esperma al espermateca de la reina.

```

68 -         prob = exp(-abs((vct_Reina(NoDepartamentos + 1) - mat_Zanganos(rnd,NoDepartamentos + 1))/vct_Reina(NoDepartamentos + 1))/Veloc);
69 -         prob_rnd = rand();
70 -         if(prob > prob_rnd)
71 -             cnt = cnt + 1;
72 -             mat_Espermateca(cnt,:) = mat_Zanganos(rnd,:);
73 -             aux_Zanganos(rnd) = 1;
74 -             mEspermateca = mEspermateca - 1;
75 -         end
76 -     end
77 - end

```

- Se selecciona aleatoriamente un espermatozoide de la espermateca de la reina. Se desmarca la mitad del genotipo del zángano y se genera una cría, cruzando el genotipo de la reina con el espermatozoide seleccionado.

```

78 - nEspermateca = cnt;
79 - mat_Crias = zeros(NoCrias,NoDepartamentos + 1);
80 - for j = 1 : NoCrias
81 -     rnd = randi(nEspermateca);
82 -     vct_Genotipo = zeros(1,NoDepartamentos + 1);
83 -     for k = 1 : fix(NoDepartamentos / 2)
84 -         rn = randi(NoDepartamentos);
85 -         while(vct_Genotipo(rn) ~= 0)
86 -             rn = randi(NoDepartamentos);
87 -         end
88 -         vct_Genotipo(rn) = mat_Espermateca(rnd,rn);
89 -     end
90 -     for k = 1 : NoDepartamentos
91 -         if(vct_Genotipo(k) == 0)
92 -             vct_Genotipo(k) = vct_Reina(k);
93 -         end
94 -     end
95 -     vct_Genoaux = zeros(1,NoDepartamentos);
96 -     for k = 1 : NoDepartamentos
97 -         vct_Genoaux(vct_Genotipo(k)) = 1;
98 -     end
99 -     vct_Cria = vct_Genotipo;
100 -    for p = 1 : NoDepartamentos
101 -        for q = p + 1 : NoDepartamentos
102 -            if(vct_Cria(p) == vct_Cria(q))
103 -                for k = 1 : NoDepartamentos
104 -                    if(vct_Genoaux(k) == 0)
105 -                        vct_Cria(q) = k;
106 -                        vct_Genoaux(k) = 1;
107 -                        break;
108 -                    end
109 -                end
110 -            end
111 -        end
112 -    end
113 -    if(TipoArea == 2)
114 -        mat_DistC = CENTROIDES(vct_Cria(1:NoDepartamentos),vct_UnidArea,mat_LnLlenado);
115 -    else
116 -        mat_DistC = mat_Dist;
117 -    end
118 -    vct_Cria(NoDepartamentos + 1) = FUNCION_OBJETIVO(mat_Flujo,mat_DistC,vct_Cria(1:NoDepartamentos));
119 -    mat_Crias(j,:) = vct_Cria;
120 - end
121 - mat_CriasMjr = zeros(NoCrias,NoDepartamentos + 1);
122 - vct_Metas = zeros(NoCrias,1);

```

- Se selecciona una trabajadora para mejorar las crías

```

123 - for j = 1 : NoCrias
124 -     trnd = randi(3);
125 -     switch(trnd)
126 -     case 1
127 -         vct_CriaMejorada = BUSQUEDA_TABU(mat_Crias(j,:),mat_Flujo,mat_Dist,NoIteracionesTabu,ListaTabu,TipoArea,vct_UnidArea,mat_LnLlenado);
128 -     case 2
129 -         vct_CriaMejorada = RECOCIDO_SIMULADO(mat_Crias(j,:),mat_Flujo,mat_Dist,NoIteracionesRecocido,TemperaturaInicial,ReduccionT,NoTemperaturas,
130 -         TipoArea,vct_UnidArea,mat_LnLlenado);
131 -     case 3
132 -         vct_CriaMejorada = ESCALADA_MONTANA(mat_Crias(j,:),mat_Flujo,mat_Dist,NoIteracionesEscalada,TipoArea,vct_UnidArea,mat_LnLlenado);
133 -     end
134 -     vct_Metas(j) = trnd;
135 -     mat_CriasMjr(j,:) = vct_CriaMejorada;
136 - end

```

- Si la mejor cría es mejor que la reina, se reemplaza la reina por la mejor cría y se retira la mejor cría de la lista de crías

```

137 -     [MjrCria,MjrMeta] = min(mat_CriasMjr(:,NoDepartamentos + 1));
138 -     mat_SlnsParciales(i,1:NoDepartamentos + 1) = mat_CriasMjr(MjrMeta,:);
139 -     mat_SlnsParciales(i,NoDepartamentos + 2) = vct_Metas(MjrMeta);
140 -     if(MjrCria < vct_Reina(NoDepartamentos + 1))
141 -         vct_Reina = mat_CriasMjr(MjrMeta,:);
142 -     end
143 - end
144 - vct_slnReina = vct_Reina;

```

- Imprimir en la reina y las mejores crías por cada vuelo de apareamiento

```

144 - vct_slnReina = vct_Reina;
145 - toc;
146 - disp(mat_SlnsParciales);
147 - disp('La Reina final');
148 - disp(vct_Reina);

```

## ANEXO B

### PROBLEMAS DE ESTUDIO

#### 1. PROBLEMA TAI12A

##### 1.1. MATRIZ DE DISTANCIAS

0	27	85	2	1	15	11	35	11	20	21	61
27	0	80	58	21	76	72	44	85	94	90	51
85	80	0	3	48	29	90	66	41	15	83	96
2	58	3	0	74	45	65	40	54	83	14	71
1	21	48	74	0	77	36	53	37	26	87	76
15	76	29	45	77	0	91	13	29	11	77	32
11	72	90	65	36	91	0	87	67	94	79	2
35	44	66	40	53	13	87	0	10	99	56	70
11	85	41	54	37	29	67	10	0	99	60	4
20	94	15	83	26	11	94	99	99	0	56	2
21	90	83	14	87	77	79	56	60	56	0	60
61	51	96	71	76	32	2	70	4	2	60	0

##### 1.2. MATRIZ DE FLUJOS

0	21	95	82	56	41	6	25	10	4	63	6
21	0	44	40	75	79	0	89	35	9	1	85
95	44	0	84	12	0	26	91	11	35	82	26
82	40	84	0	69	56	86	45	91	59	18	76
56	75	12	69	0	39	18	57	36	61	36	21
41	79	0	56	39	0	71	11	29	82	82	6
6	0	26	86	18	71	0	71	8	77	74	30
25	89	91	45	57	11	71	0	89	76	76	40
10	35	11	91	36	29	8	89	0	93	56	1
4	9	35	59	61	82	77	76	93	0	50	4
63	1	82	18	36	82	74	76	56	50	0	36
6	85	26	76	21	6	30	40	1	4	36	0

##### 1.3. SOLUCIÓN ÓPTIMA

$Z = 112208$

Secuencia: (8,1,6,2,11,10,3,5,9,7,12,4)

## 2. PROBLEMA HAD12

### 2.1. MATRIZ DE DISTANCIAS

0	1	2	2	3	4	4	5	3	5	6	7
1	0	1	1	2	3	3	4	2	4	5	6
2	1	0	2	1	2	2	3	1	3	4	5
2	1	2	0	1	2	2	3	3	3	4	5
3	2	1	1	0	1	1	2	2	2	3	4
4	3	2	2	1	0	2	3	3	1	2	3
4	3	2	2	1	2	0	1	3	1	2	3
5	4	3	3	2	3	1	0	4	2	1	2
3	2	1	3	2	3	3	4	0	4	5	6
5	4	3	3	2	1	1	2	4	0	1	2
6	5	4	4	3	2	2	1	5	1	0	1
7	6	5	5	4	3	3	2	6	2	1	0

### 2.2. MATRIZ DE FLUJOS

0	3	4	6	8	5	6	6	5	1	4	6
3	0	6	3	7	9	9	2	2	7	4	7
4	6	0	2	6	4	4	4	2	6	3	6
6	3	2	0	5	5	3	3	9	4	3	6
8	7	6	5	0	4	3	4	5	7	6	7
5	9	4	5	4	0	8	5	5	5	7	5
6	9	4	3	3	8	0	6	8	4	6	7
6	2	4	3	4	5	6	0	1	5	5	3
5	2	2	9	5	5	8	1	0	4	5	2
1	7	6	4	7	5	4	5	4	0	7	7
4	4	3	3	6	7	6	5	5	7	0	9
6	7	6	6	7	5	7	3	2	7	9	0

### 2.3. SOLUCIÓN ÓPTIMA

Z= 826

Secuencia: (3,10,11,2,12,5,6,7,8,1,4,9)

### 3. PROBLEMA HAD16

#### 3.1. MATRIZ DE DISTANCIAS

0	1	2	2	3	4	4	5	3	5	6	7	8	9	7	8
1	0	1	1	2	3	3	4	2	4	5	6	7	8	6	7
2	1	0	2	1	2	2	3	1	3	4	5	6	7	5	6
2	1	2	0	1	2	2	3	3	3	4	5	6	7	5	6
3	2	1	1	0	1	1	2	2	2	3	4	5	6	4	5
4	3	2	2	1	0	2	3	3	1	2	3	4	5	3	4
4	3	2	2	1	2	0	1	3	1	2	3	4	5	3	4
5	4	3	3	2	3	1	0	4	2	1	2	3	4	2	3
3	2	1	3	2	3	3	4	0	4	5	6	7	8	6	7
5	4	3	3	2	1	1	2	4	0	1	2	3	4	2	3
6	5	4	4	3	2	2	1	5	1	0	1	2	3	1	2
7	6	5	5	4	3	3	2	6	2	1	0	1	2	2	3
8	7	6	6	5	4	4	3	7	3	2	1	0	1	1	2
9	8	7	7	6	5	5	4	8	4	3	2	1	0	2	1
7	6	5	5	4	3	3	2	6	2	1	2	1	2	0	1
8	7	6	6	5	4	4	3	7	3	2	3	2	1	1	0

### 3.2. MATRIZ DE FLUJOS

0	3	4	6	8	5	6	6	5	1	4	6	1	5	4	5
3	0	6	3	7	9	9	2	2	7	4	7	9	6	3	2
4	6	0	2	6	4	4	4	2	6	3	6	5	6	2	6
6	3	2	0	5	5	3	3	9	4	3	6	3	4	7	8
8	7	6	5	0	4	3	4	5	7	6	7	7	3	3	3
5	9	4	5	4	0	8	5	5	5	7	5	1	8	5	4
6	9	4	3	3	8	0	6	8	4	6	7	1	8	5	6
6	2	4	3	4	5	6	0	1	5	5	3	7	5	9	4
5	2	2	9	5	5	8	1	0	4	5	2	4	5	4	5
1	7	6	4	7	5	4	5	4	0	7	7	5	6	5	5
4	4	3	3	6	7	6	5	5	7	0	9	6	5	1	8
6	7	6	6	7	5	7	3	2	7	9	0	6	5	4	5
1	9	5	3	7	1	1	7	4	5	6	6	0	5	7	4
5	6	6	4	3	8	8	5	5	6	5	5	5	0	5	3
4	3	2	7	3	5	5	9	4	5	1	4	7	5	0	8
5	2	6	8	3	4	6	4	5	5	8	5	4	3	8	0

### 3.3. SOLUCIÓN ÓPTIMA

Z= 1860

Secuencia: (9,4,16,1,7,8,6,14,15,11,12,10,5,3,2,13)

