



Universidad Popular Autónoma del Estado de Puebla

Centro Interdisciplinario de Posgrados,

Investigación y Consultoría

Departamento de Ingeniería

Doctorado en Logística y Dirección de la Cadena de Suministro

Título de la tesis

Metodologías de Solución para la Distribución de Productos con Flota Heterogénea.

Tesis que para obtener el Grado de Doctor
en Logística y Dirección de la Cadena de Suministro

Presenta

María Paz Guadalupe Acosta Quintana

Puebla, México.

Julio 2012



UPAEP – Secretaría General

Dirección General de Apoyos Académicos

Dirección del Centro de Recursos para el Aprendizaje y la Investigación.

Biblioteca Central - **Karol Wojtyla**

Tesis Digitales Restricciones de uso:

DERECHOS RESERVADOS ©

PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de textos, imágenes, gráficas, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente de donde la obtuvo mencionando el autor o autores involucrados en el documento.

Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Universidad Popular Autónoma del Estado de Puebla

Centro Interdisciplinario de Posgrados,

Investigación y Consultoría

Departamento de Ingeniería

Doctorado en Logística y Dirección de la Cadena de Suministro

Se aprueba la Tesis:

Metodologías de Solución para la Distribución de Productos con Flota
Heterogénea.

Comité de Revisión.



Dr. Elías Olivares Benítez

Director



Dr. José Luis Martínez Flores

Asesor



Dra. Claudia Malcón Cervera

Asesor

Puebla, México.

Julio 2012

DEDICATORIAS

A mi papá+ que aunque físicamente no estás conmigo este es un logro de los dos, te extraño.

A mi mamá quien me ha dado toda la fortaleza, apoyo y la compañía para el logro de esta meta.

A mis hermanos por estar siempre conmigo, los quiero.

A mi chaparrita por ser la fuente de energía que me impulsa a salir adelante, te amo mi vida.

AGRADECIMIENTOS

Al Instituto Tecnológico de Sonora por darme la oportunidad de realizar mis estudios y brindarme su apoyo para el logro de esta meta personal y profesional.

A mi Director de Tesis Dr. Elías Olivares Benítez, por su apoyo, sus conocimientos, y su tiempo para el desarrollo de este proyecto, que nos permitió compartir ideas, pero sobretodo aprender de usted. Un aspecto que tengo de usted es su amistad, regalo que cuidaré mucho. Muchas gracias por todo doctor.

A mi asesora Dra. Claudia Malcón Cervera, por su apoyo en la revisión, así como en las aportaciones y consejos durante el desarrollo de mi trabajo, muchas gracias.

A mi asesor Dr. José Luis Martínez Flores, gracias por el apoyo brindado tanto en el desarrollo de mi tesis. Gracias por darme su amistad, es un regalo valioso para mí.

Al Dr. Carlos Raúl Navarro González por su ayuda y apoyo en este proceso de esfuerzo, dedicación y sobre todo de perseverancia, gracias.

Agradezco a mi amiga Sandra A. Peñuñuri por darme siempre ese aliento que me motivaba a seguir adelante, muchas gracias amiga.

Agradezco a mis compañeros y amigos de trabajo por su apoyo incondicional.

Finalmente, a todos mis maestros de la UPATP.

INDICE GENERAL

ÍNDICE DE FIGURAS	iii
ÍNDICE DE TABLAS.....	iv
RESUMEN	v
Capítulo I. INTRODUCCIÓN.....	1
1.1 Antecedentes	1
1.2 Problema de investigación.....	7
1.3 Justificación.....	9
1.4 Objetivos de la investigación.....	10
1.4.1 Objetivo General	10
1.4.2 Objetivos Específicos.....	10
1.5 Preguntas de investigación	11
1.6 Alcances y Limitaciones.....	12
1.6.1 Alcances	12
1.6.2 Limitaciones	12
Capítulo II. MARCO TEÓRICO.....	13
2.1 Diseño de Rutas.....	13
2.2 Problema de Ruteo Vehicular (VRP)	14
2.2.1 Elementos que intervienen en el VRP	14
2.2.2 Variantes del VRP	16
2.3 Metodologías para resolver problemas de flota heterogénea	18
2.3.1 Métodos exactos	18
2.3.2 Heurísticas	21
2.3.3 Metaheurísticas.....	24
2.4 Análisis de Conglomerados	27
Capítulo III. METODOLOGIA.....	30
3.1 Diseño de instancias	31
3.2 Programación del modelo.....	31
3.3. Heurística de segmentación.....	32
3.3.1 Diseño de la Heurística.....	32

.....	33
3.3.2 Aplicación de la heurística de segmentación	35
3.4 Meta heurística: Algoritmos genéticos.....	36
3.4.1 Aplicación de algoritmos genéticos usando un vector genotipo aleatorio para la visita de los clientes.	36
3.4.2 Comparativo de soluciones generadas por las tres metodologías de solución utilizadas	42
3.4.3 Comparativo de soluciones generadas por el software GAMS utilizando como semilla inicial la generada por Algoritmos Genéticos	42
Capítulo IV. EXPERIMENTACIÓN	43
4.1 Diseño de las instancias de prueba	43
4.2 Modelo Matemático.....	44
4.3 Software comercial.....	47
4.4 Equipo de Cómputo	48
Capítulo V. RESULTADOS.....	49
5.1 Diseño de las instancias.....	49
5.2 Programación del modelo.....	49
5.3 Diseño de una heurística de segmentación.....	53
5.4. Aplicación de algoritmos genéticos usando un vector genotipo aleatorio para la visita de los clientes.....	55
5.5 Comparativo de soluciones generadas por las tres metodologías de solución utilizadas.....	58
CONCLUSIONES Y RECOMENDACIONES	61
REFERENCIAS BIBLIOGRÁFICAS	66
ANEXO A	72
ANEXO D. GRAFICAS DE LA SEGMENTACIÓN DE NODOS DE CADA INSTANCIA.....	81

ÍNDICE DE FIGURAS

Figura 1.1 Elementos de un problema de ruteo vehicular de flota heterogénea.....	4
Figura 2. 1 Ilustración de la subjetividad del análisis de conglomerados.	29
Figura 3. 1. Metodología planteada para realizar la investigación.	30
Figura 3. 2. Representación del factor de aislamiento.....	33
Figura 3. 3. Representación del factor de aislamiento.....	34
Figura 3. 4. Representación del factor de distancia al almacén.....	34
Figura 3. 5. Vector genotipo para una instancia de 18 nodos.....	36
Figura 3. 6. Vector demanda para una instancia de 18 nodos	37
Figura 3. 7. Vector genotipo que contempla al vehículo utilizado en la ruta.....	37
Figura 3. 8. Vector de genotipo y trayectorias interpretadas.....	38
Figura 3. 9. Pseudocódigo del algoritmo genético para FSMVRP	39
Figura 3. 10. Operador de mutación del algoritmo genético para FSMVRP.	40
Figura 3. 11. Operador de cruzamiento del algoritmo genético para FSMVRP.	40
Figura 3. 12. Complemento al operador de cruzamiento para el vector de tipo de vehículo para FSMVRP.....	41
Figura 5. 1. Segmentos conformados para el G30A.....	54
Figura 5. 2. Comparativo de la función objetivo arrojada por los métodos de solución aplicados	58
Figura 5. 3. Comparativo de la capacidad utilizada de los vehículos por los métodos de solución aplicados.....	59
Figura 5. 4. Comparativo del número de rutas arrojada por los métodos de solución aplicados.	60

ÍNDICE DE TABLAS

Tabla 1.1. Proyección de variables binarias enteras generadas por el modelo VRP	7
Tabla 3. 1. Datos sobre la flota de vehículos contemplada en la investigación	37
Tabla 4. 1. Datos utilizados para generar las instancias	43
Tabla 4. 2. Instancias utilizadas en la investigación	44
Tabla 5. 1. Variables que se generan para cada instancia	49
Tabla 5. 2. Resultados obtenidos mediante diversos softwares del modelo VRP para flota heterogénea	50
Tabla 5. 3. Resultados obtenidos del VRP mediante CPLEX, usando o no heurísticas y estrategias nodales.	51
Tabla 5. 4. Resultados con el software comercial GAMS	52
Tabla 5. 5. Segmentos conformados por instancia	53
Tabla 5. 6. Resultados obtenidos para cada instancia con la segmentación	54
Tabla 5. 7. Resultados obtenidos con Algoritmos Genéticos	55
Tabla 5. 8. Resultados obtenidos mediante Algoritmos Genéticos en las instancias de FSMVRP	56
Tabla 5. 9. Comparativo entre CPLEX y Algoritmos Genéticos en las instancias de FSMVRP	57

RESUMEN

En este trabajo se aborda un problema de ruteo vehicular para flota heterogénea a partir de datos reales de una empresa que tiene la necesidad de conocer si la flotilla actual es la adecuada o no. Para ello se utilizan diferentes métodos de solución, debido principalmente a la complejidad del modelo asociada al número de variables generadas por el número de clientes y los diferentes tipos de vehículos utilizados. Con el fin de validar lo relacionado al tamaño de la flota se generaron instancias partiendo de los datos reales, considerando en ellas el número de nodos, demanda expresada en peso y volumen, distancia entre nodos, así como el rendimiento por kilómetro, costos fijos, capacidades en peso y volumen de cada uno de los vehículos. Las instancias consideradas en la investigación son de 18, 30, 50, 80 y 100 nodos. Además se establecieron los siguientes criterios para el análisis de los resultados: función objetivo, uso de las capacidades de los vehículos, número de rutas y tiempo de solución, para cada una de las instancias. El modelo utilizado para este problema fue el propuesto por Olivera (2004) al cual se le realizaron adaptaciones en una de las restricciones relacionadas con la carga de tal manera que permitiera el análisis del peso y del volumen. Posteriormente se programó en Lingo versión 12, el cual no generó una solución adecuada para las instancias grandes debido a las restricciones de la licencia, por lo cual se optó en utilizar el software GAMS versión 10, arrojando resultados en tiempos no muy cortos, pero esto fue posible con el uso de estrategias y heurísticas nodales que comprobaron que este software tiene mejor capacidad para resolver este tipo de problemas aun cuando su tiempo de solución sea alto. Debido a la limitación del software se diseñó una heurística de segmentación de nodos que permitiera reducir los tiempos de solución, la cual se basa en tres factores: aislamiento, alejamiento y distancia al almacén. Una vez segmentados los nodos de cada una de las instancias se procedió a resolverlas en GAMS para analizar sus resultados. Otro método de solución utilizado fue Algoritmos Genéticos con el objetivo de tener una solución que permitiera comparar los resultados arrojados por la heurística de segmentación y por el software comercial, cuya aplicación está basada en el establecimiento de un vector genotipo que contempla demanda, tipos de vehículos y que da la solución planteada en rutas. Finalmente, se concluye que a través de los resultados obtenidos por el método exacto, la heurística y la metaheurística, la heurística de segmentación cumple en gran medida los criterios establecidos en esta investigación, permitiendo reducir los costos en la función objetivo, aprovechar las capacidades de los camiones y reduce el número de rutas y por ende el tamaño de la flota en relación a lo arrojado por los otros métodos.

Capítulo I. INTRODUCCIÓN

1.1 Antecedentes

La distribución de productos juega un rol muy importante para aquellas empresas interesadas en satisfacer a sus clientes, pero que además buscan desarrollar este tipo de actividades con costos bajos.

Toth y Vigo (2000) plantean que el problema de distribuir productos desde ciertos depósitos a sus usuarios finales juega un papel central en la gestión de algunos sistemas logísticos, y su adecuada planificación puede significar considerables ahorros. Esos potenciales ahorros justifican en gran medida la utilización de técnicas de investigación operativa como facilitadoras de la planificación.

El elemento individual mas importante en los costos de logística es la transportación de la carga para hacerla llegar al cliente (Boubeta, 2007), y que puede representar hasta el 50% del costo logístico total (Correa et al., 2008) y que comprende el costo del combustible y el costo del transporte (Ballou, 2004), por lo que es esencial la disminución de este costo de los procesos logísticos (Hasle, 2003), los cuales pueden impactar en la economía de las empresas, pues pueden llegar a representar hasta el 20% del costo del producto (Hoff et al., 2010; Mejia y Castro, 2007). Es posible la disminución de los costos de los procesos logísticos relacionados con el transporte aplicando métodos de solución y administrando de manera eficiente los recursos asociados a este servicio (Hasle, 2003), tomando decisiones relacionadas con el transporte y flotilla de vehículos; ante lo cual el problema de ruteo vehicular (VRP, vehicle routing problem) surge como una necesidad actual.

En este tipo de problemas de transporte (ruteo) es necesario determinar el tipo de recurso a utilizar, la cantidad de unidades y las rutas a seguir (Olivera, 2004), y es tratado en la literatura con un enfoque sin capacidad como el problema del agente viajero o para problemas con capacidad definida (Machado et al., 2002) es definido como el VRP, es decir el problema de ruteo de vehículos (Olivera, 2004).

El VRP, definido en los sesenta, consiste en que dado un conjunto de clientes dispersos geográficamente y una flota de vehículos, hay que determinar un conjunto de rutas de costo mínimo que comiencen y terminen en un depósito, para que los vehículos visiten a los clientes máximo una vez y satisfagan la demanda requerida (Daza et al., 2009). A esta definición se pueden agregar diversas variantes, como ventanas de tiempo para visitar a los clientes, uso de flota heterogénea en vez de homogénea para surtir las demandas, múltiple cantidad de depósitos, entregas parciales (Potvin, 2009; González y González, 2006).

El VRP se representa como un conjunto de nodos a ser visitados (clientes) y un conjunto de vehículos inicialmente ubicados en un nodo particular (depósito). El objetivo es asignar rutas a los vehículos (secuencia ordenada de nodos) tendiente a minimizar el costo total de transporte (Jaque, 2008).

La flota de vehículos susceptible de ser empleada en un problema de distribución tiene asociada un conjunto de costos dependientes de múltiples factores. El caso más general consiste en suponer que cada medio de transporte disponible presenta unas características propias que pueden o no coincidir con el resto (Yepes y Medina, 2002).

Una de las decisiones operativas que debe tomarse frecuentemente en la gestión del transporte es el diseño de rutas con las cuales se atiende la demanda de los clientes finales (Riopel et al., 2005). No es lo mismo realizar rutas de recolección de basura, que rutas de repartición de periódicos o de distribución de alimentos (Correa et al., 2007) y por lo tanto, las herramientas deben ajustarse a las características de cada situación.

En general, la flota de vehículos de una empresa de transporte no suele ser homogénea, principalmente por las aplicaciones industriales (Hoff et al., 2010). Los vehículos difieren en equipamiento, en capacidad, en antigüedad y en estructura de costos (Yadlapalli et al., 2009, Dullaert et al., 2002). La necesidad de estar presente en los diversos segmentos de mercado obliga a muchas empresas a disponer de vehículos que se adapten a diversas tipologías, como el transporte de gráneles, contenedores, etc. (Yepes y Medina, 2002). Desde el punto de vista táctico y operacional, una flota heterogénea es usualmente preferida

por su flexibilidad (Loxton y Lin, 2011). El modelo de flota heterogénea presenta dificultad creciente en la solución por métodos óptimos, incluso para problemas con apenas 20 clientes, haciendo que el enfoque de métodos heurísticos sea muy utilizado para este tipo de problemas (Pessoa et al., 2009; Renaud y Boctor, 2002).

El objetivo del Vehicle Routing Problem (VRP) para flota heterogénea es minimizar la suma de los costos fijos y variables de las rutas. El costo fijo de los vehículos se compone con los costos de adquisición y mantenimiento de la flota heterogénea. Los costos variables de las rutas están en función a la distancia total recorrida (Belfiore y Tsugunobu, 2006), y en algunos casos con la cantidad de carga transportada.

Asimismo, la disposición de unidades con diferentes capacidades de carga permite una mejor adaptación a la demanda. El problema de las rutas con una flota heterogénea y limitada de unidades de cada clase, y donde los costos de operación y los fijos dependen del tipo de vehículo, llamado en inglés “*Vehicle Routing Problem with a Heterogeneous Fleet*” (VRPHF) ha recibido muy poca atención (Taillard, 1999) en relación con el clásico VRP.

El Problema de Ruteo Vehicular de Flota Heterogénea (VRPHF) consiste en diseñar un conjunto de rutas de los vehículos, cada una comenzando y terminando en el depósito, y de tal manera que cada cliente es visitado exactamente una vez, la demanda total de un recorrido no supera la capacidad del vehículo asignado a él, y el total de los costos se reduce al mínimo, logrando determinar la composición de la ruta en orden para un servicio pre especificado de un conjunto de clientes con conocimiento de la demanda de entrega desde un almacén central (Lima et al., 2004).

En la Figura 1.1 se muestran los elementos que intervienen en un problema de ruteo vehicular de flota heterogénea, siendo estos: clientes, un solo almacén, vehículos de diferentes características, destacándose principalmente la capacidad de cada uno.

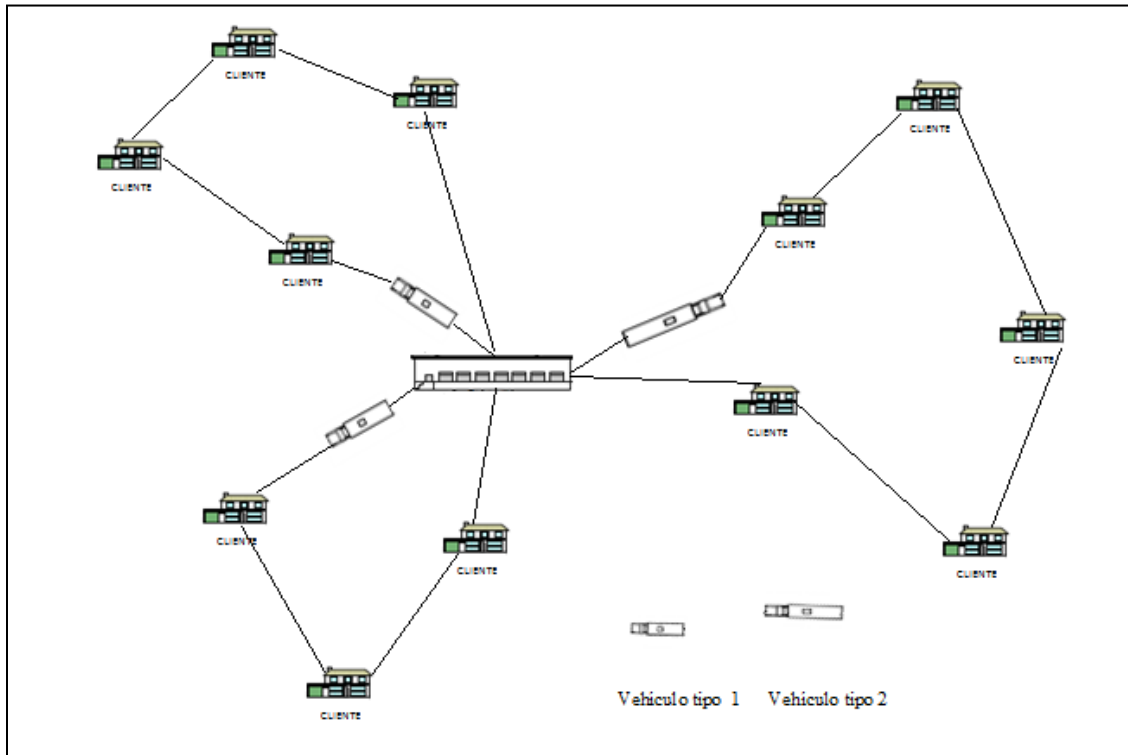


Figura 1.1 Elementos de un problema de ruteo vehicular de flota heterogénea.

Existen dos variantes para el VRP con flota heterogénea. Primeramente, el número de vehículos de cada tipo pueden ser limitados o ilimitados, cuando es necesario determinar el tamaño y la composición de la flota. En segundo lugar, los costos variables pueden o no depender del tipo de vehículo (Belfiore y Tsugunobu, 2006).

El tiempo y esfuerzo computacional requeridos para resolver este problema aumentan exponencialmente respecto al tamaño del problema, es decir, la cantidad de nodos a ser visitados por los vehículos. Para este tipo de problemas es a menudo deseable obtener soluciones aproximadas, para que puedan ser encontradas suficientemente rápido y que sean suficientemente buenas para llegar a ser útiles en la toma de decisiones. (Lüer et al., 2009).

Los problemas de múltiples rutas desde un depósito presentan una complejidad tal, que son resueltos casi exclusivamente por medio de algoritmos heurísticos. La complejidad de este problema de optimización combinatoria en la obtención de soluciones óptimas en tiempo

razonable está despertando un gran interés teórico y práctico, apoyado por las numerosas aplicaciones reales del VRP, y por la relevancia en la reducción de costos que la distribución y la logística están adquiriendo (Pereira et al., 2002).

El VRP pertenece a la clase de problemas NP-completo o NP-duro (Martínez, 2010; Choi y Tcha, 2007) y es considerado como un problema de optimización combinatoria complejo y que no tiene un algoritmo exacto que pueda resolverlo en tiempo razonable, y solo existen soluciones óptimas a este problema para pequeñas instancias (Daza et al., 2009). La complejidad del problema de ruteo vehicular crece de manera exponencial debido a la naturaleza combinatoria intrínseca del problema; que hace que incluso con pocos clientes sea complejo de resolver (Hu et al., 2007); por lo que a menudo es deseable obtener soluciones aproximadas en un tiempo adecuado y que puedan llegar a ser útiles en la toma de decisiones (Lüer et al., 2009).

En un algoritmo de programación lineal entera, el tiempo de cálculo necesario para resolver el problema se incrementa con rapidez a medida que aumenta el número de clientes n . En un caso general el número de rutas factibles que debe considerarse es $(n - 1)!/2$; así un VRP con 10 clientes tiene unas 200,000 soluciones factibles que deben ser consideradas, un problema con 20 clientes tiene alrededor de 10^{16} soluciones factibles, mientras que un problema con 50 clientes tiene alrededor 10^{62} . Y puesto que un algoritmo exacto busca producir una solución óptima explorando el espacio de búsqueda; es eficiente para problemas pequeños, pero en la práctica puede no ser útil debido al gran número de soluciones posibles para cualquier problema de tamaño razonable (Daza et al., 2009).

Debido a la complejidad del VRP diferentes métodos de aproximación han sido propuestos, que aunque no garantizan encontrar la solución óptima brindan una muy buena solución al problema (Martínez, 2010; Xu et al., 2005); estas técnicas aproximativas se dividen en técnicas de optimización local convencional (heurísticas) y técnicas de optimización local inteligente (metaheurísticas). Las heurísticas son algoritmos que permiten buenas soluciones a un costo computacional razonable, pero sin garantizar optimalidad; mientras que las “metaheurísticas” tienen como principio tomar inicialmente una solución factible,

para luego mejorarla usando algoritmos de mejoramiento embebidas en una estructura más general.

En el análisis de los resultados del problema vehicular y en el caso particular del problema de flota heterogénea se toman diferentes criterios de decisión que se analizan en las instancias generadas para ver su comportamiento. Tales criterios se proponen en el trabajo de Correa et al. (2008).

1.2 Problema de investigación

En este trabajo se resuelve un problema de ruteo vehicular con flota heterogénea. El problema está basado en información y necesidades de una empresa dedicada a la distribución de productos farmacéuticos de la región de Sinaloa. Para probar los métodos propuestos, a partir de los datos se generan instancias de diferentes tamaños.

El problema de ruteo vehicular (VRP), en muchas ocasiones está restringido por las capacidades del software de optimización utilizado. En el modelo para resolver el VRP se produce una cantidad de variables que sigue una distribución exponencial, que está definida por la cantidad de tipos de vehículos y el número de nodos considerados en el problema. En la Tabla 1.1, se presenta una proyección de las variables binarias enteras generadas, lo cual muestra la complejidad del problema a resolver.

