

**SOLUCIÓN AL PROBLEMA DE RUTEO DE VEHÍCULOS CON CAPACIDAD LIMITADA “CVRP” A
TRAVÉS DE LA HEURÍSTICA DE BARRIDO Y LA IMPLEMENTACIÓN DEL ALGORITMO GENÉTICO DE
CHU-BEASLEY**

JUAN PABLO ORREGO CARDOZO

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍA INDUSTRIAL
PEREIRA, DICIEMBRE DE 2013**

**SOLUCIÓN AL PROBLEMA DE RUTEO DE VEHÍCULOS CON CAPACIDAD LIMITADA “CVRP” A
TRAVÉS DE LA HEURÍSTICA DE BARRIDO Y LA IMPLEMENTACIÓN DEL ALGORITMO GENÉTICO DE
CHU-BEASLEY**

JUAN PABLO ORREGO CARDOZO

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

**DIRECTORA:
MSc. ELIANA MIRLEDY TORO OCAMPO**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERÍA INDUSTRIAL
PEREIRA, DICIEMBRE DE 2013**

Nota de Aceptación:

PhD. Antonio Escobar Z.
Jurado

MSc. Eliana Mirledy Toro O.
Directora

Pereira, Colombia. Diciembre de 2013

**Agradecimientos a la vida misma por darme retos,
Y a las personas que me ayudan a superarlos.**

| Contenido | Pg. |
|--|------------|
| SECCIÓN 1. REVISIÓN DEL ESTADO DEL ARTE..... | 11 |
| 1.1 Definición | 11 |
| 1.2 Tipos de VRP..... | 12 |
| 1.3 Modelos Matemáticos Básicos Para el VRP: | 14 |
| SECCIÓN 2. TÉCNICAS DE SOLUCIÓN..... | 21 |
| 2.1 Técnicas de solución utilizadas..... | 21 |
| 2.2 Técnicas de solución más frecuentes para el CVRP | 22 |
| 2.2.1 Técnicas exactas | 24 |
| 2.2.2 Técnicas Heurísticas | 25 |
| 2.2.3 Técnicas Metaheurísticas: | 29 |
| SECCIÓN 3. METODOLOGÍA IMPLEMENTADA..... | 31 |
| 3.1 Introducción | 31 |
| 3.2 Descripción General del Algoritmo | 31 |
| 3.3 Implementación de la Fase 1 | 32 |
| 3.4 Implementación de la Fase 2 | 35 |
| 3.4.1 Codificación | 35 |
| 3.4.2 Descripción Detallada del AGCB Potencializado | 36 |
| 3.5 Calibración de Parámetros | 48 |
| SECCIÓN 4. RESULTADOS | 50 |
| 4.1 Introducción | 50 |
| 4.2 Resultados en las Instancias de Augerat et al. (1998)..... | 50 |
| 4.2.1 Solución Final de los Óptimos Locales..... | 51 |
| 4.3 Resultados en las Instancias de Fisher (1994)..... | 53 |
| 4.3.1 Solución Final de los Óptimos Locales..... | 54 |
| 4.4 Resultados en las Instancias de CMT (1979) | 55 |

| | |
|--|----|
| 4.4.1 Solución Final de los Óptimos Locales..... | 56 |
| 4.5 Resultados en las Instancias de Christofides y Eilon (1969)..... | 59 |
| 4.5.1 Solución Final de los Óptimos Locales..... | 59 |
| SECCIÓN 5. CONCLUSIONES Y RECOMENDACIONES..... | 62 |
| 5.1 Conclusiones..... | 62 |
| 5.2 Recomendaciones y Trabajos Futuros | 63 |
| SECCIÓN 6. BIBLIOGRAFÍA..... | 64 |

| ÍNDICE DE FIGURAS | Pg. |
|--|-----|
| Figura 1. Variantes del VRP (González et al. (2006)) | 12 |
| Figura 2. Tour Típico TSP | 16 |
| Figura 3. Heurística de Barrido | 28 |
| Figura 4. Implementación Fase 1..... | 33 |
| Figura 5. Diagrama de Flujo Fase 1 | 34 |
| Figura 6. Esquema de Codificación..... | 35 |
| Figura 7. Diagrama de Flujo AGCB | 37 |
| Figura 8. Diagrama de Flujo Generación de la Población Inicial..... | 40 |
| Figura 9. Diagrama de Flujo Proceso de Selección | 42 |
| Figura 10. Proceso de Recombinación..... | 43 |
| Figura 11. Proceso de Mutación | 44 |
| Figura 12. Diagrama de Flujo Proceso de Recombinación | 45 |
| Figura 13. Diagrama de Flujo Proceso de Mutación..... | 46 |
| Figura 14. Diagrama de Flujo Modificación de la Población..... | 47 |

| ÍNDICE DE TABLAS | Pg. |
|--|-----|
| Tabla 1. Resumen Estado del Arte..... | 22 |
| Tabla 2. Número de Iteraciones Locales Dado un Benchmark..... | 49 |
| Tabla 3. Variación de los Arranques X y Y..... | 49 |
| Tabla 4. Resultados Benchmark de Augerat et al..... | 51 |
| Tabla 5. Resultados Benchmark de Fisher..... | 54 |
| Tabla 6. Características de las Instancias de CMT (1979)..... | 56 |
| Tabla 7. Resultados Benchmark de CMT..... | 56 |
| Tabla 8. Resultados Benchmark de Christofides y Eilon..... | 59 |

SECCIÓN 1. REVISIÓN DEL ESTADO DEL ARTE

1.1 Definición

Los problemas de ruteo de vehículos han sido objeto de investigación intensiva por más de 50 años, debido al gran interés científico que éstos tienen por ser considerados NP-completos y donde algunas instancias Euclidianas con 75 nodos aún no han sido resueltas hasta su óptimo (Prins, Christian (2004)). Por otra parte, se debe a su importancia cómo modelo de optimización, en cantidades de campos de aplicación, como son, transporte, logística, comunicaciones, manufactura, militar, entre otros. El problema “tradicional” CVRP (por sus siglas en inglés *Capacitated Vehicle Routing Problem*) puede ser descrito en su forma más sencilla como una flota de vehículos con capacidades uniformes que tiene que satisfacer la demanda de un grupo de clientes a través de un conjunto de rutas que empiezan y terminan en un almacén común y que representan el menor costo posible, así como la identificación del orden de visita a los mismos (Gendreau et al. (2012)).

Como consecuencia de los altos niveles actuales de congestión vehicular, de los fuertes consumos de energía de los vehículos de carga, así como del impacto negativo que estos últimos tienen en el medio ambiente como emisiones, accidentes, ruido y vibraciones; en los últimos años uno de los problemas que más ha interesado a la logística urbana “*City Logistics*” ha sido el de optimizar globalmente los sistemas de transporte, considerando los costos y beneficios de esquemas tanto públicos como privados. Estos conceptos de *City Logistics*, han sido implementados y son objeto de interés presentes en muchos modelos desarrollados recientemente para predecir y establecer sistemas inteligentes de transporte (Taniguchi y Thompson (2002)). Cómo caso particular se relacionan posteriormente todos los modelos generales y explícitos del VRP y sus variantes.

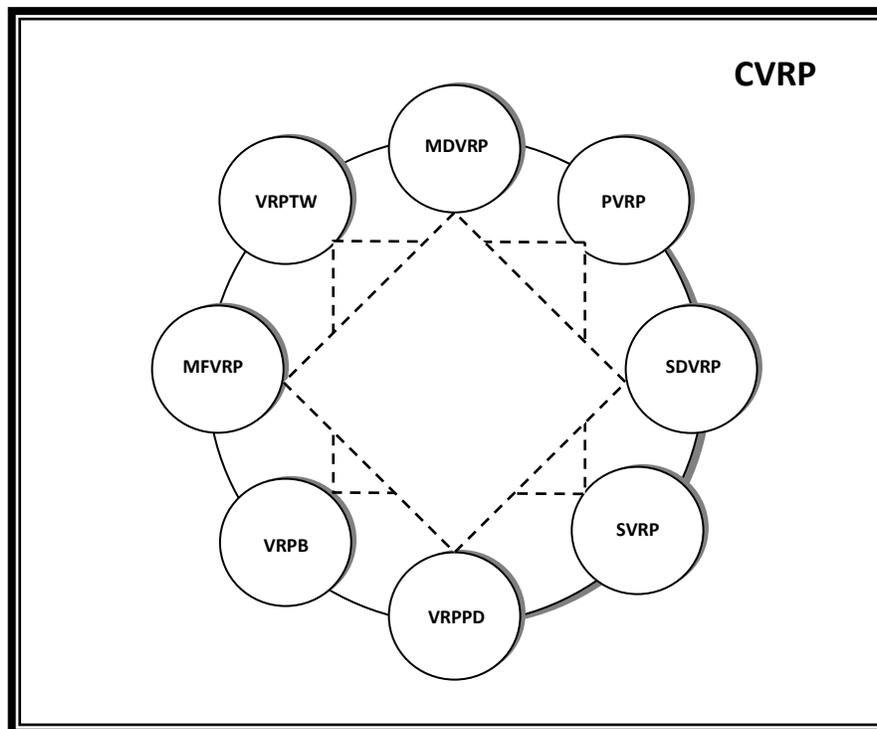
Como caso particular y elemento fundamental de las ideas expuestas anteriormente se tiene que el transporte de mercancías en ambientes urbanos juega un papel muy importante en el desarrollo sostenible de una ciudad ya que altos niveles de movimiento de bienes ocurren dentro de las ciudades (Taniguchi y Thompson(1999)). Según *Toth y Vigo, 2000.* “*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, dado que se estima que los costos del transporte representan entre el 10% y el 20% del costo final de los bienes*”. Es por esto, que dentro del presente trabajo se expondrán las diferentes variantes del VRP, además del problema específico que será abordado y su método de solución propuesto expuesto posteriormente.

1.2 Tipos de VRP

La actual diversidad de aplicaciones donde asuntos de ruteo pueden ser encontrados, conllevan a una definición de diferentes variantes del VRP con características adicionales y restricciones, que se llamarán atributos. Estos atributos buscan capturar un mayor número de detalles o tomas de decisiones, conllevando a una mayor riqueza del problema. Cómo son: estructura del sistema (depósitos, flotas de vehículos), requerimientos de los clientes (ventanas de tiempo, visitas multi-depósito), reglas de operación de vehículos (lugar de carga, restricciones en rutas, distancia o tiempo total), y contextos de decisiones (congestiones vehiculares y planeación sobre horizontes de tiempo extendidos) ((Gendreau et al. (2012))).

Los diferentes atributos y restricciones del problema generan una familia de la que vale la pena mencionar ocho casos típicos, los cuales al compartir características pueden dar lugar a todo un universo de problemas VRP. Los principales problemas de ruteo de vehículos se ilustran en la siguiente figura y pueden ser descritos así (González et al. (2006)):

Figura 1. Variantes del VRP (González et al. (2006))



- **CVRP** (*Capacitated VRP*): 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. Este problema de optimización del tipo NP-Hard, combina las características de un Bin Packing Problem (BPP), con el objetivo de asignar las cargas a los vehículos capacitados, y un problema del agente viajero (TSP) que apunta a encontrar la mejor ruta para cada vehículo. (Jansen, K. (1993)) (Shaw, P. (1998)) (Prins, C.(2004)) (Toth, P., Tramontani, A. (2008)).
- **MDVRP** (*Multiple Depot VRP*): También llamado 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. (Renaud, J. et al. (1996))(Wu, T-H. et al. (2002)) (Crevier, B. et al. (2007)) (Ho, W. et al. (2008)).
- **PVRP** (*Periodic VRP*): Contempla en su planteamiento un horizonte de operación de M días, periodo durante el cual cada cliente debe ser visitado una vez. (Drummond, L. et al. (2001)) (Francis, P. et al. (2006)) (Alonso, F. et al. (2007)) (Hemmelmayr, V. et al. (2009)).
- **SDVRP** (*Split Delivery VRP*): También llamado 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. (Belenguer, J. et al. (2000)) (Hertz, A. et al. (2006)) (Chen, S. et al. (2007)) (Jin, M. et al. (2007)).
- **SVRP** (*Stochastic 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. (Dror, M. et al. (1986))(Bertsimas, D. et al. (1991)) (Gendreau, M. et al. (1996)) (Laporte, G. et al. (2002)).
- **VRPPD** (*VRP Pickup and Delivery*): También llamado 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 a un mayor número de vehículos. (Dethloff, Jan (2001)) (Montané, A. et al. (2006)) (Bianchessi, N. et al. (2007))(Kachitvichyanukul, V. et al. (2009)).
- **VRPB** (*VRP with Backhauls*): Se trata del mismo VRPPD, pero incluye la restricción de culminar todas las entregas antes de iniciar las diversas recogidas. Este concepto, parte del hecho de que los vehículos inicialmente están cargados en su totalidad, luego re-asignar cargas a los camiones en los almacenes puede llegar a ser imposible, desde la perspectiva económica o física. (Toth y Vigo (1997)) (Mingozzi, A. et al. (1999)) (Osman, I. et al. (2002))(Brandão, José (2006)).

- **MFVRP (Mixed Fleet VRP):** Es un VRP en el que se suponen vehículos con distintas capacidades o capacidad heterogénea, por lo que es necesario considerar estas capacidades en la ruta que seguirá cada recurso, ya que un camión más grande podrá realizar una ruta más larga o que tenga mayor concentración de demanda. (Gendreau, M. et al. (1999)) (Tarantilis, C. et al. (2004))(Choi, E. et al. (2007)) (Golden, B. et al. (2007)).
- **VRPTW (VRP with Time Windows):** Es aquel en el que se incluye una restricción adicional en la que se asocia a cada cliente una ventana de tiempo, es decir, cada cliente sólo está dispuesto a recibir el bien o servicio durante un intervalo de tiempo predeterminado. (Taillard, E. et al. (1997)) (Bent, R. et al. (2004)) (Kallehauge, B. et al. (2005)) (Xiao, J. et al. (2012)).

1.3 Modelos Matemáticos Básicos Para el VRP:

Como primera instancia, es necesario conocer los conceptos fundamentales inherentes con la comprensión, desarrollo y aplicación del problema en estudio. Cómo ya fue descrito anteriormente, el desarrollo del CVRP, siendo el caso más general, depende en su medida de la fundamentación y desarrollo de dos problemas de optimización del tipo NP-Hard, los cuales serán descritos a continuación, y de este modo poder establecer los tres tipos de modelos generales planteados a hoy del VRP.

- **Bin Packing Problem (BPP)**

Debido al hecho que cada vehículo debe realizar por lo menos una ruta, se debe asumir que K (número de vehículos) no puede ser menor que un número mínimo K_{min} , que es el número mínimo de vehículos necesarios para servir a toda la demanda de los clientes. El valor de este parámetro mínimo puede ser determinado resolviendo un *BPP* asociado específicamente con el CVRP, el cual determina el mínimo numero de compartimientos (para este caso, vehículos), cada uno con capacidad C , requeridos para cargar todos los n elementos, cada uno con una demanda no negativa, denotada por $d_i, i = 1, \dots, n$.

Aún si el BPP es un problema del tipo polinomial, instancias con cientos de elementos pueden ser resueltos de forma óptima eficiente (ver Martello y Toth (1990)). Sin embargo, dado el caso de una solución optima para valores asociados al BPP con un conjunto $S \subseteq V \setminus \{0\}$, se denota cómo $r(S) = K_{min}$, el número mínimo de vehículos necesarios para servir a todos los clientes. Generalmente reemplazado por el valor trivial o límite inferior (Toth y Vigo (2000)):

$$\left\lceil \frac{d(S)}{C} \right\rceil$$

- **Problema del Agente Viajero (TSP: Traveling Salesman Problem)**

Existe un conjunto de n ciudades, $V = \{1, 2, 3, \dots, n\}$, y un conjunto de caminos (arcos) uniendo cada una de las ciudades, $(i, j) \in A$. C_{ij} es la distancia para ir de la ciudad i a la ciudad j donde $(c_{ij} = c_{ji})$ en el caso simétrico. Un agente viajero debe realizar un recorrido partiendo de una ciudad de origen, pasando por cada una de las ciudades una sola vez, y regresando a la ciudad de origen.

Este problema es fácil de entender pero su formulación matemática es compleja. Sea x_{ij} la variable de decisión del problema, donde $x_{ij} = 1$ si el arco, (i, j) es usado para hacer el tour, y $x_{ij} = 0$ en caso contrario. Así, el modelo matemático asume la siguiente forma:

$$(1) \quad \text{Min } v = \sum_{(i,j) \in A} c_{ij} x_{ij}$$

Sujeto a:

$$(2) \quad \sum_{\{i:(i,j) \in A\}} x_{ij} = 1 \quad \forall j \in V$$

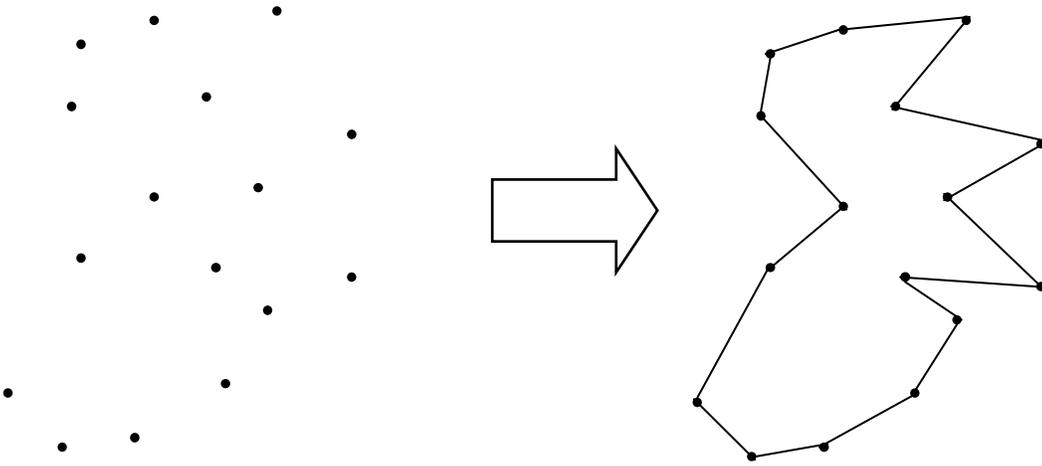
$$(3) \quad \sum_{\{j:(i,j) \in A\}} x_{ij} = 1 \quad \forall i \in V$$

$$(4) \quad \sum_{\{(i,j) \in A: i \in U; j \in (V-U)\}} x_{ij} \geq 1; \quad 2 \leq |U| \leq |V| - 2$$

Donde V es el conjunto de ciudades (vértices), A el conjunto de caminos (arcos) y $U \subset V$. El conjunto de restricciones (2) indica que se puede llegar a una ciudad dada desde una sola ciudad anterior y el conjunto (3) de restricciones indica que a partir de una ciudad se puede pasar a una única ciudad. El conjunto (4) de restricciones evita el surgimiento de subtours (Gallego et al. (2008)).

La **Figura 2** muestra un tour típico para un TSP de $n = 30$.