#### 4. PROBLEMA NUG 16A

##### 4.1. MATRIZ DE DISTANCIAS

0	1	2	3	4	1	2	3	4	5	2	3	4	5	6	3
1	0	1	2	3	2	1	2	3	4	3	2	3	4	5	4
2	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5
3	2	1	0	1	4	3	2	1	2	5	4	3	2	3	6
4	3	2	1	0	5	4	3	2	1	6	5	4	3	2	7
1	2	3	4	5	0	1	2	3	4	1	2	3	4	5	2
2	1	2	3	4	1	0	1	2	3	2	1	2	3	4	3
3	2	1	2	3	2	1	0	1	2	3	2	1	2	3	4
4	3	2	1	2	3	2	1	0	1	4	3	2	1	2	5
5	4	3	2	1	4	3	2	1	0	5	4	3	2	1	6
2	3	4	5	6	1	2	3	4	5	0	1	2	3	4	1
3	2	3	4	5	2	1	2	3	4	1	0	1	2	3	2
4	3	2	3	4	3	2	1	2	3	2	1	0	1	2	3
5	4	3	2	3	4	3	2	1	2	3	2	1	0	1	4
6	5	4	3	2	5	4	3	2	1	4	3	2	1	0	5
3	4	5	6	7	2	3	4	5	6	1	2	3	4	5	0

## 4.2. MATRIZ DE DISTANCIAS

0	0	5	0	5	2	10	3	1	5	5	5	0	0	5	4
0	0	3	10	5	1	5	1	2	4	2	5	0	10	10	3
5	3	0	2	0	5	2	4	4	5	0	0	0	5	1	0
0	10	2	0	1	0	5	2	1	0	10	2	2	0	2	1
5	5	0	1	0	5	6	5	2	5	2	0	5	1	1	1
2	1	5	0	5	0	5	2	1	6	0	0	10	0	2	0
10	5	2	5	6	5	0	0	0	0	5	10	2	2	5	1
3	1	4	2	5	2	0	0	1	1	10	10	2	0	10	2
1	2	4	1	2	1	0	1	0	2	0	3	5	5	0	5
5	4	5	0	5	6	0	1	2	0	5	5	0	5	1	0
5	2	0	10	2	0	5	10	0	5	0	5	2	5	1	10
5	5	0	2	0	0	10	10	3	5	5	0	2	10	5	0
0	0	0	2	5	10	2	2	5	0	2	2	0	2	2	1
0	10	5	0	1	0	2	0	5	5	5	10	2	0	5	5
5	10	1	2	1	2	5	10	0	1	1	5	2	5	0	3
4	3	0	1	1	0	1	2	5	0	10	0	1	5	3	0

## 4.3. SOLUCIÓN ÓPTIMA

$Z = 805$

Secuencia: (9, 14, 2, 15, 16, 3, 10, 12, 8, 11, 6, 5, 7, 1, 4, 13)

## 5. PROBLEMA ROU20

### 5.1. MATRIZ DE DISTANCIAS

0	79	32	57	68	99	97	80	90	10	11	49	38	87	70	76	89	28	27	10
79	0	96	62	55	11	79	17	28	88	62	32	34	4	5	34	57	86	81	3
32	96	0	89	21	33	4	26	75	78	22	45	94	31	42	67	42	63	7	21
57	62	89	0	23	57	68	66	32	15	12	69	34	74	25	44	81	74	84	24
68	55	21	23	0	33	84	54	95	5	15	10	81	3	51	85	63	23	13	52
99	11	33	57	33	0	14	86	29	53	97	75	19	53	23	95	99	0	43	58
97	79	4	68	84	14	0	95	74	15	85	56	43	20	14	36	86	58	2	14
80	17	26	66	54	86	95	0	34	38	79	27	21	43	6	85	77	69	49	21
90	28	75	32	95	29	74	34	0	22	80	43	14	13	92	22	5	75	37	90
10	88	78	15	5	53	15	38	22	0	41	20	14	49	42	75	73	18	1	56
11	62	22	12	15	97	85	79	80	41	0	55	29	9	78	73	34	92	6	27
49	32	45	69	10	75	56	27	43	20	55	0	86	24	88	97	67	78	56	90
38	34	94	34	81	19	43	21	14	14	29	86	0	87	62	11	63	73	79	93
87	4	31	74	3	53	20	43	13	49	9	24	87	0	46	66	50	1	15	36
70	5	42	25	51	23	14	6	92	42	78	88	62	46	0	69	72	2	40	43
76	34	67	44	85	95	36	85	22	75	73	97	11	66	69	0	87	82	54	18
89	57	42	81	63	99	86	77	5	73	34	67	63	50	72	87	0	80	74	90
28	86	63	74	23	0	58	69	75	18	92	78	73	1	2	82	80	0	14	63
27	81	7	84	13	43	2	49	37	1	6	56	79	15	40	54	74	14	0	12
10	3	21	24	52	58	14	21	90	56	27	90	93	36	43	18	90	63	12	0

## 5.2. MATRIZ DE FLUJOS

0	78	22	43	86	8	99	5	32	89	19	69	33	89	46	66	57	7	17	26
78	0	2	81	24	83	92	36	31	73	96	5	57	2	39	29	26	58	45	80
22	2	0	38	50	32	66	73	6	8	68	16	21	1	70	17	54	17	50	20
43	81	38	0	53	75	40	8	63	30	30	10	47	36	34	37	33	36	78	92
86	24	50	53	0	41	29	68	52	83	51	52	57	75	10	37	98	42	58	14
8	83	32	75	41	0	68	44	0	56	82	23	84	90	8	4	63	69	75	65
99	92	66	40	29	68	0	46	64	79	4	64	85	99	29	1	39	80	94	93
5	36	73	8	68	44	46	0	74	19	56	34	11	54	52	75	3	28	6	74
32	31	6	63	52	0	64	74	0	2	14	95	81	98	15	64	31	92	26	77
89	73	8	30	83	56	79	19	2	0	43	49	80	71	35	60	88	52	18	50
19	96	68	30	51	82	4	56	14	43	0	8	74	53	61	66	57	88	26	62
69	5	16	10	52	23	64	34	95	49	8	0	19	21	10	6	53	56	58	37
33	57	21	47	57	84	85	11	81	80	74	19	0	6	65	85	59	57	0	46
89	2	1	36	75	90	99	54	98	71	53	21	6	0	61	96	58	8	17	43
46	39	70	34	10	8	29	52	15	35	61	10	65	61	0	42	93	61	78	90
66	29	17	37	37	4	1	75	64	60	66	6	85	96	42	0	79	21	95	3
57	26	54	33	98	63	39	3	31	88	57	53	59	58	93	79	0	61	60	20
7	58	17	36	42	69	80	28	92	52	88	56	57	8	61	21	61	0	80	39
17	45	50	78	58	75	94	6	26	18	26	58	0	17	78	95	60	80	0	80
26	80	20	92	14	65	93	74	77	50	62	37	46	43	90	3	20	39	80	0

## 5.3. SOLUCIÓN ÓPTIMA

Z= 362761

Secuencia: (1,19,2,14,10,16,11,20,9,5,7,4,8,18,15,3,12,17,13,6)

## 6. PROBLEMA NUG20

### 6.1. MATRIZ DE DISTANCIAS

0	1	2	3	4	1	2	3	4	5	2	3	4	5	6	3	4	5	6	7
1	0	1	2	3	2	1	2	3	4	3	2	3	4	5	4	3	4	5	6
2	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5	4	3	4	5
3	2	1	0	1	4	3	2	1	2	5	4	3	2	3	6	5	4	3	4
4	3	2	1	0	5	4	3	2	1	6	5	4	3	2	7	6	5	4	3
1	2	3	4	5	0	1	2	3	4	1	2	3	4	5	2	3	4	5	6
2	1	2	3	4	1	0	1	2	3	2	1	2	3	4	3	2	3	4	5
3	2	1	2	3	2	1	0	1	2	3	2	1	2	3	4	3	2	3	4
4	3	2	1	2	3	2	1	0	1	4	3	2	1	2	5	4	3	2	3
5	4	3	2	1	4	3	2	1	0	5	4	3	2	1	6	5	4	3	2
2	3	4	5	6	1	2	3	4	5	0	1	2	3	4	1	2	3	4	5
3	2	3	4	5	2	1	2	3	4	1	0	1	2	3	2	1	2	3	4
4	3	2	3	4	3	2	1	2	3	2	1	0	1	2	3	2	1	2	3
5	4	3	2	3	4	3	2	1	2	3	2	1	0	1	4	3	2	1	2
6	5	4	3	2	5	4	3	2	1	4	3	2	1	0	5	4	3	2	1
3	4	5	6	7	2	3	4	5	6	1	2	3	4	5	0	1	2	3	4
4	3	4	5	6	3	2	3	4	5	2	1	2	3	4	1	0	1	2	3
5	4	3	4	5	4	3	2	3	4	3	2	1	2	3	2	1	0	1	2
6	5	4	3	4	5	4	3	2	3	4	3	2	1	2	3	2	1	0	1
7	6	5	4	3	6	5	4	3	2	5	4	3	2	1	4	3	2	1	0

## 6.2. MATRIZ DE DISTANCIAS

0	0	5	0	5	2	10	3	1	5	5	5	0	0	5	4	4	0	0	1
0	0	3	10	5	1	5	1	2	4	2	5	0	10	10	3	0	5	10	5
5	3	0	2	0	5	2	4	4	5	0	0	0	5	1	0	0	5	0	0
0	10	2	0	1	0	5	2	1	0	10	2	2	0	2	1	5	2	5	5
5	5	0	1	0	5	6	5	2	5	2	0	5	1	1	1	5	2	5	1
2	1	5	0	5	0	5	2	1	6	0	0	10	0	2	0	1	0	1	5
10	5	2	5	6	5	0	0	0	0	5	10	2	2	5	1	2	1	0	10
3	1	4	2	5	2	0	0	1	1	10	10	2	0	10	2	5	2	2	10
1	2	4	1	2	1	0	1	0	2	0	3	5	5	0	5	0	0	0	2
5	4	5	0	5	6	0	1	2	0	5	5	0	5	1	0	0	5	5	2
5	2	0	10	2	0	5	10	0	5	0	5	2	5	1	10	0	2	2	5
5	5	0	2	0	0	10	10	3	5	5	0	2	10	5	0	1	1	2	5
0	0	0	2	5	10	2	2	5	0	2	2	0	2	2	1	0	0	0	5
0	10	5	0	1	0	2	0	5	5	5	10	2	0	5	5	1	5	5	0
5	10	1	2	1	2	5	10	0	1	1	5	2	5	0	3	0	5	10	10
4	3	0	1	1	0	1	2	5	0	10	0	1	5	3	0	0	0	2	0
4	0	0	5	5	1	2	5	0	0	0	1	0	1	0	0	0	5	2	0
0	5	5	2	2	0	1	2	0	5	2	1	0	5	5	0	5	0	1	1
0	10	0	5	5	1	0	2	0	5	2	2	0	5	10	2	2	1	0	6
1	5	0	5	1	5	10	10	2	2	5	5	5	0	10	0	0	1	6	0

## 6.3. SOLUCIÓN ÓPTIMA

Z= 1285

Secuencia: (18,14,10,3,9,4,2,12,11,16,19,15,20,8,13,17,5,7,1,6)