Tabla 1.1. Proyección de variables binarias enteras generadas por el modelo VRP

	# vehiculos					
nodos	2	3	4	5	6	8
5	40	60	80	100	120	160
8	112	168	224	280	336	448
9	144	216	288	360	432	576
10	180	270	360	450	540	720
11	220	330	440	550	660	880
12	264	396	528	660	792	1056
13	312	468	624	780	936	1248
20	760	1140	1520	1900	2280	3040
25	1200	1800	2400	3000	3600	4800
26	1300	1950	2600	3250	3900	5200
27	1404	2106	2808	3510	4212	5616
32	1984	2976	3968	4960	5952	7936
51	5100	7650	10200	12750	15300	20400
81	12960	19440	25920	32400	38880	51840
101	20200	30300	40400	50500	60600	80800
nodos = # Clientes + 1 almacen						

Tal y como puede apreciarse en la tabla anterior, la proporción de variables generadas aumenta a medida que se utilizan mas vehículos. En base a lo anterior se requiere de una herramienta que ayude a resolver las instancias planteadas para la distribución de productos con una flota heterogénea que minimice los costos, permitiendo determinar el tamaño de la flota que logre un aprovechamiento de la capacidad de los vehículos considerados en cada

una de las instancias, de modo que las soluciones generadas se obtengan en tiempos adecuados y con una buena calidad.

Dada la complejidad que presenta este tipo de problemas se contempla la aplicación de una heurística y una metaheurística que permitan encontrar una solución adecuada, lo anterior como apoyo para la toma de decisiones a nivel estratégico que ayude a determinar el tamaño de la flota que se requiere y a nivel operativo lo relacionado con el diseño de las rutas de la flota. Es por ello que en esta tesis se enfocará al desarrollo de metodologías de solución que permitan apoyar a este tipo de decisiones en un tiempo razonable, y con una buena calidad de la solución.

1.3 Justificación

En consecuencia a los altos costos que genera la distribución de los productos es muy importante contar con rutas de entrega diseñadas que ayuden a minimizar el tamaño de la flota requerida para la entrega de productos, acortar la distancia recorrida, los tiempos de espera del cliente, y aprovechar al máximo la capacidad de transporte de cada vehículo de la flota, para realizar las entregas de productos, con la finalidad de cumplir con la expectativas del cliente y añadir valor al producto (Eidelman y Valdez, 2007).

Existe el interés por resolver un problema de ruteo vehicular con flota heterogénea. El problema está basado en información y necesidades de una empresa dedicada a la distribución de productos farmacéuticos de la región de Sinaloa, quien desea conocer el comportamiento que pudiera tener su flotilla considerando que el número de sus clientes se incrementa, así como su demanda, aspecto sobre el cual resulta relevante realizar una exploración utilizando instancias de diferentes tamaños, así como métodos de solución adecuados que generen soluciones buenas.

Por la complejidad en la solución de problemas de flota heterogénea en donde el número de vehículos a utilizar aumenta las variables, así como el tiempo en el que se obtiene la solución, es necesario desarrollar metodologías que permitan apoyar decisiones relacionadas con el transporte, principalmente a un nivel estratégico al definir el tamaño de la flota vehicular y a nivel operativo en lo correspondiente al diseño de las rutas que una empresa debe tener para hacerle llegar a sus clientes sus productos en el menor tiempo posible, y que garantice la disminución de los costos.

1.4 Objetivos de la investigación

1.4.1 Objetivo General

Proponer metodologías de solución enfocadas a la distribución de productos con una flota heterogénea proporcionando resultados que apoyen a la toma de decisiones relacionadas con el problema a nivel estratégico y operativo para minimizar los costos.

1.4.2 Objetivos Específicos

Los objetivos que se plantean en la investigación son los siguientes:

- Resolver el problema de ruteo vehicular para flota heterogénea por medio de diversos software comerciales que permita tener soluciones métodos exactos.
- Diseñar una heurística que permita reducir los costos usando factores asociados a la distribución relacionados con las distancias.
- Aplicar la heurística propuesta a las instancias planteadas para minimizar el tiempo de solución.
- Aplicar la metaheurística de algoritmos genéticos al problema de ruteo vehicular para flota heterogénea.
- Realizar una comparación entre las metodologías de solución usando los criterios relacionados a la distribución con flota heterogénea.

1.5 Preguntas de investigación

Con base a lo anterior se plantean las siguientes preguntas de investigación:

1. ¿Los métodos exactos proporcionan soluciones adecuadas a las instancias planteadas?
2. ¿La heurística diseñada genera soluciones apropiadas al problema de ruteo vehicular de flota heterogénea para el criterio de la distancia?
3. ¿Cómo cumplen los criterios establecidos en la investigación las metodologías de solución?
4. ¿Es posible el uso de heurísticas y metaheurísticas para resolver este tipo de problemas?
5. ¿Para qué tamaño de instancias resulta mejor el uso de las metodologías de solución?
6. ¿Es posible combinar el uso de las metodologías de solución?
7. ¿La heurística y la meta heurística proporcionan una solución adecuada al problema de ruteo vehicular de flota heterogénea proporcionando información para las decisiones estratégicas y operativas?

1.6 Alcances y Limitaciones

1.6.1 Alcances

- Se desarrollarán las metodologías de solución para las instancias diseñadas
- Se contempla el uso de software comercial LINGO versión 12 industrial para universidades y GAMS versión 10.

1.6.2 Limitaciones

- Los modelos de VRP para flota heterogénea consideran costos fijos y variables, para lo cual se determina el costo variable en base a la distancia que se recorre de un cliente a otro. Así mismo contempla restricciones enfocadas principalmente a la capacidad de los vehículos y la carga acumulada de cada uno.
- Las instancias diseñadas para esta investigación se generaron a partir de datos reales para las variables descritas y consideradas en el modelo.

Capítulo II. MARCO TEÓRICO

2.1 Diseño de Rutas

Las decisiones relacionadas con el transporte para realizar la distribución, según Ballou (2004), se expresan en una variedad de formas, las principales son: la selección del medio, el diseño de la ruta, la programación de los vehículos y la consolidación del envío.

La selección de una modalidad de transporte depende de las diferentes características del servicio y se puede usar para crear una ventaja competitiva de servicio. En cada modalidad algunos de los factores clave son la puntualidad y la disponibilidad, en tanto el contacto con la empresa, la conveniencia, la restitución y el costo son de menor importancia (Ballou, 2004).

El problema de planeación de rutas es uno de los principales problemas en la optimización de operaciones logísticas en sistemas de transporte que buscan primordialmente la reducción de costos (Soto et al., 2008).

Existen diferentes niveles en los cuales se puede realizar la distribución de los productos y uno de los problemas que se presentan es el diseño de las rutas que deben de seguir cada uno de los vehículos, lo cual es denominado como el problema de transporte que consiste en determinar un conjunto de rutas para una flota de vehículos que parten de uno o más depósitos o almacenes para satisfacer la demanda de varios clientes dispersos geográficamente los cuales se tienen claramente identificados.

Los problemas de rutas también se han estudiado en una gran variedad de contextos. Las condiciones impuestas normalmente están relacionadas con la forma en la que se desea satisfacer la demanda de un determinado servicio. Habitualmente, estos problemas se formulan en términos de problemas en grafos y pueden clasificarse en dos grandes familias

según se sitúe la demanda de los clientes sobre los arcos o los nodos del grafo, respectivamente (Albareda et al., 2003)

Las aplicaciones de los problemas de rutas por arcos son numerosas e incluyen las relacionadas con la distribución/recolección de correo, de desperdicios, limpieza de carreteras, etc., así como el trazado y mantenimiento de líneas eléctricas, de comunicación y otros. Ejemplos de problemas de rutas por nodos son todos aquellos en los que los vehículos realizan el servicio sobre los nodos del grafo, como ocurre en problemas de distribución/recogida de mercancías que aparecen casi siempre en el sector privado (Albareda et al., 2003).

2.2 Problema de Ruteo Vehicular (VRP)

El problema de ruteo vehicular (VRP) juega un rol importante dentro de la cadena de suministros donde a menudo surgen en la etapa inicial con el transporte (en la recolección de productos agrícolas, por ejemplo) o en la fase final de distribución hacia los clientes (Prins, 2004).

El VRP pertenece a la clase de problemas NP-completo o NP-duro (Martínez, 2010; Choi y Tcha, 2007) y es considerado como un problema de optimización combinatoria compleja y que no tiene un algoritmo exacto que pueda resolverlo en tiempo razonable y solo existen soluciones óptimas a este problema para pequeñas instancias (Daza et al, 2009).

La complejidad del problema de ruteo vehicular crece de manera exponencial debido a la naturaleza combinatoria intrínseca del problema; que hace que incluso con pocos clientes sea complejo de resolver (Hu et al., 2007); por lo que a menudo es deseable obtener soluciones aproximadas en un tiempo adecuado y que puedan llegar a ser útiles en la toma de decisiones (Lüer et al., 2009).

2.2.1 Elementos que intervienen en el VRP

El problema de ruteo vehicular según Olivera (2004) consiste en un conjunto de clientes y depósitos dispersos geográficamente y una flota de vehículos, para los cuales se debe

determinar un conjunto de rutas de costo mínimo que comience y termine en los depósitos, para que los vehículos visiten a todos los clientes.

Clientes

En relación a esta característica se considera que cada cliente tiene una demanda que deberá ser satisfecha por algún vehículo. La demanda es un bien que ocupa lugar en los vehículos y es usual que un mismo vehículo pueda satisfacer la demanda de todos los clientes en una misma ruta. Es común que cada cliente deba ser visitado exactamente una vez. Aunque en algunos casos se acepta que la demanda de un cliente sea satisfecha en momentos y por vehículos diferentes (Olivera, 2004).

Lo que se requiere es contar con un modelo que ayude a realizar la entrega de los productos de manera óptima, es decir que se puedan visitar en una misma ruta, un mismo vehículo, varios clientes. Los clientes podrían tener restricciones relativas a su horario de servicio. Estas restricciones se expresan en forma de intervalos de tiempo (llamados ventanas de tiempo) en los que se puede arribar al cliente (Olivera, 2004). Para el caso bajo estudio no aplican las ventanas de tiempo.

En problemas con vehículos diferentes podrían existir restricciones de compatibilidad entre éstos y los clientes. En estos casos, cada cliente sólo puede ser visitado por algunos de los vehículos (Olivera, 2004).

Otro aspecto de interés al abordar el problema del ruteo vehicular son los depósitos, que se explican a continuación.

Los Depósitos

De acuerdo a lo que plantea Olivera (2004) los vehículos como las mercancías a distribuir suelen estar ubicados en depósitos. Usualmente se exige que cada ruta comience y finalice en un mismo depósito, aunque este podría no ser el caso en algunas aplicaciones.

En los problemas con múltiples depósitos cada uno de estos tiene diferentes características, por ejemplo, su ubicación y capacidad máxima. Dada la situación puede ser que cada

depósito tenga una flota de vehículos asignada a priori o que dicha asignación sea parte de lo que se desea determinar.

Para el caso bajo estudio, solo se tiene considerada la ubicación de un almacén que tenga la función de centro logístico en el que se encuentren todos los productos terminados que serán distribuidos.

Los vehículos

La capacidad de un vehículo podría tener varias dimensiones, como por ejemplo, peso y volumen. Cuando en un mismo problema existen diferentes productos, los vehículos podrían tener compartimentos, de modo que la capacidad del vehículo dependa de los productos de que se trate (Olivera, 2004).

Cada vehículo tiene asociado un costo fijo en el que se incurre al utilizarlo y un costo variable proporcional a la distancia que recorra.

Los problemas en que los atributos (capacidad, costo, etc.) son los mismos para todos los vehículos se denominan flota homogénea, y, si hay diferencias, de flota heterogénea. La cantidad de vehículos disponibles podría ser un dato de entrada o una variable de decisión. El objetivo más usual suele ser utilizar la menor cantidad de vehículos, y minimizar la distancia recorrida ocupa un segundo lugar. (Olivera, 2004)

Los elementos del modelo del VRP y las características de los clientes, depósitos y vehículos, así como las restricciones operativas sobre las rutas, dan lugar a diferentes variantes del problema.

2.2.2 Variantes del VRP

Las aplicaciones del VRP a casos reales han aumentado debido a la consideración de diferentes aspectos que se han tomado y atendido considerándolos en los modelos a utilizar para resolver los problemas que den respuesta a estas necesidades. A continuación se describen algunas de estas variantes.

Capacitated VRP (CVRP)

EL CVRP es el VRP más general y consiste en uno o varios vehículos con capacidad limitada y constante encargados de distribuir los productos según la demanda de los clientes. (Olivera, 2004; Lee et al., 2002)

Este problema ha sido resuelto mediante búsqueda Tabú (Olivera, 2004), Algoritmos genéticos (Machado et al., 2002; Machado et al., 2003; Olivera, 2004), Algoritmos de Colonia de Hormigas (Olivera, 2004) y por otros métodos.

Multi-Depot VRP (MDVRP)

Es un modelo VRP con múltiples depósitos. Es un caso de ruteo de vehículos en el que existen varios depósitos (cada uno con una flota de vehículos independiente) que deben servir a todos los clientes.

Periodic VRP (PVRP)

Contempla en su planteamiento un horizonte de operación de M días, periodo durante el cual cada cliente debe ser visitado un número determinado de veces, problema propuesto por Francis et al. (2004) y resuelto por los mismos autores mediante relajación langrangiana.

Split Delivery VRP (SDVRP)

Se le conoce también como VRP de entrega dividida, donde se permite que un cliente pueda ser atendido por varios vehículos si el costo total se reduce, lo cual es importante si el tamaño de los pedidos excede la capacidad de un vehículo (Lee et al., 2002; Archetti et al., 2001).

Stochastic VRP (VRP)

Se trata de un VRP en que uno o varios componentes son aleatorios: clientes, demandas y tiempos estocásticos son las principales inclusiones en este tipo de problemas. El SVRP ha sido resuelto por Bianchi et al. por medio de búsqueda Tabú, recocido simulado, algoritmos de colonias de hormigas, algoritmos genéticos y otros algoritmos evolutivos.

VRP Pickup and Delivery (VRPPD)

Es un modelo VRP con entrega y recogida; es aquel en el que cabe la posibilidad de que los clientes pueden devolver determinados bienes, por tanto, se debe tener presente que estos quepan en el vehículo. Esta restricción hace más difícil el problema de planificación y puede causar una mala utilización de las capacidades de los vehículos, un aumento de las distancias recorridas o un mayor número de vehículos.

2.3 Metodologías para resolver problemas de flota heterogénea

La solución de modelos de ruteo vehicular depende de la complejidad y del número de variables que estén asociados al fenómeno bajo estudio. El VRP es un modelo clásico en el que se presentan una serie de variables y aumentando el número de restricciones, lo cual ocasiona que no se pueda obtener una solución óptima, es por ello necesario recurrir a otras técnicas que pueden dar soluciones adecuadas.

A continuación se presentan las herramientas o técnicas que se pueden utilizar para resolver un problema de flota heterogénea.

2.3.1 Métodos exactos

Los algoritmos exactos son los que producen una solución óptima empleando varias técnicas que permitan explorar el espacio de búsqueda. Estos métodos exactos incluyen los que se basan en técnicas como ramificación y acotamiento, planos cortantes y programación lógica de restricciones (Daza et al., 2009). Estos algoritmos son razonablemente eficientes para problemas de tamaño modesto (Ignizio y Cavalier, 1994); aunque con ellos es posible en principio resolver los de cualquier tamaño, en la práctica no es así, debido al gran número de soluciones posibles para cualquier problema de tamaño razonable.

Estrategias y heurísticas nodales en la programación lineal entera

El software de programación entera, en ocasiones no presenta un adecuado rendimiento, pues el proceso interactivo que es la esencia de la programación lineal (branch and bound) suele ser muy inadecuado para estos casos. Por esto es necesario presentar mejores enfoques a esta situación. Muchos de los problemas de programación entera presentan características de ser duros y difíciles de resolver (NP-Hard), para los cuales los métodos de optimización han encontrado severas dificultades de resolución en tiempos razonables (Hvattum y Lokketangen, 2012).

Existen diversas alternativas para enfrentar estos retos a la programación entera, que cada vez es más utilizada, pues a diferencia de hace 20 años donde un problema podría tener 100 variables enteras o menos, en la actualidad una instancia puede utilizar varios miles de estas, en una situación donde el tiempo de cálculo se vuelve un factor crítico.

El proceso de “branch and bound”, que es el núcleo de cualquier software de programación entera se puede entender como una búsqueda de opciones enteras dentro de cada nodo del poliedro generado, y este algoritmo termina cuando el límite inferior del óptimo global y el límite superior del óptimo global son iguales. Esta búsqueda por el poliedro o árbol de búsqueda puede ser muy larga; por lo que reducir este proceso mediante la mejora de la exploración de soluciones factibles puede ser crítico.

Para manejar las limitaciones del proceso de “branch and bound”, se contemplan una serie de estrategias, entre las cuales se incluyen ajustes a parámetros internos que indican la adición de otros procesos de búsqueda y de delimitación del óptimo, lo cual en ocasiones puede reducir los tiempos de búsqueda (Atamtürk y Salvendy, 2004); tales como técnicas de prueba, adición de planos de corte, heurísticas primarias, etc. Por lo que los software modernos para resolución de modelos lineales enteros (MIP) pueden explotar y ofrecer un conjunto amplio de alternativas para controlar y mejorar la estrategia de ramificación; e igualmente balancear con el tiempo de procesamiento, pues la generación de cortes y buenas heurísticas pueden consumirlo (Fischetti et al., 2009).

Por lo anterior, los algoritmos de resolución de problemas de programación entera existentes en el mercado, ya sea comerciales o libres mantienen una amplia gama de capacidades distintivas y por otra parte, continuamente están mejorándose, lo que permite que para modelos o problemas que antes no se podían resolver ahora sea factible lograr una solución en tiempo razonable. Durante la última década se han estado reportando avances dramáticos para los distintos softwares de programación lineal entera; mismos que de manera conservadora se puede considerar para el caso específico de CPLEX de la versión 6.5 a la 11 (Hvattum y Lokketangen, 2012). Lo cual se ha logrado por la incorporación en los programas MIP de habilidades para la mejora de algoritmos existentes y la adición de otros recientemente desarrollados, que involucran estrategias más sofisticadas de ramificación, de generación de planos de corte, pre-procesamiento, heurísticas primales, entre otras; aunque aún existe mucho trabajo que puede hacerse para mejorar y extender las capacidades de los mismos (Lodi, 2008).

Dentro de algunos softwares existen estrategias y heurísticas nodales que se pueden activar y desactivar para ver el desempeño en las soluciones, siendo algunas de ellas, las que se mencionan a continuación:

- Heurfreq: Especifica cuan frecuentemente se aplican heurísticas nodales, esta opción indica la frecuencia de su uso.
- Rinsheur: Implementación de la heurística RINS (Relaxation Induced Neighborhood Search), la cual explora en la búsqueda de soluciones en la vecindad de la solución encontrada y trata de mejorarla; esta opción indica su frecuencia de utilización.
- Mipemphasis: Esta opción controla las tácticas de ramificación y resolución para el problema de programación entero, enfatizando la búsqueda de soluciones factibles, o el óptimo global o un balance entre ambas.

2.3.2 Heurísticas

Se usa el término heurístico para referirse a un procedimiento que trata de aportar soluciones a un problema con un buen rendimiento, en lo referente a la calidad y a los recursos empleados (Melián, et al., 2003).

Una heurística es el procedimiento repetitivo que identifica una solución factible, con la cual se espera acercarse a la solución óptima (González y González, 2006). Melián et al. (2003) califican a un heurístico como un procedimiento para el que se tiene un alto grado de confianza en que encuentra soluciones de alta calidad con un costo computacional razonable, aunque no se garantice su optimalidad o su factibilidad, e incluso, en algunos casos, no se llegue a establecer lo cerca que se está de dicha situación. Se usa el calificativo heurístico en contraposición a exacto.

Las heurísticas utilizadas para resolver un problema de optimización pueden ser más *generales* o *específicas* que otras. Los métodos heurísticos específicos deben ser diseñados a propósito para cada problema, utilizando toda la información disponible y el análisis teórico del modelo (Melián et al., 2003).

Los procedimientos específicos bien diseñados suelen tener un rendimiento significativamente más alto que las heurísticas generales. Las heurísticas más generales, por el contrario, presentan otro tipo de ventajas, como la sencillez, adaptabilidad y robustez de los procedimientos.

Sin embargo, las heurísticas generales emanadas de las meta heurísticas pueden mejorar la solución utilizando recursos computacionales y estrategias inteligentes.

Métodos Heurísticos

Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución (Díaz et al., 1996).

El método heurístico más intuitivo es el de la búsqueda exhaustiva: dado un problema, analizar todas las posibles soluciones y escoger la mejor. Como puede intuirse también, este proceso será factible sólo para aquellos problemas simples que tengan un número pequeño de soluciones alternativas.

En contraposición a los métodos exactos que proporcionan una solución óptima del problema, los métodos heurísticos se limitan a proporcionar una buena solución del problema no necesariamente óptima. Lógicamente, el tiempo invertido por un método exacto para encontrar la solución óptima de un problema difícil, si es que existe tal método, es de un orden de magnitud muy superior al del heurístico (pudiendo llegar a ser tan grande en muchos casos, que sea inaplicable) (Martí, 2009).

Martí (2009) establece que los métodos heurísticos más conocidos son:

- Métodos de Descomposición.- El problema original se descompone en sub problemas más sencillos de resolver, teniendo en cuenta, aunque sea de manera general, que éstos pertenecen al mismo problema.
- Métodos Inductivos.- La idea de estos métodos es generalizar de versiones pequeñas o más sencillas al caso completo.
- Métodos de Reducción.- Consiste en identificar propiedades que se cumplen mayoritariamente en las buenas soluciones e introducirlas como restricciones del problema. El objeto es restringir el espacio de soluciones simplificando el problema. El riesgo obvio es dejar fuera las soluciones óptimas del problema original.
- Métodos Constructivos.- Consisten en construir literalmente paso a paso una solución del problema. Usualmente son métodos deterministas y suelen estar basados en la mejor elección en cada iteración. Estos métodos han sido muy utilizados en problemas clásicos como el del agente viajero.
- Métodos de Búsqueda Local.- A diferencia de los métodos anteriores, los procedimientos de búsqueda o mejora local, comienzan con una solución del problema y la mejoran progresivamente. El método finaliza cuando no existe ninguna solución accesible que mejore la anterior.

Medición de la calidad y eficiencia de los métodos heurísticos

Para medir la calidad de una heurística existen diversos procedimientos, entre los que se encuentran los siguientes:

- Comparación con la solución óptima, si se conoce o con la mejor solución disponible.
- Comparación con una cota.
- Comparación con un método exacto truncado.
- Comparación con los resultados de otras heurísticas.
- Análisis del peor caso.

Aunque se ha mencionado el caso de la resolución de un problema difícil, existen otras razones para utilizar métodos heurísticos, entre las que se destacan (Marti, 2009):

1. El problema es de una naturaleza tal que no se conoce ningún método exacto para su resolución.
2. Aunque existe un método exacto para resolver el problema, su uso es computacionalmente muy costoso.
3. El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelización.
4. El método heurístico se utiliza como parte de un procedimiento global que garantiza el óptimo de un problema. Existen dos posibilidades:
 - 4.1 El método heurístico proporciona una buena solución inicial de partida.
 - 4.2 El método heurístico participa en un paso intermedio del procedimiento, como por ejemplo las reglas de selección de la variable a entrar en la base en el método Simplex.

2.3.3 Metaheurísticas

Las metaheurísticas son estrategias maestras que permiten resolver de manera inteligente un problema (Melián et al., 2003).

Las meta heurísticas son una derivación de las heurísticas clásicas. Las heurísticas clásicas realizan una exploración limitada sobre el espacio de búsqueda y normalmente las soluciones producidas son buenas en poco tiempo. Su implementación es sencilla y son fácilmente adaptadas a problemas del mundo (Soto, et al., 2008).

Ejemplos de algunas metaheurísticas son GRASP y Búsqueda Dispersa, que se describen a continuación.

La metaheurística GRASP (Greedy Randomized Adaptive Search Procedure) es un procedimiento que consiste en dos fases: la primera es de construcción en la cual se aplica un procedimiento heurístico constructivo para obtener una buena solución inicial, la cual es mejorada en la segunda fase mediante un algoritmo de búsqueda local. La mejor solución de todas las examinadas es guardada como resultado final (Maquera et al., 2007).

La búsqueda dispersa es una metaheurística que es muy efectiva en la solución de problemas de optimización. La búsqueda dispersa opera sobre un conjunto de soluciones denominado conjunto de referencia y tiene como principio que con la combinación de soluciones, se puede obtener nuevas soluciones de modo a mejorar aquellas que la originaron (Maquera et al., 2007).

Los profesores Osman y Kelly (1996) introducen la siguiente definición:

Los procedimientos Meta heurísticos son una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que los heurísticos clásicos no son efectivos. Los Meta heurísticos proporcionan un marco general para crear nuevos algoritmos híbridos combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos.