Figura 2. Tour Típico TSP



Posteriormente a la conceptualización teórica y matemática de ambos problemas base para el desarrollo del CVRP, es posible presentar en la siguiente sección, las principales formulaciones matemáticas que pueden ser usadas para representar dicho modelo. En este orden, tres aproximaciones de modelamiento han sido propuestas en la literatura según (Toth y Vigo (2000)), y además otras variantes y adiciones de restricciones para diferentes funciones objetivo pueden ser encontradas en (Laporte y Nohert (1987)).

- **Modelos de Flujo de Vehículos (Vehicle Flow Models) (Toth y Vigo (2000))**

Los modelos de este tipo, usan variables enteras, asociadas a cada arco o extremo del grafo, el cual cuenta el número de veces que el arco o extremo es atravesado por un vehículo. Siendo este el modelo más usado para las formulaciones básicas del VRP, generalmente es utilizado para los casos en los cuales el costo de una solución puede ser expresada como la suma de los costos asociados a los arcos, y cuando las restricciones más relevantes conciernen la transición directa entre los clientes y la ruta, por lo tanto éste puede ser modelado efectivamente a través de una definición apropiada del conjunto de arcos y sus costos. Por otra parte, existen casos muy prácticos en los cuales estos modelos no deberían ser utilizados, cómo son: cuando el costo de una solución depende de la secuencia total de los vértices, o del tipo de vehículo asignado a la ruta.

Éste modelo de programación lineal del tipo *two-index vehicle flow formulation* utiliza $O(n^2)$ variables binarias de x_{ij} para indicar con un valor tomado de 1 si el arco $(i, j) \in A$ pertenece a la solución óptima y toma un valor de 0 en otro caso; es decir, toma su valor si el vehículo atraviesa o no un arco en la solución óptima. Para simplificar las notaciones, en las siguientes formulaciones se asuma que el grafo $G(V, A)$ (ó $G(V, E)$) es completo.

$$(5) \quad \min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}$$

Sujeto a:

$$(6) \quad \sum_{i \in V} x_{ij} = 1 \quad \forall j \in V \setminus \{0\}$$

$$(7) \quad \sum_{j \in V} x_{ij} = 1 \quad \forall i \in V \setminus \{0\}$$

$$(8) \quad \sum_{i \in V} x_{i0} = K,$$

$$(9) \quad \sum_{j \in V} x_{0j} = K,$$

$$(10) \quad \sum_{i \notin S} \sum_{j \in S} x_{ij} \geq r(S) \quad \forall S \subset V \setminus \{0\}, S \neq \emptyset,$$

$$(11) \quad x_{ij} \in \{0, 1\} \quad \forall i, j \in V.$$

Donde:

- $V = \{v_0, v_1, \dots, v_n\}$ es el conjunto de vértices del gráfico, donde v_0 corresponde al almacén.
- C es la matriz de distancias o costos c_{ij} entre los clientes v_i y v_j .
- K es el número de vehículos de capacidad C , necesarios para cargar toda la demanda.
- S es un subconjunto de vértices del grafo donde $S \subset V \setminus \{0\}$.
- $r(S)$ es el número mínimo de vehículos necesarios para servir a todos los clientes en S .

Las restricciones de grado de entrada y de salida (6) y (7), imponen que exactamente un arco entra y otro sale de cada vértice asociado a un cliente, para satisfacer que todos los clientes sean visitados una vez. Además las restricciones (8) y (9) imponen los requerimientos para el vértice depósito; es decir, los mismos carros que salen del depósito, regresan al depósito. Finalmente la restricción de *corte de capacidad* (10), satisface tanto la conectividad de la solución y las no violaciones de la capacidad C de cada vehículo. Dicha restricción resta aún válida si el $r(S)$ es reemplazado por el valor solución trivial del *BPP* mencionado anteriormente.

- **Modelo de Flujo de Mercancías (Commodity Flow Models)(Toth y Vigo (2000))**

Éste tipo de modelo utiliza variables continuas adicionales, asociadas con los arcos o extremos, y representan el flujo de las mercancías (demanda) por los caminos recorridos por los vehículos. La formulación necesita un grafo extendido $G'(V', A')$ el cual es obtenido agregando un vértice $n + 1$, el cuál es una copia del nodo depósito. Ahora las rutas son caminos desde el vértice 0 hasta el vértice $n + 1$. Por otra parte, dos variables de flujo no negativas y_{ij} y y_{ji} estarán asociadas con cada extremo $(i, j) \in A'$. Es decir, si el vehículo viaje desde i hasta j , entonces y_{ij} y y_{ji} darán la carga del vehículo además de su capacidad residual, respectivamente, a través del arco. A manera de ilustración $y_{ji} = C - y_{ij}$, dicha relación se mantiene estable si el vehículo viaja de j hasta i . por lo tanto, la ecuación $y_{ij} + y_{ji} = C$, es válida para todo arco $(i, j) \in A'$.

Para cada ruta de una solución factible, las variables de flujo definen dos caminos dirigidos, uno desde el vértice 0 hasta $n + 1$, cuyas variables representan la carga del vehículo, y otra desde el vértice $n + 1$ hasta el vértice 0, cuyas variables representan la capacidad residual del vehículo.

$$(12) \quad \min \sum_{(i,j) \in A'} c_{ij} x_{ij}$$

Sujeto a:

$$(13) \quad \sum_{j \in V'} (y_{ji} - y_{ij}) = 2d_i \quad \forall i \in V' \setminus \{0, n + 1\}.$$

$$(14) \quad \sum_{j \in V' \setminus \{0, n + 1\}} y_{0j} = d(V \setminus \{0, n + 1\}),$$

$$(15) \quad \sum_{j \in V' \setminus \{0, n + 1\}} y_{j0} = KC - d(V \setminus \{0, n + 1\}),$$

$$(16) \quad \sum_{j \in V' \setminus \{0, n + 1\}} y_{n+1,j} = KC,$$

$$(17) \quad y_{ij} + y_{ji} = Cx_{ij} \quad \forall (i, j) \in A',$$

$$(18) \quad \sum_{j \in V'} (x_{ij} + x_{ji}) = 2 \quad \forall i \in V' \setminus \{0, n + 1\},$$

$$(19) \quad y_{ij} \geq 0 \quad \forall (i, j) \in A',$$

$$(20) \quad x_{ij} \in [0, 1] \quad \forall (i, j) \in A',$$

La restricción de conservación (13) indica que la diferencia entre la suma de las variables de flujo de mercancías asociadas con los arcos entrantes y salientes a cada vértice i es igual al doble de la demanda de i . Las restricciones desde la (14)-(16) generan los valores correctos para las variables de flujo incidentes en dos vértices como depósitos. Finalmente, las restricciones (17) y (18) imponen la relación de las variables del flujo de vehículos y del flujo de mercancías, además del grado del vértice, respectivamente.

- **Modelo de Partición de conjuntos (Set-Partitioning Models)(Toth y Vigo (2000))**

Los modelos de este tipo tienen un número exponencial de variables binarias, cada una asociada con un circuito posible diferente. Su nombre es debido a la determinación de una colección de circuitos con costo mínimo, que servirá a cada cliente una vez, y posiblemente, satisfaga restricciones adicionales. Una de las ventajas de utilizar este tipo de modelo, es que permite la contemplación de costos generales en la ruta, por ejemplo, dependiendo de la secuencia de los arcos, o del tipo de vehículo utilizado, etc. Sin embargo su mayor desventaja es la su formulación matemática relajada es típicamente más grande que los otros modelos, ya que necesita tener en cuenta una serie de desigualdades para mantener la factibilidad de las rutas.

$$(21) \quad \min \sum_{j=1}^q c_j x_j$$

Sujeto a:

$$(22) \quad \sum_{j=1}^q a_{ij} x_j = 1 \quad \forall i \in V \setminus \{0\},$$

$$(23) \quad \sum_{j=1}^q x_j = K,$$

$$(20) \quad x_j \in \{0, 1\} \quad \forall j = 1, \dots, q.$$

Dónde:

- G es el conjunto de circuitos factibles.
- H es el conjunto de circuitos factibles por realizar, pertenecientes a G .
- q es el número absoluto de rutas pertenecientes a H .
- c_j es el costo asociado para cada circuito H_j .
- a_{ij} es una variable binaria, con valor de 1 si el vértice i es visitado por la ruta H_j y 0 en otro caso.
- x_j es una variable binaria, con valor de 1 si y sólo sí, el circuito H_j es seleccionada en la solución óptima.

Por otra parte, la restricción (22) indica que cada cliente i sea cubierto exactamente por uno de los circuitos seleccionados, y la (23) requiere que K circuitos sean seleccionados. Éste modelo general puede tener en cuenta fácilmente muchas restricciones, por ejemplo, de ventanas de tiempo.

Finalmente, es necesario conceptualizar que al enfrentarnos a estos tipos de problemas prácticos, siempre estamos expuestos a diversos tipos de dificultades o complejidades, relacionadas con aspectos como la modelación matemática, encontrar un algoritmo que conlleve a una solución óptima, disponibilidad tanto en cálculos como en tiempos del sistema para implementar el modelo (software y hardware), etc. Ya que estos tipos de problemas, enmarcados como combinatoriales, tienen una complejidad vasta en esfuerzo y dificultad para su resolución en función de la dimensión del problema. Es decir, que ésta inhibición de una búsqueda exhaustiva para la resolución del problema, conlleva a un concepto llamado **explosión combinatorial**, inherente a los problemas en los que al crecer el número de variables de decisión del problema, el número de soluciones (configuraciones) así como el esfuerzo computacional para resolverlo, crece en forma exponencial (Gallego et al. (2008)).

Dichos problemas combinatoriales pueden ser divididos en 2 grandes grupos, de acuerdo con el esfuerzo computacional requerido:

P Problemas tipo P para los cuales existen algoritmos con esfuerzo computacional de tipo polinomial para encontrar su solución óptima.

NP Problemas de tipo NP (Non Polinomial) para los cuales **no se conocen** algoritmos con un esfuerzo computacional de tipo polinomial para encontrar su solución óptima.

SECCIÓN 2. TÉCNICAS DE SOLUCIÓN

2.1 Técnicas de solución utilizadas

En el presente trabajo de investigación se propondrá una alternativa de solución para la versión básica del problema de ruteo de vehículos, el VRP *Capacitado* "CVRP". En éste caso, todos los clientes correspondientes a las entregas y las demandas son determinísticas, es decir, es un dato conocido y no puede ser particionado. Además, los vehículos son idénticos y tienen base en un mismo depósito central y la única restricción de capacidad de los vehículos es previamente impuesta. El objetivo central de éste problema es minimizar el costo total del recorrido y duración de las rutas para servir a todos los clientes. Por otra parte, un costo c_{ij} es asociado con cada arco $(i, j) \in A$ y representa el costo de viajar del vértice i a j . En éste caso, el uso de arcos lupa (i, i) no es permitido, por lo tanto se impone por definición que $c_{ii} = +\infty$ para todo $i \in V$, donde V es el conjunto de vértices y A el conjunto de arcos. Finalmente, se tienen para todos los conjuntos $(i, j) \in A$, costos $c_{ij} = c_{ji}$, es decir, el problema es del tipo simétrico ó "SCVRP", donde el conjunto de arcos A es reemplazada por un conjunto de de arcos no direccionados E . (Toth y Vigo (2000)).

Para abordar el VRP muchas han sido las técnicas de solución implementadas, pero junto con la complejidad y eficiencia de dichos métodos, el problema ha involucrado cada vez más variantes que lo hacen computacionalmente más difícil de resolver, por ejemplo, ventanas de tiempo de servicio, distribuciones de probabilidad que simulen las condiciones viales, vehículos de distintas capacidades, entre otras. Dichas variantes han ido apareciendo al analizar y tratar de resolver problemáticas reales en las ciudades. Es por esto, que el objetivo perseguido por esta tesis es el de desarrollar e implementar la combinación de técnicas heurísticas y metaheurísticas de optimización, para obtener soluciones de buena calidad en un problema de interés actual por la comunidad científica mundial.

Por lo tanto, se crea gran interés en emplear en el presente trabajo técnicas de optimización combinatorial como un alternativa interesante, ya que éstas exploran sólo una pequeña parte del espacio de soluciones de una manera controlada, y encuentran soluciones subóptimas de muy buena calidad en tiempos de cómputo razonables. Finalmente, con base a las soluciones obtenidas mediante la metodología implementada, se compararán las mejores soluciones encontradas, con diversas instancias sobre el "CVRP" tomadas de bases de datos de la literatura especializada de diversos autores de la academia para así generar las conclusiones pertinentes.

2.2 Técnicas de solución más frecuentes para el CVRP

Entre las técnicas de solución del CVRP, las más frecuentemente utilizadas son las heurísticas y metaheurísticas; ya que, ningún algoritmo exacto puede garantizar la obtención de soluciones óptimas, en tiempos de computación razonables cuando el número de ciudades es grande. Esto se debe a la característica NP-Hard del problema. A continuación se presenta un cuadro mostrando el abordaje del problema a través del tiempo por autores, posteriormente se listarán solamente diversas técnicas de solución utilizadas con frecuencia y únicamente se profundizará en aquellas que serán utilizadas en la solución del presente proyecto.

Tabla 1. Resumen Estado del Arte

| Año | Autor [Bibliografía] | Técnicas de Solución |
|------|---|---|
| 1969 | CHRISTOFIDES, N., EILON, S. <i>An Algorithm for the Vehicle Dispatching Problem</i> . Operational Research Quarterly. Volume 20, pages 309-318. | Se consideran tres métodos de solución, siendo el (C) el método superior: (A) Aproximación <i>Branch-and-Bound</i> . (B) Aproximación de los ahorros. (C) El método de los 3 tours óptimos. |
| 1993 | JANSEN, Klaus. <i>Bounds for the general capacitated routing problem</i> . Networks, volume 23, issue 3, pages 165-173. May 1993 | Route first, cluster second, utilizando un tour gigante generado por la técnica de Christofides |
| 1993 | CORNUEJOLS, Gerard., HARCHE, Farid. <i>Polyhedral study of the capacitated vehicle routing problem</i> . Mathematical Programming, volume 60, issue 1-3, pages 21-52. June 1993 | Relajación del CVRP a través del "Graphical Vehicle Routing Problem" (GVRP) y además extendiendo los resultados poliédricos conocidos para el TSP |
| 1994 | FISHER, M. <i>Optimal Solution of Vehicle Routing Problems Using Minimum k-Trees</i> . Operations Research. Volume 42, no. 4, pages 626-642. | Se modela el VRP como un problema para encontrar el costo mínimo de un K-Tree, con dos extremos K incidentes en un depósito, e imponiendo restricciones paralelas de la capacidad del vehículo y que cada cliente debe ser visitado una vez. Éstas restricciones paralelas son dualizadas para obtener un problema Lagrangiano que provee los lower-bounds para ejecutar un algoritmo <i>branch-and-bound</i> |

| | | |
|------|---|--|
| 1998 | AUGERAT, P., BELENGUER, J.M., BENAVENT, E., CORBERAN, A., NADDEF, D., RINALDI, G. <i>Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem</i> . Research Report 949-M, Université Joseph Fourier, Grenoble, France. | Implementación del algoritmo <i>branch-and-cut</i> basado en la descripción parcial poliédrica del polígono correspondiente. Además se centra en el diseño de procedimientos de separación para varias clases de desigualdades válidas. |
| 2001 | RALPHS, T.K., KOPMAN, L., PULLEYBLANK, W.R., TROTTER, L.E. <i>On the capacitated vehicle routing problem</i> . December 2001 | Metodología de “decomposition-based separation” para las restricciones de capacidad. Implementaciones a través del parallel branch, cut, y Price framework SYMPHONY. |
| 2002 | ANGEL, J. C., SOLER, D., HERVAS, A. <i>The capacitated general routing problem on mixed graphs</i> . Revista investigación operacional, volume 23, No. 1. 2002 | Heurística route first, cluster second en grafos mixtos. |
| 2002 | GENDREAU, M., LAPORTE, G., POTVIN, J-Y. <i>Metaheuristics for the capacitated VRP</i> . The vehicle routing problem, volume 9 of SIAM monographs on discrete mathematics and applications, chapter 6. 2002 | Discute las metaheurísticas que han sido aplicadas para resolver el CVRP en el tiempo: Recocido Simulado, Recocido Determinístico, Búsqueda Tabú, Algoritmos Genéticos, Colonias de Hormigas y Redes neuronales. |
| 2003 | BERGER, J., BARKAOUI, M. <i>A hybrid genetic algorithm for the capacitated vehicle routing problem</i> . Genetic and evolutionary computation – GECCO, volume 1, pages 646-656. 2003 | Algoritmo genético híbrido, evolucionando dos poblaciones de soluciones para minimizar la distancia total recorrida y utilizando diversos operadores genéticos. |
| 2004 | PRINS, Christian. <i>A simple and effective evolutionary algorithm for the vehicle routing problem</i> . Computers & Operations Research, volume 31, issue 12, pages 1985-2002. October 2004. | Algoritmo genético híbrido sin delimitadores de rutas y potencializado con operadores de búsqueda local, utilizando el operador OX para la recombinación. |
| 2006 | CHEN, A-L., YANG, G-K., WU, Z-M. <i>Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem</i> . Journal of Zhejiang University SCIENCE A, volume 7, issue 4, pages 607-614. 2006 | Algoritmo híbrido entre el uso de la optimización discreta de la partícula swarm, utilizada para la búsqueda óptima de los vecindarios globales y locales, y el recocido simulado con ciertas probabilidades para evitar quedar atrapado en óptimos locales. |
| | | |

| | | |
|------|--|--|
| 2008 | TOTH, P., TRAMONTANI, A. <i>An integer linear programming local search for capacitated vehicle routing problems</i> . The vehicle routing problem: Latest advances and new challenges, volume 2, pages 275-295. 2008 | Algoritmo de búsqueda local para el VRP, basado en la exploración de vecindarios de orden exponencial resolviendo un problema de programación lineal y una heurística de refinamiento propuesta por Franceschi et al. |
| 2009 | DAZA, J. M., MONTOYA, J.R., NARDUCCI, F. <i>Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases</i> . Revista EIA, No. 12, páginas 23-38. Diciembre 2009. | Route first, cluster second, utilizando un tour gigante generado por heurísticas de arranque y mejoradas a través de la búsqueda Tabú. Posteriormente la planificación de la flota de vehículos fue realizada mediante la técnica de scheduling. |
| 2011 | VENKATESAN, S.R., LOGENDRAN, D., CHANDRAMOHAN, D. <i>Optimization of capacitated vehicle routing problema using PSO</i> . International Journal of Engineering Science and Technology, volume 3, number 10, pages 7469-7477. October 2011 | Cluster first, route second, utilizando las heurísticas de Clark and Wright y Sweep para formar los cluster, y posteriormente resuelve cada TSP a través de la optimización de partículas swarm. |

Fuente: propia

2.2.1 Técnicas exactas

Los algoritmos exactos son aquellos que siempre producen una solución óptima. Dichas técnicas no son adecuadas en aplicaciones que requieren soluciones rápidas o que tratan de resolver instancias de problemas muy grandes. Debido a la naturaleza NP del problema VRP, la búsqueda exhaustiva de estas técnicas no resulta eficiente computacionalmente (Corona, José (2005)). Entre las que se encuentran:

- Ramificación y acotamiento (Branch and Bound, hasta 100 nodos, *Fisher, 1994*)
- Ramificación y corte (Branch and Cut, *Augerat et al., 1998*)
- Programación dinámica (*Balinski y Quandt, 1964*)
- Programación lineal entera (*Balinski y Quandt, 1964*)

2.2.2 Técnicas Heurísticas

Las técnicas denominadas como heurísticas realizan una exploración relativamente limitada en el espacio de búsqueda y comúnmente producen soluciones de buena calidad dentro de tiempos de cómputo razonables. En un macro-nivel, las heurísticas para el VRP están divididas en 4 niveles, de los cuales se hará énfasis en los dos primeros, y sólo se mencionarán los últimos dos, que son respectivamente: 1) Heurísticas de *construcción* de la solución inicial, 2) Heurísticas de procedimientos de *mejora*, 3) Heurísticas de manejo de la *población*, 4) Heurísticas de procedimientos de *aprendizaje*. (Cordeau et al. (2004)).

