Asociación Argentina de Mecánica Computacional

Mecánica Computacional Vol XXIX, págs. 9367-9377 (artículo completo)
Eduardo Dvorkin, Marcela Goldschmit, Mario Storti (Eds.)
Buenos Aires, Argentina, 15-18 Noviembre 2010

DOS ENFOQUES PARA LA SOLUCIÓN DEL PROBLEMA DE RUTEO DE VEHICULOS (CVRP): APLICACIÓN A UN CASO REAL DE RECOLECCIÓN DE RESIDUOS

Alejandra Méndez, Silvia Simón, David Palumbo, Eliana Chiachera, Mercedes Carnero

Grupo de Optimización, Facultad de Ingeniería, Universidad Nacional de Rio Cuarto,Campus Universitario, 5800 Rio Cuarto, Argentina, gop@ing.unrc.edu.ar, http://www.unrc.edu.ar

Keywords: Optimización, CVRP, Recolección de residuos Infecciosos, Algoritmos meméticos.

Abstract. Los problemas de enrutamiento, tales como el Problema de Ruteo de Vehículos con restricciones de Capacidad, CVRP; han recibido especial atención en los últimos años, debido a su gran aplicabilidad en problemas de logística. El CVRP, puede ser formulado como un problema de programación lineal entera, en el cual el número de restricciones de conectividad de la solución y los requerimientos de capacidad vehicular, crecen exponencialmente con el número de nodos a atender. Esto implica que la solución exacta al problema planteado, sólo puede ser alcanzada para instancias pequeñas del mismo y por lo tanto su utilidad decae en la resolución de problemas de la vida real.
En este escenario se presentan al menos dos opciones posibles para abordar esta dificultad: la primera consiste en considerar un conjunto de restricciones alternativas que crezcan polinomialmente con el número de nodos y resolver el nuevo problema con paquetes de software comercial. La segunda opción es la utilización de metaheurísticas, tales como algoritmos meméticos, que suelen presentar como principales desventajas la no garantía de optimalidad y los altos tiempos de cómputo asociados. En este trabajo se propone comparar el desempeño de ambos enfoques, utilizando un software comercial para el primer procedimiento y, para el segundo enfoque, un Algoritmo Memético, basado en técnicas de Computación Evolutiva equipadas con diferentes y variados mecanismos de búsqueda local que aseguran la explotación intensiva de regiones promisorias del espacio de búsqueda. Se presenta la metodología y su desempeño para la optimización en diferentes problemas test presentes en la literatura, y finalmente su aplicación a un problema real de recolección de residuos infecciosos en la ciudad de Rio Cuarto

1 INTRODUCCION

El problema de ruteo de vehículos, VRP, abarca una amplia gama de problemas que fundamentalmente consisten en encontrar un conjunto de rutas que deben ser llevadas a cabo por una flota de vehículos con el objetivo de satisfacer los requerimientos de un grupo de clientes. Dentro de ellos se encuentra el problema de ruteo de vehículos con restricciones de capacidad, CVRP, en el cual, a partir de un único depósito, se debe visitar un conjunto de n clientes, ubicados en localizaciones conocidas, y que poseen determinadas demandas. Se dispone de un conjunto de K vehículos, de capacidad idéntica y conocida, Q. El conjunto de rutas es determinado de tal manera que se optimice algún criterio de desempeño, por ejemplo, la distancia total recorrida o el costo total de transporte, sin violar la restricción de capacidad.

Este problema ha recibido especial atención en los últimos años, debido a su gran aplicabilidad en problemas de logística y a su considerable dificultad de resolución, ya que pertenece a la clase de problemas NP- duros, (Garey y Johnson 1979; Toth y Vigo 2002). Esta última característica implica que los métodos exactos de solución se podrán aplicar solamente en los problemas de un tamaño pequeño o medio. Una recopilación de los métodos exactos utilizados hasta 2002 puede encontrarse en Thot y Vigo, 2002 y más recientemente en los trabajos de Fukasawa *et al*. (2004, 2006), Baldacci *et al*. (2004) y Lysgaard *et al.* (2004).

Sin embargo, dado que los problemas que se presentan en las aplicaciones reales poseen un tamaño tal que los métodos exactos se tornan impracticables, se han propuesto en la literatura diversas heurísticas para la solución del CVRP, basadas en enfoques de búsqueda local, algoritmos evolutivos, rococido simulado, colonias de hormigas, etc. Así, puede citarse por ejemplo, la denominada búsqueda local adaptiva, (Pisinger y Ropke, 2007) aplicada tanto a CVRP como a otras variantes de VRP, donde cierto número de algoritmos simples compiten para modificar la solución corriente. En cada iteración un algoritmo es elegido para destruir la solución corriente, mientras que otro es seleccionado para repararla. La nueva solución es aceptada si satisface algún criterio preestablecido.