Los procedimientos Meta-Heurísticos se sitúan conceptualmente "por encima" de los heurísticos en el sentido que guían el diseño de estos. Así, al enfrentarnos a un problema de optimización, podemos escoger cualquiera de estos métodos para diseñar un algoritmo específico que lo resuelva aproximadamente (Osman and Kelly, 1996).

Algoritmos Genéticos

Los Algoritmos Genéticos (GA) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización generando poblaciones sucesivas a las que se aplican los operadores de mutación y cruce. Están basados en el proceso genético de los organismos vivos, donde a lo largo de las generaciones, las poblaciones evolucionan acorde con los principios de la selección natural y logrando unas características cada vez mejor adaptadas a su entorno; donde los atributos más deseables se transmiten a sus descendientes cuando éste se reproduce (Rodríguez, 2003; Ventoso, 2001).

Cada individuo representa una solución al problema, y se trata de encontrar al individuo que represente a la mejor solución. Cuanto mayor sea la adaptación de individuos al problema, mayor será la probabilidad de ser seleccionados para reproducirse, cruzando su material genético; lo cual producirá nuevos individuos, descendientes de los anteriores. De esta manera se produce una nueva población de posibles soluciones, la cual reemplaza a la anterior y verifica la interesante propiedad de que contiene una mayor proporción de buenas características en comparación con la población anterior. Así a lo largo de las generaciones las buenas características se propagan a través de la población; favoreciendo el cruce de los individuos mejor adaptados y siendo exploradas las áreas más prometedoras del espacio de búsqueda. Existe el operador de mutación que mantiene su importancia al introducir pequeñas perturbaciones en las soluciones y mantiene una diversidad en la población (Potvin, 2009).

En esta metaheurística de GA, existen 2 conceptos críticos: genotipo y fenotipo. Biológicamente el genotipo consiste en la información contenida en los genes; mientras que el fenotipo representa las características del organismo biológico, que definen su nivel de aptitud al medio (González y González, 2007). En este algoritmo, el genotipo está

representado por el Vector con valores para cada individuo o solución del problema; mientras que el fenotipo es la cuantificación de la función de aptitud, la cual suele representarse por un valor que define su nivel de optimalidad, e indica que tan buena es una solución.

La función de codificación genera e interpreta los genotipos de las sucesivas generaciones, haciendo que los descendientes sean en su mayoría viables, es decir representen soluciones factibles y cumplan con la mayor parte de las restricciones. Por lo que si el Algoritmo Genético ha sido bien diseñado, la población convergerá hacia una solución óptima del problema. Esta evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada función de codificación (Xu et al., 2005).

Para la función de codificación, se parte de la representación del genotipo en el vector de valores. El genotipo que ha gozado de más popularidad, y fundamenta de manera más natural los algoritmos genéticos y resulta sencilla de implementar es la utilización del genotipo de cadena binaria, aunque se han utilizado también números reales y letras (Caballero, 2002). En el caso de problemas con muchas restricciones que causen que se generen individuos no validos dentro del espacio de búsqueda, se puede penalizar o dejar que se crucen y muten para lograr promover la generación de individuos validos; otro procedimiento es reconstruir aquellos individuos que no verifican las restricciones, aplicando sobre ellos un nuevo algoritmo que se puede denominar “operador reparador” o (Gupta y Garg, 2009; Potvin, 2009; Mitchell et al., 2003).

Algunos autores plantean que la representación binaria del genotipo o cromosoma para un individuo no es la más adecuada para problemas tipo VRP, y en este sentido se han realizado diversas propuestas en las que el genotipo es representado por un vector de números enteros (Hao, 2010; Potvin, 2009; Xu et al., 2005).

Una vez seleccionados dos padres, sus cromosomas se combinan, utilizando habitualmente los operadores de cruce y mutación. No se pueden aplicar las formas clásicas de dichos operadores, pues el cruzamiento que promueve un intercambio de material genético entre

los padres resulta en rutas invalidas con clientes faltantes o duplicados; por lo que tal y como sugieren Potvin (2009) y Pereira (2002), es necesario diseñar procedimientos especiales de cruzamiento y mutación, que generen recurrentemente rutas validas.

2.4 Análisis de Conglomerados

El Análisis de Clúster o Análisis de conglomerados es una técnica de Análisis Exploratorio de Datos para resolver problemas de clasificación. Su objetivo consiste en ordenar objetos (personas, cosas, animales, plantas, variables, etc.) en grupos (conglomerados o clúster) de forma que el grado de asociación/similitud entre miembros del mismo conglomerado sea más fuerte que el grado de asociación/similitud entre miembros de diferentes conglomerado. Cada conglomerado se describe como la clase a la que sus miembros pertenecen (Villardón, s.f.)

El análisis de conglomerados es un método que permite descubrir asociaciones y estructuras en los datos que no son evidentes a priori pero que pueden ser útiles una vez que se han encontrado. Los resultados de un Análisis de Conglomerados pueden contribuir a la definición formal de un esquema de clasificación tal como una taxonomía para un conjunto de objetos, a sugerir modelos estadísticos para describir poblaciones, a asignar nuevos individuos a las clases para diagnóstico e identificación, etc.

Villardón (s.f.) establece que se pueden encontrar dos tipos fundamentales de métodos de clasificación: Jerárquicos y No Jerárquicos. En los primeros, la clasificación resultante tiene un número creciente de clases anidadas mientras que en el segundo las clases no son anidadas.

Los métodos pueden dividirse en aglomerativos y divisivos. En los primeros se parte de tantas clases como objetos tengamos que clasificar y en pasos sucesivos vamos obteniendo clases de objetos similares, mientras que en los segundos se parte de una única clase formada por todos los objetos que se va dividiendo en clases sucesivamente.

Los pasos de acuerdo a Villardon (s.f.) que se siguen para conformar los conglomerados son los siguientes:

- 1.- Decidir qué datos se toman para cada uno de los casos.
- 2.- Elegir una medida de la distancia entre los objetos a clasificar, que serán los conglomerados o clases iniciales.
- 3.- Buscar que conglomerado son más similares.
- 4.- Juntar estos dos conglomerados en un nuevo conglomerado que tenga al menos 2 objetos, de forma que el número de conglomerado decrece en una unidad.
- 5.- Calcular la distancia entre este nuevo conglomerado y el resto.
- 6.- Repetir desde el paso 3 hasta que todos los objetos estén en un único conglomerado.

Los distintos métodos o algoritmos dependen del método utilizado en el paso 5 para calcular la distancia entre conglomerado. Es necesario resaltar, que los distintos métodos para el cálculo de las distancias entre conglomerado producen distintas clasificaciones, por lo que no existe una clasificación correcta única.

La decisión sobre el número óptimo de conglomerado es subjetiva, especialmente cuando se incrementa el número de objetos ya que si se seleccionan pocos, los conglomerados resultantes son heterogéneos y artificiales, mientras que si se seleccionan demasiados, la interpretación de los mismos suele ser complicada.

Existen algoritmos para la conformación de los conglomerados de objetos y de datos (patrones, entidades, instancias, prácticas, unidades) en un cierto número de conglomerados (grupos, subgrupos o categorías). Sin embargo, no existe una definición universalmente aceptada y precisa del término conglomerado. Everitt et al. (2001) indican que "la definición formal (de grupo) no sólo es difícil sino que incluso puede ser fuera de lugar". A pesar de esta dificultad, varias definiciones operativas aún están disponibles, tal como lo resume Everitt (1980) y se ilustra de la siguiente manera: "Un conglomerado es un conjunto de entidades que son iguales, y las entidades de los distintos grupos no son iguales". Un grupo es "un agregado de puntos en el espacio de ensayo tal que la distancia entre dos puntos cualesquiera de la agrupación es menor que la distancia entre cualquier punto de la

agrupación y cualquier punto no en él”. Los grupos pueden ser descritos como regiones continuas de este espacio (espacio de característica dimensional) que contiene una densidad relativamente alta de puntos, diferenciándose de aquellas que contienen una densidad relativamente baja de puntos”. En la Figura 2. 1se muestran ejemplos de conglomerados conformados.

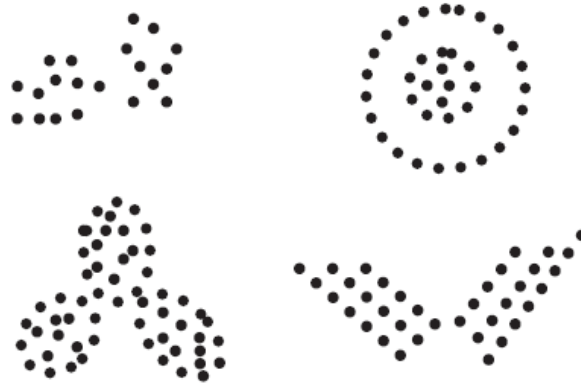


Figura 2. 1 Ilustración de la subjetividad del análisis de conglomerados.

Fuente: Xu, R, y Wunsch, D.,2009.

Es evidente que un grupo de estas definiciones se describe en términos de homogeneidad interna y la separación externa (Gordon, 1999; Hansen y Jaumard, 1997; Jain y Dubes, 1988), es decir, los objetos de datos en el mismo grupo debería ser similar a cada otro, mientras que los objetos de datos en grupos diferentes deben ser diferentes uno de otro.

Capítulo III. METODOLOGIA

Los pasos que se siguieron en la investigación se muestran de manera esquematizada en la Figura 3. 1. Manejándose posteriormente la explicación de cada uno de estos.

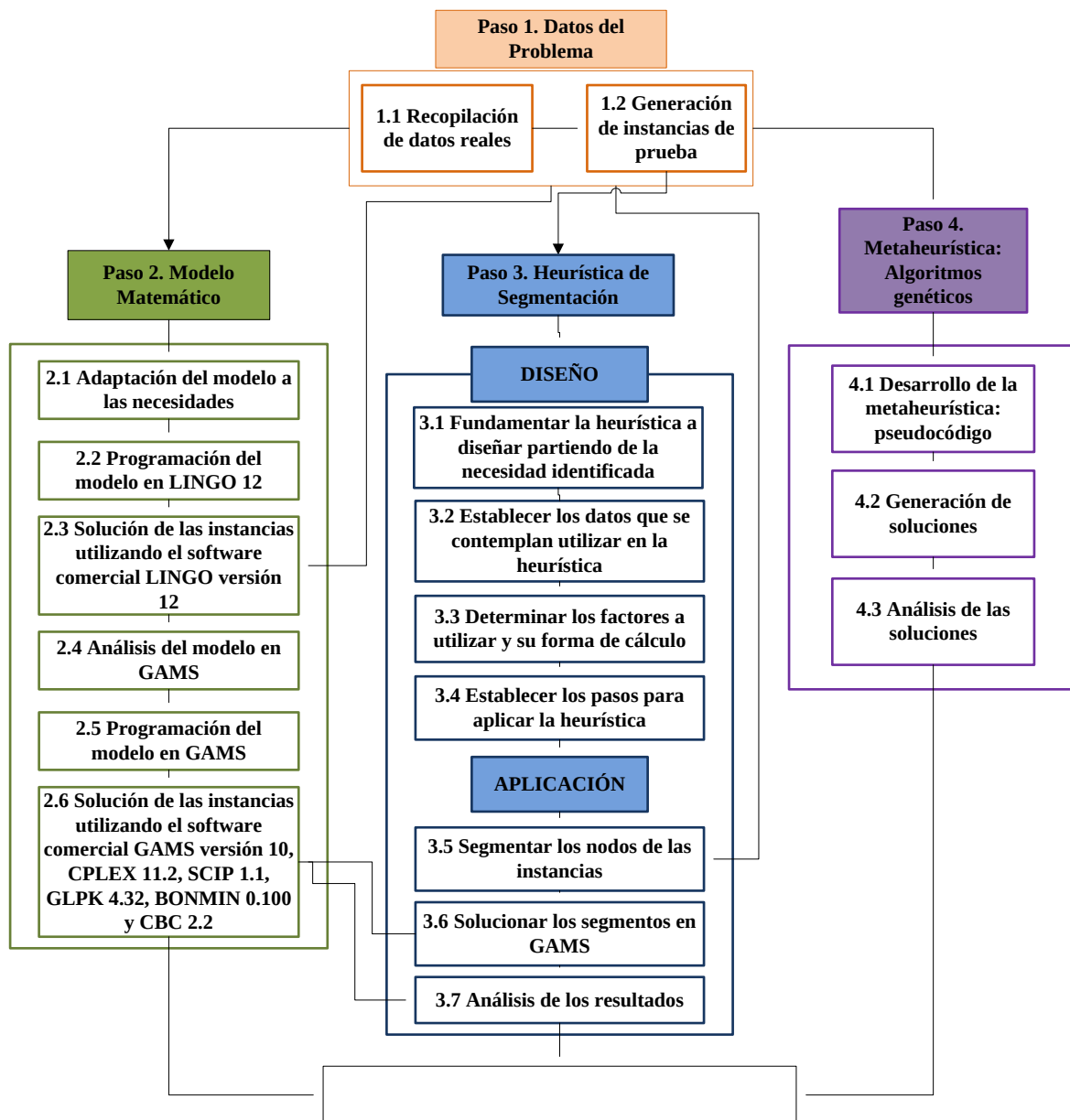


Figura 3. 1. Metodología planteada para realizar la investigación.

3.1 Diseño de instancias

Se diseñaron instancias a partir de datos reales, para 18, 30, 50, 80 y 100 nodos. Los parámetros a considerar en el diseño de las instancias en relación a los nodos serán: ubicación del nodo definiendo un punto con coordenadas (x,y) para calcular la distancia, y la demanda en peso y volumen. Con respecto a los vehículos, se definió la capacidad en peso y volumen, estableciendo además un costo fijo y los costos variables, los cuales se determina a partir de la distancia recorrida.

3.2 Programación del modelo

El modelo utilizado y que se programó para resolver las instancias planteadas es el propuesto por Olivera (2004) para problemas de ruteo vehicular de flota heterogénea. Se requiere contar con un modelo en el cual se contemple el análisis de la capacidad de los vehículos considerando el peso y el volumen.

El modelo se programó bajo diversos softwares comerciales Lingo 12, GAMS versión 22.9.2 y SCIP 1.1.

Una vez que se obtuvieron los resultados de cada uno, se realizó un comparativo de las soluciones generadas por estos, con el fin de ver cuál es de ellos genera mejores soluciones para las instancias planteadas.

Bajo el CPLEX se utilizaron las heurísticas y estrategias nodales que contempla el software para ver su comportamiento, activándolas o desactivándolas, siendo: Heurfreq, Rinsheur, y Mipemphasis.

3.3. Heurística de segmentación

Al resolver las instancias de 50, 80 y 100 nodos en el software debido a limitaciones de la licencia no fue posible obtener una solución por el número de variables que permite y cuyo número se excedía.

Debido a lo anterior se diseñaron una heurística de segmentación de nodos con el objetivo de aprovechar el modelo y resolver con el software comercial segmentos con un menor número de nodos, dentro del límite de la licencia. También se espera que este enfoque reduzca el tiempo de solución, a costa de un resultado de menor calidad en la función objetivo.

3.3.1 Diseño de la Heurística

La heurística se fundamenta en heurísticas como la de Barrido y en la Heurística de asignación generalizada de Fisher y Jaikumar.

Los datos que utiliza la heurística de segmentación son las distancias que existen entre los nodos que se desean segmentar, para lo cual se diseñó una matriz cuadrada con las distancias existentes entre nodos. Las variables que se tienen en la heurística son:

N = número de clientes

0 = nodo correspondiente al almacén

i = contador que representa al cliente i (si $i=0$ este indica al almacén). Típicamente $i = 0, \dots, N$

i_r = contador de nodos que están agrupados y pertenecen a la región r

n_r = número de nodos que se encuentra en la región r .

d_{ij} = distancia del cliente i al cliente j

La heurística de segmentación se basa en el producto de tres factores:

Factor de Aislamiento: Mientras más aislado se encuentre un cliente, más deseable se vuelve a seleccionarlo como nodo inicial, ya que de otra manera estos clientes se tienden a dejar solos y estropean el proceso de regionalización (ver Figura 3. 2). Esto se puede medir

con el promedio de su distancia a otros clientes; a mayor distancia promedio, mayor aislamiento se tiene. El factor de aislamiento se puede representar matemáticamente por:

$$f.\text{aislamiento}_i = \frac{\sum_{j=1..N} d_{i,j}}{n}$$

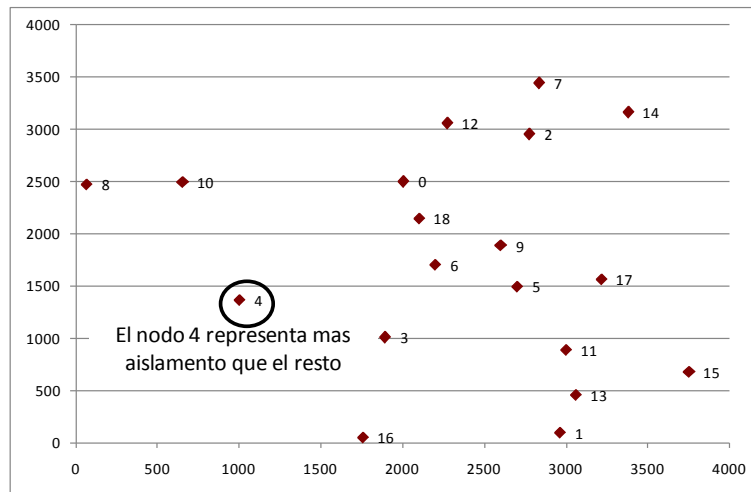


Figura 3. 2. Representación del factor de aislamiento.

Factor de Alejamiento: Mientras un cliente se encuentre en una ubicación más extrema o formando un vértice fuera de todos los puntos, más deseable se vuelve como punto de inicio, pues estos puntos suelen quedar fuera del proceso de regionalización y estropear las zonas obtenidas (ver Figura 3. 3).

El factor de alejamiento, se calcula a partir de la distancia máxima que existe de ese nodo hacia los otros (sin considerar el almacén). Su representación matemática es:

$$f.\text{alejamiento}_i = \max_{j=1..N} (d_{i,j})$$

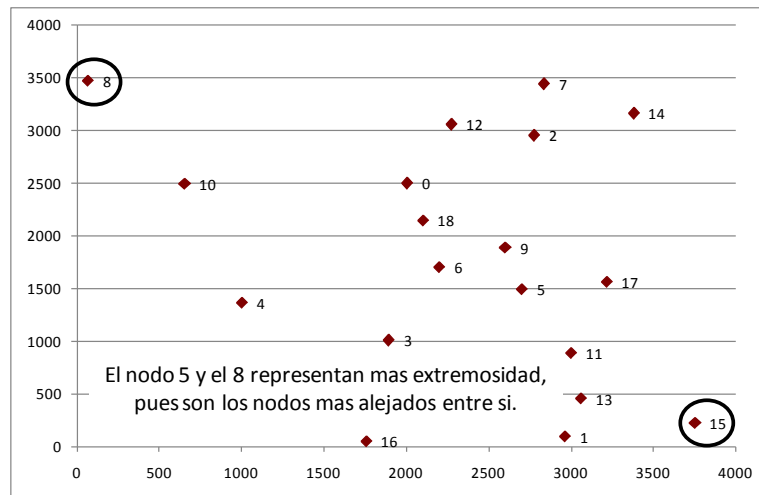


Figura 3. 3. Representación del factor de aislamiento.

Factor de distancia al almacén: Se detectó que en ocasiones iniciar el proceso de zonificación en un nodo demasiado cercano al almacén puede afectar el criterio de selección de los sucesivos nodos, por lo que se decidió incluir este criterio, el cual básicamente indica que es recomendable iniciar con algún punto que cumpla los criterios anteriores (que esté aislado y alejado con respecto a los demás nodos), pero también que este muy alejado del almacén (ver Figura 3. 4). El factor de distancia al almacén tiene la siguiente representación matemática:

$$f.dist.almacen_i = d_{i,0}$$

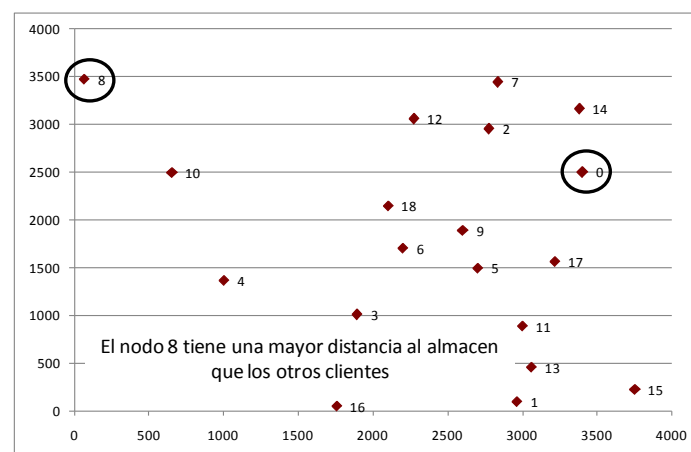


Figura 3. 4. Representación del factor de distancia al almacén.

Los pasos para la aplicación de la heurística son los siguientes:

1. Definir el número de regiones o zonas en las cuales segmentar el problema en base a la cantidad de variables que puede manejar el software para obtener una solución de manera rápida.
2. Identificar el nodo inicial. Se calcula para cada nodo el producto de los tres factores: factor de aislamiento, factor de alejamiento y factor de distancia al almacén. El nodo que tenga un producto mayor de estos factores es el que se selecciona como nodo inicial.
3. Agregar nodos a ese nodo inicial. Este paso requiere ir aplicando el criterio para seleccionar el siguiente nodo a incluir dentro de la zona o región que se está armando, para lo cual se aplica el criterio matemático de distancia promedio y alejamiento del origen. Se utiliza una potencia que determina el alejamiento entre los nodos a considerar en la región. Esta potencia puede ser modificada cuando se excluyen nodos durante la segmentación, para lo cual se recomienda cambiarlo de manera gradual cuando queden nodos fuera de los segmentos.

$$Nodo_{i_{r+1}} = \underset{\substack{j=1..N \\ j \notin region}}{\text{Min}} \left\{ \left(\sum_{i_r \in region} d_{i_r, j} / n_r \right) / (dj, o)^{0.65} \right\}$$

3.3.2 Aplicación de la heurística de segmentación

Se segmentaron los nodos para las instancias planteadas utilizando la heurística diseñada y tomando en cuenta la restricción del software en cuanto al número de variables que permite. Cada uno de los segmentos conformados en cada instancia se resolvieron con el modelo adaptado de Olivera (2004) en el software comercial utilizado y posteriormente se analizan las soluciones generadas en cada instancia tomando como criterios: función objetivo, número de rutas, capacidad del vehículo utilizada y el tiempo de solución.

3.4 Meta heurística: Algoritmos genéticos

La metaheurística se desarrolló a través de la aplicación de macros utilizando Microsoft Visual Basic 6.3, con el fin de resolver las instancias planteadas.

3.4.1 Aplicación de algoritmos genéticos usando un vector genotipo aleatorio para la visita de los clientes.

Se programó la metaheurística de algoritmos genéticos para el problema FSMVRP, para lo cual se busca realizar una interpretación de las rutas del problema de ruteo sobre un vector de genotipo, el cual debe ser interpretado por una función de codificación. Para lo cual primeramente el genotipo fue representado por un vector de cadenas binarias (en correspondencia a las variables del modelo de programación entera), representación que al ejecutarse en la metaheurística, no logró generar individuos que representaran alguna solución válida. Posteriormente se adaptó un código para aplicarles un operador reparador que corrige los cromosomas propuestos aleatoriamente para generar individuos validos; sin embargo este operador solo funcionó en instancias menores a 12 clientes; en el caso de problemas mayores generaba crecientemente individuos no validos. En un cambio de enfoque, se decidió que el genotipo seria representado por una cadena de números enteros, que al ser interpretados generarían individuos válidos.

La cadena que representa al genotipo consiste en un vector que contiene “n” números enteros generados aleatoriamente, representando a cada uno de los clientes; así para una instancia de 18 clientes este vector pudiera ser el que se muestra en la Figura 3. 5.

12	6	2	8	4	1	16	3	11	13	15	14	9	18	5	17	7	10
----	---	---	---	---	---	----	---	----	----	----	----	---	----	---	----	---	----

Figura 3. 5. Vector genotipo para una instancia de 18 nodos

Para interpretar esta secuencia, en la función de codificación, es necesario saber las demandas de cada cliente. En la Figura 3. 6 se presenta el vector demanda para la instancia de 18 nodos.