## 7. PROBLEMA TAI25A

### 7.1. MATRIZ DE DISTANCIAS

0	92	69	62	32	97	5	39	50	82	93	71	35	14	20	74	49	50	37	79	19	51	70	42	26
92	0	79	98	60	35	9	96	70	21	37	37	67	93	93	39	2	52	26	90	26	1	68	93	7
69	79	0	94	19	54	37	0	20	12	11	66	84	80	1	55	9	21	12	65	7	17	51	84	87
62	98	94	0	2	27	82	71	71	40	93	27	93	92	34	39	31	26	1	87	72	59	97	46	62
32	60	19	2	0	12	91	63	70	1	22	49	24	58	62	49	94	92	63	13	45	22	69	70	18
97	35	54	27	12	0	1	44	32	3	72	99	34	45	18	96	82	79	75	83	43	26	81	39	66
5	9	37	82	91	1	0	22	71	58	42	91	56	27	56	28	64	98	69	69	93	35	41	67	12
39	96	0	71	63	44	22	0	33	1	71	72	26	86	67	67	24	10	46	56	56	69	53	48	74
50	70	20	71	70	32	71	33	0	43	94	29	64	97	38	12	99	51	87	76	71	47	47	9	71
82	21	12	40	1	3	58	1	43	0	20	81	99	45	56	15	99	64	57	84	83	61	25	41	8
93	37	11	93	22	72	42	71	94	20	0	46	52	22	33	68	52	28	75	99	6	85	65	47	90
71	37	66	27	49	99	91	72	29	81	46	0	24	79	70	28	19	87	80	4	21	34	12	20	67
35	67	84	93	24	34	56	26	64	99	52	24	0	24	94	10	64	52	92	49	68	65	66	28	81
14	93	80	92	58	45	27	86	97	45	22	79	24	0	17	58	99	90	70	17	57	37	2	37	3
20	93	1	34	62	18	56	67	38	56	33	70	94	17	0	43	20	68	35	60	23	48	40	10	42
74	39	55	39	49	96	28	67	12	15	68	28	10	58	43	0	83	82	27	49	15	34	50	42	26
49	2	9	31	94	82	64	24	99	99	52	19	64	99	20	83	0	33	23	16	69	97	13	35	80
50	52	21	26	92	79	98	10	51	64	28	87	52	90	68	82	33	0	40	55	30	22	76	55	56
37	26	12	1	63	75	69	46	87	57	75	80	92	70	35	27	23	40	0	91	74	82	96	2	13
79	90	65	87	13	83	69	56	76	84	99	4	49	17	60	49	16	55	91	0	4	4	35	48	29
19	26	7	72	45	43	93	56	71	83	6	21	68	57	23	15	69	30	74	4	0	42	56	3	30
51	1	17	59	22	26	35	69	47	61	85	34	65	37	48	34	97	22	82	4	42	0	60	81	37
70	68	51	97	69	81	41	53	47	25	65	12	66	2	40	50	13	76	96	35	56	60	0	42	3
42	93	84	46	70	39	67	48	9	41	47	20	28	37	10	42	35	55	2	48	3	81	42	0	17
26	7	87	62	18	66	12	74	71	8	90	67	81	3	42	26	80	56	13	29	30	37	3	17	0

## 7.2. MATRIZ DE FLUJOS

0	21	95	82	56	41	6	25	10	4	63	6	44	40	75	79	0	89	35	9	1	85	84	12	0
21	0	26	91	11	35	82	26	69	56	86	45	91	59	18	76	39	18	57	36	61	36	21	71	11
95	26	0	29	82	82	6	71	8	77	74	30	89	76	76	40	93	56	1	50	4	36	27	85	2
82	91	29	0	1	15	11	35	11	20	21	61	80	58	21	76	72	44	85	94	90	51	3	48	29
56	11	82	1	0	90	66	41	15	83	96	74	45	65	40	54	83	14	71	77	36	53	37	26	87
41	35	82	15	90	0	76	91	13	29	11	77	32	87	67	94	79	2	10	99	56	70	99	60	4
6	82	6	11	66	76	0	56	2	60	72	74	46	13	20	86	4	77	15	89	48	14	89	44	59
25	26	71	35	41	91	56	0	22	57	63	6	0	62	41	62	46	25	75	76	40	66	58	30	68
10	69	8	11	15	13	2	22	0	78	91	13	59	49	85	84	8	38	41	56	39	53	77	50	30
4	56	77	20	83	29	60	57	78	0	58	55	19	85	52	34	53	40	69	12	85	72	7	49	46
63	86	74	21	96	11	72	63	91	58	0	87	58	17	68	27	21	6	67	26	82	44	35	3	62
6	45	30	61	74	77	74	6	13	55	87	0	8	51	1	91	39	87	72	45	96	7	87	68	33
44	91	89	80	45	32	46	0	59	19	58	8	0	3	21	90	45	47	25	30	43	97	33	35	61
40	59	76	58	65	87	13	62	49	85	17	51	3	0	42	36	43	7	84	6	0	0	48	62	59
75	18	76	21	40	67	20	41	85	52	68	1	21	42	0	29	94	82	29	3	3	51	67	39	15
79	76	40	76	54	94	86	62	84	34	27	91	90	36	29	0	66	42	23	62	62	28	76	66	82
0	39	93	72	83	79	4	46	8	53	21	39	45	43	94	66	0	98	35	15	17	77	44	26	76
89	18	56	44	14	2	77	25	38	40	6	87	47	7	82	42	98	0	86	60	62	62	83	91	57
35	57	1	85	71	10	15	75	41	69	67	72	25	84	29	23	35	86	0	62	36	2	2	43	65
9	36	50	94	77	99	89	76	56	12	26	45	30	6	3	62	15	60	62	0	37	49	61	5	34
1	61	4	90	36	56	48	40	39	85	82	96	43	0	3	62	17	62	36	37	0	53	96	82	48
85	36	36	51	53	70	14	66	53	72	44	7	97	0	51	28	77	62	2	49	53	0	28	31	75
84	21	27	3	37	99	89	58	77	7	35	87	33	48	67	76	44	83	2	61	96	28	0	1	95
12	71	85	48	26	60	44	30	50	49	3	68	35	62	39	66	26	91	43	5	82	31	1	0	7
0	11	2	29	87	4	59	68	30	46	62	33	61	59	15	82	76	57	65	34	48	75	95	7	0

## 7.3. SOLUCIÓN ÓPTIMA

Z= 583628

Secuencia: (9,4,6,11,5,1,15,10,14,3,17,12,19,18,23,8,21,2,22,7,16,20,24,25,13)

## 8. PROBLEMA NUG25

### 8.1. MATRIZ DE DISTANCIAS

```
0 1 2 3 4 1 2 3 4 5 2 3 4 5 6 3 4 5 6 7 4 5 6 7 8
1 0 1 2 3 2 1 2 3 4 3 2 3 4 5 4 3 4 5 6 5 4 5 6 7
2 1 0 1 2 3 2 1 2 3 4 3 2 3 4 5 4 3 4 5 6 5 4 5 6
3 2 1 0 1 4 3 2 1 2 5 4 3 2 3 6 5 4 3 4 7 6 5 4 5
4 3 2 1 0 5 4 3 2 1 6 5 4 3 2 7 6 5 4 3 8 7 6 5 4
1 2 3 4 5 0 1 2 3 4 1 2 3 4 5 2 3 4 5 6 3 4 5 6 7
2 1 2 3 4 1 0 1 2 3 2 1 2 3 4 3 2 3 4 5 4 3 4 5 6
3 2 1 2 3 2 1 0 1 2 3 2 1 2 3 4 3 2 3 4 5 4 3 4 5
4 3 2 1 2 3 2 1 0 1 4 3 2 1 2 5 4 3 2 3 6 5 4 3 4
5 4 3 2 1 4 3 2 1 0 5 4 3 2 1 6 5 4 3 2 7 6 5 4 3
2 3 4 5 6 1 2 3 4 5 0 1 2 3 4 1 2 3 4 5 2 3 4 5 6
3 2 3 4 5 2 1 2 3 4 1 0 1 2 3 2 1 2 3 4 3 2 3 4 5
4 3 2 3 4 3 2 1 2 3 2 1 0 1 2 3 2 1 2 3 4 3 2 3 4
5 4 3 2 3 4 3 2 1 2 3 2 1 0 1 4 3 2 1 2 5 4 3 2 3
6 5 4 3 2 5 4 3 2 1 4 3 2 1 0 5 4 3 2 1 6 5 4 3 2
3 4 5 6 7 2 3 4 5 6 1 2 3 4 5 0 1 2 3 4 1 2 3 4 5
4 3 4 5 6 3 2 3 4 5 2 1 2 3 4 1 0 1 2 3 2 1 2 3 4
5 4 3 4 5 4 3 2 3 4 3 2 1 2 3 2 1 0 1 2 3 2 1 2 3
6 5 4 3 4 5 4 3 2 3 4 3 2 1 2 3 2 1 0 1 4 3 2 1 2
7 6 5 4 3 6 5 4 3 2 5 4 3 2 1 4 3 2 1 0 5 4 3 2 1
4 5 6 7 8 3 4 5 6 7 2 3 4 5 6 1 2 3 4 5 0 1 2 3 4
5 4 5 6 7 4 3 4 5 6 3 2 3 4 5 2 1 2 3 4 1 0 1 2 3
6 5 4 5 6 5 4 3 4 5 4 3 2 3 4 3 2 1 2 3 2 1 0 1 2
7 6 5 4 5 6 5 4 3 4 5 4 3 2 3 4 3 2 1 2 3 2 1 0 1
8 7 6 5 4 7 6 5 4 3 6 5 4 3 2 5 4 3 2 1 4 3 2 1 0
```

## 8.2. MATRIZ DE FLUJOS

0	3	2	0	0	10	5	0	5	2	0	0	2	0	5	3	0	1	10	0	2	1	1	1	0	
3	0	4	0	10	0	0	2	2	1	5	0	0	0	0	0	1	6	1	0	2	2	5	1	10	
2	4	0	3	4	5	5	5	1	4	0	4	0	4	0	3	2	5	5	2	0	0	3	1	0	
0	0	3	0	0	0	2	2	0	6	2	5	2	5	1	1	1	2	2	4	2	0	2	2	5	
0	10	4	0	0	2	0	0	0	0	0	0	0	0	2	0	0	2	0	5	0	2	1	0	2	
10	0	5	0	2	0	10	10	5	10	6	0	0	10	2	10	1	5	5	2	5	0	2	0	1	
5	0	5	2	0	10	0	1	3	5	0	0	2	4	5	10	6	0	5	5	5	0	5	5	0	
0	2	5	2	0	10	1	0	10	2	5	2	0	3	0	0	0	4	0	5	0	5	2	2	5	
5	2	1	0	0	5	3	10	0	5	6	0	1	5	5	5	2	3	5	0	2	10	10	1	5	
2	1	4	6	0	10	5	2	5	0	0	1	2	1	0	0	0	6	6	4	5	3	2	2		
0	5	0	2	0	6	0	5	6	0	0	2	0	4	2	1	0	6	2	1	5	0	0	1	5	
0	0	4	5	0	0	0	2	0	1	2	0	2	1	0	3	10	0	0	4	0	0	4	2	5	
2	0	0	2	0	0	2	0	1	2	0	2	0	4	5	0	1	0	5	0	0	0	5	1	1	
0	0	4	5	0	10	4	3	5	1	4	1	4	0	0	0	2	2	0	2	5	0	5	2	5	
5	0	0	1	2	2	5	0	5	0	2	0	5	0	0	2	0	0	0	6	3	5	0	0	5	
3	0	3	1	0	10	10	0	5	0	1	3	0	0	2	0	0	5	5	1	5	2	1	2	10	
0	1	2	1	0	1	6	0	2	0	0	10	1	2	0	0	0	5	2	1	1	5	6	5	5	
1	6	5	2	2	5	0	4	3	0	6	0	0	2	0	5	5	0	4	0	0	0	0	5	0	
10	1	5	2	0	5	5	0	5	6	2	0	5	0	0	5	2	4	0	5	4	4	4	5	0	2
0	0	2	4	5	2	5	5	0	6	1	4	0	2	6	1	1	0	5	0	4	4	1	0	2	
2	2	0	2	0	5	5	0	2	4	5	0	0	5	3	5	1	0	4	4	0	1	0	10	1	
1	2	0	0	2	0	0	5	10	5	0	0	0	0	5	2	5	0	4	4	1	0	0	0	0	
1	5	3	2	1	2	5	2	10	3	0	4	5	5	0	1	6	0	5	1	0	0	0	0	0	
1	1	1	2	0	0	5	2	1	2	1	2	1	2	0	2	5	5	0	0	10	0	0	0	2	
0	10	0	5	2	1	0	5	5	2	5	5	1	5	5	10	5	0	2	2	1	0	0	2	0	