También han sido reportados métodos que utilizan la clásica heurística constructiva de Clark yWright combinada con simulación de Monte Carlo para producir un conjunto de rutas alternativas en una instancia dada del CVRP, (Juan, et al, 2010).

Dentro de los enfoques evolutivos han sido utilizados algoritmos genéticos híbridos, (Wang y Lu, 2009), que incorporan un mecanismo de barrido para mejorar la capacidad de búsqueda del algoritmo a la vez que utiliza métodos de superficie de respuestas para optimizar las probabilidades de cruzamiento y mutación; algoritmos genéticos celulares, (Alba y Dorronsoro, 2006) en los cuales la población está estructurada en una topología particular, y además el algoritmo está dotado de mecanismos de búsqueda local.

Estrategias de cúmulo de partículas, con múltiples estructuras de aprendizaje social, han sido aplicadas utilizando dos tipos diferentes de representación y sus correspondientes decodificadores, (Ai y Kachitvichyanukul, 2009).

El objetivo del presente trabajo es comparar el desempeño de una heurística inspirada en el paradigma evolutivo e hibridizada con mecanismos de búsqueda local frente a la resolución exacta del CVRP con un algoritmo comercial, de modo tal de establecer la calidad de las soluciones obtenidas por el algoritmo diseñado, sus posibilidades de escalamiento y el esfuerzo computacional que insume cada método.

2 MODELADO DEL PROBLEMA: MÉTODOS DE RESOLUCIÓN EXACTOS

La utilización de métodos exactos para la resolución del CVRP, tales como los utilizados en el software comercial *General Algebraic Modeling System*, GAMS, involucra como primer paso contar con una formulación del mismo que permita su modelado en este entorno de trabajo. En la literatura se presentan diferentes formulaciones para el CVRP, que pueden ser agrupadas básicamente en tres tipos: *vehicule flow formulations* que utiliza variables binarias asociadas con cada enlace del grafo, para indicar si un vehículo atraviesa dicho enlace en la solución óptima, *commodity flow formulation* que utiliza variables enteras adicionales asociadas a los arcos y que representan el flujo de *commodities* dentro de los caminos atravesados por los vehículos. Finalmemte el CVRP puede ser formulado como un *set partitioning problem* que permite determinar una colección de circuitos de mínimo costo (Thot y Vigo, 2002).

A continuación se describen dos clases de formulaciones básicas para CVRP que corresponden al primer tipo, las cuales emplean variables binarias doblemente subindicadas, x_{ij}, que asumen el valor 1 si el enlace (i,j) es atravesado por un vehículo en la solución óptima y el valor 0 en caso contrario. El número de tales variables es del $\boldsymbol{O}(n^2)$, para un problema de ruteo que deba atender exactamente a n clientes.

Sea $G=(V,A)$ un grafo completamente conectado, en el cual $V=\{0,1,.....,n\}$ es el conjunto de vértices y A el conjunto de enlaces que conectan los vértices. Los vértices $i=1,\ldots,n$ corresponden a los clientes geográficamente dispersos, mientras que el nodo 0 está asociado al déposito.

El costo asociado al recorrido de un enlace cualquiera (i,j) viene dado por los elementos de una matriz $\mathbf{C}$, mientras que cada elemento q_i. del vector $\mathbf{q}\in\mathbb{R}^n$, representa la demanda asociada i-ésimo cliente.

El CVRP formulado como un problema de programación lineal entera (IP) puede entonces ser escrito como:

Formulación F1:

$$\min \sum_{i\in V}\sum_{j\in V} c_{ij}x_{ij} \tag{1}$$

Sujeto a:

$$\sum_{i\in V} x_{ij} = 1 \qquad \forall\, j\in V\setminus\{0\} \tag{2}$$

$$\sum_{j\in V} x_{ij} = 1 \qquad \forall\, i\in V\setminus\{0\} \tag{3}$$

$$\sum_{i\in V} x_{i0} = K \tag{4}$$

$$\sum_{j\in V} x_{0j} = K \tag{5}$$

$$\sum_{i\notin S}\sum_{j\in S} x_{ij} \ge r(S) \qquad \forall\, S\subseteq V\setminus\{0\}, S\neq\varnothing \tag{6}$$

$$x_{ij}\in\{0,1\} \qquad \forall\, i,j\in V \tag{7}$$

Donde K es el número de vehículos disponibles para realizar el recorrido y r(S) es el número mínimo de vehículos necesarios para servir a todos los vértices incluidos en S

Las restricciones 2 y 3 imponen que sólo un arco entra y deja cada vértice del grafo G, mientras que las ecuaciones 4 y 5 restringen a K el grado asociado al depósito.