2.2.2.1 Heurísticas de construcción

Parten de un problema y una solución “vacía” para que a partir de ella se pueda construir una solución factible pero que casi nunca resulta óptima. Las ideas detrás de la mayor parte de éste tipo de heurísticas se encuentran muy bien documentadas en (Laporte y Semet (2002)).

❖ Método del ahorro (Clarke and Wright (1964))

De igual modo que es una heurística extremadamente simple, es sin duda la que ha sido más utilizada en la práctica debido a su simplicidad y flexibilidad. Para éste caso, se supone que el agente viajero parte del nodo 1 y que la matriz de distancias C es simétrica, es decir, que $c_{ij} = c_{ji}$ para todo par de nodos i y j . La noción de economía explotada por éste método es la siguiente: se genera una economía por ir del nodo j a partir del nodo i , en lugar de hacer dos viajes ida y vuelta a partir del nodo 1, como se muestra a continuación (Martel, Alain (2003)):

$$s_{ij} = (c_{1i} + c_{i1} + c_{1j} + c_{j1}) - (c_{1i} + c_{ij} + c_{j1}) = c_{i1} + c_{1j} - c_{ij}$$

❖ Algoritmo del ahorro (Versión paralela y versión secuencial) (Martel, Alain (2003))

Dos variantes de la heurística expuesta anteriormente son encontradas a continuación. Utilizando ésta herramienta, nos encontramos resolviendo problemas de afectación de clientes y agentes viajeros simultáneamente. A continuación se presentan las dos versiones:

1. Para cada par de clientes, calcular las economías:

$$s_{ij} = c_{0i} + c_{j0} - c_{ij}$$

2. Organizar los arcos en orden decreciente por sus economías, y organizarlos en una lista. Llamar el arco en lo alto de la lista cómo (p, q) .

Versión Paralela:

3. Agregar el arco (p, q) a una ruta si no hay alguna restricción violada. (En el inicio se supone que hay tantas rutas cómo clientes, es decir que se puede realizar un viaje ida y vuelta para cada cliente)
4. Pasar al próximo arco en lista y llamarlo (p, q) . Repetir la etapa 3 hasta que no sea posible agregar más arcos.

Versión Secuencial:

3. Agregar el arco (p, q) a la ruta en curso si no hay alguna restricción violada, y si el cliente p o el cliente q se encuentran en el extremo de la ruta.
4. Pasar al próximo arco sobre la lista y llamarlo (p, q) . Repetir la etapa 3 hasta que no se puedan agregar más arcos. Escoger el primer arco de la lista, para comenzar una nueva ruta.
5. Continuar con las etapas 3 y 4 hasta que no se puedan escoger más arcos.

❖ Métodos de inserción

Éstas heurísticas de construcción se desarrollan insertando clientes que no han sido asignados a una ruta dentro de las ya existentes o nodos libres. Inicialmente se escogen dos puntos de partida para comenzar la ruta del agente viajero, y posteriormente según el criterio del método seleccionado, se van evaluando las diversas alternativas, ganando el lugar de ingreso a la ruta la que tenga un valor objetivo mejor a las demás posibilidades.

Entre los diversos métodos de resolución de éste problema por ésta metodología, se encuentran:

- 1) *Vecino más cercano, aleatorio o no.*
- 2) *Inserción del más lejano*
- 3) *Inserción la menos costosa*

❖ Heurísticas de dos fases (Corona, José (2005)) (Gillet et al. (1974))

Estas técnicas, parten de un problema y una solución “vacía” para que a partir de ella se pueda construir una solución factible pero que casi nunca resulta óptima. Para la obtención de la solución, las heurísticas de dos fases dividen al problema de VRP en dos etapas: una de asignación de clientes a vehículos, y otra para la determinación del orden de visita a dichos clientes. Existen luego dos grandes grupos de solución, y a los cuales pertenecen diversos métodos específicos de solución, como se agruparán a continuación.

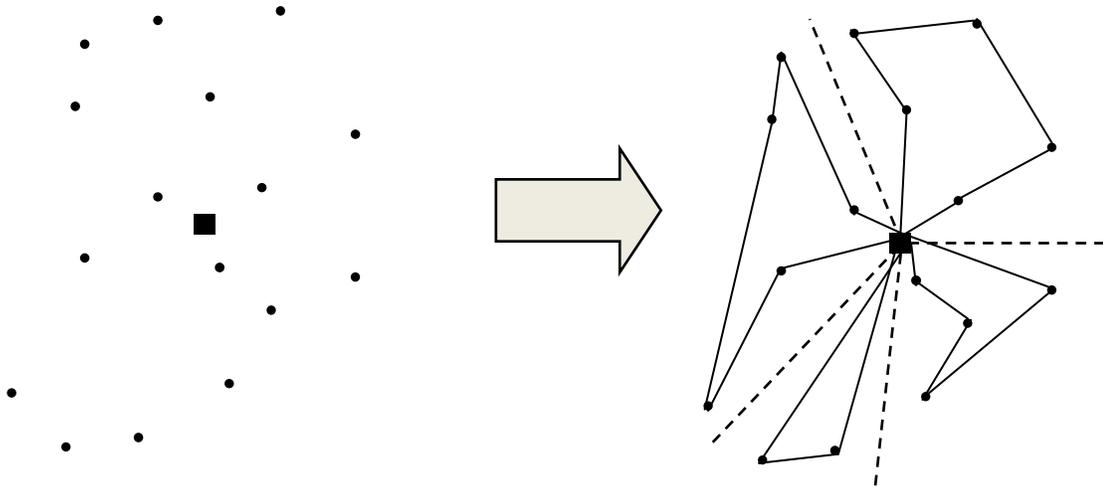
- *Métodos de asignar primero y rutear después*

En los métodos asignar primero y rutear después (*cluster first, route second*) primero se busca generar grupos de clientes, también llamados clusters, que estarán en una misma ruta en la solución final. Luego, para cada cluster, se crea una ruta que visite a todos sus clientes. Las restricciones de capacidad son consideradas en la primera etapa, asegurando que la demanda total de cada cluster es un TSP, que, dependiendo de la cantidad de clientes en el cluster, se puede resolver en forma exacta o aproximada.

Heurística de barrido o Sweep: Esta heurística será la utilizada en el desarrollo del presente proyecto, por lo tanto se ampliará la información sobre la misma. En dicho método los clusters se forman girando una semirrecta con origen en el depósito e incorporando los clientes barridos por dicha semirrecta hasta que se viole la restricción de capacidad. Cada cluster es luego ruteado resolviendo un TSP de forma exacta o aproximada. Este algoritmo puede aplicarse en problemas planos, es decir, en los que cada nodo corresponde a un punto en el plano y las distancias entre ellos se definen como su distancia euclidiana. Se supone que cada cliente v_i está dado por sus coordenadas polares (ρ_i, θ_i) en un sistema que tiene el depósito como origen. A continuación se enuncian los pasos de solución:

- *Paso 1 (Inicialización)*: Ordenar los clientes según θ de manera creciente. Si dos clientes tienen igual valor de θ , colocar primero el de menor valor de ρ . Seleccionar un cliente v_i para comenzar y hacer $k = 1$ y $R_k = \{v_i\}$.
- *Paso 2 (Selección)*: Si todos los clientes pertenecen a algún cluster, ir a 3. Si no, seleccionar el siguiente cliente v_{i+1} . Si v_{i+1} puede ser agregado a R_k sin violar las restricciones de capacidad, hacer $R_k = R_k \cup v_{i+1}$. Si no, hacer $k = k + 1$ y crear un nuevo cluster $R_k = \{v_i\}$. Ir a 2.
- *Paso 3 (Optimización)*: Para cada cluster R_k donde $k = 1, \dots, m$, resolver un TSP con sus clientes.

Figura 3. Heurística de Barrido



- *Métodos de Rutear primero y asignar después*

En los métodos asignar primero y rutear después (*Route first, cluster second*), Primeramente se calcula una ruta que visita a todos los clientes resolviendo un TSP gigante, sin respetar las restricciones del problema, para después, particionar dicha ruta en varias sub-rutas, cada una de las cuales es factible bajo el parámetro de capacidad del camión.

2.2.2.2 Heurísticas de mejora iterativa

Una vez que se tiene una solución para un problema VRP, se puede intentar mejorar mediante algún procedimiento de búsqueda local. Para cada solución s se define un conjunto de soluciones vecinas $N(s)$. Un procedimiento de búsqueda local parte de una solución s , la reemplaza por una solución $s' \in N(s)$ de menor costo y repite el procedimiento hasta que la solución no pueda ser mejorada. Al terminar, se obtiene una solución localmente óptima respecto a la definición de la vecindad. Entre ellas se encuentran (Corona, José (2005)) (Gendreau et al. (2013)):

- ❖ λ - intercambio (Osman (1993))
- ❖ El operador Or-opt (Or (1976))
- ❖ Operadores String Exchange y String Relocation (Van Breedam (1994))
- ❖ GENI y GENIUS (Gendreau, Hertz y Laporte (1992))
- ❖ Transferencias cíclicas (Thompson y Psaraftis (1993))
- ❖ Intercambios r -opt (Lin (1965))
- ❖ Mecanismos unstringing y stringing (US) (Gendreau et al. (1992))

2.2.3 Técnicas Metaheurísticas:

El énfasis en las técnicas metaheurísticas está en realizar una exploración profunda de las regiones más prometedoras del espacio solución. La calidad de las soluciones producidas por estos métodos es mucho mejor que la obtenida por las heurísticas clásicas, aunque su tiempo de cómputo es por lo general mucho mayor.

Existen dos tipos principales de familias de técnicas heurísticas avanzadas (*metaheurísticas*), que son, las basadas en métodos de búsqueda local, y las que explotan las poblaciones de soluciones (Algoritmos genéticos y meméticos). A continuación se presentarán ideas claves y las técnicas más utilizadas bajo cada familia.

❖ *Métodos de búsqueda local o por vecindario*

Su principio de funcionamiento de base, parte del hecho que hay una solución inicial realizable, y a ésta se le aplican una serie de modificaciones locales a la solución, entre tanto, dichas modificaciones mejoren el objetivo del problema en estudio. Dichos métodos, se basan en la utilización de heurísticas de mejora iterativa, y tienen diversos problemas, como son: el hecho de encontrar óptimos locales, la calidad de las soluciones y su tiempo de cálculo dependen de la riqueza de las transformaciones permitidas. Por otra parte es necesario definirles la forma en la cual está definido su espacio de búsqueda explorado por el algoritmo, los cuales serán definidos a continuación:

- *Espacio de búsqueda*: Es simplemente el conjunto de todas las soluciones que pueden ser consideradas (visitadas) durante la ejecución del algoritmo. Puede corresponder también, al dominio realizable de un problema, con cada punto que corresponde a una solución del mismo, respetando todas las restricciones.
- *Vecindario*: En cada iteración, las transformaciones que pueden ser aplicadas a la solución en curso S , definen un conjunto de soluciones vecinas de S , y esto es lo que se llama vecindario $N(S)$ de S . En general, para todo problema, existen más posibles definiciones de vecindarios que de espacios de búsqueda.

Entre los principales métodos de búsqueda local se encuentran:

- Búsqueda local simple
- Búsqueda local múltiple
- Rococido simulado
- Variable Neighborhood Search (VNS)
- Variable Neighborhood Descent (VND)
- Métodos de búsqueda con tabús (simples y probabilísticos)

Actualmente, las tendencias recientes, están trabajando el llamado paralelismo e híbridos, los cuales respectivamente, trabajan con cálculos paralelos para acelerar las fases más exigentes en cálculo o que intercambian información para encontrar mejores soluciones, ver para más información el capítulo del libro de (Enrique Alba (2005)). Y posteriormente, la otra tendencia es la de realizar conjunciones entre diversos métodos de optimización (Branch-and-Bound, algoritmos genéticos, de hormigas, meméticos, programación por restricciones y métodos de búsqueda local), los cuales han demostrado ser los métodos más eficientes a hoy de ésta rama de optimización matemática.

❖ *Métodos de exploración poblacional*

Las metaheurísticas basadas en poblaciones o poblacionales, son aquellas que emplean un conjunto de soluciones (población) en cada iteración del algoritmo, en lugar de utilizar una única solución como las metaheurísticas de trayectoria o de búsqueda local. Estas metaheurísticas proporcionan de forma intrínseca un mecanismo de exploración paralelo del espacio de soluciones, y su eficiencia depende en gran medida de cómo se manipule dicha población (Duarte, A. (2007)).

Entre los principales métodos basados en manejo de poblaciones se encuentran:

- Algoritmos genéticos
- Algoritmos meméticos
- Optimización mediante cúmulo de partículas (Swarm)
- Evolución Diferencial
- Optimización por colonia de hormigas
- Algoritmos de estimación de distribución
- Algoritmos de búsqueda dispersa (Scatter Search y Path Relinking)

SECCIÓN 3. METODOLOGÍA IMPLEMENTADA

3.1 Introducción

Inicialmente, basados en las experiencias de la literatura especializada y del análisis del problema según sus características, se establece que el método de solución apropiado para evaluar su desempeño para este problema, es la implementación de técnicas heurísticas y metaheurísticas de optimización, aprovechando las características combinatoriales del mismo. Ahora bien, ¿Por qué utilizar un método aproximado y no un método exacto de solución? La respuesta se encuentra en la dependencia que existe entre la topología de la red del problema y la complejidad en tanto a esfuerzo y dificultad para la resolución del mismo.

Cuando existe dicha dificultad, se dice que el problema está enmarcado como combinatorial, es decir, la inhibición de una posible búsqueda exhaustiva para la solución del mismo, conlleva al concepto llamado **explosión combinatorial** (Gallego et al. (2008)). Es aquí donde radica el interés por encontrar métodos eficientes aproximados, no sólo porque los problemas de ruteo de vehículos son considerados NP-completos; si no que, también, al día de hoy, algunas instancias Euclidianas con menos de 75 nodos aún no han sido resueltas hasta su óptimo cómo se verá más adelante, en el desarrollo del trabajo (Prins, Christian (2004)).

Dentro de éstas técnicas que se utilizarán para el desarrollo del trabajo actual, se encuentran las heurísticas de barrido, ahorros, golosa y vecino más cercano aleatorio, además de la evolución de la población de posibles soluciones por medio del algoritmo genético de Chu-Beasley potencializándolo con elementos propuestos por autores como Christian Prins y Mauricio Resende en diversos elementos del proceso iterativo del código, cómo se documentará a continuación, y así poder evaluar su desempeño comparándolo con las experiencias de otros investigadores.

Para el desarrollo del trabajo presentado y para dar solución a los casos de prueba seleccionados del tipo *SCVRP* (*Symmetric Capacitated Vehicle Routing Problem*), la metodología de solución será implementada en un algoritmo desarrollado en el software de ingeniería MATLAB, con datos de entrada como la instancia de prueba a resolver y de salida cómo son las soluciones de cada cluster por vehículo.

3.2 Descripción General del Algoritmo

La metodología de solución implementada para el desarrollo del trabajo actual es la mencionada anteriormente como de dos fases, es decir; está guiada inicialmente por la aplicación de la heurística de barrido o Sweep(Corona, José (2005)) (Gillet et al. (1974)) generando los Clusters de afectación por vehículo, y luego la solución a cada uno como un problema de agente viajero (TSP) a través del AGCB. A continuación se hace referencia a cada una de las fases y posterior análisis detallado de cada una de ellas.

Fase 1: Creación de los Clusters de afectación para cada uno de los vehículos, dependiendo de su capacidad y ubicación de los clientes respecto al depósito.

Fase 2: Se da solución a cada cluster establecido en la etapa anterior, implementando el AGCB adaptado a un problema de optimización del agente viajero simétrico.

3.3 Implementación de la Fase 1

En la primera fase de la metodología implementada se procede con la importación de los datos de las instancias seleccionadas del tipo *SCVRP (Symmetric Capacitated Vehicle Routing Problem)* Euclidiano, es decir; la información de entrada de cada cliente es su posición en (x, y) y su respectiva demanda, además de la capacidad general de los camiones. Posteriormente se procede a generar el agrupamiento de los clientes que serán solucionados en la segunda fase del código, de la siguiente manera.

- Cómo la información de los clientes está dada en sus coordenadas (x, y) es necesario realizar una matriz cuadrada, con infinitos en la diagonal principal ya que la distancia entre un mismo cliente no es considerada, y realizar el cálculo de la distancia entre todos los puntos a través de un razonamiento pitagórico, así: $\sqrt{\Delta x_{ij}^2 + \Delta y_{ij}^2} = Distancia_{ij}$ Y de ésta forma obtener una matriz simétrica con las distancias entre clientes de la forma $Distancia(i, j) = Distancia(j, i)$.
- Ahora bien, se procede a calcular el ángulo de cada cliente respecto a una recta arbitraria que parte del depósito central, y se organizan los clientes de menor a mayor ángulo.
- Luego, tomando un camión k no utilizado, y partiendo del cliente no afectado con el ángulo θ_i más pequeño, adicionar a la ruta del camión dicho cliente, en tanto que la capacidad del mismo no sea excedida. Si restan clientes sin atender se debe recomenzar tomando un nuevo camión k .
- Si al intentar incluir en un camión k no utilizado un cliente n en el cluster en creación y el mismo no puede ser incluido debido a la violación de la restricción de capacidad del vehículo. Se procede a evaluar el siguiente cliente $n+1$ en la lista, dando un salto, y de esta forma determinar si el mismo puede ser incluido, esto con el fin de evitar que al final queden clientes sin adicionar en un cluster determinado.
- El paso anterior se realiza hasta máximo tres veces para intentar incluir un cliente, si no se encuentra alguno, se toma un nuevo camión k ; pero si sí se logra, se re-organiza la matriz de ángulos de los clientes, y se vuelve a comenzar la clusterización desde el cliente saltado.
- Finalmente se genera un vector para cada cluster con los clientes que pertenecen a él, para proceder en la siguiente etapa a solucionar el orden de visitas según la metodología seleccionada.