## 8.3. SOLUCIÓN ÓPTIMA

Z= 1872

Secuencia: (5,11,20,15,22,2,25,8,9,1,18,16,3,6,19,24,21,14,7,10,17,12,4,23,13)

## ANEXO C

### DISEÑO DE EXPERIMENTOS

#### 1. RESULTADOS PROBLEMAS DE REFERENCIA

A continuación se presentan los resultados arrojados por el algoritmo HBMO en la solución de los 8 problemas.

##### 1.1. PROBLEMAS DE 12 DEPARTAMENTOS

Resultados para problema de 12 Departamentos (tai12a)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	112208	112208	112208	112208
6	112208	112208	112208	112208
9	112208	112208	112208	112208
12	112208	112208	112208	112208

Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=12 (tai12a)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	0,000%	0,000%	0,000%	0,000%
6	0,000%	0,000%	0,000%	0,000%
9	0,000%	0,000%	0,000%	0,000%
12	0,000%	0,000%	0,000%	0,000%

Resultados para problema de 12 Departamentos (had12)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	828	826	826	826
6	826	828	826	826
9	826	826	826	826
12	826	826	826	826

Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=12 (had12)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	0,242%	0,000%	0,000%	0,000%
6	0,000%	0,242%	0,000%	0,000%
9	0,000%	0,000%	0,000%	0,000%
12	0,000%	0,000%	0,000%	0,000%

## 1.2. PROBLEMAS DE 16 DEPARTAMENTOS

Resultados para problema de 16 Departamentos (had16)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	1861	1860	1861	1861
6	1861	1860	1861	1860
9	1860	1861	1860	1860
12	1861	1861	1860	1860

Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=16 (had16)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	0,054%	0,000%	0,054%	0,054%
6	0,054%	0,000%	0,054%	0,000%
9	0,000%	0,054%	0,000%	0,000%
12	0,054%	0,054%	0,000%	0,000%

**Resultados para problema de 16 Departamentos (nug16a)**

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	817	817	805	815
6	817	816	817	805
9	815	805	805	805
12	805	811	805	805

**Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=16 (nug16a)**

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	1,491%	1,491%	0,000%	1,242%
6	1,491%	1,366%	1,491%	0,000%
9	1,242%	0,000%	0,000%	0,000%
12	0,000%	0,745%	0,000%	0,000%

### 1.3. PROBLEMAS DE 20 DEPARTAMENTOS

**Resultados para problema de 20 Departamentos (rou20)**

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	369904	369301	369098	369278
6	368956	369193	369518	368956
9	367971	364378	366650	364378
12	364378	367971	363406	363406

**Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=20 (rou20)**

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	1,969%	1,803%	1,747%	1,796%
6	1,708%	1,773%	1,863%	1,708%
9	1,436%	0,446%	1,072%	0,446%
12	0,446%	1,436%	0,178%	0,178%

Resultados para problema de 20 Departamentos (nug20)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	1326	1309	1301	1303
6	1303	1318	1298	1298
9	1301	1285	1298	1287
12	1301	1287	1285	1285

Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=20 (nug20)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	3,191%	1,868%	1,245%	1,401%
6	1,401%	2,568%	1,012%	1,012%
9	1,245%	0,000%	1,012%	0,156%
12	1,245%	0,156%	0,000%	0,000%

1.4. PROBLEMAS DE 25 DEPARTAMENTOS

Resultados para problema de 25 Departamentos (tai25a)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$	$Z_{HBMO}$
3	604939	602592	601619	601075
6	607174	598793	600930	600183
9	598793	595560	600545	600183
12	595250	598793	594197	593629

Porcentaje de diferencia entre  $Z_{HBMO}$  y  $Z_{OPT}$  - Departamentos=25 (tai25a)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$	$\Delta Z\%$
3	3,651%	3,249%	3,083%	2,989%
6	4,034%	2,598%	2,965%	2,837%
9	2,598%	2,044%	2,899%	2,837%
12	1,991%	2,598%	1,811%	1,714%

### Resultados para problema de 25 Departamentos (nug25)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	Z <sub>HBMO</sub>	Z <sub>HBMO</sub>	Z <sub>HBMO</sub>	Z <sub>HBMO</sub>
3	1963	1958	1899	1873
6	1976	1891	1875	1872
9	1891	1875	1873	1872
12	1892	1877	1875	1872

### Porcentaje de diferencia entre Z<sub>HBMO</sub> y Z<sub>OPT</sub> - Departamentos=25 (nug25)

No. CRÍAS	5	10	15	20
No. VUELOS DE APAREAMIENTO	ΔZ%	ΔZ%	ΔZ%	ΔZ%
3	4,861%	4,594%	1,442%	0,053%
6	5,556%	1,015%	0,160%	0,000%
9	1,015%	0,160%	0,053%	0,000%
12	1,068%	0,267%	0,160%	0,000%

## 2. DISEÑO DE EXPERIMENTOS

### 2.1. TABLAS ANOVA

#### Anova – Problemas de 20 Departamentos

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,002 <sup>a</sup>	15	,000	5,489	,001
Intersección	,004	1	,004	218,956	,000
NumVuelo	,001	3	,000	18,831	,000
NumCrias	,000	3	8,33E-005	4,163	,023
NumVuelo * NumCrias	,000	9	2,97E-005	1,484	,235
Error	,000	16	2,00E-005		
Total	,006	32			
Total corregida	,002	31			

<sup>a</sup>. R cuadrado = ,837 (R cuadrado corregida = ,685)

## Anova – Problemas de 25 Departamentos

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	15	,000	1,339	,284
Intersección	,013	1	,013	59,533	,000
NumVuelo	,002	3	,001	2,551	,092
NumCrias	,002	3	,001	2,304	,116
NumVuelo * NumCrias	,001	9	,000	,613	,770
Error	,003	16	,000		
Total	,021	32			
Total corregida	,008	31			

a. R cuadrado = ,557 (R cuadrado corregida = ,141)

## 2.2. FACTOR PROBLEMA

### 2.2.1. Escenario de 25 Departamentos

- *Factores: Numero de vuelos – Problema*

## Tabla ANOVA – Problemas de 25 Departamentos

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	7	,001	3,091	,018
Intersección	,013	1	,013	75,297	,000
NumVuelos	,002	3	,001	3,227	,040
Problema	,002	1	,002	10,050	,004
NumVuelos * Problema	,000	3	,000	,635	,600
Error	,004	24	,000		
Total	,021	32			
Total corregida	,008	31			

a. R cuadrado = ,474 (R cuadrado corregida = ,321)

Gráfico MediaDesvOpt Vs. Factor Problema - (25 Dptos)

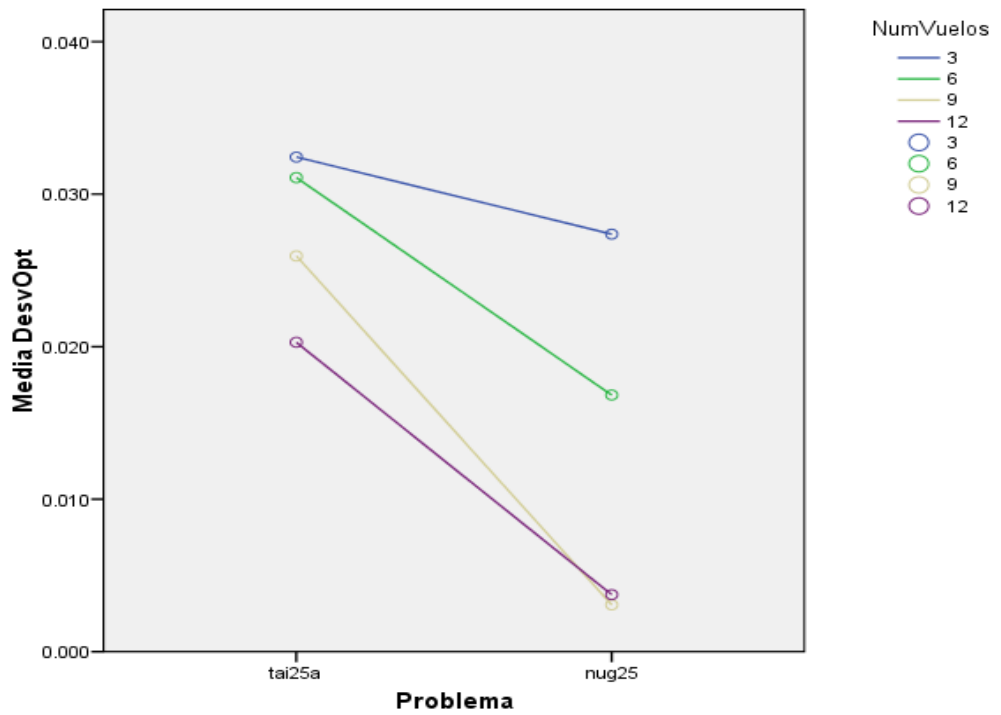
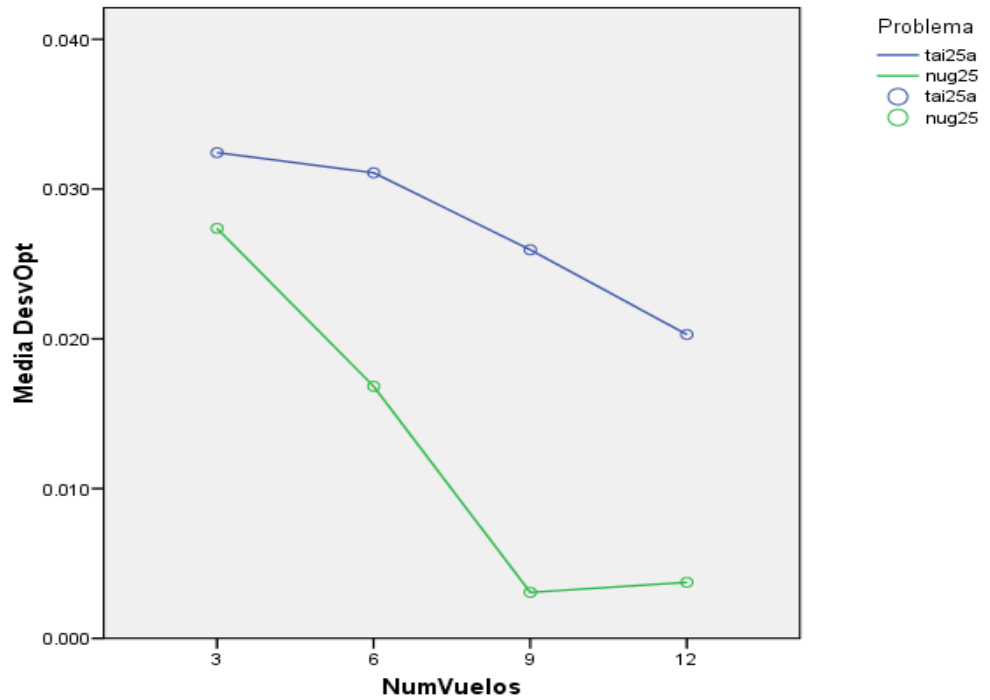