Cliente	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Peso	2505	7964	2812	787	2384	5181	2937	8339	8233	7116	7486	7168	3250	7845	1652	5278	642	4383

Figura 3. 6. Vector demanda para una instancia de 18 nodos

Así pues según el vector del genotipo, para la primera entrega, el vehículo parte del almacén o bodega, que denotaremos con el número cero, posteriormente viaja hacia el cliente 12, llega al cliente 6, luego al 2 y así sucesivamente, según lo permita la capacidad del vehículo asignado. Para esta instancia de FSMVRP, se considera una flota de cuatro vehículos, con distintas capacidades, costos y rendimientos. En la Tabla 3. 1 se presentan los datos utilizados para la flota de vehículos utilizados.

Tabla 3. 1. Datos sobre la flota de vehículos contemplada en la investigación

Vehículo		1	2	3	4
Capacidad	Kg	18,074	27,300	27,698	35,110
costos fijos	\$/Ruta	71	120	220	300
Rendimiento	\$/m	1.065	2.434	2.130	3.043

Para especificar cual vehículo se utilizará en una determinada ruta, se agrega a la cadena del genotipo, un segundo vector de cuatro posiciones, las cuales indican respectivamente a partir de su posición, cuando se usa el vehículo correspondiente. De esta manera el genotipo completo se muestra en la Figura 3. 7

12	6	2	8	4	1	16	3	11	13	15	14	9	18	5	17	7	10	0	4	7	11
----	---	---	---	---	---	----	---	----	----	----	----	---	----	---	----	---	----	---	---	---	----

Figura 3. 7. Vector genotipo que contempla al vehículo utilizado en la ruta

Donde el segundo vector indica que a partir de la posición 0 se puede utilizar el primer vehículo, a partir de la posición 4 el segundo vehículo, a partir de la posición 7 se puede usar el vehículo 3 y a partir de la 11va posición se puede usar el vehículo tipo 4, esto se ilustra en la siguiente Figura 3. 8.

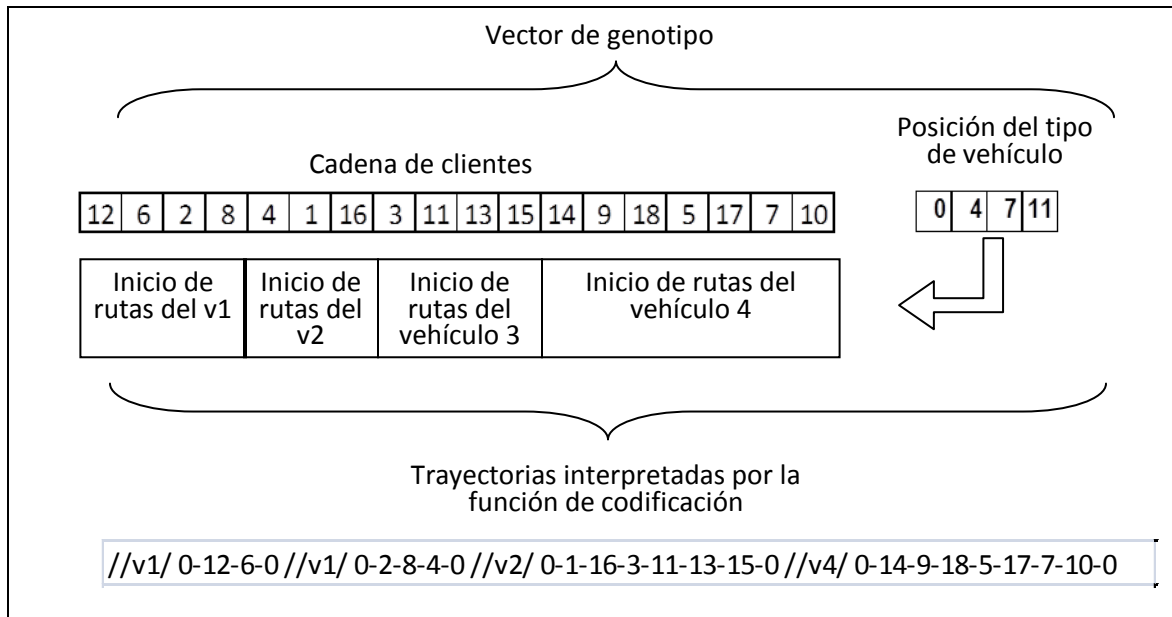


Figura 3. 8. Vector de genotipo y trayectorias interpretadas

Así pues según el vector del genotipo, para la primera entrega, el vehículo tipo 1, parte del almacén o bodega, que denotaremos con el número cero, posteriormente viaja hacia el cliente 12, llega al cliente 6; luego por falta de capacidad del vehículo, se trunca la ruta, e inicia con el cliente 2, que aun corresponde al vehículo tipo 1, por lo que la siguiente ruta es 0-2-8-4-0, y así sucesivamente. En el momento en la que su capacidad pretenda ser excedida, termina el recorrido, regresando al origen e inicia otra ruta. Tal y como puede observarse las trayectorias interpretadas fácilmente pueden cuantificarse para generar el fenotipo que representa su nivel de optimalidad, en la función de aptitud, utilizando el enfoque de costos fijos y costos variables para cada recorrido, según el tipo de vehículo.

El pseudocódigo del algoritmo genético se ilustra en la Figura 3. 9.

```

BEGIN /* Algoritmo Genético para VRP */
  Generar una población inicial compuesta del genotipo de vectores aleatorio de enteros
  Computar la función de codificación para cada individuo, interpretando las rutas asociadas
  Aplicar la función de aptitud para cada individuo, representando su nivel de optimalidad, en cuanto a
    las distancias recorridas y los costos asociados a ella.
  WHILE NOT limite de Generaciones DO
    BEGIN /* Producir nueva generación */
      FOR Tamaño población * Incremento generacional DO
        BEGIN /*Ciclo reproductivo */
          Seleccionar como padres a dos individuos de la población, para el cruce
            (aleatoriamente)
          Cruzar con cierta probabilidad los dos individuos obteniendo dos descendientes, en
            una posición aleatoria para intercambio de genes
          Mutar los descendientes luego de cierto número de iteraciones e introducir cambios
            aleatorios en ellos
          Insertar los dos descendientes en la población extendida
          Computar la función de codificación para cada individuo, interpretando las rutas
            asociadas
          Aplicar la función de aptitud para cada individuo, representando su nivel de
            optimizad, en cuanto a las distancias recorridas y los costos asociados a ella.
          Reducir la población extendida descartando aleatoriamente según un factor de
            eliminación algunos individuos cuyos costos asociados estén arriba de la media
            aritmética obtenida por toda la población.

        END
      END
    END
  END

```

Figura 3. 9. Pseudocódigo del algoritmo genético para FSMVRP

Se consideran dos categorías de operadores genéticos: Cruzamiento y mutación, que deben ser capaces de trabajar con la cadena de enteros que representa al genotipo del recorrido y generar siempre rutas validas sin eliminar o duplicar clientes. Los cuales se describen a continuación:

Operador de Mutación: Este operador debe introducir cambios aleatorios en el vector del genotipo, lo cual se puede lograr mediante la generación de 2 posiciones aleatorias y el intercambio de clientes en esas ubicaciones. A modo de ejemplo se muestra el proceso en la figura 3.10.

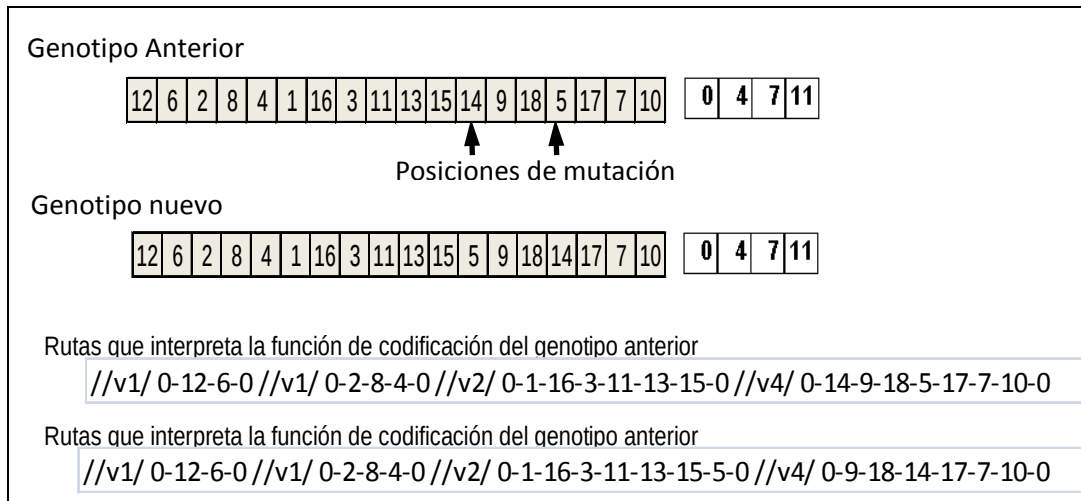


Figura 3. 10. Operador de mutación del algoritmo genético para FSMVRP.

Operador de Cruzamiento: Este operador debe promover un intercambio de material genético entre dos padres y generar descendientes que resulten en rutas validas, sin clientes faltantes o duplicados. Esto hace que el material genético del genotipo del segundo padre sea revisado y sorteado previo a su acomodo; todo esto a partir de una posición aleatoria de intercambio de material genético. Esto se ilustra en la Figura 3. 11.

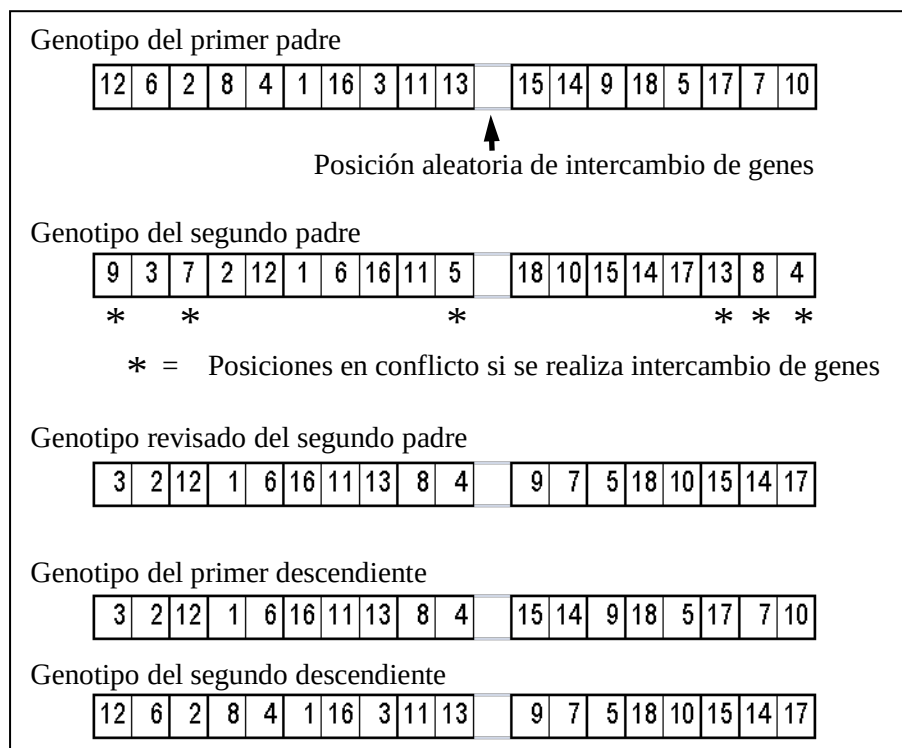


Figura 3. 11. Operador de cruzamiento del algoritmo genético para FSMVRP.

El operador de cruzamiento, también debe involucrar actividades para decidir sobre el manejo de la flota de vehículos, en este sentido Choi y Tcha (2007) hacen una interesante propuesta en su análisis para obtener límites inferiores a costos de ruteo; en su propuesta, indican que usualmente en problemas VRP con flota heterogénea existe un vehículo dominante (el que tenga un mejor costo proporcional a la capacidad del mismo). Para promover el uso del vehículo más económico posterior al cruzamiento de los vectores genotipo de los clientes, se realiza en los hijos una operación de deslizamiento, que consiste en ubicar en el vector de rutas, la ruta con mayor uso u ancho y ampliar ese ancho de banda de uso hacia los lados, en una unidad, promoviendo con ello de manera paulatina que el vehículo que más se use en los padres se use aun mas en sus descendientes; esto aunado al operador de reducción, que se indica en el pseudocódigo, asegura que solo los vehículos más económicos, vayan aumentando su uso. Esto se ilustra en la Figura 3. 12.

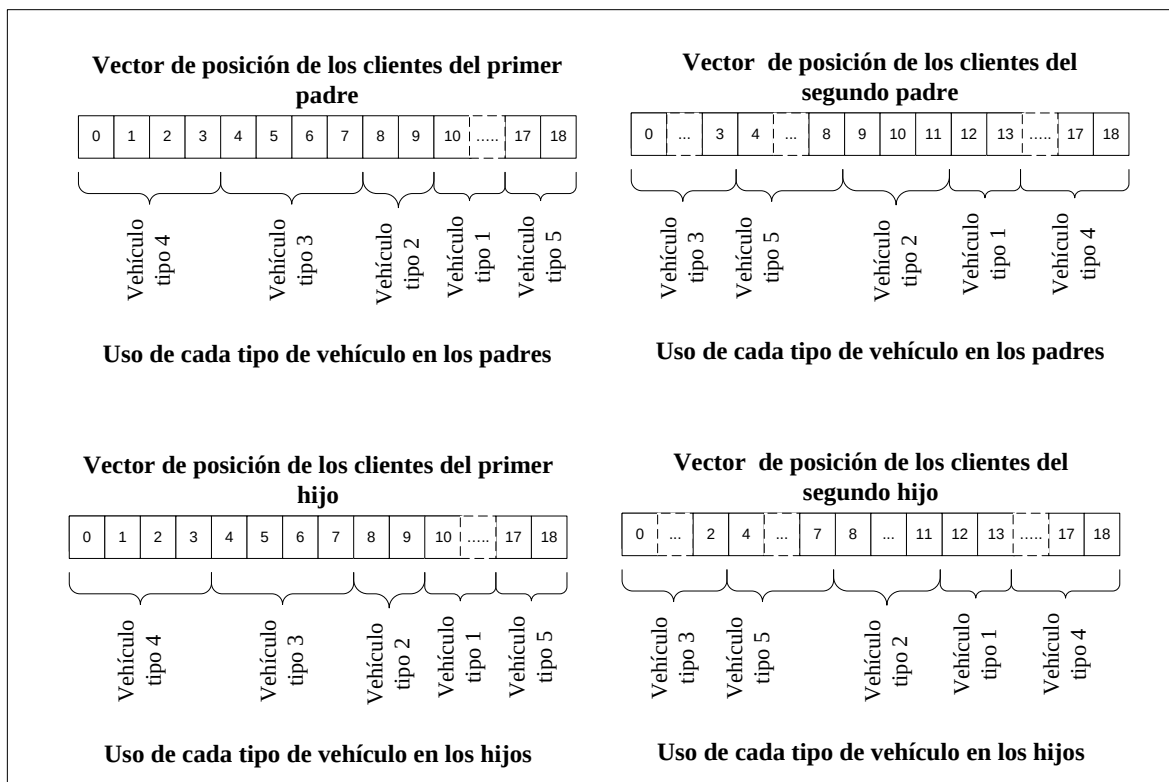


Figura 3. 12. Complemento al operador de cruzamiento para el vector de tipo de vehículo para FSMVRP

3.4.2 Comparativo de soluciones generadas por las tres metodologías de solución utilizadas

Con los resultados obtenidos por las tres metodologías de solución se analizaron utilizando los criterios considerados en la investigación, siendo estos: función objetivo, capacidad utilizada de los vehículos y número de rutas, lo cual se realizó de manera gráfica.

3.4.3 Comparativo de soluciones generadas por el software GAMS utilizando como semilla inicial la generada por Algoritmos Genéticos

Con el fin de tener un comparativo entre la meta heurística y las soluciones generadas por el software comercial GAMS, se pretende utilizar como semilla inicial para trabajarla en el entorno de este software con el fin de ver si este mejora las soluciones generadas por la meta heurística. Para lo cual se programará el software adaptándolo con valores iniciales.

Capítulo IV. EXPERIMENTACIÓN

4.1 Diseño de las instancias de prueba

Las instancias se generaron a partir de datos reales proporcionados por una empresa (Ver Anexo A), siendo estos: número de clientes, ubicación de cada uno, distancias entre los nodos que conforman la red de clientes, demanda semanal, costos fijos de los vehículos y el rendimiento de combustible por vehículo.

Los datos utilizados para generar las instancias se presentan en la Tabla 4. 1. Datos utilizados para generar las instancias.

Tabla 4. 1. Datos utilizados para generar las instancias

Valor	Demanda de cada cliente		Distancia (m)
	Peso (Kg)	Volumen (cm3)	
Mínimo	52	0.31	10
Máximo	34633	207.795	3730

El valor utilizado para generar las capacidades de los camiones se basó en el mínimo y máximo de las demandas generadas, lo cual fue aplicable al peso y volumen.

Los datos de las coordenadas (x,y) para la ubicación de los nodos, la demanda en peso y volumen, la capacidad de los vehículos en cada una de las instancias se generaron utilizando la función de Excell “aleatorio.entre“, una vez definida la ubicación de cada uno de los nodos se determinó la distancia euclidiana entre ellos, utilizando la siguiente fórmula:

$$d_{i,j \#} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$$

Para cada uno de los tamaños de 18, 30, 50, 80 y 100 nodos se generaron cinco instancias para realizar las pruebas con el fin de validar los resultados de una manera más amplia, en

donde el parámetro que se varió fue la distancia entre los nodos; el resto de los datos quedó igual (Ver Archivo Instancias Prueba).

Los datos utilizados para la investigación fueron instancias conformadas por diferente número de nodos, tipo de vehículos y costos asociados a la distancia que existe entre nodos. En la Tabla 4. 2 se presentan las instancias diseñadas para la investigación:

Tabla 4. 2. Instancias utilizadas en la investigación

Grupos de Instancias	Número de Nodos	Tipos de Vehículos	Características de los vehículos	
			Peso	Volumen
1	18	2	18074	108
			27300	164
2	31	4	18074	108
			27300	164
			27698	166
			35110	211
3	50	4	22511	211
			30887	185
			41953	164
			48990	166
4	80	5	102662	615
			199891	1199
			222598	1335
			274958	1649
			306211	1837
5	100	5	17316	103.896
			18350	110.1
			21310	127.86
			24156	144.936
			37512	225.072

4.2 Modelo Matemático

El modelo utilizado para la investigación se basó en el problema de ruteo vehicular para flota heterogénea propuesto por Olivera (2004).

En el modelo de flota heterogénea los costos y las capacidades de los vehículos varían, existiendo un conjunto $T = \{1, \dots, T\}$ de tipos de vehículo, V es el conjunto de clientes que

existen y E es el conjunto que contiene los caminos posibles para ir de un nodo i hacia el nodo j en la red de transporte. La capacidad de los vehículos $k \in T$ es q^k y su costo fijo es f^k . Los costos para cada tipo de vehículo son c_{ij}^k . Se asume que los índices de los vehículos están ordenados en forma creciente por capacidad (es decir $q^{k_1} \leq q^{k_2}$ para $k_1, k_2 \in T$, $k_1 < k_2$).

Las variables binarias x_{ij}^k indican si el arco (i,j) es utilizado por el vehículo k y las variables r_i y v_i positivas indican la carga acumulada en peso y volumen respectivamente en la ruta correspondiente hasta el nodo i (incluyéndolo). $\Delta^+(i)$ y $\Delta^-(i)$ representan el conjunto de nodos adyacentes e incidentes al nodo i .

A continuación se presentan las ecuaciones del modelo:

Función Objetivo

$$\min \sum_{k \in T} f^k + \sum_{j \in \Delta^+(0)} x_{0j}^k + \sum_{k \in T} \sum_{(i,j) \in E} c_{ij}^k x_{ij}^k \quad (1)$$

Sujeto a

$$\sum_{k \in T} \sum_{i \in \Delta^+(j)} x_{ij}^k = 1 \quad \forall j \in V \setminus \{0\} \quad (2)$$

$$\sum_{j \in \Delta^+(i)} x_{ij}^k - \sum_{j \in \Delta^-(i)} x_{ji}^k = 0 \quad \forall i \in V, \forall k \in T \quad (3)$$

$$r_0 = 0$$

Para el Peso

$$r_j - r_i \geq (d_j + q^{|T|}) \sum_{k \in T} x_{ij}^k - q^{|T|} \quad \forall i \in V \setminus \{0\}, \forall j \in \Delta^+(i) \quad (4a)$$

$$r_j - r_0 \geq (d_j + q^{|T|}) \sum_{k \in T} x_{0j}^k - q^{|T|} \quad \forall j \in V, \forall j \in \Delta^+(0) \quad (4b)$$

Para el Volumen

$$v_j - v_i \geq (d_j + q^{|T|}) \sum_{k \in T} x_{ij}^k - q^{|T|} \quad \forall i \in V \setminus \{0\}, \forall j \in \Delta + (i) \quad (5a)$$

$$v_j - v_0 \geq (d_j + q^{|T|}) \sum_{k \in T} x_{0j}^k - q^{|T|} \quad \forall j \in V, \forall j \in \Delta + (i) \quad (5b)$$

$$r_j \leq \sum_{k \in T} \sum_{i \in \Delta^-(j)} q_k x_{ij}^k \quad \forall i \in V \setminus \{0\} \quad (6)$$

$$x_{ij}^k \in \{0,1\} \quad \forall (i, j) \in E, \forall k \in T \quad (7)$$

$$r_j \geq 0 \quad \forall j \in V$$

La función objetivo (1) mide el costo total de la solución incluyendo los costos fijos y variables. La ecuación (2) establece que todo cliente debe ser visitado por algún vehículo. La ecuación (3) indica que si un vehículo de tipo k visita al nodo i , entonces un vehículo del mismo tipo debe abandonarlo. Las ecuaciones (4a,4b, 5a, 5b y 6) fijan los valores de las variables r_i y v_i actúan como restricciones de eliminación de subtours, mientras que la capacidad de los vehículos se establece en la Ec. (7).

Esta formulación asume que la cantidad de vehículos de cada tipo es ilimitada. El problema correspondiente se denomina *Fleet Size and Mix Vehicle Routing Problem* (o *FSMVRP*).

El modelo se adaptó principalmente porque al desarrollarlo en un software comercial no generó resultados coherentes. Los cambios se describen a continuación:

La Ec. (2) aplica $\forall j \in V \setminus \{0\}, i \neq j$.

En base a las necesidades observadas en la investigación, se requirió plantear las restricciones considerando el peso y el volumen, pero además se plantearon un conjunto de restricciones tanto de peso como de volumen para todos los nodos excepto el origen y para i

- j y otro conjunto exclusivamente para el origen en donde $i \bullet j$, con lo cual el modelo dio resultados adecuados. Para las diferentes instancias, la capacidad de los diferentes vehículos exhibe una correlación positiva entre el peso y el volumen.

4.3 Software comercial

El primer software comercial utilizado fue Lingo 12, Versión Industrial, la cual permite manejar 16000 restricciones, 32000 variables reales y 3200 variables enteras. Mismo que tiene un costo de 795 dólares para instituciones educativas o 2995 dólares para uso comercial (Lindo, 2012). En el Anexo B se presenta el modelo programado en LINGO 12 para el grupo de instancias de 18 Nodos.

El segundo software comercial usado fue CPLEX 11.2, ejecutado desde el entorno de desarrollo de GAMS (General Algebraic Modeling System) bajo la versión 22.9.2, CPLEX adquirido directamente de IBM tiene un costo de 8490 dólares (Cplex, 2012a), pero se indica que tiene un programa para académicos y estudiantes bajo la iniciativa “ILOG Optimization in the IBM Academic Initiative” (Cplex, 2012b), pero ejecutándolo bajo el entorno GAMS/CPLEX tiene un costo de 12800 dólares para uso comercial y 1920 dólares para uso académico (GAMS, 2012). En el Anexo C se muestra el modelo programado en GAMS para el grupo de instancias de 18 nodos.