Figura 4. Implementación Fase 1

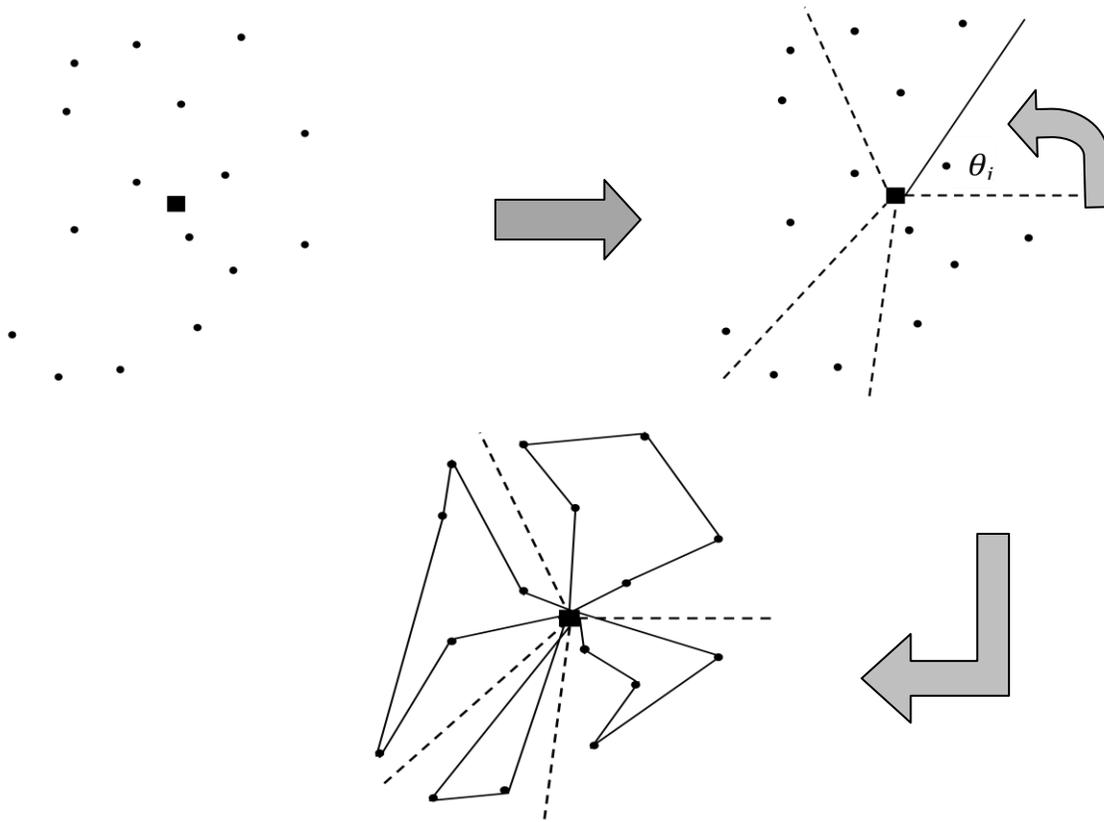
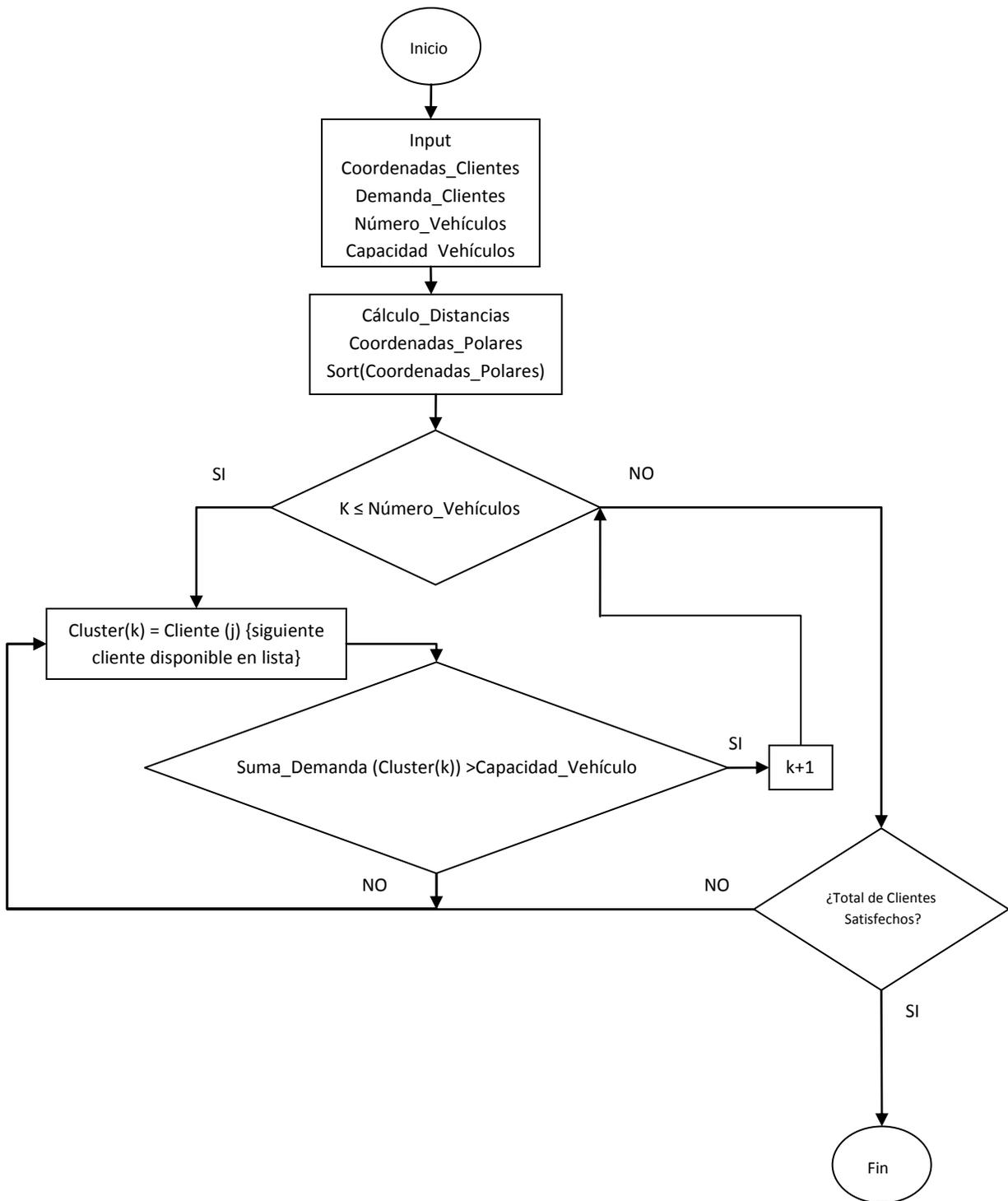


Figura 5. Diagrama de Flujo Fase 1



3.4 Implementación de la Fase 2

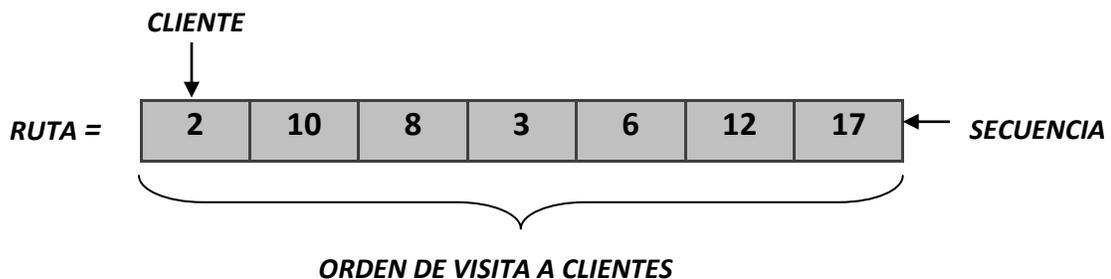
En ésta segunda etapa de la metodología de solución, se opta por optimizar la ruta de cada camión por cluster de forma separada resolviendo un problema del agente viajero TSP (como se mostró en la sección anterior) implementando un *Algoritmo Genético Modificado de Chu-Beasley* (Chu-Beasley (1997)), y potencializándolo con la calibración de parámetros para éste tipo de códigos de manejos poblacionales propuesto por (Christian Prins (2004)) y con algunas claves parciales aleatorias para algoritmos genéticos "*Biased Random-Key Genetic Algorithms*" propuestas por (Mauricio Resende, (2011)).

Por lo tanto, a continuación se realizará una explicación detallada de lo que es cada elemento mencionado anteriormente y su forma de aplicación en el código de solución del SCVRP, y posteriormente se listarán los procedimientos realizados dentro de la última fase del código, en su respectivo orden de aplicación dentro del mismo.

3.4.1 Codificación

Debido a la utilización de la heurística mencionada en la sección anterior de *dos fases*, la solución final al problema será un listado de n vectores, donde n es el número de Clusters generados en el agrupamiento inicial por la heurística de barrido. Luego, cada vector representa la solución óptima del recorrido del vehículo para el cluster $i = 1, \dots, n$ y finalmente cada elemento "gen" del vector representará el orden secuencial en que serán visitados cada uno de los clientes de dicho agrupamiento; el tamaño de cada solución n_i depende del número de clientes pertenecientes a cada grupo, limitado por la demanda de los mismos y la capacidad del vehículo.

Figura 6. Esquema de Codificación



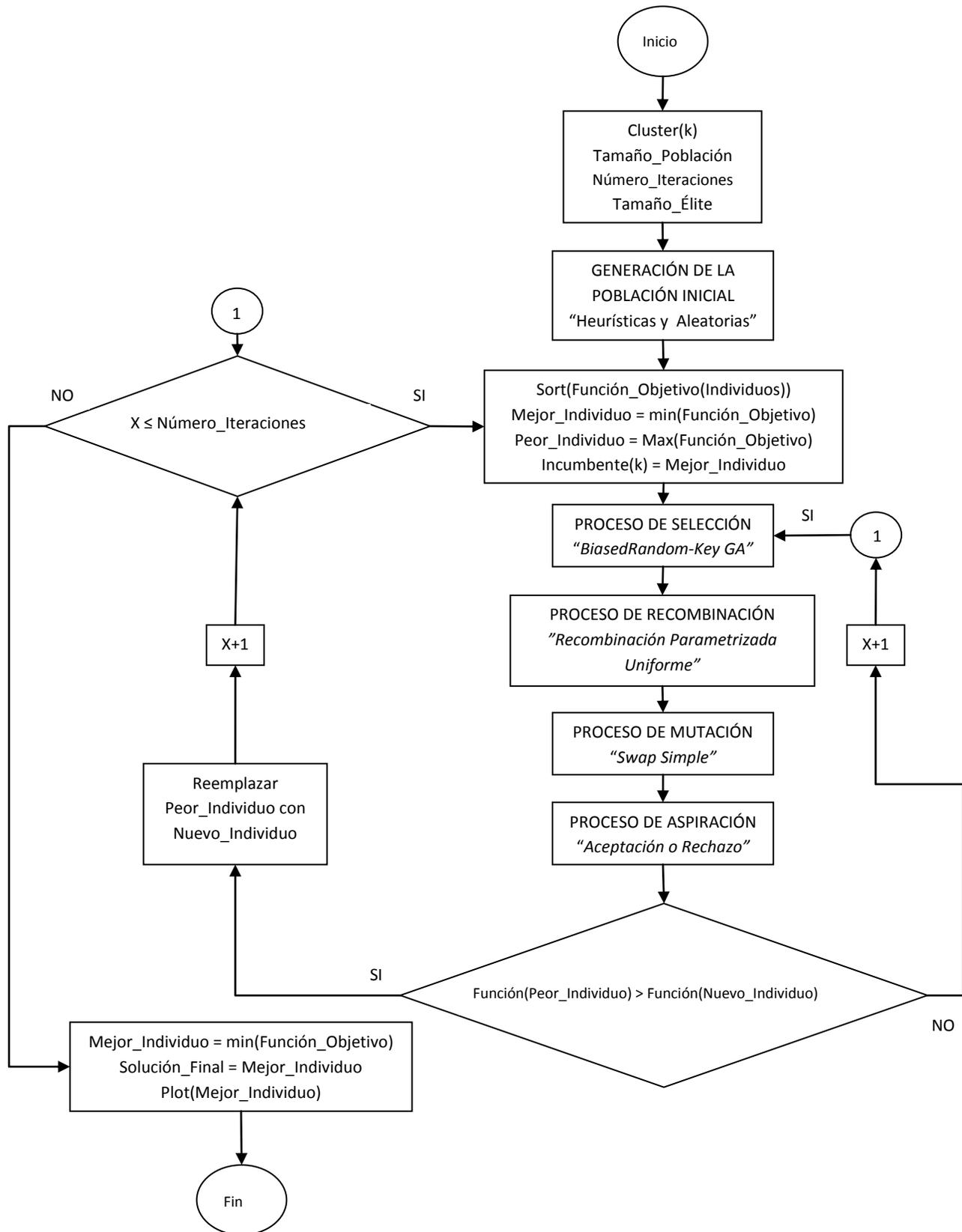
3.4.2 Descripción Detallada del AGCB Potencializado

El AGCB está basado y utiliza la teoría fundamental y los conceptos del algoritmo genético básico, pero presenta algunas diferencias que lo hacen muy competitivo para evaluar sistemas de gran tamaño y complejidad de forma más eficiente. En la figura 7 se documenta en un diagrama de flujo para mejor comprensión del lector. A continuación se enuncian las características más importantes y que tendrán inherencia en el desarrollo del presente trabajo (Gallego et al. (2008)).

- Utiliza la función objetivo para identificar el valor de la solución de mejor calidad, y lo utiliza para el proceso de reemplazo de una solución generada a través del proceso de selección – recombinación – mutación, por otra que se encuentra en la población actual.
- A diferencia del algoritmo genético propuesto por Holland, el algoritmo propuesto por Chu y Beasley sólo genera y sustituye un individuo a la vez en la población, en cada ciclo generacional.
- Es un algoritmo elitista, ya que un padre será reemplazado por un descendiente en la próxima generación, si y sólo si, el descendiente tiene una función objetivo de mejor calidad que el padre.
- Puede incluir una etapa de mejoramiento después de realizar selección, recombinación y mutación. Esto permite explotar la solución descendiente antes de determinar si se puede reemplazar a un individuo de la población actual.
- No se permiten soluciones infactibles en el desarrollo del código.

A continuación se presentan en más detalle algunas de las características del AGCB, y se explicarán en su momento de participación, los elementos potencializadores de ésta metodología mejorada propuesta.

Figura 7. Diagrama de Flujo AGCB



3.4.2.1 Población Inicial

Una forma de mejorar el desempeño del algoritmo genético, en los casos en que el problema es de gran tamaño y complejidad matemática, es generar de manera eficiente la población inicial. Para esto, puede generarse parte de la población inicial usando algoritmos heurísticos constructivos y la otra parte usando soluciones aleatorias controladas (Gallego et al. (2008). Es aquí donde en la generación de la población inicial, se utilizarán los conceptos para la calibración de parámetros publicados por (Christian Prins (2004)) quien documenta para los AG:

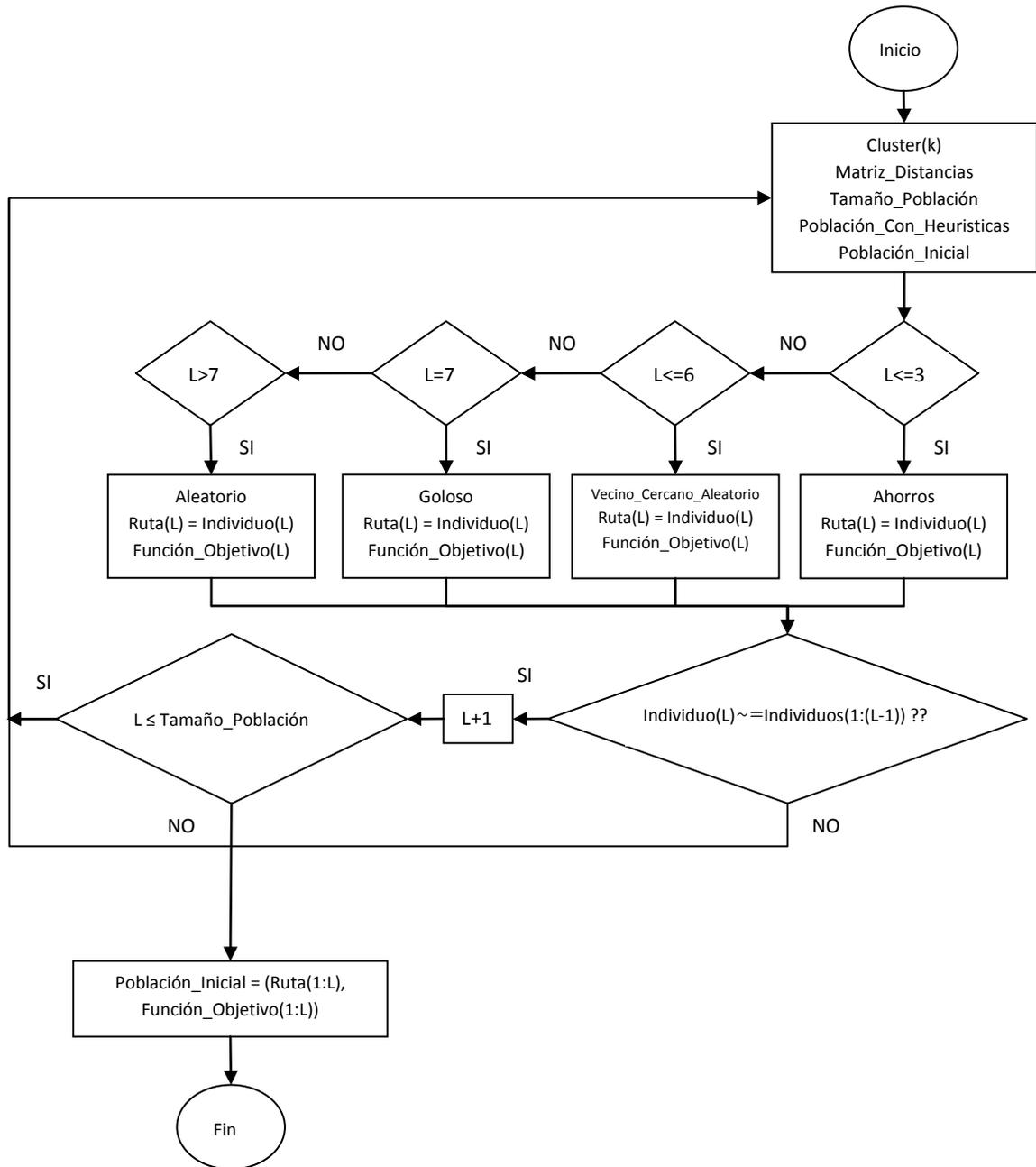
- ✓ *Tamaño de la población:* En total una población inicial de 30 individuos para una metodología utilizada con la heurística de barrido, es decir de dos fases. De los cuales 7 individuos serán generados utilizando algoritmos heurísticos constructivos, y el resto de forma aleatoria controlada. Siempre verificando que cada individuo generado sea diferente de los demás.
Heurísticas: Las heurísticas de construcción utilizadas para la generación de la población inicial serán: *vecino más cercano aleatorio, ahorros y goloso*, cada una con una participación de 3, 3, 1 individuos generados respectivamente.
Aleatorias: Respecto a las construcciones de la población inicial generadas por ésta metodología, serán de forma controlada, es decir, se tiene un conocimiento previo del problema y de las soluciones que pueden aparecer, generando los elementos del vector de cada cluster de forma factible y con una distribución de probabilidad para la generación de los números aleatorios de cada "gen" distribuidos uniformemente entre 0 y 1.
- ✓ *Tasa de mutación:* La tasa de mutación designada para éste tipo de problemas es entre 0.005 y 0.01 Sin embargo, por decisión del investigador, para la solución de éste problema, serán realizados swaps aleatorios a la población modificando el esquema de una ruta actual intercambiando dos posiciones aleatorias de la misma. Si el individuo resultante es mejor que el peor de todos, será reemplazado, de lo contrario la modificación será ignorada.
- ✓ *Iteraciones Locales:* Se plantean un máximo total de 5.000 iteraciones locales. Para cada TSP mayor a 4 clientes a optimizar.
- ✓ *Corridas por instancia:* Se establecen un total de 10 corridas por cada instancia trabajada.