Gráfico MediaDesvOpt Vs. Número de vuelos de apareamiento - (25 Dptos)



- Factores: Numero de crías – Problema

**Tabla ANOVA – Problemas de 25 Departamentos**

Variable dependiente: DesvOpt

Fuente	Suma de cuadrados tipo III	gl	Media cuadrática	F	Significación
Modelo corregido	,004 <sup>a</sup>	7	,001	3,729	,007
Intersección	,013	1	,013	82,668	,000
NumCrias	,002	3	,001	3,200	,041
Problema	,002	1	,002	11,034	,003
NumCrias * Problema	,001	3	,000	1,823	,170
Error	,004	24	,000		
Total	,021	32			
Total corregida	,008	31			

a. R cuadrado = ,521 (R cuadrado corregida = ,381)

**Gráfico MediaDesvOpt Vs. Factor Problema - (25 Dptos)**

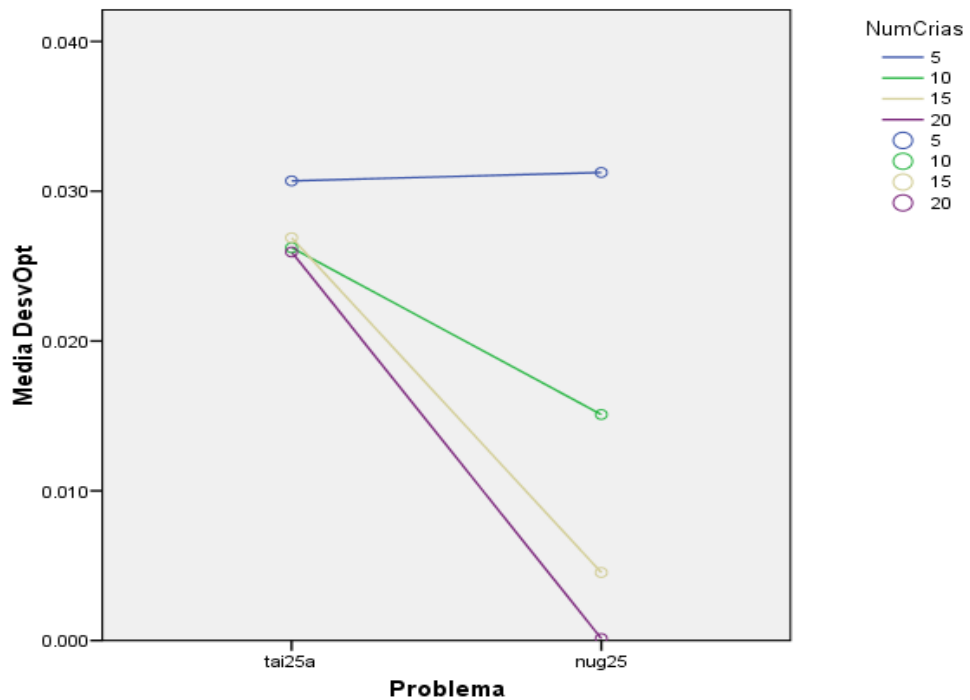
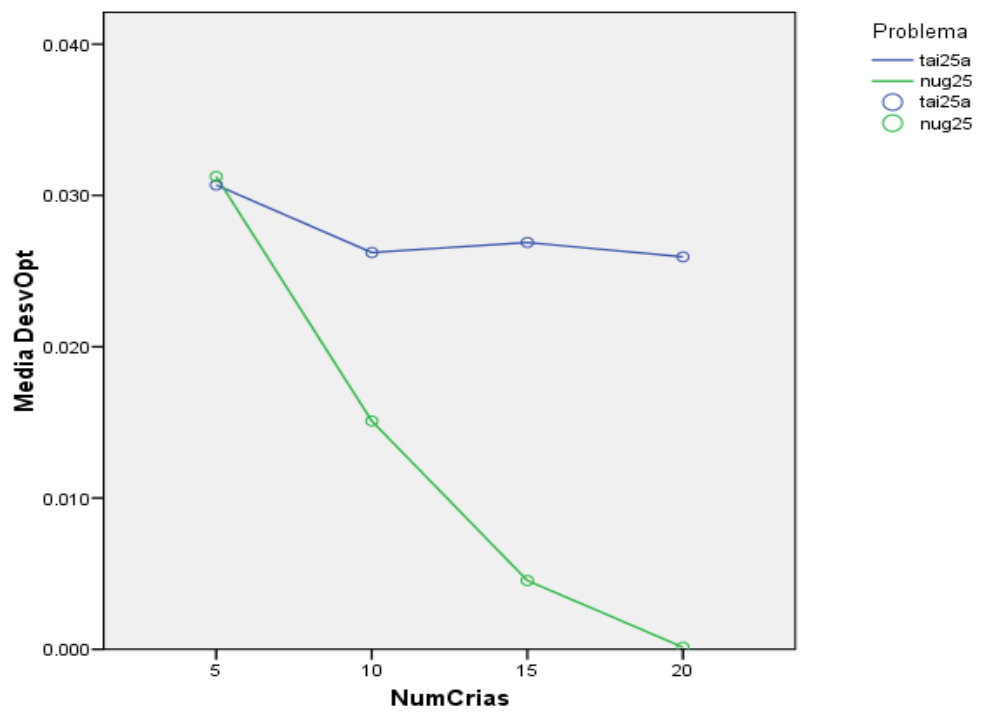


Gráfico MediaDesvOpt Vs. Número de crías - (25 Dptos)



**ANEXO D**  
**ARTÍCULO ACADÉMICO**

**(EN LA SIGUIENTE PAGINA)**

# **“METAHEURISTICA HONEY-BEES MATTING OPTIMIZATION (HBMO) APLICADA A LA SOLUCION DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTA (FLP) DE UN SOLO NIVEL Y DEPARTAMENTOS DE AREAS IGUALES O DESIGUALES”**

**DIANA CAROLINA MENDOZA MORENO**

Ingeniería Industrial

Universidad Industrial de Santander

dianac614@hotmail.com

**GINNA MARCELA COTE RAMIREZ**

Ingeniería Industrial

Universidad Industrial de Santander

ginna\_cote@hotmail.com

## **RESUMEN**

La distribución de plantas es un problema muy común de optimización combinatoria, lo que hace difícil su tratamiento por métodos convencionales. Este trabajo trata el problema distribución de planta (FLP) de un solo nivel y departamentos de áreas iguales o desiguales. El enfoque utilizado para darle solución es mediante la metaheurística Honey-Bees Matting Optimization (HBMO).

A partir de la investigación y documentación del problema, se estructura de acuerdo a las características de la planta, la formulación y modelación matemática, así como el diseño del algoritmo para la solución del mismo. Se desarrolla una herramienta en MATLAB, capaz de solucionar el problema documentado usando el algoritmo HBMO. Se resuelve un conjunto de problemas y posteriormente se examinan los resultados obtenidos mediante un diseño de experimentos con el fin de concluir acerca del desempeño de la herramienta diseñada y de la metaheurística aplicada, respecto a otros métodos utilizados en la literatura.

**PALABRAS CLAVE:** Distribución de planta (FLP), metaheurísticas, Honey Bees Matting Optimization (HBMO).

## **ABSTRACT**

The facility layout problem is a very common problem of combinatorial optimization, which makes it difficult to treat by conventional methods. This paper addresses the facility layout problem (FLP) with one floor and departments with equal or unequal areas. The approach used for solving is by metaheuristic Honey-Bees Optimization Matting (HBMO).

From the research and documentation of the problem, is structured according to the characteristics of the plant, the formulation and mathematical modelling as well as algorithm design for solving it. It develops a MATLAB tool, able to solving the problem using the algorithm HBMO. It solves a set of problems and then discusses the results by the experimental design in order to conclude about the performance of the tool designed and metaheuristic applied with respect to other methods used in the literature.

**KEY WORDS:** Facility layout problem (FLP), metaheuristic, Honey Bees Matting Optimization (HBMO).

## 1. INTRODUCCIÓN

El Problema de Distribución de Plantas (FLP), consiste en ubicar de forma óptima los departamentos, ya sean oficinas, máquinas o centros de trabajo en un área disponible. Sin embargo, la distribución óptima de una planta puede ser interpretada como un problema de optimización combinatoria, cuyo espacio de soluciones está conformado por todas las distribuciones factibles a realizarse, por lo tanto a medida que aumenta el número de departamentos el problema se hace exponencialmente complejo.

Encontrar una buena distribución de planta ha sido un área de la investigación de operaciones que ha despertado un gran interés de estudio a través del tiempo, debido a que un diseño de planta bien hecho contribuye a reducir tiempos y minimizar costos dentro de la empresa. Este proyecto de investigación se concentra en el estudio de la metaheurística Honey-Bees Matting Optimization (HBMO), la definición de sus parámetros y su aplicación al problema planteado mediante el diseño de un algoritmo codificado en MATLAB versión 2009, el software suministra herramientas suficientes para dar solución al problema, cuenta con un lenguaje de programación sencillo y buen desempeño en el procesamiento de datos.

Para conocer el desempeño de la herramienta diseñada y la eficiencia del método de solución elegido, se resuelve un conjunto de problemas históricos y posteriormente se examinan los resultados obtenidos mediante un diseño de experimentos.

## 2. CARACTERÍSTICAS DE LA PLANTA

El problema de distribución de plantas, es la determinación de la disposición más eficiente de los departamentos dentro de una instalación. Departamento se refiere a todo centro de trabajo, oficina, máquina o grupo de máquinas que pertenezcan a la empresa, mientras instalación hace referencia al espacio físico disponible en la planta, en el que será ubicado uno de los departamentos definidos.

Hay diferentes ejemplos de distribución tales como plantas industriales, edificios administrativos, oficinas, instalaciones de servicios, etc. La disposición eficiente de los recursos (máquinas, departamentos o personas) dentro de una instalación permite que el flujo del proceso sea coordinado y no existan retrocesos en el proceso productivo. Para plantas manufactureras, el costo de manejo de materiales es la medida más importante para determinar la eficiencia de un diseño. Según Tompkins [31], el

costo de manejo de materiales representa del 20 al 50% del costo total de operación y del 15 al 70% del costo total de producción de un producto. El costo de manejo de materiales se determina con base en los flujos de los materiales a través del proceso y las distancias entre los distintos departamentos.