El tercer software corresponde SCIP 1.1, desarrollado recientemente por Achterberg (2007 y 2009), bajo un entorno académico y de investigación, mismo que puede adquirirse gratuitamente para este fin (SCIP, 2012). SCIP que es acrónimo de “Solving Constraint Integer Programs” aclama ser uno de los softwares no-comerciales más rápidos; y que puede ser adquirido comercialmente de varias maneras, para ser ejecutado dentro del entorno de desarrollo GAMS a un costo de \$ 640 para uso académico y \$ 3200 dólares para uso comercial (GAMS, 2012); bajo la misma cuota, pueden ejecutarse en el entorno GAMS, tanto para uso académico como comercial, además se utilizaron los siguientes softwares: GLPK (GNU Linear Programan Kit) versión 4.32, BONMIN (Basic Open-source Nonlinear Mixed Integer programming) versión 0.100 y CBC (COIN-OR Branch and Cut) versión 2.2.

4.4 Equipo de Cómputo

Se utilizaron dos equipos de cómputo, teniendo las siguientes características:

Equipo 1. Procesador Intel Core de 2.27 GHz, memoria (RAM) 4.00 GB y el tipo de sistema operativo de 32 bits.

Equipo 2. El software de programación lineal Lingo 12 y CPLEX 11.2, SCIP 1.1, GLPK 4.32, BONMIN 0.100 y CBC 2.2 se corrieron en una computadora Dell Optiplex 745, Intel a 1.86 Ghz y 2 Gb de RAM.

Capítulo V. RESULTADOS

5.1 Diseño de las instancias

Los datos de cada una de las instancias diseñadas se presentan en el anexo A. Se presentan los datos generados en cada una de las instancias, siendo estos la demanda en peso y volumen, las características de los camiones en cuanto a peso, volumen, tipos, y la matriz con las distancias entre los nodos.

5.2 Programación del modelo

Se corrieron las instancias de 18, 31, 50, 80 y 100 para tener una solución con un método exacto.

Al modelo propuesto por Olivera se le hicieron adaptaciones debido a que en la investigación se consideró analizar la demanda en peso y volumen, por lo cual se agregó otro tipo adicional de restricción, además de hacerle una modificación en relación a la condición para cual aplica esta restricción. Esto se explicó en el capítulo de Metodología en la sección correspondiente al modelo matemático 4.2.

Al correr las diversas instancias en los distintos softwares de programación lineal entera, se observa que no en todos los casos se encuentra una solución factible, debido a la gran complejidad del modelo y al número de variables, como se muestra en la Tabla 5. 1. Variables que se generan para cada instancia.

Tabla 5. 1. Variables que se generan para cada instancia

Grupo de Instancia- Número de Nodos	Vehículos	Número de variables enteras
3-50 Nodos	4	10200
4-80 Nodos	5	32400
5-100 Nodos	5	50500

La tabla 5.2, muestra un análisis de los resultados obtenidos con los softwares de programación lineal entera considerados: Lingo 12 y CPLEX 11.2, SCIP 1.1, GLPK 4.32, BONMIN 0.100 y CBC 2.2. Las instancias consideradas en las pruebas fueron las instancias A para cada tamaño. En la Tabla 5. 2, se presenta un código que indica el grupo, el tamaño y la instancia, por ejemplo G50-E, la G50 indica el grupo de la instancia de acuerdo a su tamaño correspondiendo a 50 nodos, y E es para la instancia E

Tabla 5. 2. Resultados obtenidos mediante diversos softwares del modelo VRP para flota heterogénea

Instancia prueba	Número de clientes	Tiempo	Solución factible encontrada					
			CPLEX 11.2	SCIP 1.1	GLPK 4.32	BONMIN 0.100	CBC 2.2	LINGO 12
G18A	18	24 min	381.22	383.23	388.75	443.66	640.99	Ninguna
G31A	31	24 min	681.31	682.86	820.33	Ninguna	Ninguna	Ninguna
G50A	50	30 min	768.70	764.57	855.46	Ninguna	Ninguna	Ninguna
G80A	80	60 min	3018.73	3611.43	Ninguna	Ninguna	Ninguna	Ninguna
G100A	100	60 min	2550.94	Ninguna	Ninguna	Ninguna	Ninguna	Ninguna

Tal y como puede apreciarse en la tabla anterior, se puede evidenciar primeramente la complejidad combinatoria del modelo de VRP. También muestra la incapacidad del software LINGO 12 de resolverlo. Es interesante observar que los dos mejores softwares encontrados son de licencia comercial, el primero (CPLEX) y el otro de desarrollo abierto (SCIP). En la tabla 5.2 también se pueden apreciar los resultados con los softwares actuales para la resolución del problema VRP.

Situación que se presenta su grado de desarrollo e involucramiento de las estrategias y heurísticas nodales en sus capacidades de procesamiento.

A modo de comparativo, se presenta la Tabla 5. 3, donde se muestran los resultados obtenidos mediante la utilización de los parámetros que trae por defecto CPLEX contra los obtenidos al desactivar heurísticas y estrategias nodales.

Tabla 5. 3. Resultados obtenidos del VRP mediante CPLEX, usando o no heurísticas y estrategias nodales.

Instancia prueba	Instancia	Número de clientes	Tiempo	Solución factible encontrada, en CPLEX 11.2	
				Con heurísticas y estrategias nodales	Sin heurísticas y estrategias nodales
G18A	1	18	24 min	381.22	595.10
G31A	2	31	24 min	681.31	Ninguna
G50A	3	50	30 min	768.70	Ninguna
G80A	4	80	60 min	3018.73	Ninguna
G100A	5	100	60 min	2550.94	Ninguna

Como puede apreciarse en la tabla 5.3, al desactivar los algoritmos heurísticos y nodales en el CPLEX, su desempeño es similar al de cualquier software gratuito, tal como BONMIN y CBC, aunque se presenta aun superior a LINGO.

Se puede concluir de la tabla 5.3 que las heurísticas y estrategias nodales son parte crucial para un software de programación lineal; pues su adecuado involucramiento puede agilizar mucho la solución de problemas MIP, especialmente los que tengan las características de alta complejidad combinatoria, tal y como corresponde a los problemas de ruteo vehicular (VRP).

Con los resultados proporcionados, se seleccionó CPLEX y se analizaron los resultados para cada una de las instancias considerándose criterios a analizar: función objetivo (que combina costos fijos y variables), capacidad utilizada de los camiones, número de rutas (que permite determinar el tamaño de la flota) y tiempo de solución. Se realizaron concentrados de los resultados bajo los criterios mencionados. En el disco en el archivo

Concentrado de resultados se presentan lo obtenido a través de correr el modelo en GAMS para todos los grupos de instancias.

En la Tabla 5. 4, se muestran los resultados arrojados por el software comercial GAMS bajo la plataforma de CPLEX.

Tabla 5. 4. Resultados con el software comercial GAMS

Grupo de Instancias	Instancia	Función objetivo	Capacidad Utilizada de los vehículos	Número de rutas	Tiempo de solución promedio	Tipo de vehículo usado
1 (18 Nodos)	A	387.76	97.12%	5	15 min	El 100% de las rutas se realizan con el vehículo 1
	B	384.69	94.49%	5	15 min	
	C	384.45	93.32%	5	15 min	
	D	382.19	90.67%	5	15 min	
	E	360.29	99.96%	5	15 min	
2 (30 Nodos)	A	681.31	91.22%	9	24 min	El 97.72 % de las rutas se realizan con el vehículo 1 y 2.28% de las rutas con el vehículo 2
	B	690.07	91.22%	9	24 min	
	C	680.17	96.27%	8	24 min	
	D	679.46	91.22%	9	24 min	
	E	694.85	91.22%	9	24 min	
3 (50 Nodos)	A	768.7	94.75%	10	30 min	El 100% de las rutas se realizan con el vehículo 1
	B	776.64	93.02%	10	30 min	
	C	777.25	94.75%	10	30 min	
	D	784.82	93.08%	10	30 min	
	E	787.82	94.75%	10	30 min	
4 (80 Nodos)	A	3018.73	81.35%	37	60 min	El 87 % de las rutas con vehículo 1 y el 13% de las rutas con el vehículo 2
	B	3196.61	83.81%	40	60 min	
	C	3051.65	86.96%	38	60 min	
	D	3157.5	84.43%	40	60 min	
	E	3039.95	90.39%	35	60 min	
5 (100 Nodos)	A	2550.94	74.36%	34	60 min	En todas las rutas se usa el vehículo 1
	B	2429.75	81.28%	32	60 min	
	C	2579.84	76.50%	34	60 min	
	D	2538.97	76.50%	34	60 min	
	E	2658.44	76.34%	35	60 min	

5.3 Diseño de una heurística de segmentación

Una vez diseñada la heurística de segmentación y aplicándola a las instancias definidas se obtuvo lo que se muestra en Tabla 5. 5. Cabe mencionar que el número de nodos a incluir en cada segmento fue considerando la restricción que pone el software comercial, pero además se considerado que el número de nodos a incluir en cada segmento fuera el mismo, es decir del mismo tamaño.

Tabla 5. 5. Segmentos conformados por instancia

Grupo de Instancias	Número de nodos	Número de segmentos conformados	Nodos contemplados en cada segmento
1	18	2	9
2	30	2	15
3	50	2	25
4	80	4	20
5	100	5	20

En la Figura 5. 1. Segmentos conformados para el G30Ase muestra una gráfica con los segmentos conformados para el grupo de instancia de 30 nodos y para la instancia A (G30A) con la aplicación de la heurística de segmentación, utilizando un color para cada segmento para las instancias consideradas en la investigación. En el **anexo D** se muestran las graficas para cada instancia de los segmentos conformados para el grupo de instancias de planteadas en la investigación.

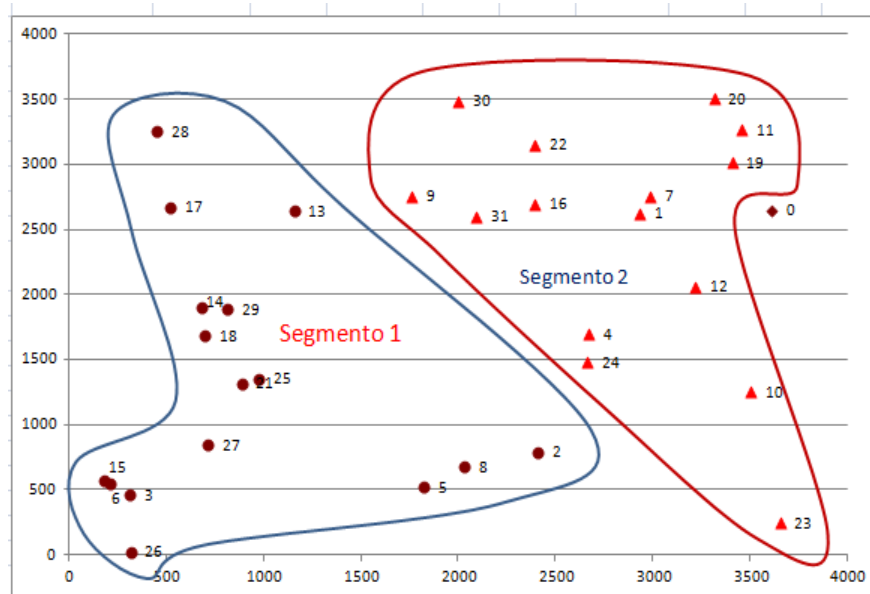


Figura 5. 1. Segmentos conformados para el G30A

Una vez conformados los segmentos para todos los grupos de instancias e instancias, se procedió resolver para cada uno con el modelo de programación matemática a plantearlos en el modelo integrando en los indicadores de la solución global resultados de todos los segmentos. En el disco adjunto se encuentran los resultados una vez segmentadas las instancias y que se corrieron en GAMS, cuyo archivo es Concentrado de Resultados. En la Tabla 5. 6, se presentan los resultados para cada instancia con la segmentación.

Tabla 5. 6. Resultados obtenidos para cada instancia con la segmentación

Grupo de Instancias	Instancia	Función objetivo	Capacidad Utilizada de los vehículos	Número de rutas	Tiempo de solución promedio	Tipo de vehículos usados
1 (18 Nodos)	A	382.54	87.42%	5	12 min	El 92.31% de las rutas se utiliza el vehículo 1 y el 7.69% el vehículo 2
	B	458.60	75.85%	6	12 min	
	C	432.85	73.97%	5	12 min	
	D	312.21	81.06%	4	12 min	
	E	444.75	73.33%	5	12 min	
2 (30 Nodos)	A	664.68	91.10%	8	12 min	En el 88% de las rutas se utiliza el vehículo 1 y
	B	690.82	91.48%	9	12 min	
	C	729.33	95.16%	8	12 min	
	D	715.34	90.74%	8	12 min	

	E	689.58	91.67%	9	12 min	el 12% el vehículo 2
3 (50 Nodos)	A	762.67	93.28%	10	15 min	En el 96 % de las rutas se utiliza el vehículo 1 y el 4% de las rutas utiliza el vehículo 2
	B	774.69	94.75%	10	15 min	
	C	770.90	93.60%	10	15 min	
	D	833.68	89.91%	10	15 min	
	E	836.34	91.39%	10	15 min	
4 (80 Nodos)	A	2735.77	91.56%	24	15 min	En el 42.86% se utiliza el vehículo 1, en el 57.14% se utiliza el vehículo 2 y el 0.79% se utiliza el vehículo 3
	B	2906.86	86.48%	28	15 min	
	C	2571.19	92.99%	23	15 min	
	D	2761.81	89.96%	25	15 min	
	E	2872.02	88.15%	26	15 min	
5 (100 Nodos)	A	2181.77	87.57%	29	12 min	En el 100% de las rutas se usa el vehículo 1
	B	2129.36	92.38%	28	12 min	
	C	2278.47	86.16%	30	12 min	
	D	2168.52	88.38%	30	12 min	
	E	2212.79	88.76%	29	12 min	

5.4. Aplicación de algoritmos genéticos usando un vector genotipo aleatorio para la visita de los clientes.

A partir de la configuración establecida para la aplicación de Algoritmos Genéticos considerando como vector las rutas, se obtuvieron resultados, los cuales se presentan en la Tabla 5. 7

Tabla 5. 7. Resultados obtenidos con Algoritmos Genéticos

Grupo de Instancias	Instancia	Función objetivo	Capacidad Utilizada de los vehículos	Número de rutas
1 (18 Nodos)	A	381.69	93.62%	5
	B	379.22	91.87%	5
	C	385.39	90.34%	5
	D	381.38	87.03%	5
	E	388.47	95.11%	5
2 (30)	A	688.2	96.70%	8
	B	694.10	89.81%	9

Nodos)	C	691.30	91.21%	9
	D	690.65	91.21%	9
	E	697.02	91.21%	9
3 (50 Nodos)	A	808.30	94.83%	10
	B	809.10	94.83%	10
	C	812.09	94.75%	10
	D	810.10	94.75%	10
	E	805.10	93.84%	10
4 (80 Nodos)	A	3244.19	82.90%	32
	B	3139.67	82.04%	35
	C	2863.29	89.87%	24
	D	3103.37	88.12%	22
	E	3123.33	81.27%	35
5 (100 Nodos)	A	2376.84	86.52%	30
	B	2259.36	89.69%	29
	C	2494.70	85.77%	29
	D	2260.57	89.12%	29
	E	2429.01	89.14%	29

La aplicación del algoritmo genético para problemas de ruteo vehicular bajo flota heterogénea, proporciona los resultados que se presentan en la Tabla 5. 8.

Tabla 5. 8. Resultados obtenidos mediante Algoritmos Genéticos en las instancias de FSMVRP

Grupo de Instancia	Instancia	Número de clientes	Algoritmos genéticos		
			Tiempo	Número de generaciones	Solución encontrada
1	A	18	0.75 min	200	391.29
2	A	31	1.10 min	200	698.13
3	A	50	1.79 min	200	812.67
4	A	80	2.70 min	200	2851.91
5	A	100	3.85 min	200	2217.66

Al comparar los resultados en las tablas 5.2, 5.3 y 5.8, se puede observar que en lo referente a softwares comerciales, el CPLEX 11.2 presenta una clara superioridad al LINGO 12, aunque desafortunadamente CPLEX 11.2 es tres veces más caro que LINGO y representa una erogación más grande para cualquier empresa.

Se puede observar que según los datos presentados en la tabla 3 y 4, se puede contrastar que las soluciones encontradas mediante algoritmos genéticos para las distintas instancias son

equiparables a los encontrados mediante el software comercial de CPLEX 11.2, lo cual se muestra en la **Tabla 5. 9**.

Tabla 5. 9. Comparativo entre CPLEX y Algoritmos Genéticos en las instancias de FSMVRP

Instancia	Número de clientes	Solución encontrada en CPLEX*	Solución encontrada por Algoritmos genéticos	Diferencia porcentual
G18A	18	381.22	391.29	2.6 %
G31A	31	681.31	698.13	2.5 %
G50A	50	768.70	812.67	5.7 %
G80A	80	3018.73	2851.91	- 5.5%
G100A	100	2550.94	2217.66	-13.1 %

*Los resultados obtenidos fueron con un tiempo límite

Tal y como se puede apreciar en la tabla 5.9 para instancias pequeñas el software comercial de optimización proporciona resultados mejores que con el método de algoritmos genéticos; situación que se revierte para las últimas dos instancias que se resolvieron con un tiempo límite de 60 minutos en el optimizador comercial. Los valores de la función objetivo son los obtenidos con el tiempo límite, lo cual se debe a que al correr el modelo sin tiempo límite tarda en generar una solución óptima. La solución obtenida es la que genera el optimizador como la mejor, pero no la óptima.

Observando la tabla 5.9, y realizando un comparativo en los tiempos de ejecución que se muestran en las tablas 5.3 y 5.8 es fácil concluir que la propuesta de meta heurística de algoritmos genéticos presenta una superioridad en cuanto al tiempo en que genera la solución con respecto al software CPLEX; y si a esto se agrega los resultados observados en la tabla 5.2, igualmente se puede ampliar la afirmación de que la metaheurística propuesta presenta superioridad a los softwares comerciales de CPLEX 11.2 y LINGO 12 al generar una solución en un tiempo corto y cuya calidad en la solución es buena. Especialmente si se considera los costos de adquisición de estos softwares, contra la prácticamente nula erogación (solo del Microsoft EXCEL) para el uso de los algoritmos genéticos; esto adicional a su rápido desenvolvimiento, pues llega a una solución buena en menos de 5 minutos.

5.5 Comparativo de soluciones generadas por las tres metodologías de solución utilizadas

A partir de los resultados obtenidos se realizaron comparativos entre lo obtenido con el software, la heurística y metaheurística, para lo cual se compararon de acuerdo a lo arrojado en la función objetivo, capacidad utilizada de los vehículo y el número de rutas. En la Figura 5. 2 se presenta el comparativo de la función objetivo obtenido con las tres metodologías de solución.

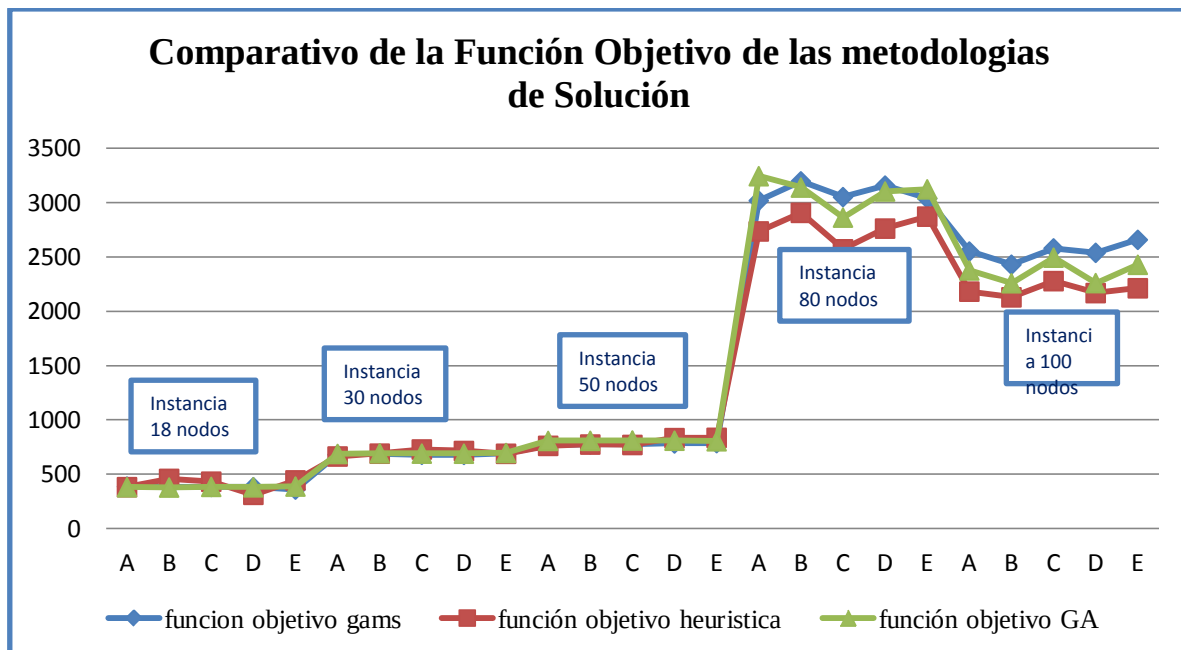


Figura 5. 2. Comparativo de la función objetivo arrojada por los métodos de solución aplicados

De acuerdo a lo que se muestra en la Figura 5. 2, se puede observar que la función objetivo que generó la aplicación de la heurística de segmentación se comporta de manera similar a los resultados obtenidos para el grupo de instancias de 18, 31 y 50 nodos y para las instancias de 80 y 100 nodos supera a los arrojados por el software comercial y la metaheurística de Algoritmos genéticos. Los segmentos son de tamaño tal que pueden resolverse rápidamente y con una buena calidad, asegurada por los métodos exactos del software comercial.

Otro criterio analizado fue la capacidad utilizada de los vehículos, para ello se presenta la Figura 5. 3.

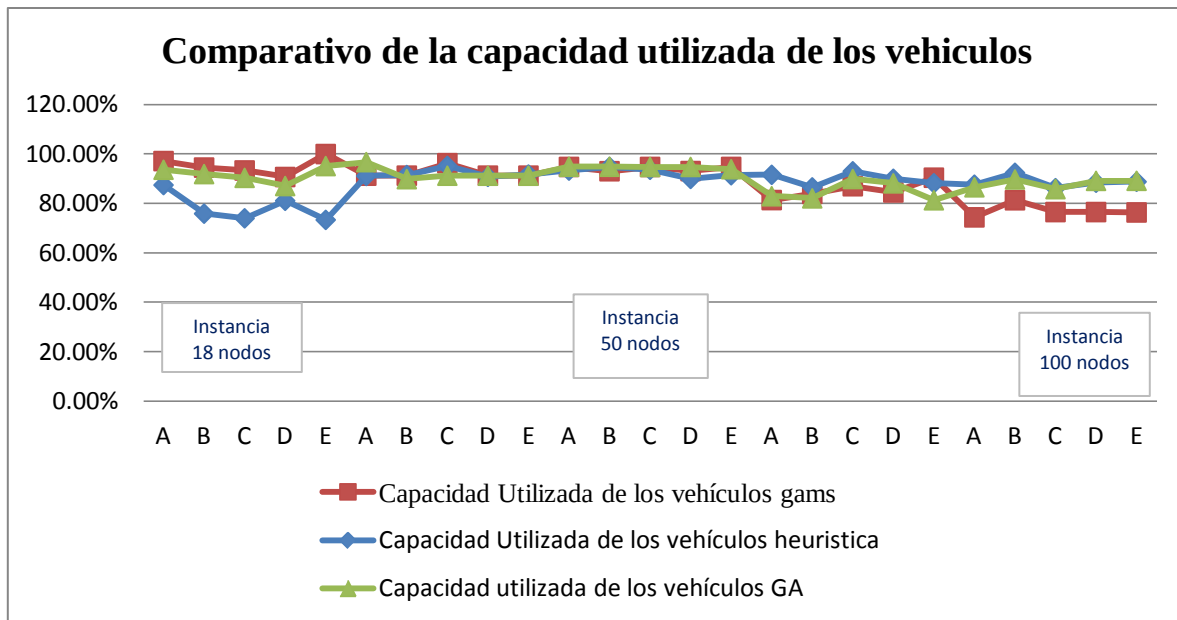


Figura 5. 3. Comparativo de la capacidad utilizada de los vehículos por los métodos de solución aplicados

Para las instancias de 18 nodos, los resultados de la heurística están por debajo de proporcionados por el software comercial manejando un tiempo límite y algoritmos genéticos, y para las instancias de 30, 50 y 80 nodos los resultados son muy parecidos. Para la instancia de 100 nodos se tiene un mejor aprovechamiento de la capacidad por el software comercial y algoritmos genéticos que son muy similares.