En la Figura 8 podemos observar un diagrama de flujo para una mejor comprensión del proceso.

3.4.2.2 Función Objetivo e Infactibilidad

En el AGCB, para cada individuo de la población se calcula la función de costo, que es la función objetivo en éste caso. Además, no se generan soluciones infactibles por la utilización de la función *Randperm* de MATLAB que garantiza que no hayan repeticiones de clientes, ni sub-tours. Es por esto, que la función objetivo es la utilizada para implementar el mecanismo de selección y también en el proceso de sustitución de un individuo de la población, cuando toda la población, por la razón expresada anteriormente, es factible. Esta característica resulta ser una de las principales ventajas de este algoritmo, ya que no se requiere definir un factor de penalidad, el cual resulta siempre ser un valor subjetivo que afecta el desempeño del mismo. La función objetivo para cada solución de la población inicial es calculada como se muestra en el siguiente diagrama de flujo.

Figura 8. Diagrama de Flujo Generación de la Población Inicial



3.4.2.3 Selección

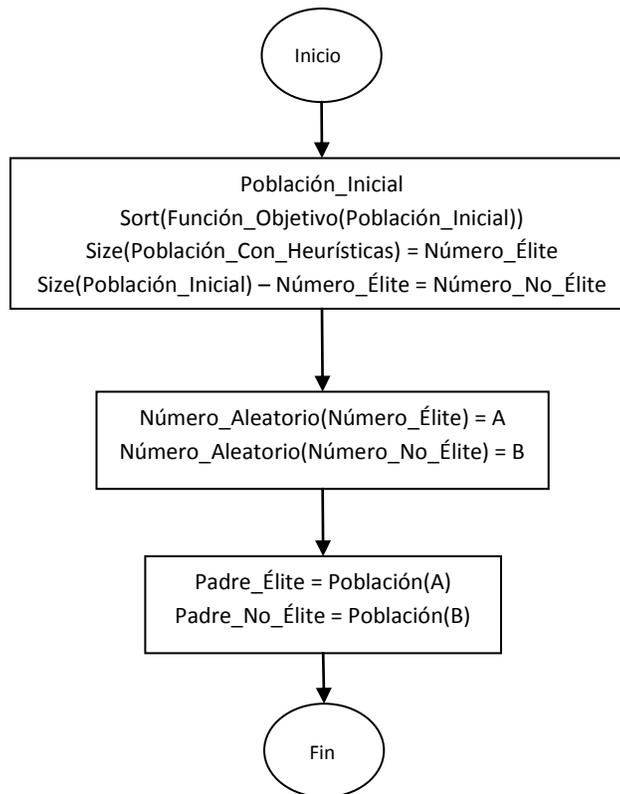
En el AGCB tradicional se utiliza el método de selección por torneo, en el cual, se realizan dos torneos, en cada uno de los cuales participan k individuos de la población actual (entre 2 y 4 generalmente). El proceso ocurre de la siguiente forma: se seleccionan aleatoriamente k individuos de la población actual, se comparan sus funciones objetivo y aquel que posea la mejor se almacena en la posición reservada para el padre número 1. Luego el proceso se realiza nuevamente para determinar el padre número 2. Posterior a la selección de los dos, ambos padres pasan a la siguiente fase llamada de recombinación (Gallego et al. (2008)).

Sin embargo, en el desarrollo planteado en este documento, se construye una relación para el proceso de selección del algoritmo, entre, la metodología antes mencionada de (Mauricio Resende (2011)) y (Christian Prins (2004)), donde el primero plantea que la población debe ser netamente aleatoria y el segundo sugiere unos parámetros mencionados en el literal anterior sobre el manejo de la población inicial, dando como resultado la siguiente metodología de solución.

Luego que la función de adaptación de cada elemento de la población inicial, entre los generados por heurísticas constructivas o de forma aleatoria, son decodificadas, la población es dividida en dos grupos de individuos, un pequeño grupo de individuos élite p_e , que son aquellos 7 individuos que tienen los mejores valores de función objetivo, y otro conjunto de individuos no-élite, siendo estos los faltantes de la relación entre $p - p_e$. Ahora bien, el método propuesto de selección difiere del tradicional AGCB también, en la forma como los padres son seleccionados para ser recombinados. Se había mencionado que el método tradicional es por torneo, mientras que en éste caso, un elemento padre es seleccionado de la partición de soluciones élite de la generación actual, y el otro de la partición no-élite, ambas selecciones de forma aleatoria. Estos son los únicos dos elementos que compartirán material genético en la siguiente etapa para dar paso a un nuevo elemento de la generación $k+1$. Además, se dice que el algoritmo es parcial, ya que un padre siempre es del segmento élite de la población y por otra parte, las repeticiones en la selección de las parejas son permitidas, por lo tanto, un individuo puede producir más de un descendiente en la misma generación.

A continuación se representa de forma gráfica el proceso de selección adaptado a este problema.

Figura 9. Diagrama de Flujo Proceso de Selección



3.4.2.4 Recombinación

Esta fase del algoritmo, denominada recombinación, es donde se genera el intercambio de material genético entre los dos padres seleccionados, el de la población élite y no-élite para este caso. Una de las ventajas del código utilizado, es que se suprime la necesidad de eliminar de forma indiscriminada a uno de los dos descendientes generados en el proceso en cuestión, ya que la metodología *Parameterized Uniform Crossover* (Spears y DeJong (1991)) nos evita este inconveniente. Esto debido a que se finaliza el proceso con un solo descendiente compartiendo material genético tanto del padre 1 como del padre 2, quien finalmente será la solución evolucionada que pasará a la próxima generación $k+1$ reemplazando al peor de los elementos de la población actual, según los criterios antes mencionados; y de ésta forma, recomenzar el proceso de selección del código. A continuación, se explicará en detalle la recombinación planteada con dichas ventajas.

3.4.2.4.1 Parameterized Uniform Crossover (*Spears y DeJong (1991)*)

En ésta metodología implementada para potencializar los algoritmos con claves parciales aleatorias (*Biased Random-Key Genetic Algorithms*), se establece $p_e = 0.7$ como la probabilidad que tiene un hijo de adquirir el material genético proveniente de las componentes del vector de su padre élite.

Sea n el número de componentes del individuo a recombinar, y para $i = 1, \dots, n$. El i -ésimo componente $c(i)$ del vector descendiente c toma el valor del i -ésimo componente $e(i)$ del padre élite e con la probabilidad p_e y el valor del i -ésimo componente $\bar{e}(i)$ del padre no-élite \bar{e} con una probabilidad de $1 - p_e$. Debido al tipo de codificación implementado, donde la información de cada *gen* significa necesariamente el nodo demandante que será visitado en ese orden de forma imperativa, se pueden presentar problemas de repetición de elementos que ya están incluidos en la solución en curso. Por lo tanto, se propone la metodología de que cuando se presenten estos casos, se deja el gen del descendiente vacío, y al final de la recombinación, se completarán los espacios en estas condiciones de forma sencilla. Es decir, se reemplazará con el nodo no visitado en orden ascendente de acuerdo a su posición en la tabla organizada de los ángulos respecto al depósito.

Finalmente, el hijo es reemplazado bajo los criterios establecidos en el literal “*Modificación de la Población*”, en la posición del último individuo que tendrá la peor función de adaptación. Luego, éste, con su nueva ruta y valor de función objetivo hará parte de un nuevo ciclo generacional. A continuación en la Figura 12 se detalla el proceso antes descrito para mejor comprensión.

Figura 10. Proceso de Recombinación

| | | | | | |
|---------------------|-----|-----|-----|-----|-----|
| P1 = | 3 | 1 | 5 | 4 | 2 |
| P2 = | 1 | 2 | 4 | 5 | 3 |
| Aleatorios = | 0.5 | 0.2 | 0.9 | 0.8 | 0.1 |
| Hijo = | 3 | 1 | 4 | 5 | 2 |

3.4.2.5 Mutación

Una vez obtenido el descendiente y reemplazado bajo el criterio de diversidad explicado a continuación en la población de la corrida del algoritmo. Se lanza en todas las iteraciones un proceso de mutación clásico, como se ilustra en la siguiente figura, llamado *Swap*, en el cuál se seleccionan dos puntos aleatorios (genes) y se cambia el uno por el otro. Dicho elemento mutado en cada iteración, es comparado con el peor de los individuos bajo el criterio de modificación de la población, si es diferente y con mejor función objetivo, éste es reemplazado y el elemento mutado entra a hacer parte de la nueva generación $k+1$. Ver el diagrama de flujo en la figura 13.

Figura 11. Proceso de Mutación

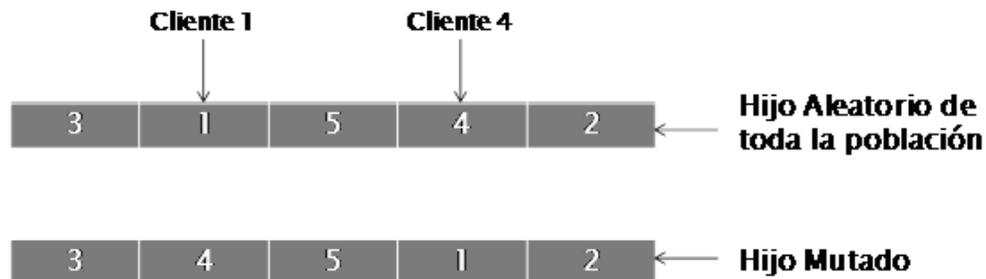


Figura 12. Diagrama de Flujo Proceso de Recombinación

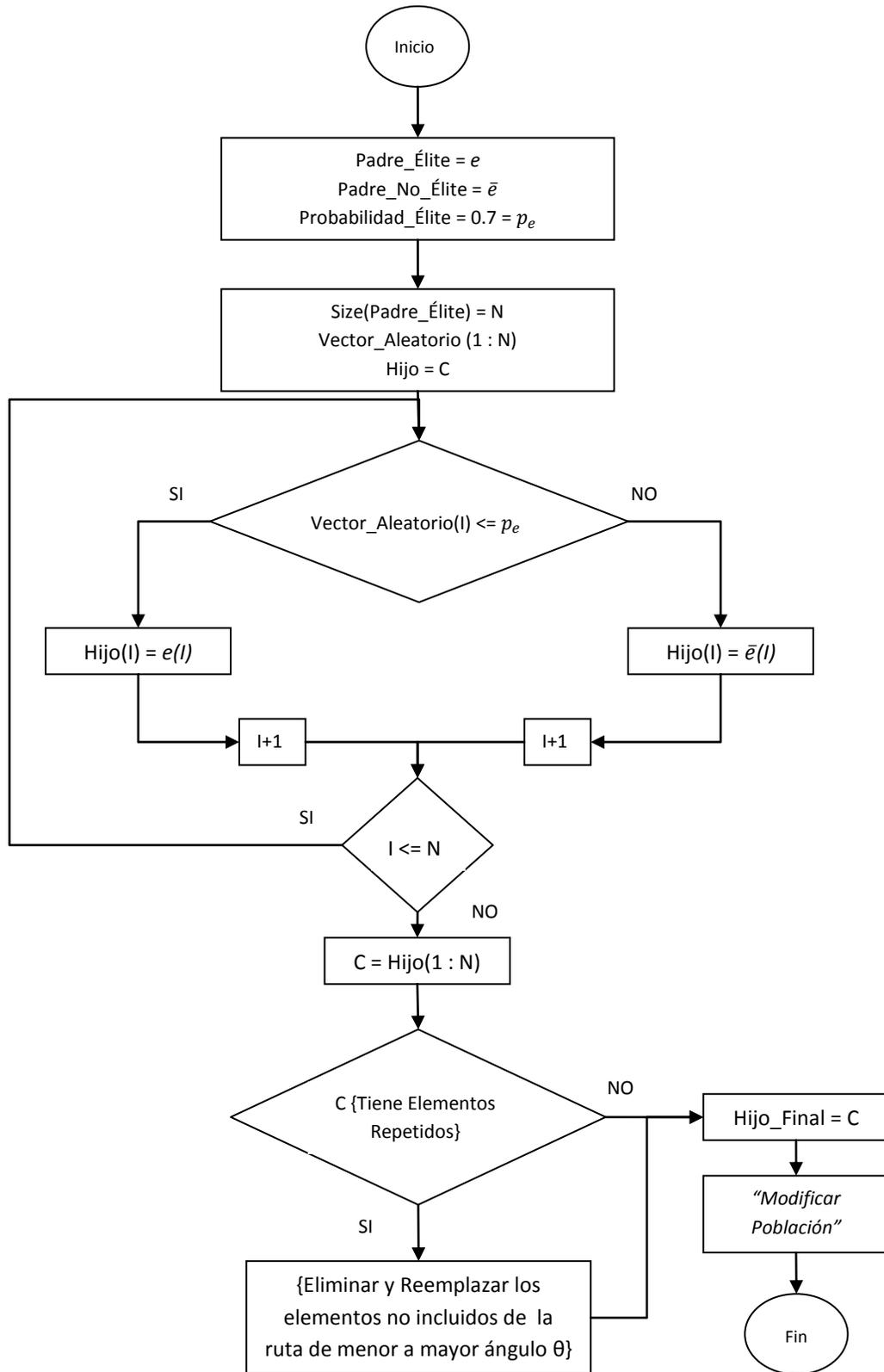
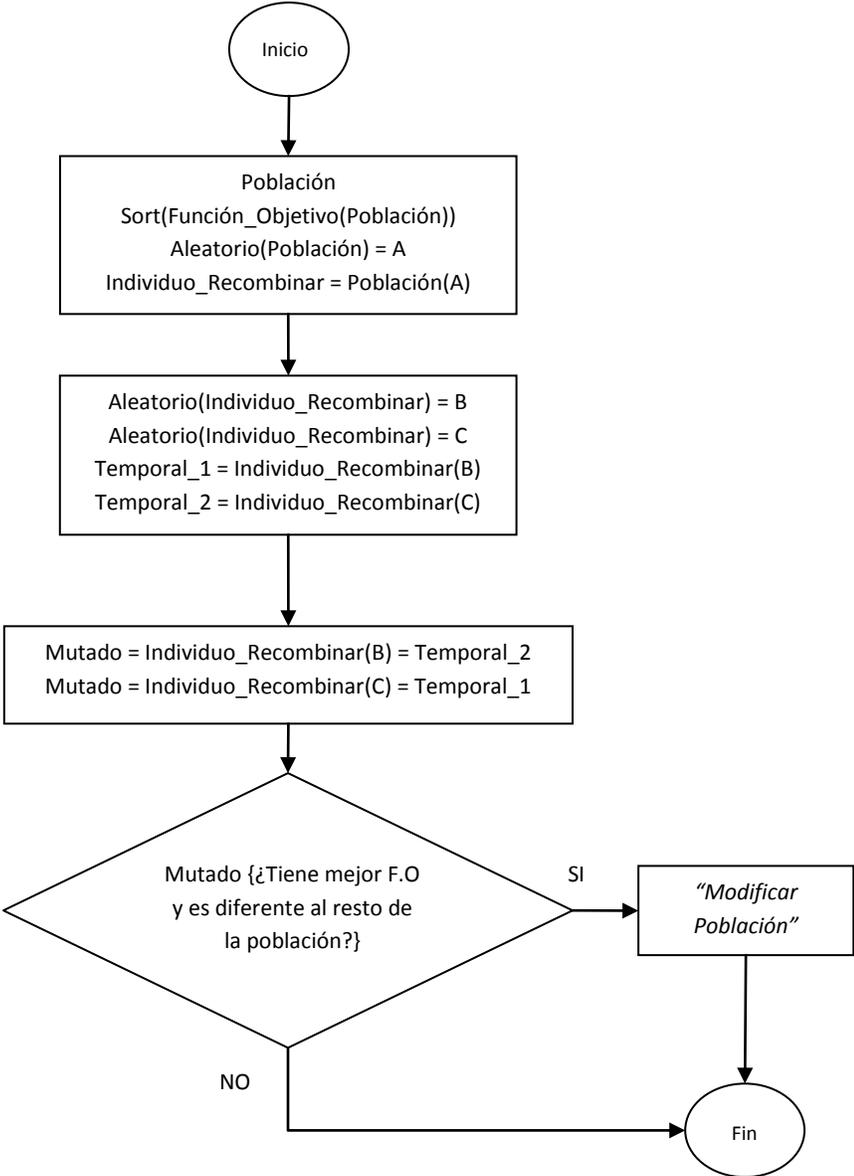


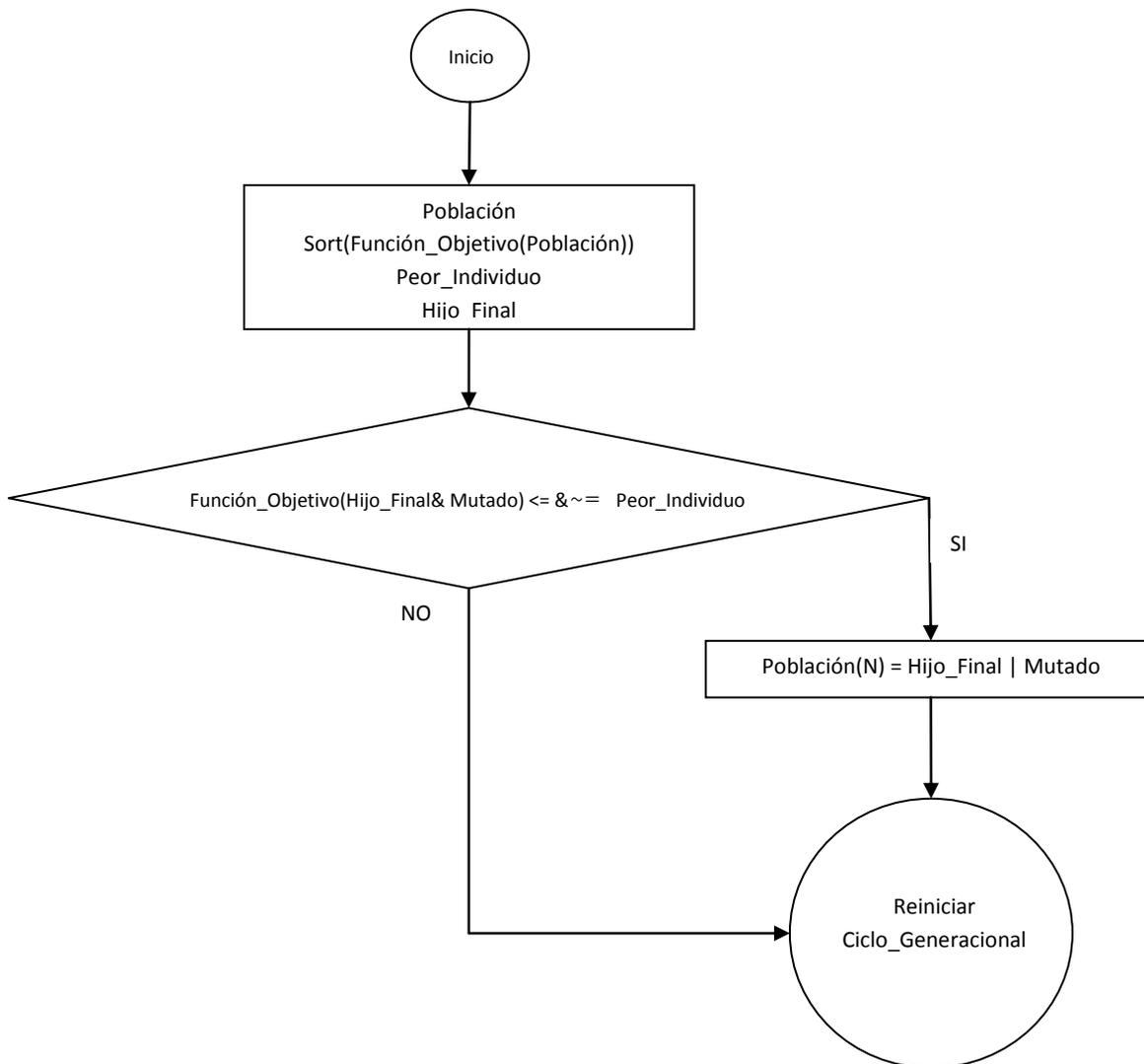
Figura 13. Diagrama de Flujo Proceso de Mutación



3.4.2.6 Modificación de la Población

En el AGCB potencializado, sólo se cambia un individuo de la población en cada ciclo generacional, a diferencia del AG tradicional quien en cada ciclo modifica prácticamente toda la población. El descendiente debe sustituir al elemento de la población de peor calidad siempre y cuando éste sea de mejor calidad que el peor de la población actual. Para éste caso además, como todos los elementos son factibles, se verifica únicamente los valores de sus funciones objetivo y que sea completamente diferente.