En la actualidad las empresas deben considerar que el diseño de una planta debe estar en capacidad de adaptarse fácilmente a los cambios que competitivamente exige el entorno; como aumento de la demanda, mejoras en el diseño de un producto, etc. Por lo tanto se debe prestar atención a la naturaleza dinámica que debe tener la distribución de plantas, esto se refiere que el flujo de proceso varía en diferentes periodos de tiempo, es decir, que el diseño varía con el tiempo. Este problema se conoce como el problema de distribución de planta dinámico (FDLP). Sin embargo cuando la empresa establece que no se van a presentar dichos cambios o que estos no son significativos, se puede realizar una distribución estática en la cual el diseño se hace para un largo periodo de tiempo.

## **2.1. CARACTERÍSTICAS ESPACIALES DE LOS DEPARTAMENTOS**

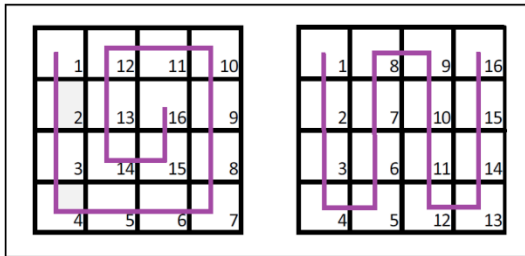
Un departamento está definido físicamente por su área o por su forma, dependiendo del proceso productivo que se llevara a cabo, las máquinas necesarias y el espacio disponible para el manejo y ubicación de materiales que garanticen el flujo normal del proceso.

**2.1.1. Características de área.** Según su área los departamentos se pueden clasificar en departamentos de áreas iguales y departamentos de áreas desiguales. De acuerdo a esta clasificación, el procedimiento para solucionar el FLP se hace de manera diferente para cada una de éstas. Para áreas iguales el problema se considera de asignación cuadrática (QAP) y para áreas desiguales, recibe el nombre de problema del conjunto de cobertura (SCP).

De acuerdo con Kyu-Yeul Lee et ál [23], el problema de áreas iguales consiste en ubicar dentro de un conjunto de instalaciones discretas, un conjunto discreto de departamentos, de manera que a cada instalación se le asigne un solo departamento, también llamado problema de asignación uno a uno o asignación cuadrática. Mientras que el problema de áreas desiguales consiste en asignar todas las instalaciones en un plano o área disponible. En este problema, un departamento es representado por un polígono que puede tomar cualquier forma o número de instalaciones, dependiendo de las unidades de área que requiera para su operación.

Para solucionar el problema de áreas desiguales, se toma la planta a distribuir y se construye sobre ella una cuadrícula formada por cuadros o rectángulos (unidad de área). A partir de este momento, el problema se puede resolver de dos formas diferentes dependiendo del tipo de distribución que se vaya a realizar; un problema de distribución basado en bloques de una cuadrícula o un problema de distribución de bloques continuos. En el primero, a cada departamento se asigna el número de unidades de

área que necesite, el objetivo es distribuir éstas unidades de área en el espacio total disponible. Para garantizar que las unidades de área correspondientes a cada departamento queden juntas en la distribución, se penaliza la función objetivo. En el segundo, se manejan los mismos departamentos con sus unidades de área correspondientes pero para garantizar que dichas unidades de área queden juntas en la distribución, se maneja una curva de llenado (SFC), ésta fue desarrollada por Bartholdi y Platzman [5] para el problema del agente viajero (TSP). Esta curva consiste en una línea continua que atraviesa cada una de las unidades de área y va definiendo la secuencia en la que van a ser ubicados los departamentos, garantizando que aquellos que estén constituidos por más de una unidad de área, no vayan a ser divididos.



**Fig. 2. Ejemplo líneas de llenado [9]**

**2.1.2. Características de forma.** Cuando un departamento se clasifica según su forma, se puede decir que ésta puede ser regular o irregular. Regular, si es de forma cuadrada o rectangular; definida únicamente por dos dimensiones: ancho y largo, de lo contrario, se considerará un departamento de forma irregular (Fig. 1).



**Fig. 1. Figuras Regulares e Irregulares**

### 3. FORMULACIÓN DEL PROBLEMA DE DISTRIBUCIÓN DE PLANTAS

El criterio de decisión se define a través de la función objetivo que puede ser una única ó se pueden definir también múltiples funciones para un mismo caso, dependiendo del alcance definido. A su vez el número de restricciones para un problema, estarán determinadas por los objetivos que se definan. Los objetivos más usados ó planteados para la solución del problema FLP son:

- Minimización del costo de manejo y manipulación del material.
- Minimización de los costos de adaptación de las instalaciones.
- Minimización del número de retrocesos y bypass.
- Minimización del tráfico de material.
- Minimización de las formas irregulares de las instalaciones.

De igual forma se definen las restricciones para los objetivos planteados, estas restricciones pueden ser:

- Ubicación de ciertas instalaciones fijas.
- Distancia mínima entre las instalaciones.
- Orientación de los puestos de trabajo.

- Puntos de entrada y salida de material establecidos.
- No traslape entre las instalaciones.

### 3.1. FORMULACIÓN DISCRETA

Cuando el diseño es considerado como discreto, el problema de optimización es tratado como un Problema de Asignación Cuadrática (QAP). El espacio disponible para la planta se divide en bloques rectangulares con la misma área y forma, y cada bloque se asigna a una instalación. El objetivo, es asignar cada departamento en una instalación de tal forma que la distancia recorrida del material, producto en proceso o producto terminado sea mínima.

Una formulación típica, al determinar la ubicación relativa de las instalaciones a fin de reducir al mínimo el costo de manejo de materiales es el siguiente[2]:

$$\text{Min} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \sum_{l=1}^N f_{ik} d_{jl} X_{ij} X_{kl} \quad (1)$$

s.a:

$$\sum_{i=1}^N X_{ij} = 1, \quad j = 1, \dots, N \quad (2)$$

$$\sum_{j=1}^N X_{ij} = 1, \quad i = 1, \dots, N \quad (3)$$

Donde  $N$  es el número de departamentos a distribuir y siendo las matrices  $F$  y  $D$  de orden  $n \times n$ ;  $F = (f_{ik})$  y  $D = (d_{jl})$ ,  $f_{ik}$  es el flujo entre el departamento  $i$  al departamento  $k$ ,  $d_{jl}$  la distancia entre la instalación  $j$  y la instalación  $l$  y  $X_{ij}$  es una variable binaria  $[0,1]$  para localizar el

departamento  $i$  en la instalación  $j$ . La función objetivo (1) representa la suma de las distancias totales recorridas dado que el departamento  $i$  esté ubicado en la instalación  $j$ . En la ecuación (2) que a cada instalación sólo se le asigne un departamento y en la ecuación (3) se asegura que cada departamento esté ubicado en una instalación [20].

### 3.2. FORMULACIÓN CONTINUA

El objetivo de esta formulación, es ubicar las instalaciones de acuerdo a las coordenadas del centroide  $(x_i, y_i)$  y sus dimensiones (ancho y largo). A diferencia de la formulación discreta, ésta permite establecer restricciones especiales para cada escenario que se presente. La formulación continua es usada para solucionar el problema de áreas desiguales llamado también problema del conjunto de cobertura (SCP).

Una formulación apropiada para el problema de distribución de planta con departamentos de áreas desiguales e irregulares puede ser [12]:

$$\text{Minimizar } C = \sum_{i=1}^N \sum_{j=1}^N f_{ij} (|x_j - x_i| + |y_j - y_i|)$$

Donde  $N$  es el número de instalaciones,  $f_{ij}$  el costo del flujo de material desde centroide de la instalación  $i$  hasta el centroide de la instalación  $j$ ,  $(x_i, y_i)$  las coordenadas del centroide de la instalación  $i$  y  $(x_j, y_j)$  las coordenadas del centroide de la instalación  $j$ .

También es necesario al establecer el diseño de una planta, tener en cuenta que el área total

disponible no debe ser superior a la suma del área de todas las instalaciones.

#### **4. METAHEURÍSTICA HONEY-BEES MATING OPTIMIZATION**

La metaheurística HBMO está basada en el comportamiento y las características que presenta una colonia de abejas, donde principalmente se modela el proceso de apareamiento de las mismas. Las abejas son insectos eusociales, las principales características de los insectos eusociales son que estos cooperan en el cuidado de la cría y el nido, que existe solapamiento de generaciones y se da la división del trabajo reproductivo.

##### **4.1. ESTRUCTURA DE LA COLONIA**

Una colonia de abejas puede llegar a tener hasta 80.000 individuos, en una colmena hay miles de trabajadoras, cientos de zánganos y una reina, cada uno de estos tiene una labor asignada dentro de la colonia.

La reina es la madre de la colmena, única con capacidad reproductiva y es la encargada de poner los huevos, la reina se alimenta con “jalea real”, una sustancia de color blanco lechoso, que la hace más grande que cualquier otra abeja en la colmena.

Los zánganos son los machos reproductores, es decir los padres de la colonia, tienen ojos notablemente más grandes que los demás miembros de la colmena, con el fin de identificar a la reina cuando ella está en su vuelo de apareamiento.

Las trabajadoras son exclusivamente abejas hembras, las más numerosas de la colonia, son pequeñas, se encargan del cuidado de la colmena, de recolectar néctar, polen, agua y ciertas resinas vegetales utilizadas en la construcción de la colmena, trabajan sin tregua y son las encargadas del cuidado de las crías.

Las crías provienen de huevos que pueden ser fecundados o no fecundados, aquellas que provienen de huevos fecundados serán futuras reinas u trabajadoras y los que provienen de huevos no fecundados serán zánganos [32]. Una colonia de abejas puede llegar a tener hasta 80.000 individuos, en una colmena hay miles de trabajadoras, cientos de zánganos y una reina, cada uno de estos tiene una labor asignada dentro de la colonia.

##### **4.2. EL VUELO DE APAREAMIENTO**

El proceso de apareamiento comienza con una danza interpretada por la reina, la cual se hace lejos de la colmena, luego durante el vuelo de apareamiento los zánganos siguen a la reina en el aire para lograr aparearse con ella, la reina se aparee con entre 7 a 20 zánganos y almacena de cada uno de ellos su material genético en una bolsa llamada espermateca para así formar la reserva genética de la colonia, en seguida del apareamiento se da la muerte del zángano. La reina puede poner más de 1.500 huevos por día y esta obtiene una mezcla al azar de material acumulado en la espermateca para fertilizar cada uno, generándose las crías [32].

### 4.3. MODELO ANÁLOGO

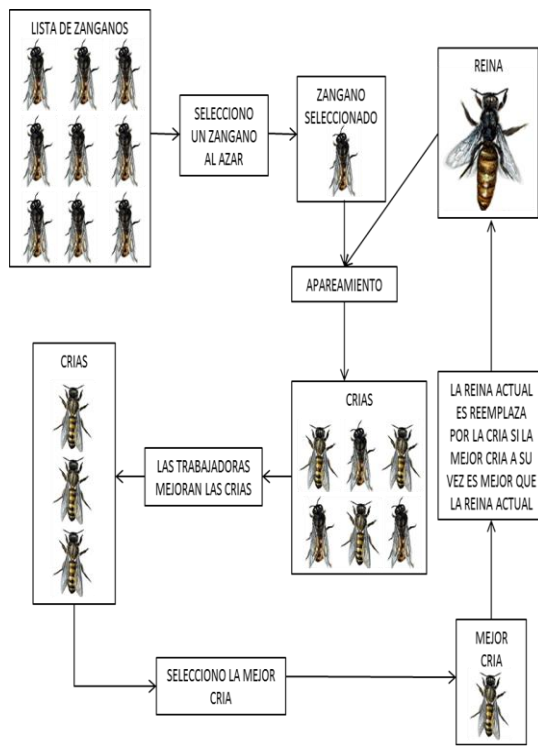


Fig. 3. Estructura de una colonia

Los procesos principales del algoritmo HBMO según Teo y Abbas[30] son:

- El vuelo de apareamiento de las reinas con los zánganos para seleccionar a los padres de las futuras crías.
- El cruce de material genético de la(s) abeja(s) reina con los zánganos seleccionados, para generar las crías.
- Mejora del aptitud (características) de las crías bajo el cuidado de las trabajadoras.
- Adaptación de la aptitud de las trabajadoras, basado en la cantidad de mejoras logradas en crías.