En relación al número de rutas que establece el tamaño de la flota se analizaron el número de rutas que genera el software comercial y el obtenido por la heurística, en la Figura 5. 4 se muestra este comportamiento.

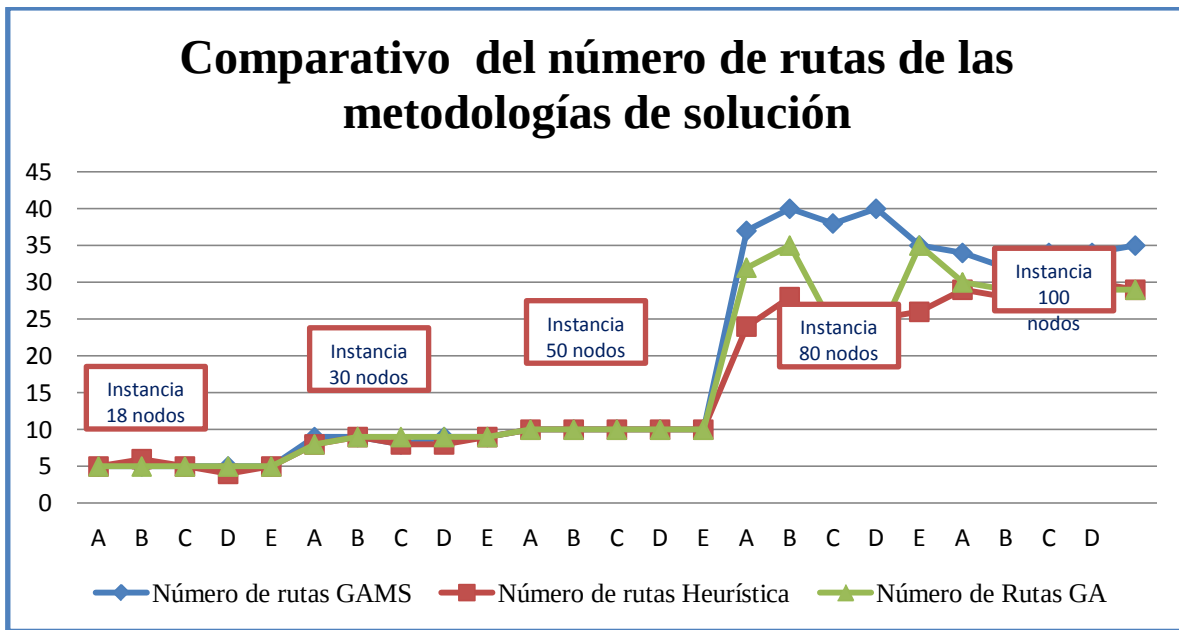


Figura 5. 4. Comparativo del número de rutas arrojada por los métodos de solución aplicados.

De acuerdo a los resultados que se muestran en la gráfica anterior, se puede establecer que a través de la aplicación de heurística de segmentación se tiene un menor número de rutas, impactando en un tamaño de flota mínima pequeña, lo cual permite tener la requerida para atender la demanda de los clientes a un bajo costo. Este comportamiento fue similar para todas las instancias.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

En cuanto al problema de ruteo vehicular, en esta investigación se abordan las decisiones estratégicas relacionadas al tamaño de la flota, que puede ser homogénea o heterogénea; se concluye que para tener la atención requerida no es necesario tener más de tres tipos de vehículos; los cuales permiten atender la demanda a un bajo costo, con un número mínimo de rutas en donde se aprovecha la capacidad de los vehículos.

Un aspecto muy importante a considerar al seleccionar el tipo de flotilla, es que a medida que la cantidad de clientes aumenta (50 o más) la tendencia es a utilizar un solo tipo de vehículo; pero contrariamente a lo que otros autores establecen sobre que es más conveniente utilizar un vehículo muy grande (ya que el costo es distribuido entre el número de unidades), en esta investigación no fue así, ya que la selección fue en general sobre el vehículo más pequeño y económico; debido a que daba el mayor rendimiento y el menor costo fijo.

En lo referente a la solución de problemas VRP con métodos exactos, en esta investigación, se muestra que al plantear el modelo de Olivera para flota heterogénea, el número de variables y de restricciones se incrementa geométricamente, lo cual impacta directamente en el tiempo que requiere el modelo para generar soluciones.

Se evidenció que en problemas de ruteo vehicular al usar métodos exactos, es posible encontrar soluciones factibles, pero no llegar al óptimo; lo cual se concluyó al evaluar diversos softwares. El software Lingo 12 en su versión industrial, tiene serias limitaciones por la gran cantidad de variables enteras que requiere el planteamiento del VRP y presenta aun para casos reducidos, incapacidad de resolver problemas de ruteo vehicular con flota heterogénea. Los softwares CPLEX 11.2 y el SCIP 1.1 son las mejores alternativas para la solución de los problemas de ruteo vehicular; siendo el primero la mejor

alternativa, aunque presenta un costo comercial de casi \$9000 usd que es tres veces mayor al de SCIP 1.1

Se concluye que la diversidad de desempeños en los software analizados Lingo 12, CPLEX 11.2 y SCIP 1.1, está relacionada a las heurísticas y estrategias nodales que este contemple en su ejecución, por lo que estas representan una característica crucial para un software de programación lineal; pues su adecuado involucramiento puede agilizar mucho la resolución de problemas MIP con características de alta complejidad combinatoria, tal y como corresponde a los problemas de ruteo vehicular (VRP). En este sentido CPLEX es claramente muy superior a las otras alternativas estudiadas.

En cuanto al uso de heurísticas en la solución de problemas VRP, esta investigación mostró que el diseño de la heurística considerando como criterio para la conformación de los segmentos: la distancia, permitió agrupar los nodos a partir de considerar un almacén y atendiendo los factores considerados en el diseño de la misma, permitió reducir los costos, lo cual se debe a que el costo variable está asociado a la distancia que recorre cada uno de los vehículos para visitar a los clientes. La heurística generó mejores resultados en un tiempo menor debido que al segmentar y correrlos en el software comercial se obtuvieron mejores soluciones en un corto tiempo.

Una de las desventajas de la heurística de segmentación es que se requiere tiempo para conformar los segmentos. Además de la disminución de costos, se visualiza que el número de vehículos que se requieren para la distribución disminuye y se aprovecha la capacidad de los vehículos, lo cual representa un aspecto importante en la administración de la flota.

En lo referente al uso de metaheurísticas en la solución de problemas de ruteo vehicular, en esta investigación se demuestra que el uso de la metaheurística propuesta y que se basa en algoritmos genéticos resultó superior en su rendimiento al ser comparado con los métodos exactos, pues aun para instancias grandes produce en pocos minutos resultados aceptables. Además la utilización de esta metaheurística puede representar una

ventaja para la industria considerando la inversión inicial que representa un software comercial.

Se demuestra que para la adecuada resolución de problemas de ruteo vehicular con flota heterogénea (FSMVRP) por medio de la metaheurística de algoritmos genéticos (GA) resulta adecuado el enfoque de representar el vector de genotipo mediante una cadena de números enteros, junto con un vector de utilización de tipo de vehículos y además de una función de codificación que al interpretarse rutinariamente genere individuos válidos.

Se evidencia que al usar una metaheurística de algoritmos genéticos adaptando el vector genotipo a partir de generar rutas aleatorias, permite tener soluciones en tiempos cortos y de buena calidad en relación a las generadas por el método exacto en tiempos largos. Y en base a los resultados de colocar como semilla inicial la generada por la metaheurística en el software CPLEX, la solución obtenida por este último se mejoró en un porcentaje bajo, lo cual indica que la metaheurística genera soluciones adecuadas.

El resolver este problema por los tres métodos de solución permite ver la diferencia que existe en las soluciones al compararlas de acuerdo a los criterios considerados. El uso de la metaheurística propuesta de algoritmos genéticos es más rápida y económica, pero es mejor en calidad la heurística de segmentación, y es relativamente razonable en tiempo, aún para los casos de 100 nodos, con la desventaja de tener que formar los segmentos como un paso inicial.

Recomendaciones

El problema de ruteo vehicular está adquiriendo cada vez más relevancia en la reducción de los costos logísticos relacionados con los procesos de transportación de materiales y de productos; por lo que el empleo de estrategias que permitan reducir estos

impacta en una administración eficiente de los recursos. En estas estrategias el tiempo de solución es un factor clave, por lo que se requiere el empleo de herramientas de solución que garanticen la optimización (o un buen sub-óptimo) y tener con ello un respaldo que apoye a la toma de decisiones: mismo que deberá ser realizado a un costo que no resulte prohibitivo para las empresas; lo cual en su conjunto abre las puertas a proyectos novedosos y áreas de investigación.

El enfoque de resolución de problemas de ruteo vehicular con flota heterogénea, con la utilización de distintos softwares de programación lineal entera puede ser adecuado para problemas de tamaño reducido, pero insuficiente para problemas de tamaño moderado (más de 50 clientes) y prohibitivo para problemas de mayor capacidad (más de 80 clientes); no solo por los resultados obtenidos, sino por el tiempo de obtención de los mismos. Se considera adecuada la utilización de métodos alternos en forma de una heurística de segmentación y de una metaheurística de algoritmos genéticos para abordar el problema de ruteo vehicular; lo que demuestra que el estudio y análisis de heurísticas y metaheurísticas para el problema de ruteo vehicular representa un área de interés para desarrollar y lograr involucrar distintas restricciones que pueden encontrarse en problemáticas reales. Uno de los trabajos a desarrollar es una revisión exhaustiva de la formulación matemática de estos modelos que generan el no obtener soluciones en corto tiempo. Una de los trabajos no considerados en la investigación y que se pueden realizar es el desarrollar la aplicación de algoritmos genéticos a través del código binario para tamaños de instancias grandes, lo cual para este tipo de problemas resulta complejo al momento de establecer el tipo de vehículo a usar cuando son más de dos vehículos.

Este tipo de investigaciones provee de información en las decisiones de tipo estratégicos relacionadas a la adquisición de un tamaño y tipo de flotilla acorde a sus necesidades, que permita a las empresas atender la demanda de sus clientes a bajos costos; y en decisiones de tipo operativo para la planeación de rutas en donde intervienen factores como costos, aprovechamiento de la capacidad de los vehículos, tamaño de la flota y entregas a clientes en tiempo y forma; evitando situaciones en las que se tengan vehículos sin ser utilizados. Es importante que al considerar el uso de una flota que varíe en capacidades se tome en cuenta

el uso de metodologías de solución que ayuden no solo a resolver la problemática específica, sino que esto sea en tiempo razonable y a un costo que no sea prohibitivo para los empresarios como lo representan la mayoría de los métodos exactos; por lo que la investigación y avances en esta área son muy importantes y tienen una gran proyección en el largo plazo. Partiendo de lo anterior, es necesario realizar un análisis de la forma en cómo se estiman los costos considerados en el modelo, así como la forma el modelo selecciona al vehículo en términos de costos, debiéndose a que en la bibliografía se establece que siempre se selecciona al vehículo más grande ya que de esta manera se optimizan los costos.

En estos tiempos en que los costos de hidrocarburos mantienen un aumento escalonado, puede ocurrir que el vehículo con menor capacidad sea el que puede representar mayor rendimiento y menor costo fijo, siendo el más competitivo en relación a los demás, mientras que el vehículo más grande, el rendimiento más bajo y no sea opción en la minimización de costos tal como se enfoca el modelo. Dado que en el modelo se contemplan costos fijos y variables, asociados a la distancia se tiene que selecciona aquel vehículo con costo fijo más bajo en vez de minimizar las distancias recorridas, atendiendo la capacidad del mismo. Esto requiere un mayor estudio y genera una línea de investigación, ya que algunas empresas lo consideran como la estrategia a seguir principalmente porque se parte que se prorratea el costo en un mayor número de unidades, bajando por ende el costo de la distribución de estos productos.

REFERENCIAS BIBLIOGRÁFICAS

Achterberg, T., (2007), “Constraint Integer Programming”, Ph.D. thesis, Berlin, http://opus.kobv.de/tuberlin/volltexte/2007/1611/pdf/achterberg_tobias.pdf, con acceso en enero 2012.

Achterberg, T., (2009), “SCIP: Solving constraint integer programs”, Mathematical Programming Computation 2(1):1-41.

Albareda, M., Díaz, J., Fernández, E. y Martínez, C. (2003). Problemas combinados de localización de rutas. V Seminario Español de Localización “Avances en localización de centros de servicio y sus aplicaciones”, Lorca, Disponible en: <http://www.eio.upc.es/personal/homepages/elena/Web/memoria2.htm>, Noviembre 2003.

Archetti, C., Mansini, R. and Speranza, M., (2001). The Split Delivery Vehicle Routing Problem with Small Capacity, November 7, 2001, Disponible en: http://fausto.eco.unibs.it/~www_megu/ricerca/quaderni/201.pdf

Atamtürk, A., Salvelsbergh, M., “INTEGER–PROGRAMMING SOFTWARE SYSTEMS”, Springer, Annals of Operations Research, 140(1):67-124

Ballou, R., (2004). Busssines Logistics/Supply Chain Management. 5ª ed. Prentice Hall. Upper Saddle River. New Jersey.

Bianchi, L., Metaheuristics for the Vehicle Routing Problem with Stochastic Demands; S/F., Disponible en: <http://www.idsia.ch/idsiareport/IDSIA-06-04.pdf>.

Boubeta, A., (2007), “Distribución logística y comercial. La logística en la empresa”, Gesbiblo S. L, España.

Caballero, R., Molina, J., Gómez, T., Luque, M., Torrico, A., (2002), “A Genetic Algorithm to Solve an Integer Goal Programming Model for the Higher Education”, EU/ME European Chapter On Metaheuristics, France.

Choi, E. and Tcha D., (2007), “A column generation approach to the heterogeneous fleet vehicle routing problem”, Journal Computers and Operations Research, 34(7):2080-2095

CPLEX, (2012a), “IBM ILOG CPLEX Optimization Studio - View US prices & buy”, http://www42.ibm.com/software/dre/ecatalog/detail.wss?locale=en_US&synkey=D159451Q14719W46 con acceso en Enero 2012.

CPLEX, (2012b), “IBM ILOG CPLEX Optimization Studio Preview Edition Trial”, <http://www-01.ibm.com/software/websphere/products/optimization/cplex-studio-preview-edition/> con acceso en Enero 2012.

CPLEX, (2012c), “IBM ILOG CPLEX V12.1, User's Manual for CPLEX”, ftp://ftp.software.ibm.com/software/websphere/ilog/docs/optimization/cplex/ps_usrman_cplex.pdf, con acceso en Enero 2012.

Correa, F., Gómez, H., Loaiza, J., Lopera, C., Villegas, J., (2008), “Características del diseño de rutas de distribución de alimentos en el Valle de Aburrá”, *Revista Facultad de Ingeniería Universidad de Antioquia*, 45:172-183.

Correa, F., Gómez, H., Loaiza, J., Villegas, J., (2007). “Aplicaciones prácticas de los problemas de rutas de vehículos”. Reporte de Investigación, Departamento de Ingeniería Industrial, Facultad de Ingeniería – Universidad de Antioquia.

Daza, J, Montoya, J., Narducci, F., (2009), “Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases”, *Revista EIA, Escuela de Ingeniería de Antioquia*, 12:23-38, Colombia.

Díaz, A., Glover, F., Ghaziri, M., (1996). *Optimización Heurística y Redes Neuronales*. Madrid, Paraninfo.

Dullaert, W., Janssens, G. K., Sörensen, K., Vernimmen, B. (2002). “New heuristics for the Fleet Size and Mix Vehicle Routing Problem with Time Windows”, *Journal of the Operational Research Society*, 53(11):1232-1238.

Eidelman, P. y Valdez, A., (2007), “Sistema de optimización para el ruteo dinámico de vehículos con ventanas de tiempo”, Tesis, encontrada en <http://www.midnightsort.com.ar/personales/alejandrovaldez/tesis/archivos/tesis-informe.doc> con acceso en Enero 2012.

Everitt, B., (1980). *Cluster analysis*, 2nd edition. London: Social Science Research Council.

Everitt, B., Landau, S., and Leese, M. (2001). *Cluster analysis*, 4th edition. London :

Francis, P., Smilowitz, K. and Tzur, M., (2004). The period vehicle routing problem with service choice; 2004., Disponible en: http://www.iems.northwestern.edu/images/PDF/WP_04_005.pdf,

Fischetti, M., Lodi, A., Salvagnin, D., (2009), “Just MIP it!”, en “Matheuristics: Hybridizing Metaheuristics and Mathematical Programming”, Maniezzo V., Stautzle T., Voss S., Eds, *Annals of Information Systems* 10:39-70.

GAMS, (2012), “GAMS Sales Support”, <http://www.gams.com/sales/sales.htm> con acceso en Enero 2012.

González, G. y González, F., (2006), Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 1: formulación del problema, Ingeniería e Investigación, Universidad Nacional de Colombia, Colombia, pp. 149-156.

González, G. y González F., (2007),” Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 2: algoritmo genético, comparación con una solución heurística”, Ingeniería e Investigación, 27(1):149-157

Gordon, A. (1999). Classification, 2nd edition . London, UK : Chapman and Hall/CRC Press.

Gupta, S., Garg, L., (2009), “An Improved Genetic Algorithm Based on Adaptive Repair Operator for Solving the Knapsack Problem”, Journal of Computer Science, 5(8):544-547.

Hao, X., (2010), “A fuel consumption objective of VRP and the genetic Algorithm”, (ICEEE) International Conference on E-Product E-Service and E-Entertainment, pp. 1-4

Hansen, P. and Jaumard, B., (1997). Cluster analysis and mathematical programming, Mathematical Programming , 79 : 191 – 215 .

Hasle, G., (2003). Heuristics for Rich VRP Models ROUTE 2003. International Workshop On Vehicle Routing. Skodsborg. Denmark. 2003. www1.ctt.dtu.dk/route2003/presentations/Geir%20Hasle.ppt . Consultada el 15 de noviembre de 2007.

Hoff, A., Andersson H., Christiansen M., Hasle G., Løkketangen A., (2010), “Industrial aspects and literature survey: Fleet composition and routing”, Computers and Operations Research, 37(12):2041-2061

Hu, X., Huang, M. and Zeng Z., (2007), “AN INTELLIGENT SOLUTION SYSTEM FOR A VEHICLE ROUTING PROBLEM IN URBAN DISTRIBUTION”, International Journal of Innovative Computing, 3(1):189-198

Hvattum, M., Lokketangen A., (2012), “Comparisons of Commercial MIP Solvers and an Adaptive Memory (Tabu Search) Procedure for a Class of 0-1 Integer Programming Problems”, en proceso para aparecer en Annals of Operations Research, http://leeds-faculty.colorado.edu/glover/500%20-%20BOOP3_ANOR%20v3%2001.pdf, con acceso en Enero 2012.

Ignizio, J. and Cavalier, T., (1994). Linear programming. Englewood Cliffs, NJ: Prentice Hall.

Jain, A. and Dubes , R., (1988). Algorithms for clustering data . Englewood Cliffs, NJ : Prentice Hall.

Jaque, A., (2008), “Métodos Aproximados para la Solución del Problema de Enrutamiento de Vehículos”, texto encontrado en <http://andresjaquep.files.wordpress.com/2008/12/estado-del-arte-vrp1.pdf> con acceso en enero 2012

Lee, Ch., Epelman, M., Chelsea, C. and Bozer Y., (2002). A Shortest Path Approach to the Multiple-Vehicle Routing Problem with Split Pick-Ups, 2002, Disponible en: http://osiris.tuwien.ac.at/~wgarn/VehicleRouting/mvrpsp_ts.pdf.

Lindo, (2012). “Pricing information - Lindo Systems Inc”, http://www.lindo.com/index.php?option=com_content&view=article&id=114&Itemid=40 con acceso en Enero 2012.

Lima, C., Goldbarg M., Goldbarg E., (2004), “A Memetic Algorithm for the Heterogeneous Fleet Vehicle Routing Problem”, *Electronic Notes in Discrete Mathematics*, 18(5-6): 171-176.

Lodi, A., (2008), “MIP Computation and beyond”, ARRIVAL Project, “50 Years of Integer Programming 1958–2008”, Liebling T., Naddef D., Pulleyblank W., Reinelt G., Rinaldi G., Wolsey L., Eds., Springer.

Loxton, R. and Lin, Q., (2011), “Optimal Fleet Composition Via Dynamic Programming And Golden Section Search”, *Journal Of Industrial And Management Optimization*, 7(4):875-890

Lüer, A., Benavente, M., Bustos, J., Venegas B., (2009), “El Problema de Rutas de Vehículos: Extensiones y Métodos de Resolución, estado del Arte”, Conferencia en Workshop internacional del 3er Encuentro de Informatica y Gestion, EIG 2009

Machado, P., Tavares, J., Pereira, F. and Costa, E., (2002), “GVR: a New Genetic Representation for the Vehicle Routing Problem”, 13th Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2002), pp. 95-102, Ireland.

Machado, P., Tavares, J., Pereira, F. y Costa, E., (2003). Crossover and Diversity: A study about GVR. (a) Memorias de Proceedings of the Analysis and Design of Representations and Operations a bird-of-a-feather workshop at the Genetic and Evolutionary Computation Conference. Chicago.

Marti, R., (2009) Procedimientos Metaheurísticos en Optimización Combinatoria. <http://www.uv.es/~rmarti/>. Departamento de Estadística e Investigación de Operaciones, Universidad de Valencia.

Martínez, A., (2010), “Ant Colony System y Asynchronous Teams para el Problema de Ruteo Capacitado de Vehículos”, *REVISTA ICHIO*, 1(2):1-24

Mejía, G. and Castro, E., (2007), “Optimización del proceso logístico en una empresa de colombiana de alimentos congelados y refrigerados”, *Revista de Ingeniería*, 26: 47-54

Melián, B., Pérez, J., (2003), *Metaheurísticas: una visión global*. Inteligencia Artificial, *Revista Iberoamericana de Inteligencia Artificial*. N.19 pp. 7-28 ISSN: 1137-3601. © AEPIA (2003). <http://www.aepia.org/revista>.

Mitchell, G., O'Donoghue, D., Barnes, D. and McCarville, M., (2003), “GeneRepair - A Repair Operator for Genetic Algorithms”, *GECCO- Genetic and Evolucionary Computation Conference*.

Olivera, A., (2004), “Heurísticas para problemas de ruteo vehículos”, *Universidad de la república, Uruguay*

Osman, I.H. and Kelly, J.P. (eds.) (1996), *Meta-Heuristics: Theory and Applications*, Ed. Kluwer Academic, Boston.

Pereira, F.B., Tavares, J., Machado, P. and Costa, E., (2002), “GVR: a New Genetic Representation for the Vehicle Routing Problem”, *13th Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2002)*, pp. 95-102, Ireland.

Pessoa, A., Uchoa, E. and Poggi of Aragão, M. (2009), “A robust branch-cut-and-price algorithm for the heterogeneous fleet vehicle routing problem”. *Networks*, 54(4):167–177

Potvin, Y., (2009), “State-of-the Art Review—Evolutionary Algorithms for Vehicle Routing”, *INFORMS Journal on Computing* Fall, 21:518-548

Prins, C., (2004), “A Simple and Effective Evolutionary Algorithm for the Vehicle Routing Problem”, *journal of Computers & Operations Research*, 31 (2004): 1985–2002

Renaud, J. and Boctor, F., (2002), “A sweep-based algorithm for the fleet size and mix vehicle routing problem”, *European Journal of Operational Research*, 140(3):618-628.

Riopel, D., Langevin, A. and Campbell, J., (2005). “The Network of Logistics Decisions”. *Logistics Systems: Design and Optimization*. New York. Springer. pp. 1-38.

SCIP, (2012), “SCIP: License - What Are The License Terms?”, <http://scip.zib.de/licence.shtml> con acceso en enero 2012.

Rodríguez, V.M.B., (2003), “Manipulación Genética aplicada al problema del agente viajero”, *Tesis de maestría, Instituto Politécnico Nacional*

Soto, D., Soto, W. y Pinzón, Y., (2008), Una metaheurística híbrida aplicada a un problema de planificación de rutas. *Revista Avances en Sistemas e Informática*, Vol. 5 No. 3, Medellín.