Figura 14. Diagrama de Flujo Modificación de la Población



Finalmente, para permitir que la metodología de solución sea más competitiva y tenga mayor posibilidad de búsqueda en el espacio de soluciones, se plantea un cambio en el ángulo de la recta arbitraria de arranque para la generación de los Clusters; la cual está inicialmente en $\theta = 0^\circ$. Posteriormente se aplicará a cada Cluster generado según la variación del ángulo de arranque, el código del AGCB Potencializado y guardando las mejores soluciones de los Clusters con mejor función objetivo global.

Para ejemplificar la variación propuesta en el código, se mantendrá un factor de desplazamiento de X según el tamaño de las instancias probadas, y un espacio de arranque distinto de Y entre clientes. Es decir, si la instancia es de 30 clientes, se tienen tres vehículos disponibles y se manejan factores de desplazamiento $X = 0.2$ y $Y = 0.1$, el número de veces que se realizará un desplazamiento es de 6, y el espacio entre arranques de 3 clientes. En otras palabras, los primeros 3 Clusters generados arrancando del cliente 1, dan una F.O.1 Global; los siguientes tres diferentes arrancando del cliente 4, dan una F.O.2; así sucesivamente hasta realizar el cambio de arranque 6 veces, obteniendo una F.O.6 Global. Finalmente se escoge el conjunto de Clusters con su solución óptima según la metodología que arroje una menor F.O Global.

3.5 Calibración de Parámetros

A continuación se presenta un listado y una tabla resumen de la calibración de parámetros manejada para un mejor aprovechamiento del código implementado.

- **Tamaño de la Población:** La población generada será constante y tendrá un total de 30 individuos, de los cuales 7 serán generados a través de algoritmos heurísticos constructivos y 13 serán generados de forma aleatoria controlada (Christian Prins (2004)).
- **Probabilidad de Recombinación de un Cromosoma Élite (P_e):** Dicha probabilidad de seleccionar un elemento del padre proveniente de la sección élite de la población inicial es de 0.7 (Mauricio Resende (2011)).
- **Mutación:** En cada iteración del AGCB se realizará un *Swap* doble.
- **Número de Iteraciones locales:** Dado que el código generado presenta tanta variabilidad en el mejor número de iteraciones a realizar, se definió una escala de repeticiones dependiendo del tamaño de las instancias propuestas por un autor determinado.

Tabla 2. Número de Iteraciones Locales Dado un Benchmark

| Benchmark | Número Clientes | Número Iteraciones |
|-----------------------------------|-----------------|--------------------|
| Augerat et al. | $N \leq 50$ | 1500 |
| | $N < 78$ | 2000 |
| | $N \geq 78$ | 3000 |
| Fisher | $N < 50$ | 3000 |
| | $N \geq 50$ | 5000 |
| Christofides, Mingozi and Toth | $N < 100$ | 2000 |
| | $N \geq 100$ | 3000 |
| Christofides y Eilon | $N \leq 76$ | 2000 |
| | $N > 76$ | 3000 |

- **Número de Corridas:** El número de corridas por cada instancia será manejado según lo propuesto por Christian Prins, quien especifica un total de 10 corridas por benchmark a realizar.
- **Variación de los Arranques X y Y:** Los arranques como ya fueron explicados anteriormente, varían según el tamaño de la instancia a trabajar. El propósito de variarlos, es el de lograr un cubrimiento (una vuelta completa) del total de clientes de las instancias, y de esta forma evaluar la mayor cantidad de funciones objetivo obtenidas con cada variación en el arranque. A continuación se muestran los valores porcentuales de X y Y según el tamaño de clientes por instancia.

Tabla 3. Variación de los Arranques X y Y

| Tamaño Instancia | Valor X | Valor Y |
|------------------|---------|---------|
| $N \leq 50$ | 20% | 10% |
| $N \leq 100$ | 15% | 7% |
| $N \leq 150$ | 12% | 6% |
| $N \leq 200$ | 10% | 5% |
| $N \leq 250$ | 7% | 6% |

SECCIÓN 4. RESULTADOS

4.1 Introducción

Con el fin de validar la metodología propuesta, es necesario evaluar casos de prueba de la literatura especializada a partir de los cuales se pueda medir, tanto la eficiencia computacional, como la precisión que tiene el algoritmo para obtener resultados de buena calidad. La variable cuantitativa que servirá como punto comparativo con los resultados propuestos por diversos investigadores, es la función objetivo obtenida en las corridas del algoritmo que busca minimizar las distancias totales recorridas por todos los camiones. Con la cual se obtiene el GAP, que es la diferencia porcentual que existe entre el valor de la función objetivo de la mejor solución conocida (BKS.FO) y la solución de la metodología propuesta.

Se plantearán por lo tanto dos tipos de GAP, uno relativo a la diferencia con el promedio de las 10 corridas calculadas a través de la metodología propuesta (GAP.Promedio), y uno respecto al mejor valor encontrado en las 10 corridas (GAP.Min). En las tablas presentadas a continuación para cada benchmark realizado, se presenta además una columna (OPT(*)), la cual indica que si hay un * presente en la columna, la fila que corresponde a dicha posición es una instancia para la cual ya se ha encontrado su valor óptimo global de solución. Dichos valores óptimos globales han sido encontrados a través de metodologías de solución exactas de exploración exhaustiva del espacio de soluciones, cómo se mencionaron anteriormente, especialmente con las técnicas de *branch and cuty branch and bound*. (Augerat et al. (1998)) (Blasum et al. (2000)) (Christofides&Eilon (1969)) (Fisher (1994)) (Fukasawa et al. (2003)) (Ralphs et al. (2003)).

Además, las tablas a presentar cuentan con información adicional para la comprensión e interpretación de los resultados, cómo; el nombre de la instancia tratada, el número de iteraciones realizadas para dicho cálculo, el promedio de las funciones objetivo encontradas, el valor mínimo presente entre las soluciones, y la información presentada anteriormente.

Los 20 casos de prueba trabajados en el presente documento pueden ser descargados de la página web del VRP (NEO Research Group (2013)), donde se pueden encontrar tanto las instancias, como los BKS de las mismas e información pertinente sobre el VRP.

4.2 Resultados en las Instancias de Augerat et al. (1998)

Un total de 7 instancias del grupo propuesto por Augerat en 1998 fueron documentadas para éste trabajo. Dicho grupo cuenta con 3 clases de variantes. Para las del grupo “A” tanto las ubicaciones como las demandas de los clientes son aleatorias. Las del grupo “B” son instancias clusterizadas, y finalmente las del grupo “P” son versiones modificadas de instancias presentes en la literatura. En éste caso sólo fueron trabajadas instancias del grupo “A” y “B”.

La codificación de éste grupo de instancias se presenta de la siguiente manera; **X-NYY-KZ**. Dónde, **X** es igual a “A, B ó P” que indica el autor de la instancia, para éste caso Augerat del grupo respectivo a la letra escogida. **YY** es el número de nodos del sistema; es decir, el número de clientes + 1. Finalmente **Z** es el número de vehículos a utilizar.

Todas las instancias de éste grupo de estudio, cuentan con su valor óptimo global, representados en la tabla presentada a continuación con un (*). Dichos valores fueron encontrados a través Implementación del algoritmo *branch-and-cut*(ver tabla número 1).

Tabla 4. Resultados Benchmark de Augerat et al.

| Nombre | OPT (*) | BKS.FO | # Iteraciones | Tipo.Medida | Promedio | Min | GAP.Min | GAP.Promedio |
|-----------|---------|--------|---------------|-------------|----------|---------|---------|--------------|
| A-N48-K7 | * | 1073 | 1500 | F.O | 1119,07 | 1119,07 | 4% | 4% |
| A-N48-K7 | * | 1073 | 1500 | Tiempo | 42,42 | 42,21 | | |
| A-N46-K7 | * | 914 | 1500 | F.O | 999,01 | 997,74 | 9% | 9% |
| A-N46-K7 | * | 914 | 1500 | Tiempo | 48,44 | 48,23 | | |
| B-N44-K7 | * | 909 | 1500 | F.O | 1074,93 | 1074,30 | 18% | 18% |
| B-N44-K7 | * | 909 | 1500 | Tiempo | 34,12 | 33,96 | | |
| A-N62-K8 | * | 1288 | 2000 | F.O | 1536,83 | 1536,70 | 19% | 19% |
| A-N62-K8 | * | 1288 | 2000 | Tiempo | 69,34 | 68,97 | | |
| B-N57-K9 | * | 1598 | 2000 | F.O | 1764,20 | 1764,20 | 10% | 10% |
| B-N57-K9 | * | 1598 | 2000 | Tiempo | 71,60 | 64,20 | | |
| A-N80-K10 | * | 1763 | 3000 | F.O | 2106,54 | 2105,80 | 19% | 19% |
| A-N80-K10 | * | 1763 | 3000 | Tiempo | 165,56 | 161,24 | | |
| B-N78-K10 | * | 1221 | 3000 | F.O | 1331,52 | 1327,50 | 9% | 9% |
| B-N78-K10 | * | 1221 | 3000 | Tiempo | 161,96 | 161,12 | | |

4.2.1 Solución Final de los Óptimos Locales

A continuación se presentan las rutas encontradas con menor función objetivo, por cada camión para cada instancia trabajada.

A-N48-K7

F.O: **1119,07**

| | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|
| Ruta 1: | 15 | 18 | 48 | 11 | 3 | 42 | | | |
| Ruta 2: | 29 | 30 | 22 | 31 | 14 | 47 | 34 | 17 | |
| Ruta 3: | 46 | 28 | 16 | 43 | 12 | 5 | 25 | 10 | 35 |
| Ruta 4: | 41 | 8 | 9 | 40 | 27 | 21 | 4 | 38 | |
| Ruta 5: | 45 | 36 | 33 | 39 | 26 | 20 | 37 | | |
| Ruta 6: | 24 | 44 | 32 | 2 | 7 | 23 | 19 | | |
| Ruta 7: | 6 | 13 | | | | | | | |

A-N46-K7

F.O: 997,74

| | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|
| Ruta1: | 20 | 21 | 28 | 2 | 41 | 35 | 32 | 30 | 42 |
| Ruta2: | 12 | 9 | 34 | 43 | 17 | 25 | 16 | | |
| Ruta3: | 37 | 40 | 26 | 19 | 36 | 13 | | | |
| Ruta4: | 6 | 23 | 8 | 45 | 11 | 4 | 33 | 39 | |
| Ruta5: | 14 | 5 | 31 | 3 | 38 | | | | |
| Ruta6: | 10 | 46 | 7 | 44 | 27 | 15 | 24 | | |
| Ruta7: | 29 | 22 | 18 | | | | | | |

B-N44-K7

F.O: 1074,3

| | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|---|
| Ruta1: | 26 | 29 | 4 | 13 | 2 | 41 | | | |
| Ruta2: | 36 | 28 | 32 | 44 | 34 | 16 | 21 | | |
| Ruta3: | 19 | 20 | 12 | 27 | 25 | 40 | 15 | 37 | 3 |
| Ruta4: | 17 | 31 | 11 | 10 | 33 | 30 | | | |
| Ruta5: | 24 | 39 | | | | | | | |
| Ruta6: | 8 | 14 | 43 | 23 | 9 | 5 | 42 | | |
| Ruta7: | 6 | 18 | 38 | 35 | 22 | 7 | | | |

A-N62-K8

F.O 1536,7

| | | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 29 | 57 | 61 | 26 | 50 | 5 | 12 | 6 | 15 | 49 | | |
| Ruta2: | 16 | 14 | 19 | 2 | 55 | 28 | 24 | 42 | 22 | 10 | 13 | 54 |
| Ruta3: | 39 | 20 | 40 | 4 | 43 | 31 | | | | | | |
| Ruta4: | 3 | 47 | 25 | 48 | 46 | 23 | | | | | | |
| Ruta5: | 37 | 17 | 38 | 45 | 53 | 52 | | | | | | |
| Ruta6: | 8 | 11 | 9 | 44 | 30 | 33 | 7 | 62 | 21 | | | |
| Ruta7: | 56 | 27 | 59 | 18 | 51 | 58 | 41 | | | | | |
| Ruta8: | 60 | 34 | 36 | 32 | 35 | | | | | | | |

B-N57-K9

F.O 1764,2

| | | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|--|--|--|--|--|
| Ruta1: | 4 | 29 | 10 | 47 | 23 | | | | | | | |
| Ruta2: | 32 | 56 | 28 | 19 | 18 | 11 | 57 | | | | | |
| Ruta3: | 2 | 46 | 25 | 53 | 44 | | | | | | | |
| Ruta4: | 5 | 31 | 39 | 49 | 34 | 55 | 13 | | | | | |
| Ruta5: | 50 | 36 | 15 | 35 | 20 | 45 | 43 | | | | | |
| Ruta6: | 8 | 27 | 40 | 14 | 12 | 21 | | | | | | |

| | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|---|--|--|
| Ruta7: | 37 | 30 | 38 | 24 | 17 | 3 | 54 | | | |
| Ruta8: | 9 | 16 | 26 | 52 | 42 | 22 | 48 | 6 | | |
| Ruta9: | 33 | 51 | 41 | 7 | | | | | | |

A-N80-K10

F.O 2105,8

| | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 50 | 39 | 59 | 33 | 5 | 46 | 74 | | | |
| Ruta2: | 37 | 68 | 67 | 71 | 77 | 51 | 23 | 55 | 73 | 54 |
| Ruta3: | 43 | 34 | 10 | 56 | 70 | 66 | 57 | 16 | 65 | 78 |
| Ruta4: | 32 | 26 | 36 | 48 | 42 | 47 | 61 | 40 | 4 | |
| Ruta5: | 75 | 18 | 21 | 76 | 58 | 27 | 20 | | | |
| Ruta6: | 14 | 30 | 28 | 60 | 62 | 17 | 44 | 79 | 6 | 45 |
| Ruta7: | 13 | 31 | 69 | 9 | 7 | 25 | 24 | | | |
| Ruta8: | 63 | 12 | 35 | 38 | 3 | 53 | 64 | | | |
| Ruta9: | 8 | 29 | 80 | 2 | | | | | | |
| Ruta10: | 11 | 72 | 49 | 19 | 15 | 22 | 41 | | | |

B-N78-K10

F.O 1327,5

| | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 57 | 9 | 23 | 72 | 13 | 35 | 47 | 52 | | |
| Ruta2: | 16 | 46 | 32 | 74 | 77 | 33 | 8 | 48 | | |
| Ruta3: | 58 | 21 | 55 | 4 | 30 | 76 | 66 | | | |
| Ruta4: | 64 | 31 | 71 | 38 | 50 | 28 | | | | |
| Ruta5: | 63 | 62 | 34 | 36 | 56 | 6 | 3 | | | |
| Ruta6: | 51 | 40 | 49 | 14 | 60 | 29 | | | | |
| Ruta7: | 67 | 61 | 42 | 15 | 37 | 5 | 75 | 20 | 10 | 7 |
| Ruta8: | 17 | 19 | 26 | 11 | 69 | 54 | 41 | 65 | 27 | 24 |
| Ruta9: | 78 | 45 | 12 | 2 | 25 | 53 | 22 | 44 | | |
| Ruta10: | 18 | 59 | 70 | 68 | 39 | 73 | | | | |

4.3 Resultados en las Instancias de Fisher (1994)

El total de las 3 instancias del grupo propuesto por Fisher en 1994 fueron documentadas para éste trabajo. Las tres variantes presentes, son respectivamente de tamaño pequeño (45 nodos), mediano (72 nodos) y grande (135 nodos).

La codificación de éste grupo de instancias se presenta también de la siguiente manera; **X-NYY-KZ**. Dónde, **X** es igual a “F” que indica el autor de la instancia, para éste caso Fisher. **YY** es el número de nodos del sistema; es decir, el número de clientes + 1. Finalmente **Z** es el número de vehículos a utilizar.

Todas las instancias de éste grupo de estudio, cuentan con su valor óptimo global, representados en la tabla presentada a continuación con un (*). Dichos valores fueron encontrados a través de la implementación del algoritmo *branch-and-bound* y utilizando los costos mínimos de K-Trees (ver *tabla número 1*).

Tabla 5. Resultados Benchmark de Fisher

| Nombre | OPT (*) | BKS.FO | # Iteraciones | Tipo.Medida | Promedio | Min | GAP.Min | GAP.Promedio |
|-----------|---------|--------|---------------|-------------|----------|---------|---------|--------------|
| F-N45-K4 | * | 724 | 3000 | F.O | 900,65 | 889,94 | 23% | 24% |
| F-N45-K4 | * | 724 | 3000 | Tiempo | 55,58 | 54,01 | | |
| F-N72-K4 | * | 237 | 5000 | F.O | 325,51 | 315,35 | 33% | 37% |
| F-N72-K4 | * | 237 | 5000 | Tiempo | 120,77 | 119,75 | | |
| F-N135-K7 | * | 1162 | 5000 | F.O | 1489,44 | 1454,94 | 25% | 28% |
| F-N135-K7 | * | 1162 | 5000 | Tiempo | 353,37 | 350,25 | | |

4.3.1 Solución Final de los Óptimos Locales

A continuación se presentan las rutas encontradas con menor función objetivo, por cada camión para cada instancia trabajada.