- Sustitución de las reinas débiles por las mejores crías encontradas.

Todos estos procesos, son igualmente relevantes en el algoritmo, por lo cual cada uno va generando la estructura de la solución final que entregará el algoritmo.

Antes de comenzar el proceso de apareamiento, el usuario debe definir el número que corresponde al tamaño de la espermoteca de la reina. Este número corresponde al número máximo de apareamientos de la reina en un único vuelo de apareamiento. Cada vez que la reina se aparea exitosamente con un zángano, el genotipo del zángano es almacenado y esta variable se incrementa en 1, hasta que la espermoteca está llena. Otros dos parámetros deben ser definidos, el número de reinas y el número de crías que nacerán de las reinas. En la implementación del algoritmo HBMO, el número de reinas es uno, porque en la vida real solo una reina sobrevivirá en una colmena.

Una representación general del algoritmo HBMO original propuesto por Abbas [1], se presenta en la Fig. 4.

---

Define  $W$  y  $B$  son el número de trabajadoras y crías respectivamente  
 Define  $M$ , será el tamaño de la espermoteca  
 Define  $E(t)$  y  $S(t)$  energía y velocidad de la abeja reina respectivamente  
 Inicializa cada trabajadora como una única heurística  
 Genera aleatoriamente el genotipo de la reina  
 Selecciona aleatoriamente una trabajadora para mejorar el genotipo de la reina  
**mientras** el criterio de parada no se satisfaga  
      $t = 0$   
     inicie  $E(t)$  y  $S(t)$

```

inicie  $g = \frac{0.5 \times E(t)}{M}$ 
genere un zángano D aleatoriamente
mientras  $E(t) > 0$ 
  evalúe el genotipo del zángano
  si el zángano pasa la condición de
  probabilidad, entonces
    si el espermateca de la
    reina no está llena,
    entonces
      adicione su esperma al
      espermateca de la reina
    fin si
  fin si
   $E(t + 1) = E(t) - g$ 
   $S(t + 1) = S(t) \times 0.9$ 
  Desmarque la mitad del genotipo
  del zángano
fin mientras
para  $cría = 1$  hasta B
  Seleccione un esperma del espermateca de la
  reina, aleatoriamente
  Genere una cría, cruzando el genotipo de la
  reina con el esperma seleccionado
  Seleccione una trabajadora para mejorar la
  cría
fin para
mientras la mejor cría sea mejor que la reina
  Reemplace la reina por la mejor cría
  Retire la mejor cría de la lista de crías
fin mientras
  Mate todas las crías

```

**fin mientras**

**Fig. 4. Representación general del algoritmo HBMO**

Al comienzo del vuelo de apareamiento, la reina inicia con un contenido de energía. Inicialmente tanto la energía como la velocidad de la reina son generadas aleatoriamente. La reina regresa a su nido cuando la energía se acerca a cero o la espermateca está llena.

La probabilidad de que un zángano se aparee con la reina está dada por la siguiente función de recocido [1]:

$$Prob(D) = e^{\left[\frac{-\Delta(f)}{S(t)}\right]}$$

Donde  $Prob(D)$  es la probabilidad de colocar el esperma del zángano en la espermateca de la reina, es decir, la probabilidad de un apareamiento exitoso,  $\Delta(f)$  es la diferencia absoluta entre la aptitud del zángano  $D$  y la aptitud de la reina y  $S(t)$  es la velocidad de la reina en un tiempo  $t$ . La probabilidad de apareamiento es alta cuando la reina se encuentra en el comienzo de su vuelo de apareamiento, por lo tanto su velocidad es alta, o cuando la aptitud del zángano es tan buena como la de la reina. Después de cada transición en el espacio, la velocidad y la energía de la reina disminuyen de acuerdo a las siguientes ecuaciones [1]:

$$S(t + 1) = \alpha * S(t)$$

$$E(t + 1) = E(t) - g$$

Donde  $\alpha$  y  $g$ , corresponden a los factores de reducción de la velocidad y la energía de la reina en el instante  $t$  respectivamente,  $\alpha$  es un factor  $C(0,1)$ . Una serie de vuelos de apareamiento se realizan. Si el apareamiento es exitoso, el esperma del zángano es almacenado en el espermateca de la reina. Cruzando los genotipos del zángano con los de la reina, se forma una nueva cría (solución de prueba), la cual después puede ser mejorada utilizando trabajadoras, que en la implementación del algoritmo son procedimientos heurísticos que conducen a una búsqueda. Una de las mayores diferencias del HBMO con los algoritmos evolutivos clásicos es que desde que la reina almacena un número de espermatozoides en su espermateca ella puede usar partes del genotipo de

diferentes zánganos para crear una nueva solución lo cual da la posibilidad de tener las crías más aptas.

En la vida real el papel de las trabajadoras es restringido al cuidado de la cría y por esta razón las trabajadoras no son miembros separados de la población pero ellas son usadas como procedimientos de búsqueda local a fin de mejorar las crías producidas por el vuelo de apareamiento de la reina. Cada una de las trabajadoras, tiene diferentes capacidades y la selección de dos trabajadoras diferentes puede producir diferentes soluciones. Cada una de los trabajadoras tiene la posibilidad de activar o desactivar una serie de diferentes características. Cada una de las crías se elige al azar, una trabajadora alimenta esta con jalea real, esto corresponde a la fase de búsqueda local y finalmente se tiene como resultado la posibilidad de reemplazar la reina si la solución de la cría es mejor que la solución de la reina actual. Si la cría no reemplaza la reina, en el siguiente vuelo de apareamiento de la reina esta cría será uno de los zánganos [1].

#### 4.4. ALGORITMO HBMO PROPUESTO PARA EL PROBLEMA FLP

El genotipo de la reina, los zánganos y las crías son soluciones al problema que se está desarrollando, el cual está representado por una cadena de genes, donde el número de genes que conforma dicho genotipo es igual al número de departamentos (N) del problema.

Un ejemplo de un genotipo, de un zángano, una cría o la reina para un problema de 12 departamentos es:

2	6	3	8	5	10	7	1	12	4	11	9
---	---	---	---	---	----	---	---	----	---	----	---

Fig. 5. Ejemplo, genotipo de un zángano

##### 4.4.1. Parámetros utilizados en el algoritmo

Los parámetros generales con los que se trabaja en el algoritmo propuesto, son:

- La cantidad de abejas reina (Q): este valor, representa el número de abejas reina que se tendrán al iniciar cada vuelo de apareamiento. Como se dijo anteriormente en el desarrollo de este problema se trabajara con una sola reina.
- La cantidad de zánganos (D): este valor, representa el número máximo de zánganos con que la reina podrá aparearse en un vuelo. El valor que se ha asignado es 100 zánganos en cada vuelo.
- El tamaño de la espermateca (M): este valor, representa la cantidad de material genético de zánganos, que puede almacenar la abeja reina en cada vuelo. Este valor se estableció en 100.

El parámetro “factor de reducción de energía de la abeja reina”, de determina utilizando la expresión introducida por Teo y Abbas [30], expresada como:

$$g = \frac{0,5 * E(t)}{M}$$

Donde  $E(t)$ , corresponde a la energía de la abeja reina en un tiempo  $t$  y  $M$ , al tamaño de la

espermateca respectivamente. La energía inicial y la velocidad inicial de la abeja reina, se inician aleatoriamente tomando valores entre 0,5 y 1. En la Tabla 1, se muestra un resumen de los parámetros utilizados en el algoritmo HBMO propuesto en este estudio.

**Tabla 1. Parámetros definidos para la implementación del algoritmo HBMO**

PARAMETRO	VALOR
Cantidad de abejas reina	1
Cantidad de zánganos	100
Velocidad inicial de la abeja reina	Aleatorio entre 0,5 y 1
Energía inicial de la abeja reina	Aleatorio entre 0,5 y 1
Factor de reducción de energía de la reina	$g$
Factor de reducción de velocidad de la reina	0,9
Tamaño de la espermateca	100
Número de vuelos	3, 6, 9 y 12
Numero de crías	5, 10, 15 y 20

**Fuente: Autores**

#### 4.4.2. Obtención de la población inicial.

Como ya se definió se tendrá una abeja reina, y se estableció que la solución inicial o abeja reina inicial, se obtendrá aleatoriamente. Luego la abeja reina inicial, es mejorada por medio de la utilización de las trabajadoras, es decir, por medio de procedimientos heurísticos, para así asegurar que la reina represente una buena solución. Para el desarrollo del proyecto se eligieron tres metaheurísticas de búsqueda local: escalada de montaña, búsqueda tabú y recocido simulado.

Siguiendo el modelo inicial propuesto por Abbas [1], la población inicial de zánganos también se obtiene de manera aleatoria.

#### 4.4.3. Selección del material genético del zángano.

Anteriormente se presentó la función de recocido, para representar la probabilidad de que el material genético de un zángano sea agregado a la espermateca de abeja reina. Esta probabilidad debe ser comparada con un número aleatorio generado entre 0 y 1, si la probabilidad de apareamiento es mayor que el número generado, la abeja reina acepta el material genético de dicho zángano, añadiendo su genotipo a la espermateca, dando como resultado un apareamiento exitoso. Cuando la aptitud de algún zángano es mejor que la aptitud de la reina, este reemplazara a la reina.

#### 4.4.4. Genotipo de los zánganos.

Cada zángano posee un genotipo y un operador llamado “marcador de genotipo”, el cual determina el “genotipo marcado” del zángano, cuando el número de genes  $N$  (número de departamentos) es par, se marca aleatoriamente la mitad de los genes en el genotipo del zángano, es decir  $N/2$  genes, mientras que cuando el número de genes es impar, se marca  $(N-1)/2$  genes. El “genotipo marcado” del zángano es el utilizado en el cruce con el genotipo de la reina para crear una nueva cría, esto garantiza que solo la mitad de los genes del zángano sean usados, ya que se considera que los zánganos son haploides.

A continuación se muestra el genotipo de un zángano, el operador “marcador de genotipo” y el “genotipo marcado”, este último será el aportado a la cría en el momento del cruce con el genotipo de la reina. La demarcación  $m$ , indica que el gen

correspondiente ha sido marcado, la demarcación nm indica que el gen no ha sido marcado y el asterisco (\*), representa un gen inexistente.