Taillard, E., (1999), “A heuristic column generation method for the heterogeneous fleet VRP”, *RAIRO Rech Opér.* 33(1), pp 1-14 .

Toth, P., Vigo, D. (2000). An Overview of Vehicle Routing Problems. Monographs on Discrete Mathematics and Applications. In: The Vehicle Routing Problem. SIAM

Ventoso, J., (2001), “ Algoritmos Genéticos (I)”, *Revista Sólo Programadores*, 89:40-46

Villardón, J. (s.f). Introducción al análisis de clúster.

Xu, L., Lim M.H. and Er M.J., (2005), “Investigation on Genetic Representations for Vehicle Routing Problem”, *IEEE Internacional conference on systems, Man and cybernetics*, 4:3083-3088

Yadlapalli, K., Rathinam, S. and Darbha S., (2009), “An approximation algorithm for a 2-depot, heterogeneous vehicle routing problem”, *ACC'09 Proceedings of the 2009 conference on American Control Conference*, 1730-1735.

Yepes, V y Medina, J.,(2002). Optimización de rutas con flotas heterogéneas y múltiples usos de vehículos VRPHEMSTW, SEMNI. España.

ANEXO A

Datos reales

Demanda de los clientes expresada en peso (kg) y volumen (cm3)

DESTINO	PESO	VOLUMEN
1	242	1
2	159	1
3	389	2
4	374	2
5	277	2
6	34081	204
7	34633	208
8	14084	85
9	13274	80
10	52	0
11	81	0
12	105	1
13	2334	14
14	2311	14
15	1642	10
16	2861	17
17	14730	88
18	14671	88
19	1216	7
20	1168	7
21	2885	17
22	1781	11
23	707	4
24	768	5
25	209	1

Información sobre los vehículos que contempla la empresa

TIPOS DE CAMIONES	CAPACIDAD	PESO VEHICULAR CAJA SIN CARGA	VOLUMEN	GASTO DE GAS X KM (NUEVO)	DETERIORO (Reemplazo)
1 TON	1 TON	800 KGS	5 M3	8 km	MOTOR 5 AÑOS
3 TON	3.5 TONS	3 TONS	12 M3	3.5 km	MOTOR 6 AÑOS
5 TONS	5 TONS	5 TONS	22M3	4 km	UNIDAD 6 AÑOS
RABON (Solo trae un eje)	8 TONS	8 TONS	40 M3	2.8 km	MOTOR A LOS 800,000 KMS
TORTON (trae 2 ejes y motor más grande)	15 TONS	8 TONS	40 M3	2.5 km	MOTOR A LOS 800,000 KMS
TRAILER	28 TONS	8 TONS	100 M3	2.5 km	MOTOR AL 1,000,000 KMS

ANEXO B. MODELO PROGRAMADO EN LINGO 12

Grupo de Instancia de 18 Nodos

MODEL:

! Modelo de flota heterogénea con 18 nodos;

SETS:

! Definición de atributos para datos y variables;

NODO_ORIGEN: R; !R = CARGA ACUMULADA EN LA RUTA;

NODO_DESTINO : DEMANDA;

VEHICULO: Q, C_fijos; !Q = CAPACIDAD DEL CAMIÓN;

VARIABLES(NODO_ORIGEN, NODO_DESTINO, VEHICULO)|

NODO_ORIGEN(&1) #NE# NODO_DESTINO(&2): C, X;

Vcosto(NODO_ORIGEN, NODO_DESTINO): Costo;

! Definición de índices para las restricciones;

NORIG_VEH(NODO_ORIGEN, VEHICULO);

NDEST_VEH(NODO_DESTINO, VEHICULO);

! pp;

ENDSETS

DATA:

! Aquí están los parámetros;

NODO_ORIGEN = @ole('c:\temp\Costos31N.xls', 'NODOS');

NODO_DESTINO = @ole('c:\temp\Costos31N.xls', 'NODOS');

VEHICULO = 1 ,2, 3;

c_fijos = 71 , 120 , 220;

Q = 18074, 27300, 27698; !Deben estar de menor a mayor carga;

costo = @OLE('c:\temp\Costos31N.xls', 'costo');

DEMANDA = @OLE('c:\temp\Costos31N.xls', 'Demanda');

ENDDATA

CALC:

ALMACEN = @SIZE(NODO_ORIGEN); !En este caso ALMACEN = 5;

VEH_GRANDE = @SIZE(VEHICULO); !En este caso VEH_GRANDE = 2;

R(@SIZE(NODO_ORIGEN)) = 0;

@FOR(NODO_ORIGEN(I):@for(nodo_destino(J)| I#NE#j: c(i,j,1)=costo(i,j)));

@FOR(NODO_ORIGEN(I):@for(nodo_destino(J)| I#NE#j: c(i,j,2)=costo(i,j)));

@FOR(NODO_ORIGEN(I):@for(nodo_destino(J)| I#NE#j: c(i,j,3)=costo(i,j)));

!@FOR(NODO_ORIGEN(I):@for(nodo_destino(J)| I#NE#j: c(i,j,4)=costo(i,j)));

ENDCALC

! Objetivo;

! No se requiere "I#NE#J" porque ya esta filtrado en la sección

SETS en VARIABLES;

[OBJ] MIN = CC1+CC2+CC3+CC4 + CCF;

cc1=((@SUM(VARIABLES(I,J,K) | K #EQ# 1: c(I,J,K) * x(I,J,K))/1000)/8)*8.52 ;
 cc2=((@SUM(VARIABLES(I,J,K) | K #EQ# 2: c(I,J,K) * x(I,J,K))/1000)/3.5)*8.52 ;
 cc3=((@SUM(VARIABLES(I,J,K) | K #EQ# 3: c(I,J,K) * x(I,J,K))/1000)/4)*8.52 ;
 !cc4=((@SUM(VARIABLES(I,J,K) | K #EQ# 4: c(I,J,K) * x(I,J,K))/1000)/2.8)*8.52 ;

CCF=@SUM(VARIABLES(I,J,K)| I #EQ# @size(nodo_origen) : c_fijos(K) * x(I,J,K));

@sum(variables(I,j,k): X (I,J,K)) =58;

!Se requiere "I #NE# J" porque no esta filtrado en la sección
 SETS en NORIG_VEH;

**!! Coincide con las fórmulas, pero no coincide con el modelo de
 4nodos BASE y son equivalentes a las siguiente instrucción:**
 @FOR(NODO_DESTINO(J)| J#NE#ALMACEN: [CLIENTE]
 @SUM(NORIG_VEH (I,K)| I #NE# J: X(I,J,K)) = 1);
 @FOR(NODO_ORIGEN(I)| I#NE#ALMACEN: [CLIENTE]
 @SUM(NDEST_VEH (J,K)| I#NE#J: X(I,J,K)) = 1);

!Se requiere "I #NE# J" porque no esta filtrado en la sección
 SETS por separado en NODO_ORIGEN, VEHICULO y NODO_DESTINO.
 En "I#NE#ALMACEN", el 5 se refiere al nodo M;
 @FOR(NODO_ORIGEN(I)| I#NE#ALMACEN:
 @FOR(VEHICULO(K): [DESTINO]
 (@SUM(NODO_DESTINO(J)| I#NE#J: X(I,J,K)) -
 @SUM(NODO_DESTINO(J)| I#NE#J: X(J,I,K))) = 0));

!R(5) = 0 que es equivalente a;

!R(@SIZE(NODO_ORIGEN)) = 0; !Está mejor en la sección CALC;

!Se requiere "I #NE# J" porque no esta filtrado en la sección
 SETS por separado en NODO_ORIGEN y NODO_DESTINO.
 En "I#NE#ALMACEN", el 5 se refiere al nodo M.

Q(2) es la carga del vehículo 2, que debe ser la carga del vehículo
 más grande;

@FOR(NODO_ORIGEN(I)| I#NE#ALMACEN: ! Para I = A,B,C y D;
 @FOR(NODO_DESTINO(J)| I#NE#J #AND# J#NE#ALMACEN: [CARGA1]
 R(J) - R(I) >= (DEMANDA(J) + Q(VEH_GRANDE)) * @SUM(VEHICULO(K):
 X(I,J,K))
 - Q(VEH_GRANDE)));

!Se requiere "I #NE# J" porque no esta filtrado en la sección
 SETS por separado en NODO_ORIGEN y NODO_DESTINO.

En "I#EQ#ALMACEN", el 5 se refiere al nodo M.

Q(2) es la carga del vehículo 2, que debe ser la carga del vehículo más grande;

```
@FOR( NODO_ORIGEN(I)| I#EQ#ALMACEN: ! Para I = M;
@FOR( NODO_DESTINO(J)| I#NE#J: [CARGA2]
  R(J) - R(I) >= (DEMANDA(J) + Q(VEH_GRANDE)) * @SUM( VEHICULO(K):
X(I,J,K))
  - Q(VEH_GRANDE) ));
```

!Se requiere "I#NE#J" porque no esta filtrado en la sección

SETS en NORIG_VEH;

```
@FOR( NODO_DESTINO(J)| J#NE#ALMACEN: [CARGA_ACUM]
  R(J) <= @SUM( NORIG_VEH (I,K)| I#NE#J: Q(K) * X(I,J,K) ) );
```

! No se requiere "I#NE#J" porque ya esta filtrado en la sección

SETS en VARIABLES;

```
@FOR( VARIABLES(I,J,K): [BINARIA]
  @BIN(X(I,J,K)) );
```

END

ANEXO C. MODELO PROGRAMADO EN GAMS 10

Grupo de Instancia de 18 Nodos

*FSMVRP 18 NODOS

sets

i cities /

\$include 18n_cli.txt

0 /

g(i) origen /0/

h(i) clientes /

\$include 18n_cli.txt /

k camiones /1*4/

alias (i,j);

table dd(i,j) distancias

\$include 18nE_dis.txt

parameter d(i) demanada por ciudad /

\$include 18n_dem_uni.txt /

parameter q(k) capacidad del vehiculo

/1 18074

2 25303

3 32533

4 39762/;

parameter f(k) Costo fijo por usar vehiculo

/1 71

2 99.4

3 127.8

4 156.2/;

parameter c(i,j,k) cost coefficients ;

parameter dd(i,j) ;

c(i,j,'1')=dd(i,j)/1000/8.0*8.52;

c(i,j,'2')=dd(i,j)/1000/3.5*8.52;

c(i,j,'3')=dd(i,j)/1000/4.0*8.52;

c(i,j,'4')=dd(i,j)/1000/2.8*8.52;

variables x(i,j,k)

z

r(i)

w(j,k);

binary variable x;

positive variable r;

equations objective

eqtn_1(j)

eqtn_2(i,k)

eqtn_3(i)

eqtn_4(i,j)

eqtn_5(i,j)

eqtn_6(j);

*1.15

objective.. z =e= sum((i,j,k),f(k)*x(i,j,k)\$g(i))+(sum((i,j,k), c(i,j,k)*x(i,j,k)));

```

*1.16
eqtn_1(j)$h(j).. sum(i, sum(k, x(i,j,k))) =e= 1;
*1.17
eqtn_2(i,k)$h(i).. sum(j, x(i,j,k)) - sum(j, x(j,i,k)) =e= 0;
*1.18
eqtn_3(i)$g(i).. r(i) =e= 0;
*1.19
eqtn_4(i,j)$h(i)*h(j).. (r(j)-r(i)) =g= ((d(j)+ smax(k,q(k))) * sum(k,x(i,j,k)))-
(smax(k,q(k)));
*1.19.1
eqtn_5(i,j)$g(i).. (r(j)-r(i)) =g= ((d(j)+ smax(k,q(k))) * sum(k,x(i,j,k)))-(smax(k,q(k)));
*1.20
eqtn_6(j)$h(j).. r(j) =l= sum(i,sum(k,x(i,j,k)*q(k)));
x.fx(i,i,k) = 0;
model fsmvrp / all /;
option optcr=0.01;
fsmvrp.reslim= 15*60;
option decimals = 2;

solve fsmvrp min z using mip;
display x.l,r.l,z.l;