F-N45-K4

F.O 889,94

| | | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|----|----|---|
| Ruta1: | 2 | 39 | 43 | 40 | 35 | 33 | 28 | 7 | 6 | 8 | 36 | 4 |
| | 5 | 37 | 38 | | | | | | | | | |
| Ruta2: | 25 | 11 | 19 | 12 | 13 | 14 | 15 | 18 | 3 | 16 | 10 | |
| Ruta3: | 21 | 22 | 9 | 20 | 23 | 27 | 26 | 17 | | | | |
| Ruta4: | 24 | 29 | 30 | 34 | 32 | 41 | 42 | 31 | 45 | 44 | | |

F-N72-K4**F.O 315,35**

| | | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 12 | 19 | 72 | 7 | 11 | 6 | 4 | 10 | | | | |
| Ruta2: | 21 | 30 | 24 | 27 | 25 | 26 | 29 | 28 | 43 | 45 | 44 | 47 |
| | 54 | 46 | 53 | 49 | 48 | 51 | 50 | 23 | 22 | 31 | 5 | 8 |
| | 9 | | | | | | | | | | | |
| Ruta3: | 32 | 35 | 62 | 59 | 60 | 64 | 63 | 65 | 66 | 67 | 68 | 70 |
| | 38 | 39 | 41 | 69 | 40 | 57 | 58 | 71 | 52 | 42 | 56 | 55 |
| | 33 | | | | | | | | | | | |
| Ruta4: | 36 | 37 | 34 | 15 | 2 | 14 | 13 | 17 | 18 | 3 | 16 | 20 |
| | 61 | | | | | | | | | | | |

F-N135-K7**F.O 1454,94**

| | | | | | | | | | | | | |
|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|
| Ruta1: | 18 | 19 | 130 | 128 | 110 | 109 | 108 | 107 | 115 | 132 | 117 | |
| Ruta2: | 114 | 121 | 122 | 129 | 127 | 72 | | | | | | |
| Ruta3: | 81 | 111 | 123 | 124 | 125 | 112 | 126 | 113 | 82 | 133 | 67 | |
| Ruta4: | 74 | 75 | 78 | 65 | 64 | 79 | 134 | 34 | 69 | 70 | 71 | 68 |
| | 80 | 35 | 49 | 76 | 2 | 63 | 51 | 50 | 33 | 135 | 77 | 48 |
| | 73 | | | | | | | | | | | |
| Ruta5: | 61 | 62 | 55 | 56 | 58 | 106 | 57 | 104 | 105 | 99 | 98 | 97 |
| | 41 | 96 | 38 | 36 | 37 | 100 | 101 | 102 | 103 | 54 | 52 | 53 |
| Ruta6: | 60 | 32 | 59 | 94 | 95 | 46 | 39 | 40 | 44 | 45 | 4 | 42 |
| | 43 | 3 | 6 | 7 | 8 | 9 | 10 | 5 | 11 | 12 | 13 | 14 |
| | 17 | 93 | 30 | 31 | 29 | 28 | 27 | 25 | 24 | 23 | | |
| Ruta7: | 91 | 90 | 16 | 15 | 89 | 88 | 21 | 83 | 87 | 86 | 85 | 84 |
| | 66 | 131 | 120 | 116 | 118 | 20 | 119 | 47 | 22 | 26 | 92 | |

4.4 Resultados en las Instancias de CMT (1979)

Un total de 4 instancias del grupo propuesto por Christofides, Mingozzi y Toth en 1979 fueron documentadas para éste trabajo. Dichas instancias, llamadas como, " *los 14 problemas clásicos del VRP*", oscilan entre 50 – 199 clientes. Las instancias 6 – 10 y 13, 14 tienen restricciones de duración del largo de la ruta, además de tener tiempo de servicio diferentes a 0; por lo tanto, no

son consideradas para el desarrollo del presente trabajo. Mientras que las instancias 1 – 5 y 11, 12 son VRP puros; luego, de éste grupo son seleccionados los 4 problemas trabajados.

La codificación de éste grupo de problemas, no está relacionada con las características del mismo; es por esto, que es necesario realizar una tabla donde se especifiquen las características de las instancias documentadas.

Tabla 6. Características de las Instancias de CMT (1979)

| Instancia | Número de Nodos | Número de Vehículos |
|-----------------|-----------------|---------------------|
| vrpnc 1 | 51 | 5 |
| vrpnc 4 | 151 | 12 |
| vrpnc 11 | 121 | 7 |
| vrpnc 12 | 101 | 10 |

Sólo para 2 de las 14 instancias presentadas por Christofides se han encontrado sus óptimos globales debido a su alta complejidad matemática. Esto es para los casos de prueba vrpnc 1 y vrpnc 7. Para el resto de los casos, se han encontrado los mejores óptimos locales a través de técnicas metaheurísticas propuestas por diversos autores cómo algoritmos genéticos, búsquedas tabú y recocidos simulados. Para tener más información al respecto ver Christian Prins (2004) páginas 1996 -1998.

Tabla 7. Resultados Benchmark de CMT

| Nombre | OPT (*) | BKS.FO | # Iteraciones | Tipo.Medida | Promedio | Min | GAP.Min | GAP.Promedio |
|---------|---------|---------|---------------|---------------|----------|---------|---------|--------------|
| vrpnc1 | * | 524,61 | 2000 | F.O | 571,64 | 569,08 | 8% | 9% |
| vrpnc1 | * | 524,61 | 2000 | Tiempo | 55,41 | 55,11 | | |
| vrpnc4 | | 1028,42 | 3000 | F.O | 1136,28 | 1120,27 | 9% | 10% |
| vrpnc4 | | 1028,42 | 3000 | Tiempo | 349,52 | 346,48 | | |
| vrpnc11 | | 1042,11 | 3000 | F.O | 1295,57 | 1285,12 | 23% | 24% |
| vrpnc11 | | 1042,11 | 3000 | Tiempo | 180,67 | 179,78 | | |
| vrpnc12 | | 819,56 | 3000 | F.O | 1014,23 | 1011,52 | 23% | 24% |
| vrpnc12 | | 819,56 | 3000 | Tiempo | 174,13 | 173,61 | | |

4.4.1 Solución Final de los Óptimos Locales

A continuación se presentan las rutas encontradas con menor función objetivo, por cada camión para cada instancia trabajada.

vrpnc 1

F.O 569,08

| | | | | | | | | | | | |
|---------------|----|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 47 | 6 | 50 | 11 | 31 | 40 | 34 | 46 | 16 | 13 | |
| Ruta2: | 3 | 36 | 30 | 22 | 35 | 51 | 17 | 10 | 39 | 12 | |
| Ruta3: | 28 | 9 | 27 | 32 | 29 | 4 | 37 | 21 | 23 | 2 | 33 |
| Ruta4: | 14 | 26 | 15 | 25 | 44 | 8 | 24 | 49 | 7 | | |
| Ruta5: | 48 | 19 | 5 | 42 | 41 | 20 | 43 | 45 | 38 | 18 | |

vrpnc 4

F.O 1120,27

| | | | | | | | | | | | | |
|----------------|-----|-----|-----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| Ruta1: | 112 | 51 | 103 | 34 | 82 | 36 | 35 | 79 | 130 | 80 | 4 | 78 |
| | 77 | | | | | | | | | | | |
| Ruta2: | 70 | 123 | 21 | 67 | 104 | 72 | 66 | 137 | 136 | 121 | 10 | 52 |
| | 2 | 133 | | | | | | | | | | |
| Ruta3: | 28 | 128 | 102 | 71 | 31 | 129 | 132 | 33 | 91 | 64 | 127 | 109 |
| | 11 | 32 | | | | | | | | | | |
| Ruta4: | 147 | 89 | 149 | 63 | 108 | 12 | 65 | 50 | 144 | 37 | 20 | 124 |
| | 8 | 107 | 53 | | | | | | | | | |
| Ruta5: | 61 | 84 | 115 | 9 | 126 | 46 | 47 | 125 | 48 | 49 | 83 | 19 |
| Ruta6: | 90 | 148 | 119 | 6 | 85 | 18 | 114 | 87 | 62 | 100 | 7 | |
| Ruta7: | 97 | 105 | 94 | 86 | 17 | 142 | 92 | 99 | 60 | 113 | | |
| Ruta8: | 14 | 118 | 98 | 43 | 143 | 15 | 39 | 141 | 45 | 120 | 101 | 38 |
| | 93 | 96 | 95 | | | | | | | | | |
| Ruta9: | 138 | 88 | 145 | 3 | 116 | 58 | 44 | 16 | 146 | 42 | 23 | 134 |
| | 74 | 41 | 59 | 54 | | | | | | | | |
| Ruta10: | 106 | 22 | 73 | 75 | 76 | 57 | 24 | 68 | 40 | 140 | 26 | 111 |
| Ruta11: | 27 | 150 | 5 | 56 | 131 | 55 | 135 | 25 | 30 | 81 | 151 | 110 |
| | 13 | 139 | | | | | | | | | | |
| Ruta12: | 69 | 122 | 117 | 29 | | | | | | | | |

vrpnc 11

F.O 1285,12

| | | | | | | | | | | | | |
|---------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|-----|
| Ruta1: | 103 | 102 | 100 | 101 | 74 | 77 | 78 | 80 | 81 | 57 | 64 | 61 |
| | 59 | 56 | 54 | 69 | 104 | 105 | | | | | | |
| Ruta2: | 89 | 106 | 107 | 108 | 68 | 72 | 75 | 79 | 73 | 76 | 71 | 70 |
| | 121 | 120 | 82 | 83 | | | | | | | | |
| Ruta3: | 112 | 85 | 114 | 6 | 7 | 8 | 10 | 15 | 16 | 12 | 11 | 5 |
| | 4 | 2 | 3 | 84 | 118 | 113 | | | | | | |
| Ruta4: | 87 | 86 | 14 | 9 | 13 | 23 | 25 | 34 | 28 | 20 | 17 | 109 |
| | 119 | 19 | 90 | 88 | | | | | | | | |
| Ruta5: | 94 | 95 | 110 | 22 | 21 | 24 | 27 | 29 | 33 | 36 | 30 | 37 |
| | 35 | 32 | 31 | 26 | 18 | 115 | 91 | 92 | 93 | | | |
| Ruta6: | 96 | 97 | 98 | 116 | 38 | 42 | 45 | 48 | 47 | 50 | 51 | 52 |
| | 49 | 43 | 40 | 39 | | | | | | | | |
| Ruta7: | 99 | 53 | 55 | 58 | 62 | 63 | 65 | 67 | 66 | 60 | 46 | 44 |
| | 41 | 111 | 117 | | | | | | | | | |

vrpnc 12

F.O 1011,52

| | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-----|----|----|-----|----|----|----|
| Ruta1: | 11 | 9 | 12 | 13 | 15 | 17 | 20 | 16 | 14 | 24 | | |
| Ruta2: | 21 | 23 | 25 | 26 | 28 | 30 | 31 | 19 | 18 | 29 | 27 | 22 |
| Ruta3: | 33 | 34 | 32 | 36 | 38 | 39 | 40 | 37 | 35 | | | |
| Ruta4: | 53 | 51 | 52 | 49 | 46 | 45 | 60 | 61 | 56 | 41 | 42 | 43 |
| | 47 | 50 | 48 | 44 | | | | | | | | |
| Ruta5: | 67 | 65 | 55 | 54 | 57 | 59 | 58 | 69 | 70 | | | |
| Ruta6: | 68 | 66 | 63 | 62 | 73 | 81 | 74 | 78 | 80 | 75 | 64 | |
| Ruta7: | 82 | 79 | 77 | 71 | 72 | 83 | 84 | 87 | 88 | 90 | 92 | 91 |
| Ruta8: | 89 | 86 | 85 | 93 | 94 | 95 | 96 | 99 | | | | |
| Ruta9: | 6 | 4 | 8 | 7 | 5 | 101 | 98 | 97 | 100 | 3 | 2 | 76 |
| Ruta10: | 10 | | | | | | | | | | | |

4.5 Resultados en las Instancias de Christofides y Eilon (1969)

Un total de las 6 instancias del grupo propuesto por Christofides y Eilon en 1969 fueron documentadas para éste trabajo. Dos de tamaño pequeño, dos de tamaño mediano y finalmente fueron trabajadas las dos instancias de tamaño grande presentes.

La codificación de éste grupo de instancias se presenta también de la siguiente manera; **X-NYY-KZ**. Dónde, **X** es igual a "E" que indica el autor de la instancia, para éste caso Eilon y Christofides. **YY** es el número de nodos del sistema; es decir, el número de clientes + 1. Finalmente **Z** es el número de vehículos a utilizar.

La mayor parte de las instancias de éste grupo de estudio, cuentan con su valor óptimo global, representados en la tabla presentada a continuación con un (*). Dichos valores fueron encontrados a través de la implementación de los algoritmos de los tres tours óptimos y *branch-and-bound* (Ver tabla número 1).

Tabla 8. Resultados Benchmark de Christofides y Eilon

| Nombre | OPT (*) | BKS.FO | # Iteraciones | Tipo.Medida | Promedio | Min | GAP.Min | GAP.Promedio |
|------------|---------|--------|---------------|---------------|----------|---------|---------|--------------|
| E-N101-K14 | | 1071 | 3000 | F.O | 1204,34 | 1203,01 | 12% | 12% |
| E-N101-K14 | | 1071 | 3000 | Tiempo | 311,46 | 303,45 | | |
| E-N101-K8 | | 815 | 3000 | F.O | 888,56 | 880,56 | 8% | 9% |
| E-N101-K8 | | 815 | 3000 | Tiempo | 205,43 | 203,06 | | |
| E-N76-K10 | * | 830 | 2000 | F.O | 871,06 | 870,15 | 5% | 5% |
| E-N76-K10 | * | 830 | 2000 | Tiempo | 112,65 | 109,40 | | |
| E-N51-K5 | * | 521 | 2000 | F.O | 571,41 | 566,36 | 9% | 10% |
| E-N51-K5 | * | 521 | 2000 | Tiempo | 56,78 | 55,17 | | |
| E-N33-K4 | * | 835 | 2000 | F.O | 875,59 | 872,16 | 4% | 5% |
| E-N33-K4 | * | 835 | 2000 | Tiempo | 28,14 | 25,43 | | |
| E-N23-K3 | * | 569 | 3000 | F.O | 647,55 | 623,70 | 10% | 14% |
| E-N23-K3 | * | 569 | 3000 | Tiempo | 18,80 | 15,75 | | |

4.5.1 Solución Final de los Óptimos Locales

A continuación se presentan las rutas encontradas con menor función objetivo, por cada camión para cada instancia trabajada.

E-N101-K14

F.O 1203,01

| | | | | | | | | | | | | |
|----------------|----|----|-----|----|----|-----|----|----|----|----|--|--|
| Ruta1: | 29 | 78 | 69 | 30 | 25 | 81 | 13 | | | | | |
| Ruta2: | 77 | 4 | 80 | 79 | 35 | 36 | 82 | 34 | | | | |
| Ruta3: | 2 | 52 | 67 | 66 | 72 | 10 | 51 | | | | | |
| Ruta4: | 28 | 70 | 71 | 31 | 21 | 33 | 91 | 64 | 11 | | | |
| Ruta5: | 32 | 89 | 63 | 12 | 65 | 20 | 8 | | | | | |
| Ruta6: | 53 | 49 | 48 | 50 | 37 | 47 | | | | | | |
| Ruta7: | 90 | 61 | 84 | 85 | 62 | 18 | 46 | 9 | 83 | 19 | | |
| Ruta8: | 7 | 97 | 100 | 17 | 87 | 6 | | | | | | |
| Ruta9: | 60 | 92 | 45 | 86 | 94 | | | | | | | |
| Ruta10: | 95 | 96 | 93 | 99 | 39 | 101 | 38 | 98 | | | | |
| Ruta11: | 14 | 88 | 43 | 15 | 44 | 16 | 42 | 58 | 3 | | | |
| Ruta12: | 59 | 74 | 75 | 23 | 76 | 22 | 41 | 54 | | | | |
| Ruta13: | 5 | 68 | 24 | 57 | 73 | | | | | | | |
| Ruta14: | 40 | 26 | 56 | 55 | 27 | | | | | | | |

E-N101-K8

F.O 880,56

| | | | | | | | | | | | | |
|---------------|----|----|-----|----|----|----|-----|----|----|----|----|----|
| Ruta1: | 28 | 70 | 71 | 11 | 33 | 91 | 64 | 65 | 12 | 20 | 63 | 8 |
| | 89 | 32 | | | | | | | | | | |
| Ruta2: | 90 | 61 | 85 | 18 | 46 | 47 | 37 | 50 | 48 | 49 | 83 | 9 |
| | 84 | 19 | 53 | | | | | | | | | |
| Ruta3: | 7 | 97 | 100 | 94 | 86 | 92 | 45 | 87 | 17 | 62 | 6 | |
| Ruta4: | 95 | 96 | 60 | 93 | 99 | 38 | 101 | 15 | 39 | 44 | 43 | 98 |
| | 14 | | | | | | | | | | | |
| Ruta5: | 54 | 59 | 41 | 22 | 74 | 73 | 57 | 76 | 75 | 23 | 42 | 16 |
| | 58 | 3 | 88 | | | | | | | | | |
| Ruta6: | 13 | 81 | 30 | 25 | 56 | 26 | 68 | 24 | 40 | 5 | 55 | 27 |
| | | | | | | | | | | | | |
| Ruta7: | 29 | 77 | 78 | 69 | 4 | 80 | 79 | 35 | 36 | 82 | 34 | 51 |
| | | | | | | | | | | | | |
| Ruta8: | 31 | 21 | 67 | 66 | 72 | 10 | 52 | 2 | | | | |
| | | | | | | | | | | | | |

E-n76-k10

F.O 870,15

| | | | | | | | | | | | |
|---------|----|----|----|----|----|----|----|----|----|----|--|
| Ruta1: | 8 | 54 | 12 | 60 | 15 | 36 | | | | | |
| Ruta2: | 27 | 59 | 39 | 67 | 66 | 11 | 73 | | | | |
| Ruta3: | 41 | 10 | 26 | 56 | 32 | 40 | 13 | | | | |
| Ruta4: | 18 | 45 | 33 | 51 | 19 | 25 | 4 | | | | |
| Ruta5: | 7 | 34 | 64 | 42 | 57 | 24 | 50 | 17 | 52 | | |
| Ruta6: | 3 | 63 | 23 | 65 | 43 | 44 | 2 | 74 | | | |
| Ruta7: | 69 | 29 | 62 | 22 | 75 | 31 | 76 | | | | |
| Ruta8: | 49 | 48 | 37 | 70 | 72 | 61 | 71 | 38 | 6 | 30 | |
| Ruta9: | 5 | 46 | 16 | 21 | 58 | 55 | 14 | 28 | | | |
| Ruta10: | 68 | 47 | 9 | 20 | 53 | 35 | | | | | |