Genotipo del zángano	2	6	3	8	5	10	7	1	12	4	11	9
Marcador de genotipo	m	nm	nm	M	nm	m	m	nm	m	nm	nm	m
Genotipo marcado	2	*	*	8	*	10	7	*	12	*	*	9

Fig. 6. Genotipo de los zánganos

#### 4.4.5. Cruce de material genético y obtención de las crías.

Para el cruce entre el “genotipo marcado” del zángano y el genotipo de la abeja reina, se toma el “genotipo marcado” y en las posiciones donde no hay genes existentes, se completa con el gen correspondiente en dicha posición del genotipo de la abeja reina. Luego se revisan cuales genes se repiten a lo largo del genotipo y cuales genes hacen falta en el mismo, para así llevar a cabo la corrección del genotipo de la cría, donde un genotipo que se repite es remplazado por un genotipo faltante. A continuación se ilustra un ejemplo:

“Genotipo marcado”	2	*	*	8	*	10	7	*	12	*	*	9
--------------------	---	---	---	---	---	----	---	---	----	---	---	---

Fig. 7. Genotipo marcado

Genotipo de la reina	3	5	1	8	10	4	12	6	2	11	9	7
----------------------	---	---	---	---	----	---	----	---	---	----	---	---

Fig. 8. Genotipo de la reina

Cruce	2	5	1	8	10	10	7	6	12	11	9	9
-------	---	---	---	---	----	----	---	---	----	----	---	---

Fig. 9. Cruce de material genético

4.4.6. Mejora de las crías utilizando las trabajadoras. Se consideraron 3 trabajadoras, las cuales son las encargadas de realizar el

mejoramiento tanto de la abeja reina como de las crías obtenidas después de realizar el cruce de material genético del zángano y de la reina. Las trabajadoras utilizadas son las metaheurísticas búsqueda tabú, recocido simulado y escalada de montaña.

#### 4.4.7. Actualización de la población.

Después de realizado cada vuelo y una vez las crías generadas mediante el cruce de material genético han sido mejoradas a través de la aplicación de las trabajadoras (metaheurísticas), se evalúa la aptitud de cada una de las crías mejoradas. Si la aptitud de alguna de las crías mejorada es mejor que la aptitud de la abeja reina actual ( $Z_{cría} < Z_{reina}$ ), entonces se reemplaza la abeja reina por la cría de mejor aptitud, siendo esta ultima la nueva reina.

#### 4.4.8. Condiciones de detención.

El criterio de parada que se estableció, es alcanzar cierto número de iteraciones, el cual es representado por el número de vuelos, que dicho anteriormente variara entre 3, 6, 9 y 12 vuelos. La regla de detención para las trabajadoras utilizadas en el experimento es 10 iteraciones en búsqueda tabú y 7 el tamaño de la lista tabú; 20 iteraciones en escalada de montaña y en recocido simulado 10 iteraciones, 10 temperaturas, temperatura inicial igual a 100 y factor de reducción de la temperatura igual a 0.9.

## 5. RESULTADOS

De los problemas de prueba proporcionados por Burkard, Karisch y Rendl en 1994, disponibles en

la librería electrónica QAPLIB [8], fueron seleccionados ocho problemas para probar la eficiencia del algoritmo HBMO propuesto en este trabajo para resolver el problema de distribución de planta (FLP).

Se analizaron 4 tamaños de planta diferente; 12, 16, 20 y 25 departamentos. De los ocho problemas se corrieron dos para cada instancia. Cada problema fue resuelto con 3, 6, 9 y 12 vuelos de apareamiento y 5, 10, 15 y 20 crías para cada vuelo.

La herramienta diseñada ofrece muy buenos resultados, tiene la capacidad de solucionar problemas como los del experimento y además problemas para instancias mayores. Las desviaciones del óptimo son muy pequeñas para todos los casos e incluso del 0% para los casos de 12 y 16 departamentos, lo que corrobora el buen desempeño del algoritmo HBMO.

Para instancias de 20 y 25 departamentos, la definición de los parámetros es independiente para cada problema, es decir que el rendimiento del algoritmo depende del problema particular que se esté resolviendo.

## 6. LINEAS FUTURAS DE INVESTIGACIÓN

- Ampliar la investigación, considerando mayor número de restricciones, acercándose de esta manera a lo que es una planta real.
- Se recomienda aplicar la metaheurística al Problema de Distribución de Planta Dinámico (DFLP), de tal manera que se consideren los

cambios en la demanda a través del tiempo, lo cual tiene incidencia en el Diseño de la Planta.

- Incorporar otro tipo de trabajadoras al algoritmo HBMO, a fin de mejorar el desempeño de la metaheurística.
- Desarrollar proyectos de este tipo en empresas, de tal forma que se caracterice la planta siguiendo sus condiciones específicas y se haga la formulación y la implementación del algoritmo para el caso particular.

## 7. REFERENCIAS BIBLIOGRAFICAS

- [1] Abbass H. A, 2001, MBO: Marriage in Honey Bees Optimization A Haplometrosis Polygynous Swarming Approach. IEEE Congress on Evolutionary Computation 2001; Seoul, pp. 207-214
- [2] AMINE, Drira., HENRI, Pierreval., SONIA, Hajri-Gabouj. Facility layout problems: A survey. No. 31 (2007); p. 255 – 267
- [3] ARMOUR, G.C., BUFFA, E.S., A heuristic algorithm and simulation approach to relative allocation of facilities. Management Science. Vol. 9, No. 2, 1963; p. 294–309
- [4] BALAKRISHNAN J., CHENG C.H., CONWAY Y LAUD., A hybrid genetic algorithm for the dynamic plant layout problem. International journal of production economics. Vol. 2, No. 86 (2003); p. 107 -120
- [5] BARTHOLDI, J.J. y PLATZMAN, L.K., An Onlong planar traveling salesman heuristic based on Space filling curves. En : Operation research letters. Vol. 1, No. 4 (1982); p. 121 – 125
- [6] BLUM, Christian y ROLI, Andrea, Metaheuristics in Combinatorial Optimization:

- Overview and Conceptual Comparison. En: ACM Computing Survey. Vol. 35, No. 3, (2003); p. 268-308
- [7] BRITO, Julio; CAMPOS, Clara; GARCIA, Félix; BATISTA, Belén & MORENO, José A., Metaheurísticas: una revisión actualizada. Grupo de Computación Inteligente, Universidad de La Laguna, La Laguna, 2004
- [8] BURKARD R.E., ÇELA E., S.E. KARISCH & F. RENDL, QAPLIB - A Quadratic Assignment Problem Library. Disponible en: <http://www.opt.math.tu-graz.ac.at/qaplib/>
- [9] CHASE, JACOBS & AQUILANO, Administración de Operaciones: Producción y Cadena de Suministro. 12 ed. México: McGraw Hill, 2009; p. 221 – 222
- [10] CHWIF, L., PEREIRA BARRETTO, M.R. & MOSCATO, L.A., A solution to the facility layout problem using simulated annealing. En : Computers in Industry. Vol. 2, No. 36 (1998); p. 125 – 132
- [11] DUARTE & CALDERON, Alternativas de Solución al Problema de Distribución de Planta (FLP) de un Solo Nivel y Departamentos Iguales ó Desiguales, Por Medio Del Análisis De Métodos Metaheurísticos. 2010
- [12] Dunker, T., Radonsb, G., & Westkaïmpera, E. (2005). Combining evolutionary computation and dynamic programming for solving a dynamic facility layout problem. European Journal of Operational Research, 165(1), 55–69.
- [13] HERAGU, S.S., Facilities design. En : Boston: BWS. (1997); p. 3.
- [14] HILLER & LIBERMAN, Introducción a la investigación de Operaciones. 8 ed. México: McGraw Hill, 2006; p. 617, 619
- [15] GALLEGO CARRILLO, Micael; DUARTE MUÑOZ, Abraham; PANTRIGO FERNÁNDEZ, Juan José, Metaheurísticas. 1ª ed. Madrid: Editorial Dykinson S.L., 2008; p. 9-12
- [16] GAREY, M. R. & JOHNSON, D. S., Computers and intractability: A guide to the theory of NP-completeness. New York: WH Freeman, 1979
- [17] GLOVER, F., Future paths for integer programming and links to artificial intelligence. En: Computers & operations Research. Vol. 3, No. 1 (1986); p. 533 – 49
- [18] Guignard, M. (2002), Lagrangian Relaxation, Handbook of Applied Optimization, Oxford University Press, pag. 465-474
- [19] KIM, J.G., & KIM, Y.D, Layout planning for facilities with fixed shapes and input and output points. En : International journal of Production Research. Vol. 18, No. 38 (2000); p. 4635 – 4653
- [20] KOCHHAR, Jasmit Singh., FOSTER, Bryce. T. y HERAGU, Sunderesh S., Hope: A genetic algorithm for the unequal area facility layout problem. En : Computers operation research. Vol. 25, No. 7/8 (1998); p. 583 – 594
- [21] KOMARUDIN, KUAN YEW, WONG., Applying Ant System for solving Unequal Area Facility Layout Problems. European Journal of Operational Research. Vol. 202, (2010); p. 730–746
- [22] KOOPMANS, T.C., & BECKMANN, M., Assignment problems and the location of economic activities. Econometrica. Vol. 1, No. 25, 1957; p. 53-76
- [23] LEE, Kyu – Yeul., HAN, Seong – nam y ROH, Myung – LL., An improved genetic algorithm for facility layout problems having

- inner structure walls and passages. En : Computers and operations research. No. 30 (2003); p. 117 – 38
- [24] MELLER, R.D., CHEN, W., SHERALI, H.D., Applying the sequence-pair representation to optimal facility layout designs. Operations Research Letters. Vol. 35, No. 5 (2007); p. 651–659
- [25] MELLER, R.D., GAU, K.Y., The facility layout problem: Recent and emerging trends and perspectives. Journal of Manufacturing Systems. Vol. 15, No. 5, 1996; p. 351–366
- [26] MELLER, R.D., NARAYANAN, V., VANCE, P.H., Optimal facility layout design. Operations Research Letters Vol. 23 No. 3–5, 1999; p. 117–127
- [27] MIR, M. e IMAM, M.H. A hybrid optimization approach for layout design of unequal – area facilities. En : Computers and industrial engineering. Vol. 1 -2, No. 39 (2001); p. 49 – 63
- [28] RIOJAS CAÑARI, Alicia Cirila. Búsqueda Tabú: conceptos, algoritmo y aplicación al problema de las N-reinas. Monografía (Licenciatura en Investigación Operativa), UNIVERSIDAD MAYOR DE SAN MARCOS, Facultad ciencias Matemáticas. Lima, 2005
- [29] SHAYAN, E., & CHIHITILAPPILLY, A. Genetic algorithm for facilities layout problems based on slicing tree structure. En : International Journal of Production Research. Vol. 19, No. 42 (2004); p. 4055-4067
- [30] TEO J., ABBASS H.A., An Annealing Approach to the Mating-Flight Trajectories in the Marriage in Honey Bees Optimization Algorithm, Technical Report CS04/01, School of Computer Science, University of New South Wales at ADFA, 2001
- [31] TOMPKINS, J.A., WHITE, J.A., BOZER, Y.A, FRAZELLE, E.H., TANCHOCO, J.M., & TREVINO, J. Facilities planning. En : New York: Wiley. (1996).
- [32] Yannis Marinakis, Magdalene Marinaki and Nikolaos Matsatsinis. A Hybrid Clustering Algorithm based on Honey Bees Mating Optimization and Greedy Randomized Adaptive Search Procedure. Lecture Notes in Computer Science. Vol. 5313, (2008); p. 138-152
- [33] WELGAMA P.S. y GIBSON P.R. A construction algorithm for the machine layout problem with fixed pick-up and drop-off points. En : International journal of production research. Vol. 11, No. 31 (1993); p. 2575 – 2590.