El conjunto de restricciones agrupadas en la ecuación 6 establecen tanto restricciones de capacidad como de conectividad de la solución. El número de restricciones del tipo 6 es de $\boldsymbol{O}(2^n)$ ya que es necesario generar toda la familia de partes de un conjunto de n elementos. Dicho de otro modo, esta formulación establece un conjunto de restricciones que crece exponencialmente con el tamaño del problema y por lo tanto la modelización del mismo se torna impracticable aún para instancias de tamaño pequeño.

A modo de ejemplo, para el problema test denominado eil13, que considera un conjunto de 12 clientes a atender el número de restricciones de conectividad y capacidad asciende a 4095. Esta formulación fue inicialmente modelada en el entorno MatLab, logrando expresar a todo el conjunto de restricciones en forma matricial, de manera tal que problema IP queda descripto en su forma canónica y pudo fácilmente ser transcripto al lenguaje GAMS.

Sin embargo esta metodología no pudo ser aplicada en instancias de tamaño mayor, ya que para un problema de 24 nodos el número de restricciones superaba 16 millones, excediendo tanto la capacidad de MatLab, como el número de restricciones que puede manejar GAMS.

Una alternativa es formular el CVRP como un problema de programación lineal entera mixta, (MIP) que incluye variables auxiliares continuas para modelar las restricciones de conectividad y capacidad, (Kara y Bektas, 2003, Kara et al, 2004):

$$u_i + (Q - \overline{q}_i - q_i)x_{0i} - \overline{q}_i x_{i0} \le Q - \overline{q}_i \qquad i = 1, 2, ..., n \tag{8}$$

$$u_i + \overline{q}_i x_{0i} \ge q_i + \overline{q}_i \qquad i = 1, 2, ..., n \tag{9}$$

$$u_i - u_j + Qx_{ij} + (Q - q_i - q_j)x_{ji} \le Q - q_j \qquad i \ne j, \;\; j = 1, 2, ..., n \tag{10}$$

Donde: u_i es una variable auxiliar continua que representa la carga del vehículo después de visitar al nodo i y $\overline{q}_i = \min_{j, j \ne i} \{q_j\}$

Las restricciones 8 y 9 introducen restricciones de capacidad sobre las rutas, mientras que la ecuación 10 representa tanto restricciones de capacidad como de eliminación de subrutas, esto es recorridos que no comiencen y terminen en el depósito visitando sólo una vez cada nodo. La segunda formulación, F2, consiste entonces en resolver, conjuntamente, la función objetivo (1), sujeto a las restricciones (2) a (6), (8), (9) y (10).

3 HEURÍSTICAS PARA CVRP

La otra metodología utilizada fue una metaheurística inspirada en el paradigma evolutivo y donde se incorporaron aspectos distintivos de búsqueda local. A partir de una población inicial de potenciales soluciones (individuos), generada en forma aleatoria, las mismas son sometidas a un mecanismo de selección local por ranking para generar una población intermedia o conjunto de apareamiento. Dicho conjunto sufre un proceso de transformación a través del operador de cruzamiento para luego, con una probabilidad dada, realizar un procedimiento de búsqueda local sobre los individuos. Finalmente se reinserta la población intermedia para conformar la próxima generación.

En las siguientes subsecciones se describen los componentes fundamentales del algoritmo memético (AM) utilizado: la representación de los individuos, los operadores genéticos, el diseño de la función de aptitud y los mecanismos de búsqueda local incorporados, (Mendez, *et al*, 2008). Posteriormente se completa la implementación de la estrategia seleccionada.

3.1 Representación de los individuos

Los algoritmos genéticos para los problemas de ruteo usan una representación cuasi-directa de las soluciones, como secuencias de tareas, a través de un vector **a** de números enteros de tal manera que un elemento a_j indica el cliente que corresponde atender en el j-ésimo lugar.

Si no se utilizan separadores de rutas, este tipo de representación se combina con una técnica de separación óptima, a través de la cual se consigue para cada secuencia su mejor partición en rutas, es decir obtiene la forma en que se debe dividir el vector **a** para que la suma de los costos de las rutas resultantes sea mínimo.