E-N51-K5

F.O 566,36

| | | | | | | | | | | | |
|--------|----|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 13 | 16 | 46 | 34 | 40 | 31 | 11 | 50 | 6 | 47 | |
| Ruta2: | 39 | 10 | 51 | 35 | 22 | 30 | 36 | 3 | 17 | 12 | |
| Ruta3: | 33 | 2 | 23 | 21 | 37 | 4 | 29 | 32 | 27 | 9 | 28 |
| Ruta4: | 14 | 26 | 15 | 25 | 44 | 8 | 24 | 49 | 7 | | |
| Ruta5: | 48 | 18 | 38 | 45 | 43 | 20 | 41 | 42 | 5 | 19 | |

E-N33-K4

F.O 872,16

| | | | | | | | | | | | | |
|--------|----|----|----|----|----|----|----|----|----|----|----|----|
| Ruta1: | 32 | 28 | 17 | 29 | 30 | 31 | | | | | | |
| Ruta2: | 15 | 16 | 18 | 26 | 27 | | | | | | | |
| Ruta3: | 4 | 3 | 13 | 12 | 22 | 21 | 24 | 25 | 23 | 20 | 19 | 14 |
| | | 2 | | | | | | | | | | |
| Ruta4: | 5 | 8 | 9 | 10 | 11 | 33 | 7 | 6 | | | | |

E-N23-K3

F.O 623,7

| | | | | | | | | | | | | |
|--------|----|----|----|----|----|----|----|----|---|---|----|----|
| Ruta1: | 13 | 9 | 5 | 6 | 14 | 12 | 7 | 2 | 3 | 4 | 17 | 16 |
| | 15 | 18 | 23 | 21 | 20 | 19 | | | | | | |
| Ruta2: | 22 | 8 | 10 | 23 | 18 | 15 | 16 | 17 | 4 | 3 | 2 | 7 |
| | 12 | 14 | 6 | 5 | 22 | 13 | | | | | | |
| Ruta3: | 11 | | | | | | | | | | | |

SECCIÓN 5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Se desarrolló una metodología de solución de dos fases para el SCVRP. Basada en la implementación de la heurística constructiva de barrido (Sweep) para la generación de los Clusters, y el AGCB para dar solución a la ruta óptima de cada camión dentro del mismo. Tomando como medida de desempeño, la minimización de la distancia total recorrida por el conjunto de vehículos.
- Se reporta la ventaja que tiene la metodología propuesta para dar solución aproximada a las instancias seleccionadas de Christofides y Eilon, con GAPs para las instancias pequeñas entre 4% - 10%, y para las grandes entre 5% - 12% ambas en el mejor de los casos. Además, se obtuvieron buenos resultados al trabajar con las instancias de CMT, vrpnc 1 y vrpnc 4, ambas con un GAP menor al 10%.
- Para el caso del benchmark realizado a los casos de prueba propuestos por Augerat et al. Se observa la necesidad de realizar movimientos inter-rutas e intra-rutas para poder escapar de óptimos locales y poder explorar aún más el espacio de soluciones. Ya que para instancias de tamaños similares se tienen GAPs muy distantes, como es el caso de las instancias A-N48-K7 (4% GAP) y B-N44-K7 (18% GAP). Caso similar ocurre con las tres instancias propuestas por Fisher, donde para los tres casos, la diferencia promedio con las BKS está por encima del 20%.
- Se valida la necesidad de encontrar y desarrollar métodos aproximados que obtengan buenas soluciones en un tiempo de cómputo admisible, dada la dificultad de utilizar los diversos métodos existentes exactos de solución. Ya que el VRP se caracteriza como un problema del tipo NP-Hard, no sólo con el aumento del tamaño de las instancias, sino también con la topología de las mismas.
- Se sugiere utilizar técnicas tanto heurísticas constructivas como aleatorias controladas al momento de generar los individuos pertenecientes a la población inicial; ya que esto ayuda a disminuir el número de iteraciones locales necesarias para dar solución a cada cluster a través del AGCB; dado que hay mejores individuos desde el inicio del algoritmo.

5.2 Recomendaciones y Trabajos Futuros

- Desarrollar la metodología de solución en el lenguaje de programación C++, con el fin de lograr mayor eficiencia en los tiempos de ejecución.
- Incluir estrategias de búsqueda local y de vecindario, para realizar movimientos intra-ruta e inter-ruta, esto con el fin de explorar mejor el espacio de soluciones y no quedar atrapado en óptimos locales de forma prematura.
- Evaluar un mayor componente aleatorio al momento de generar los Clusters y los diferentes arranques dados por las variables X y Y, para cada corrida de la instancia. Luego, de esta forma, se evita generar siempre los mismos Clusters, que el único elemento diferenciador que tendrían para mejorar la función objetivo, sería la solución del TSP en cada cluster a través del AGCB. (CLUSTERS ALEATORIOS)
- Utilizar diversas herramientas y técnicas disponibles que pueden ser utilizadas para potencializar la eficiencia de los algoritmos genéticos, como es el caso de la recombinación parametrizada uniforme y la variante de los AG propuesta por Chu y Beasley, entre otras.

SECCIÓN 6. BIBLIOGRAFÍA

ALONSO, F., ALVAREZ, M., BEASLEY, J., 2007. *A tabu search algorithm for the periodic vehicle routing problem with multiple vehicle trips and accessibility restrictions*. Journal of the Operational Research Society. Volume 59, pages 963-976.

ANGEL, J. C., SOLER, D., HERVAS, A. 2002. *The capacitated general routing problem on mixed graphs*. Revista investigación operacional, volume 23, No. 1.

AUGERAT, P., BELENGUER, J.M., BENAVENT, E., CORBERAN, A., NADDEF, D., RINALDI, G., 1998. *Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem*. Research Report 949-M, Université Joseph Fourier, Grenoble, France.

BELENGUER, J., MARTINEZ, M., MOTA, E., 2000. *A lower bound for the split-delivery vehicle routing problem*. Operations Research: A journal of the institute for operations research and the management sciences. Volume 48, issue 5, pages 801-810.

BENT, R., HENTENRYCK, P., 2004. *A two-stage hybrid local search for the vehicle routing problem with time windows*. Transportation Science: A journal of the institute for operations research and the management sciences. Volume 38, issue 4, pages 515-530.

BERGER, J., BARKAOUI, M. 2003. *A hybrid genetic algorithm for the capacitated vehicle routing problem*. Genetic and evolutionary computation – GECCO, volume 1, pages 646-656.

BERTSIMAS, D., RYZIN, G., 1991. *A stochastic and dynamic vehicle routing problem in the Euclidean plane*. Operations Research: A journal of the institute for operations research and the management sciences. Volume 39, issue 4, pages 601-615.

BIANCHESSI, N., RIGHINI, G., 2007. *Heuristics algorithms for the vehicle routing problem with simultaneous pick-up and delivery*. Computers & Operations Research. Volume 34, issue 2, pages 578-594.

BLASUM, U., HOCHSTÄTLER, W., 2000. *Application of the Branch and Cut Method to the Vehicle Routing Problem*. Zentrum für Angewandte Informatik Köln Technical Report. Zpr2000-386.

BRANDÃO, J., 2006. *A new tabu search algorithm for the vehicle routing problem with backhauls*. European Journal of Operational Research. Volume 173, issue 2, pages 540-555.

CHEN, A-L., YANG, G-K., WU, Z-M. 2006. *Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem*. Journal of Zhejiang University SCIENCE A, volume 7, issue 4, pages 607-614.

CHEN, S., GOLDEN, B., WASIL, E., 2007. *The split-delivery vehicle routing problem: Applications, algorithms, test problems, and computational results*. Networks: An international journal. Volume 49, issue 4, pages 318-329.

- CHOI, E., TCHA, D-W., 2007. *A column generation approach to the heterogeneous fleet vehicle routing problem*. Computers & Operations Research. Volume 34, issue 7, pages 2080-2095.
- CHRISTOFIDES, N., EILON, S., 1969. *An Algorithm for the Vehicle Dispatching Problem*. Operational Research Quarterly. Volume 20, pages 309-318.
- CHRISTOFIDES, N., MINGOZZI, A., TOTH, P., 1979. *The Vehicle Routing Problem*. In: Combinatorial Optimization (CHRISTOFIDES, N., MINGOZZI, A., TOTH, P. and SANDI, C., editors). Chichester: John Wiley & Sons Ltd. Pages 315-338.
- CHU, P. C., BEASLEY, J. E., 1997. *A Genetic Algorithm for the Generalized Assignment Problem*. Computers & Operations Research. Volume 24, issue 1, pages 17-23.
- CORDEAU, J. F., GENDREAU, M., HERTZ, A., LAPORTE, G., SORMANY, J. S., 2004. *New heuristics for the vehicle routing problem*. Les cahiers du GERAD.
- CORNUEJOLS, Gerard., HARCHE, Farid. 1993. *Polyhedral study of the capacitated vehicle routing problem*. Mathematical Programming, volume 60, issue 1-3, pages 21-52.
- CREVIER, B., CORDEAU, J-F., LAPORTE, G., 2007. *The multi-depot vehicle routing problem with inter-depot routes*. European Journal of Operational Research. Volume 176, issue 2, pages 756-773.
- DAZA, J. M., MONTOYA, J.R., NARDUCCI, F. 2009. *Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases*. Revista EIA, No. 12, páginas 23-38.
- DETHLOFF, J., 2001. *Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up*. OR-Spektrum. Volume 23, issue 1, pages 79-96.
- DROR, M., TRUDEAU, P., 1986. *Stochastic vehicle routing with modified savings algorithm*. European Journal of Operational Research. Volume 23, issue 2, pages 228-235.
- DRUMMOND, L., OCHI, L., VIANNA D., 2001. *An asynchronous parallel metaheuristic for the period vehicle routing problem*. Future Generation Computer Systems. Volume 17, issue 4, pages 379-386.
- DUARTE, Abraham. 2007. *Metaheurísticas*. Librería-Editorial Dykinson. Volumen 22 de Ciencias Experimentales y Tecnología.
- FISHER, M. 1994. *Optimal Solution of Vehicle Routing Problems Using Minimum k-Trees*. Operations Research. Volume 42, no. 4, pages 626-642.
- FRANCIS, P., SMILOWITZ, K., 2006. *Modeling techniques for periodic vehicle routing problems*. Transportation Research Part B: Methodological. Volume 40, issue 10, pages 872-884.

FUKASAWA, R., POGGI DE ARAGAO, M., REIS, M., UCHOA, E., 2003. *Robust Branch-and-Cut-and-Price for the Vehicle Routing Problem*. Relatorios de Pesquisa em Engenharia de Producao. RPEP Vol. 3 No. 8.

GALLEGO, R, Ramón., ESCOBAR, Z, Antonio. *Statical Planning of Colombia's Transmission Systems Using Genetics Algorithm*. 16th Internacional Conference on CAD/CAM Robotic & Factories of the Future, Trinidad y Tobago, Junio 2000.

GALLEGO, R, Ramón., ESCOBAR, Z, Antonio., TORO, O, Eliana., 2008. *Técnicas metaheurísticas de optimización*. Universidad Tecnológica de Pereira. Textos Universitarios, ISBN: 978-958-722-007-0, Ed. 2.

GALLEGO R, Ramón., ROMERO, L, Rubén., ESCOBAR, Z, Antonio. "Algoritmos Genéticos", texto guía en Maestría en Ingeniería Eléctrica. Universidad Tecnológica de Pereira.

GENDREAU, M., LANGEVIN, A., FRAYRET, J, M., 2013. *Notes du cours "Réseaux Logistiques", séance 6 : Planification du transport par camions*. École Polytechnique de Montréal.

GENDREAU, M., LAPORTE, G., MUSARAGANYI, C., TAILLARD, E., 1999. *A tabu search heuristic for the heterogeneous fleet vehicle routing problem*. Computers & Operations Research. Volume 26, issue 12, pages 1153-1173.

GENDREAU, M., LAPORTE, G., POTVIN, J-Y., 2002. *Metaheuristics for the capacitated VRP*. The vehicle routing problem, volume 9 of SIAM monographs on discrete mathematics and applications, chapter 6.

GENDREAU, M., LAPORTE, G., SÉGUIN, R., 1996. *Stochastic vehicle routing*. European Journal of Operational Research. Volume 88, issue 1, pages 3-12.

GENDREAU, M., VIDAL, T., CRAINIC, T.G., PRINS, C., 2012. *Heuristics for multi-attribute vehicle routing problems: A survey and synthesis*. CIRRELT.

GILLET, B. E., MILLER, L. R., 1974. *A Heuristic Algorithm for the Vehicle Dispatch Problem*. Operations Research. Volume 22, pages 340-349.

GOLDEN, B., LI, F., WASIL, E., 2007. *A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem*. Computers & Operations Research. Volume 34, issue 9, pages 2734-2742.

GONZÁLEZ, G., GONZÁLEZ F., 2006. *Metaheurísticas aplicadas al ruteo de vehículos. Un caso de estudio. Parte 1: Formulación del problema*. Bogotá: Universidad Nacional de Colombia. Ingeniería e Investigación. Vol. 26, número 003, pp. 149-156.

HEMMELMAYR, V., DOERNER, K., HARTL, R., 2009. *A variable neighborhood search heuristic for periodic routing problems*. European Journal of Operational Research. Volume 195, issue 3, pages 791-802.

HERTZ, A., ARCHETTI, C., SPERANZA, M., 2006. *A tabu search algorithm for the Split-delivery vehicle routing problem*. *Transportation Science: A journal of the institute for operations research and the management sciences*. Volume 40, issue 1, pages 64-73.

HO, W., HO, G., LAU, H., 2008. *A hybrid genetic algorithm for the multi-depot vehicle routing problem*. *Engineering Applications of Artificial Intelligence*. Volume 21, issue 4, pages 548-557.

JANSEN, Klaus. 1993. *Bounds for the general capacitated routing problem*. *Networks*, volume 23, issue 3, pages 165-173.

JIN, M., LIU, K., BOWDEN, R., 2007. *A two-stage algorithm with valid inequalities for the split-delivery vehicle routing problem*. *International Journal of Production Economics*. Volume 105, issue 1, pages 228-242.

KACHITVICHYANUKUL, V., AL, T-J., 2009. *A particle swarm optimization for the vehicle routing problem with simultaneous pick-up and delivery*. *Computers & Operations Research*. Volume 36, issue 5, pages 1693-1702.

KALLEHAUGE, B., LARSEN, J., MADSEN, O., SOLOMON, M., 2005. *Vehicle routing problem with time windows*. *Column Generation*. Book, pages 67-98.

LAPORTE, G., LOUVEAUX, F., HAMME, L., 2002. *An integer L-Shaped algorithm for the capacitated vehicle routing problem with stochastic demand*. *Operations Research: A journal of the institute for operations research and the management sciences*. Volume 50, issue 3, pages 415-423.

LAPORTE, G., NOBERT, Y., 1987. *Exact algorithms for the vehicle routing problem*. *Annals of Discrete Mathematics*, 31:147-184.

MARCELO, S., TOTH, P., 1990. *Knapsack Problems: Algorithms and Computer Implementations*. Wiley, Chichester, UK.

MARTEL, Alain., 2003. *Le pilotage des flux*. CENTOR, Université Laval. DF-3.4.1

MINGOZZI, A., GIORGI, S., BALDACCI, R., 1999. *An exact method for the vehicle routing problem with backhauls*. *Transportation Science: A journal of the institute for operations research and the management sciences*. Volume 33, issue 3, pages 315-329.

MONTANÉ, F., GALVAO, D., 2006. *A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service*. *Computers & Operations Research*. Volume 33, issue 3, pages 595-619.

NEO Research Group. <http://neo.lcc.uma.es/vrp/vrp-instances/>

OSMAN, I., WASSAN, N., 2002. *A reactive tabu search meta-heuristic for the vehicle routing problem with back-hauls*. *Journal of Scheduling*. Volume 5, issue 4, pages 263-285.

PAPADIMITRIOU, Christos, H., (1982). *Combinatorial Optimization*. Prentice Hall.

PRINS, Christian. 2004. *A simple and effective evolutionary algorithm for the vehicle routing problem*. Computers & Operations Research, volume 31, issue 12, pages 1985-2002.

RALPHS, T.K., KOPMAN, L., PULLEYBLANK, W.R., TROTTER JR, L.E., 2001. *On the Capacitated Vehicle Routing Problem*. Mathematical Programming Series.B 94, 343.

RENAUD, J., LAPORTE, G., BOCTOR, F., 1996. *A tabu search heuristics for the multi-depot vehicle routing problem*. Computers& Operations Research. Volume 23, issue 3, pages 229-235.

RESENDE, Mauricio. 2011. *BiasedRandom-Key Genetic Algorithms With Applications In Telecommunications*. AT&T Labs Research Technical Report.Pages 1-6.

SHAW, P., 1998. *Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems*. Memorias de "Proceedings of the Fourth International Conference on Principles and Practice of Constraint Programming". pp. 417-431.

SOTO, M, José, A., (2011). *Fundamentos teóricos de simulación discreta*. Texto guía Maestría en investigación operativa y estadística. Facultad de ingeniería industrial. Universidad Tecnológica de Pereira.

SPEARS, W. M., DE JONG, K. A. 1991. *On The Virtues of Parameterized Uniform Crossover*.Proceedings of the Fourth International Conference on Genetic Algorithms, eds. R. Belew and L. Booker, San Mateo, CA: Morgan Kaufmann, 230-236.

TAILLARD, E., BADEAU, P., GENDREAU, M., GUERTIN, F., POTVIN, J-Y., 1997. *A tabu search heuristic for the vehicle routing problem with soft time windows*.Transportation Science: A journal of the institute for operations research and the management sciences. Volume 31, issue 2, pages 170-186.

TANIGUCHI, E., THOMPSON, R., 1999. *Modeling city logistics*. City Logistics 5, pp. 3–37.

TANIGUCHI, E., THOMPSON, R., 2002. *Modeling city logistics*.Transportation Research Record: Journal of the transportation research board, volume 1790, pp. 45-51.

TARANTILIS, C., KIRANOUDIS, C., VASSILIADIS, V., 2004. *A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem*. European Journal of Operational Research. Volume 152, issue 1, pages 148-158.

TOTH, P., TRAMONTANI, A. 2008. *An integer linear programming local search for capacitated vehicle routing problems*.The vehicle routing problem: Latest advances and new challenges, volume 2, pages 275-295.

TOTH, P., VIGO, D., 1997. *An exact algorithm for the vehicle routing problem with backhauls*.Transportation Science: A journal of the institute for operations research and the management sciences. Volume 31, issue 4, pages 372-385.

TOTH, P., VIGO, D., 2000. *An Overview of Vehicle Routing Problems, Monographs on Discrete Mathematics and Applications. In: The Vehicle Routing Problem.* SIAM, 2000, pp. 1-26.

VENKATESAN, S.R., LOGENDRAN, D., CHANDRAMOHAN, D. 2011. *Optimization of capacitated vehicle routing problem using PSO.* International Journal of Engineering Science and Technology, volume 3, number 10, pages 7469-7477.

WU, T-H., LOW, C., BAI, J-W., 2002. *Heuristics solutions to multi-depot location-routing problems.* Computers & Operations Research. Volume 29, issue 10, pages 1393-1415.

XIAO, J., LU, B., 2012. *Vehicle routing problem with soft time windows.* Advances in Computer Science and Information Engineering. Volume 168, pages 317-322.