```

ANEXO D. GRAFICAS DE LA SEGMENTACIÓN DE NODOS DE CADA
INSTANCIA

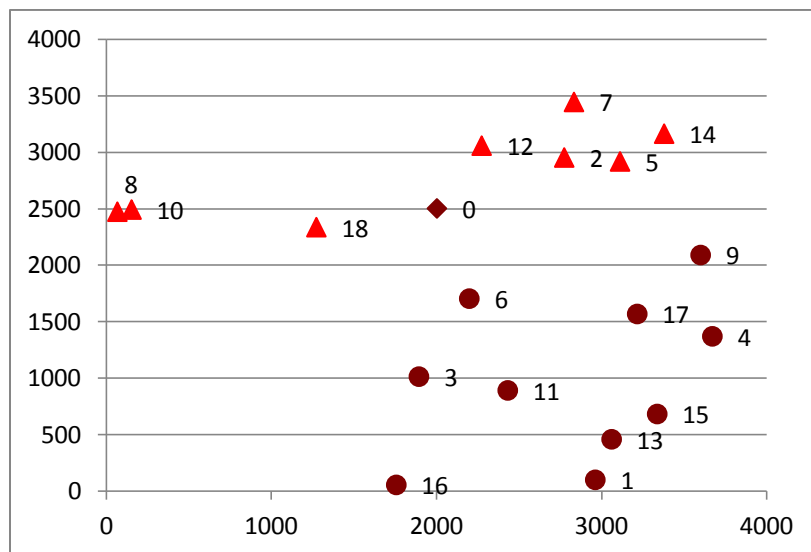


Figura D1 .Segmentación de 18 nodos de la instancia prueba A

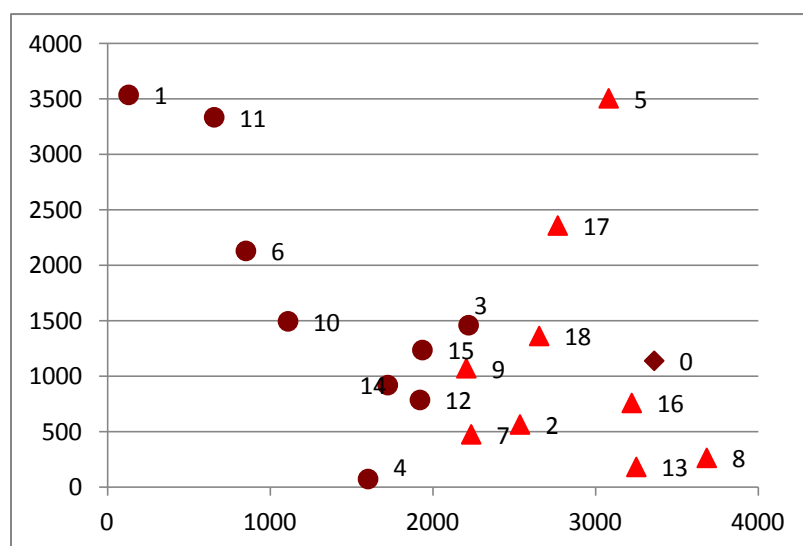


Figura D2. Segmentación de 18 nodos de la instancia prueba B

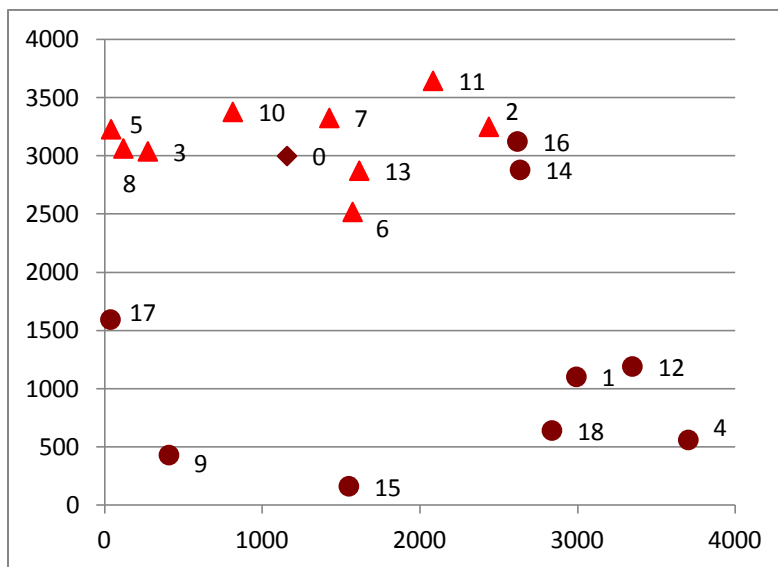


Figura D3. Segmentación de 18 nodos de la instancia prueba C

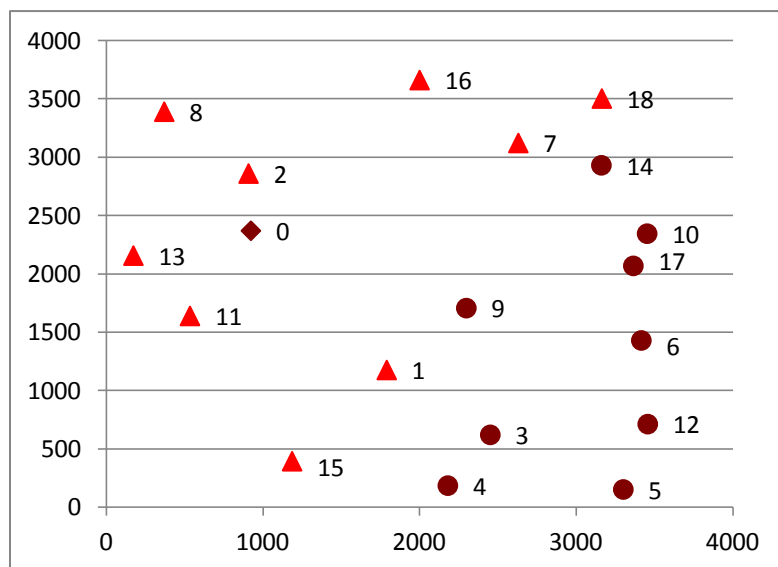


Figura D4. Segmentación de 18 nodos de la instancia prueba D

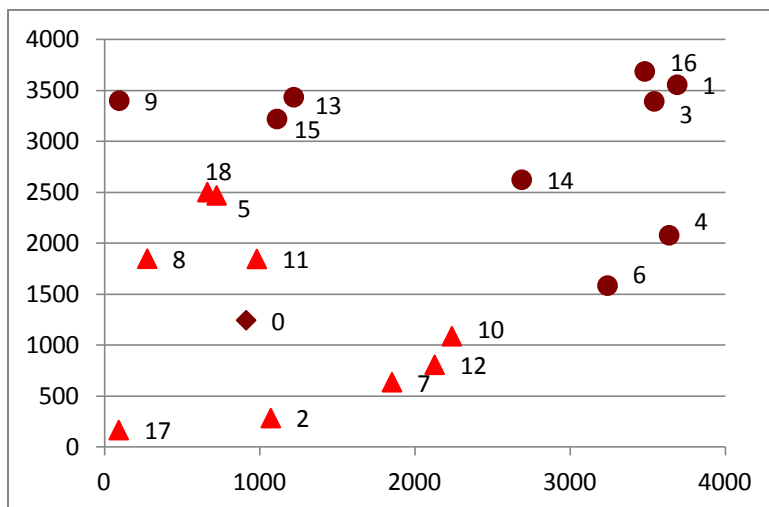


Figura D5. Segmentación de 18 nodos de la instancia prueba E

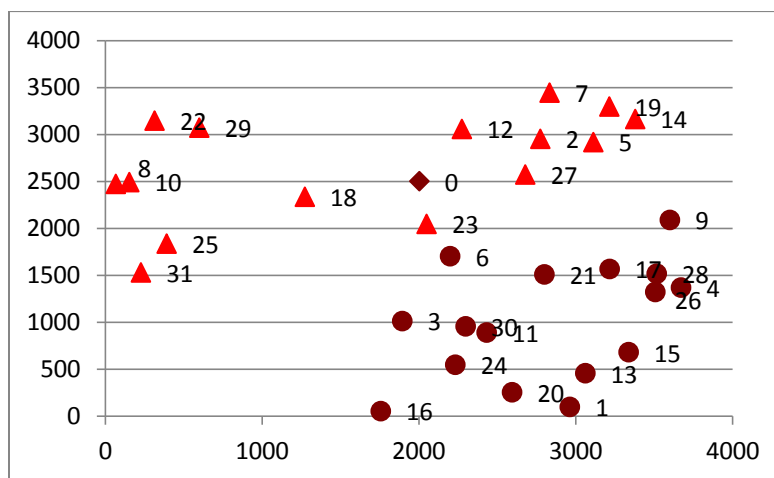


Figura D6. Segmentación de 30 nodos de la instancia prueba A

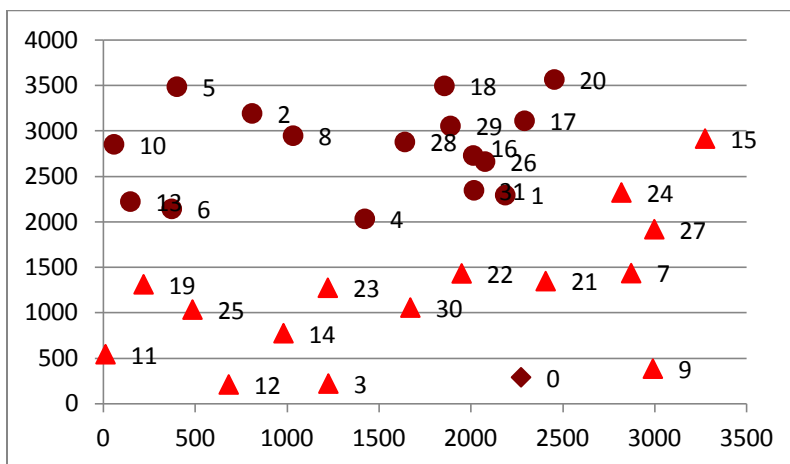


Figura D7. Segmentación de 30 nodos de la instancia prueba B

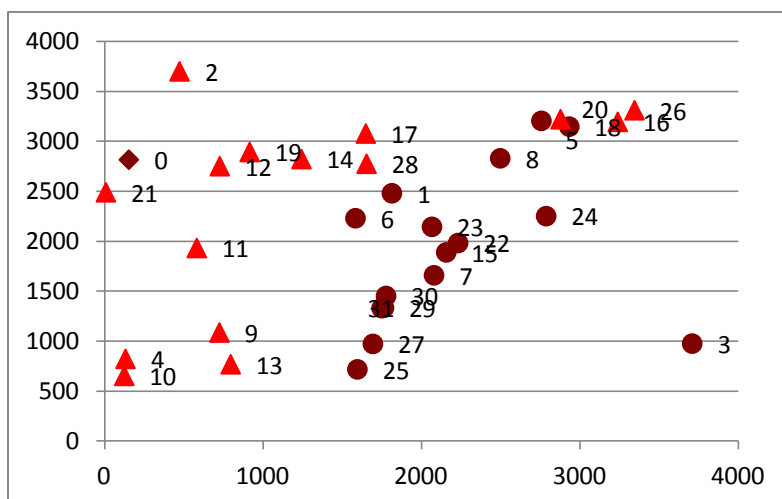


Figura D8. Segmentación de 30 nodos de la instancia prueba C

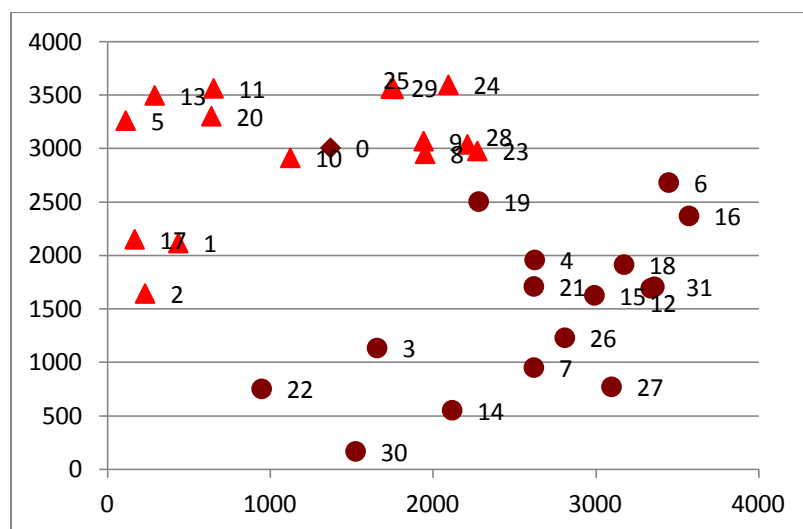


Figura D9. Segmentación de 30 nodos de la instancia prueba D

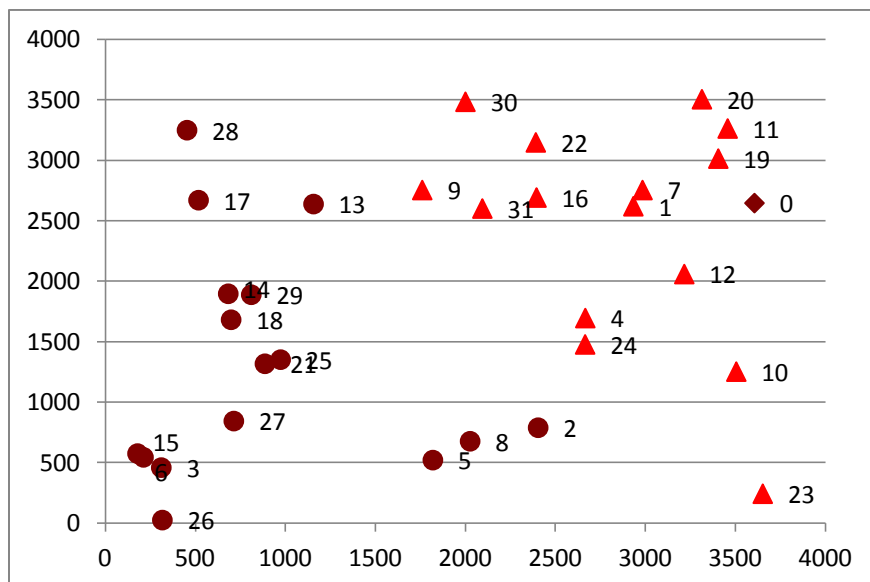


Figura D10. Segmentación de 30 nodos de la instancia prueba E

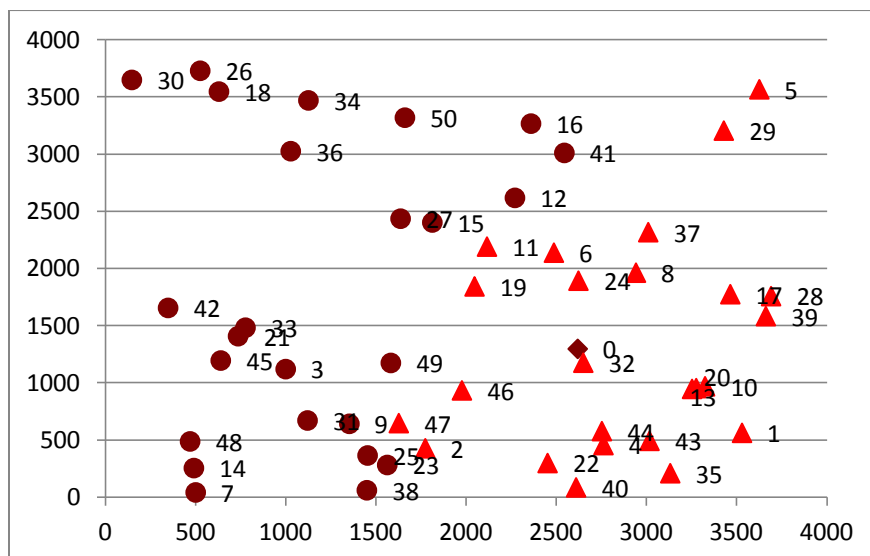


Figura D11. Segmentación de 50 nodos de la instancia prueba A

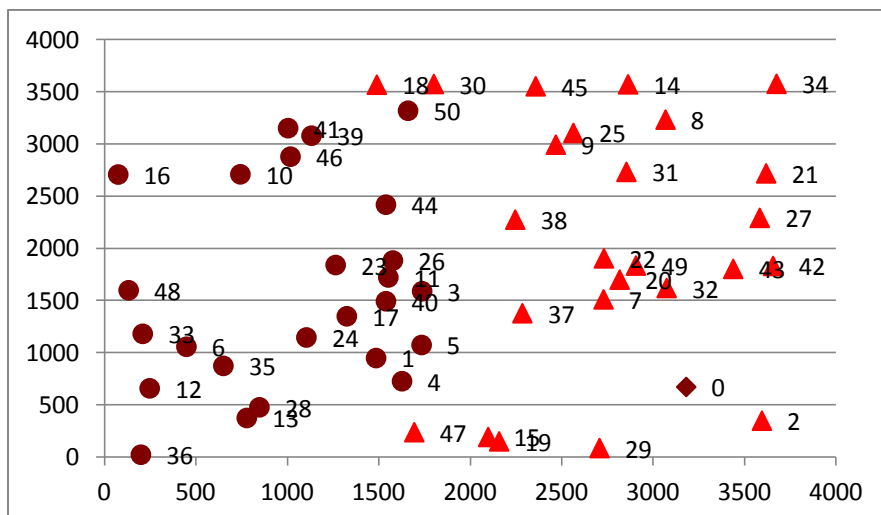


Figura D12. Segmentación de 50 nodos de la instancia prueba B

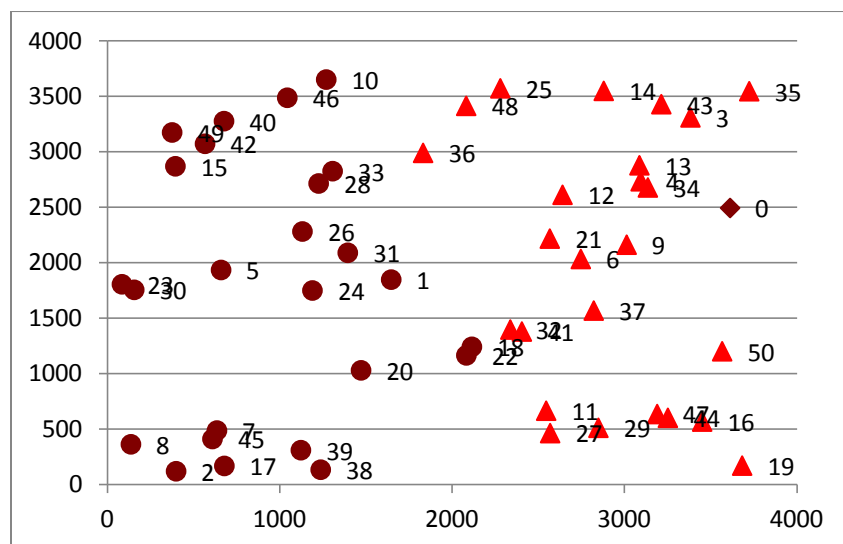


Figura D13. Segmentación de 50 nodos de la instancia prueba C

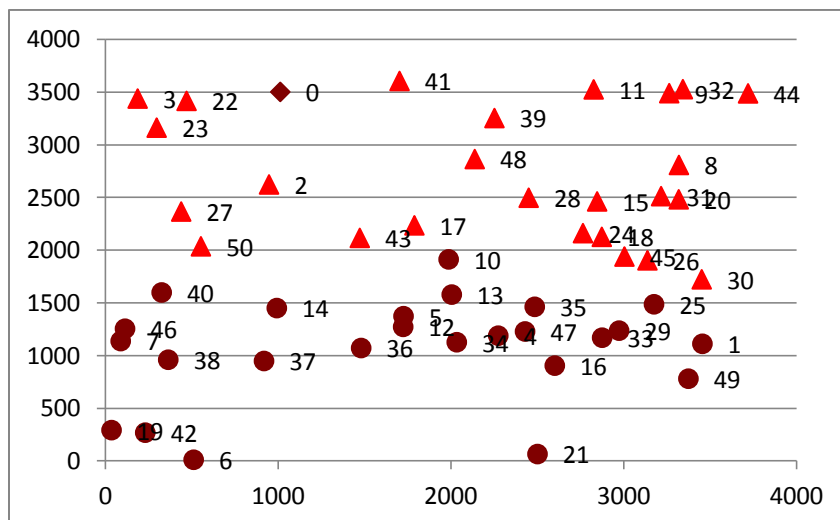


Figura D14 .Segmentación de 50 nodos de la instancia prueba D

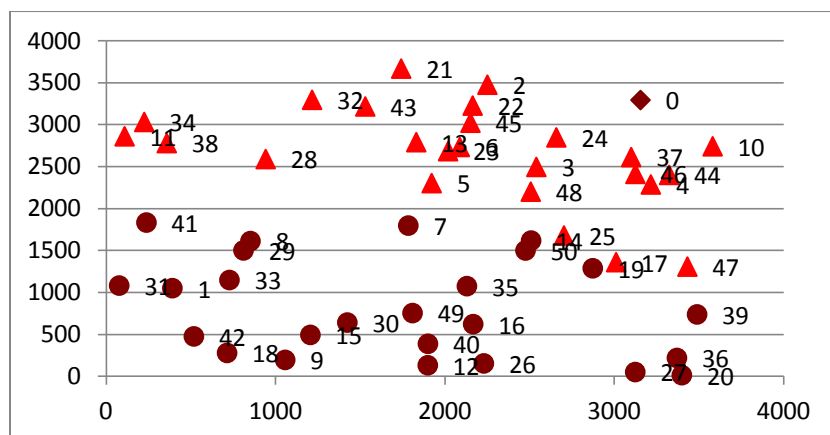


Figura D15.Segmentación de 50 nodos de la instancia prueba E

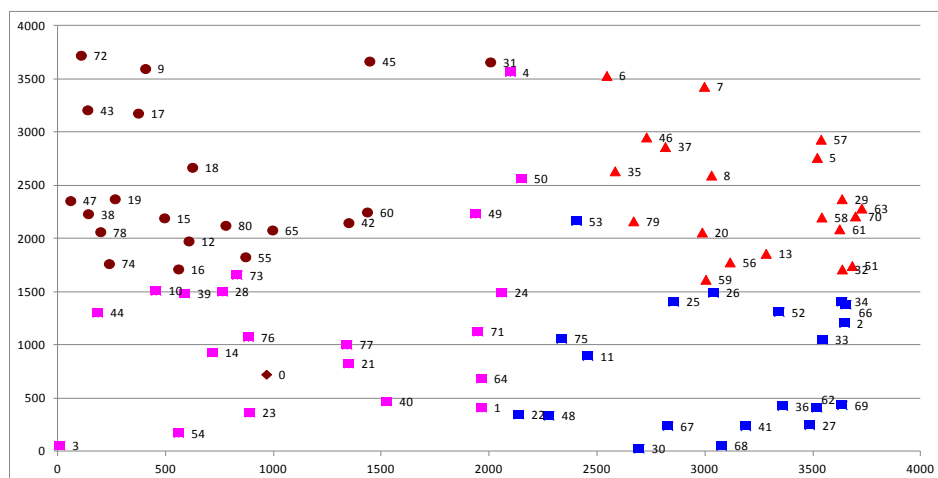


Figura D16. Segmentación de 80 nodos de la instancia prueba A

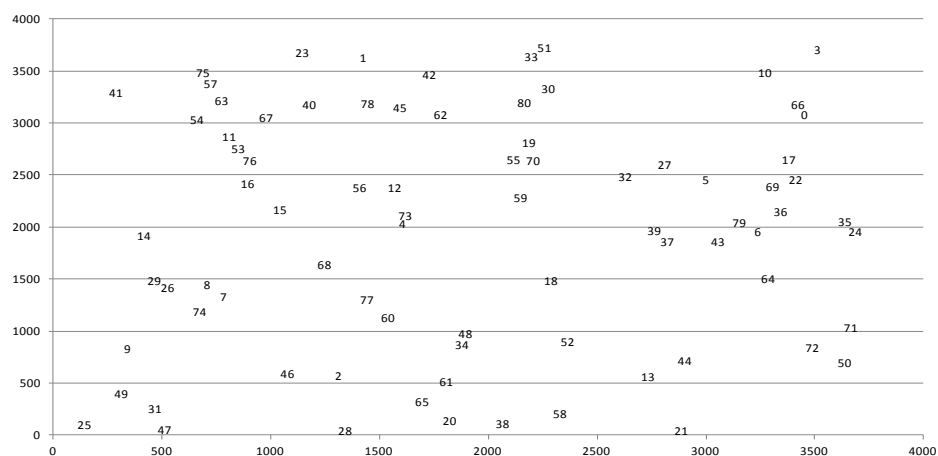


Figura D17. Segmentación de 80 nodos de la instancia prueba B

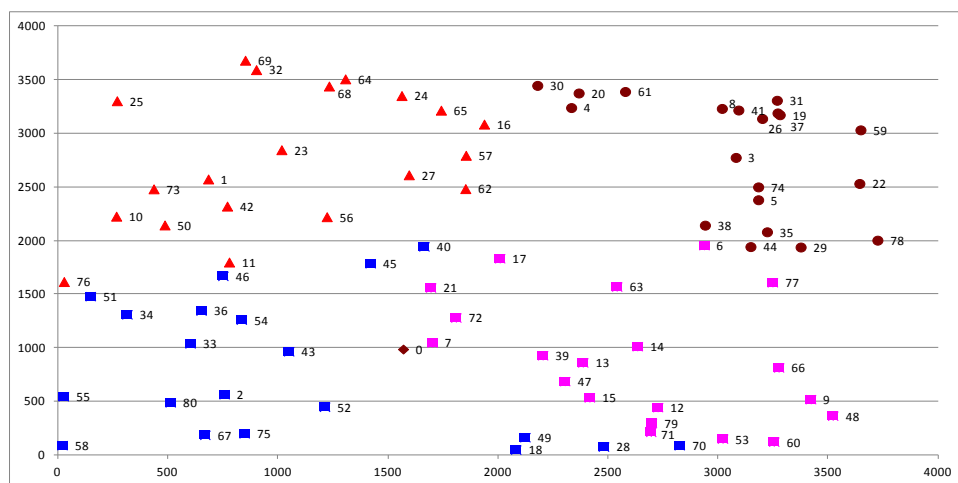


Figura D18. Segmentación de 80 nodos de la instancia prueba C

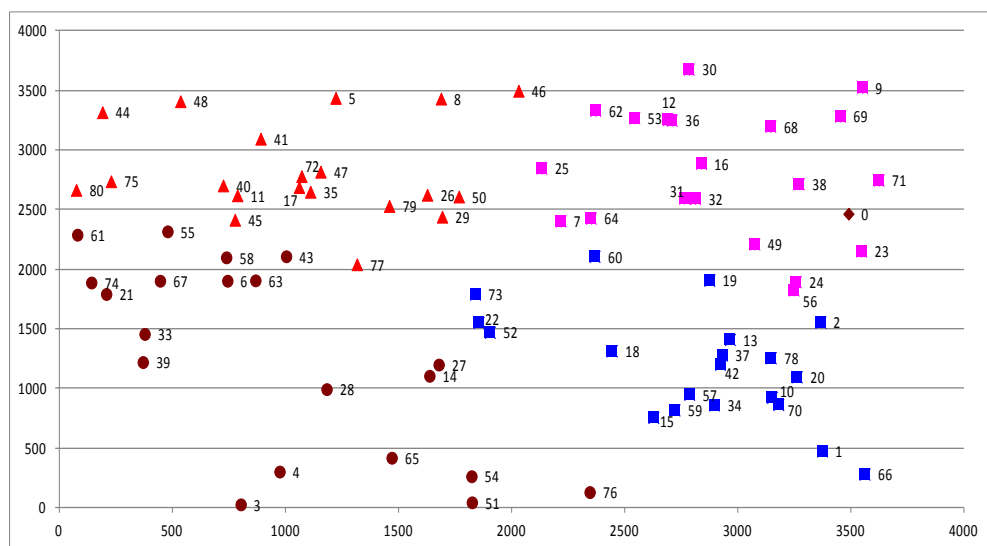


Figura D19. Segmentación de 80 nodos de la instancia prueba D

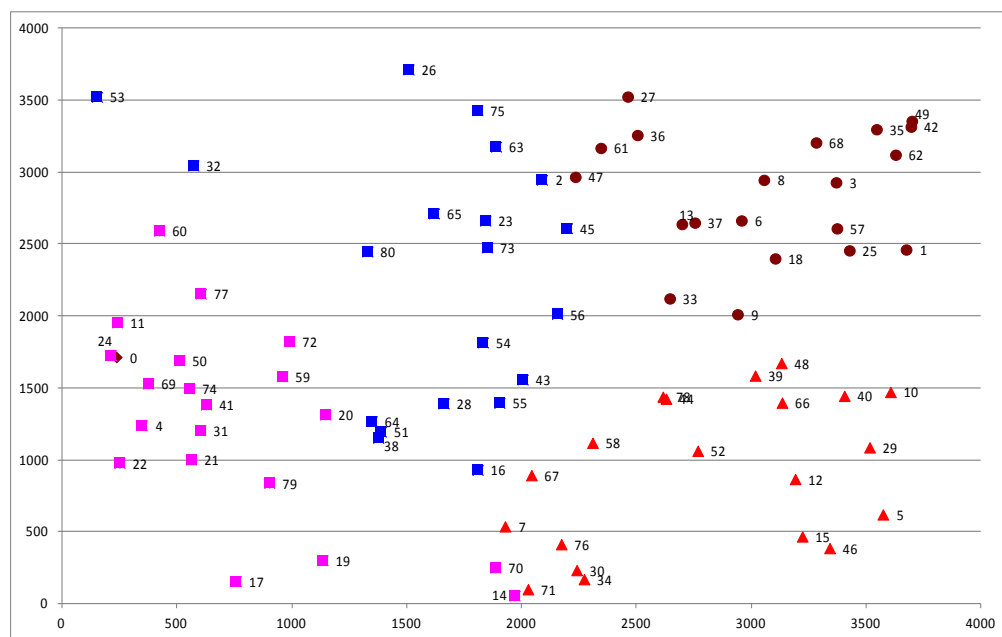


Figura D20. Segmentación de 80 nodos de la instancia prueba E

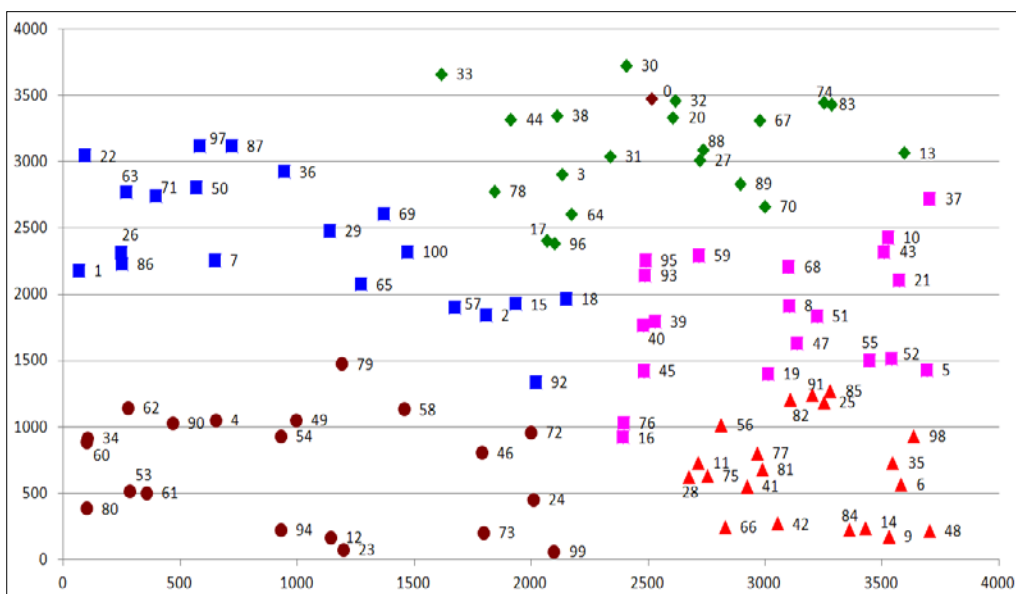


Figura D23. Segmentación de 100 nodos de la instancia prueba C

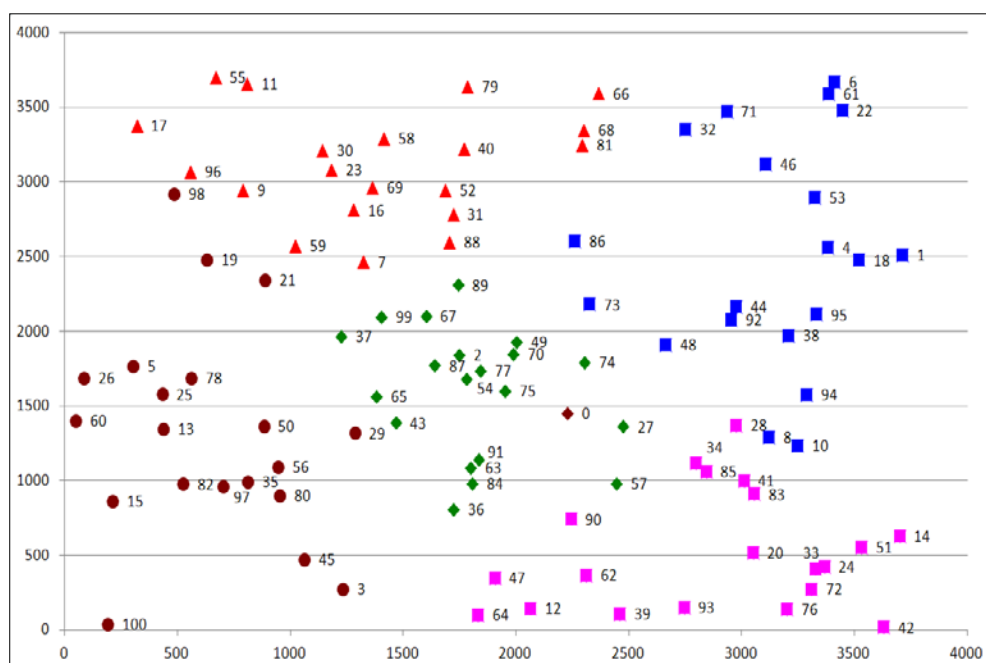


Figura D24. Segmentación de 100 nodos de la instancia prueba D

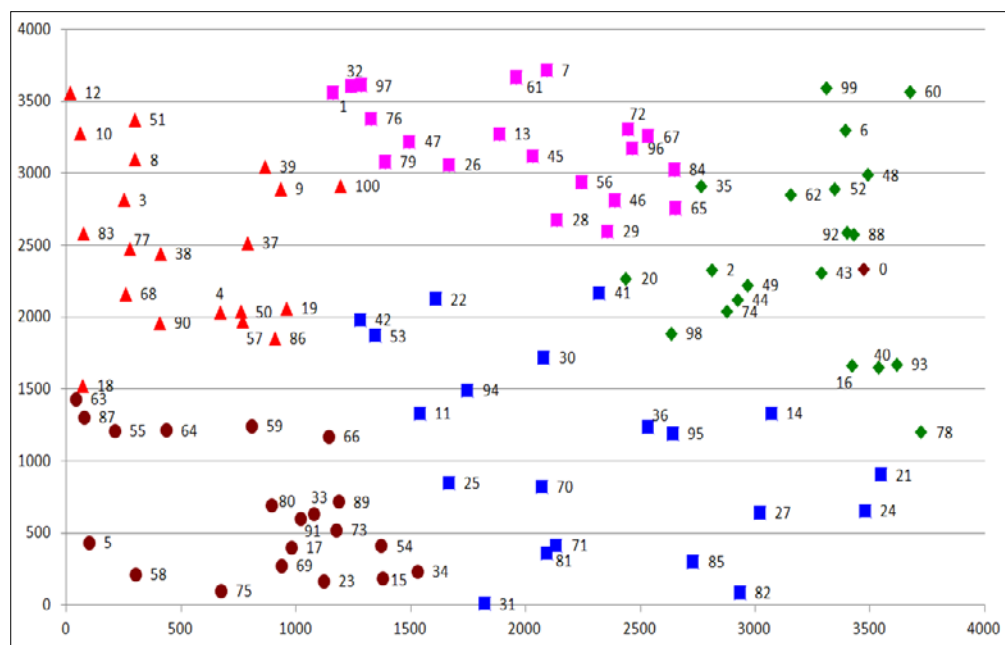


Figura D25. Segmentación de 100 nodos de la instancia prueba E