Presentado de este modo, **a** no representa directamente una solución del CVRP, pero puede ser visualizada como un viaje gigante ignorando los limites de capacidad del vehículo. El procedimiento de separación óptima es aplicado sobre **a** para obtener una solución del CVRP al momento de evaluar el costo que tiene asociado.

3.2 Función de aptitud

La bondad de una solución potencial es evaluada a través de una función que devuelve el costo asociado al conjunto de rutas que genera el algoritmo de separación. En la representación utilizada no existen separadores que permitan la determinación de las diferentes rutas involucradas en una propuesta de solución. De este modo en la evaluación del costo será necesario determinar los límites de cada ruta. Para esto se utiliza un procedimiento de división de la secuencia (Lacomme, 2004) que asegura obtener el corte óptimo de la secuencia, es decir para una secuencia dada encuentra como se debe cortar para producir la división en rutas de mínimo costo, cumpliendo las restricciones de capacidad. Este procedimiento propuesto por Lacomme se basa en la heurística de Ulusoy (1985).

Una vez obtenido el conjunto de rutas el valor de la función de aptitud o *fitness, fa,* viene dado por:

$$fa = \sum_{t=1}^{K} r_t \tag{11}$$

Donde r_t representa el costo de una ruta t, K el número total de vehículos disponibles y los costos de atravesar los enlaces entre nodos para cada ruta son obviamente extraídos de la matriz **C**

$$r_t = c_{0,i_s} + \sum_{i_s=1}^{z\text{-}1} c_{i_s,i_{s+1}} + c_{i_z,0} \qquad i_s = 1,....,n \tag{12}$$

z: cantidad de clientes en la ruta t

3.3 Operador de cruzamiento

Para el tipo de cromosomas sin delimitadores de rutas se pueden experimentar los cruzamientos clásicos de los cromosomas que se utilizan para problemas de secuencia: X1 (Cruzamiento con un punto de corte), LOX (*Linear Order Crossover*), OX (*Order Crossover*).

Estos cruces se realizan para generar permutaciones válidas y para preservar hasta cierto punto las características transmitidas por los padres. Se implemento un cruzamiento LOX en este nivel.

3.4 Búsqueda Local versus Mutación

En optimización combinatoria, es conocido que el AG básico (Holland, 1975) con una simple mutación no puede competir con Simulated Annealing (SA) y Tabú Search (TS). El AG puede ser efectivo hibridizado con una búsqueda local, dando un AG híbrido (HGA), Algoritmo Memético (AM) o Genetic Local Search (GLS), Moscatto (1999). Con una dada probabilidad, cada individuo en un AM es sometido a una búsqueda local (BL). Algunos movimientos que conducen la búsqueda local en este tipo de problemas pueden encontrarse en Lacomme (2001) y otras posibles combinaciones pueden ser propuestas para acentuar la búsqueda.

La Búsqueda Local (BL) trabaja tanto sobre cromosomas como sobre soluciones; en este último caso será necesario aplicar el procedimiento de separación óptima para convertir un individuo del algoritmo en una solución del CVRP.

La BL ejecutada consiste en diversas fases que son implementadas en distintos esquemas dependiendo de la etapa del algoritmo en que es requerida. Las fases realizan movimientos que involucran nodos individuales (u), pares de nodos consecutivos (u,x) y bloques de rutas.

A continuación se exponen brevemente los movimientos aplicados, una descripción detallada de los mismos puede encontrarse en Mendez *et al*, 2008.

Fase 1: mover al cliente o tarea *u* después del cliente *v*, o antes si *v* es el primer cliente de su ruta.
Fase 2: mover los clientes adyacentes *(u,x)* después de las tareas *(v,y)* o antes si *v* es la primera tarea de su ruta.
Fase 3: Intercambio (*swap*) de los clientes *u* y *v*.
Fase 4: intercambio entre dos rutas con movimiento de bloques de tareas, en todas las modalidades se consideran dos viajes, y se intercambian el bloque de la ruta que queda después de *u* con el bloque que sigue a *v* en su ruta. Esta fase opera sobre soluciones a diferencia de las anteriores y ha sido implementado en dos modalidades:
Los clientes *u* y *v*, que corresponden a distintas rutas, permiten localizar dos bloques formados por todos los clientes siguientes a *u* y *v* hasta el final de las rutas correspondientes. Estos bloques serán intercambiados en sus posiciones, como un intercambio (*swap* de la fase 3) pero esta vez como bloques de clientes. De esta forma se produce el primer movimiento de esta fase.

Otra alternativa consiste en invertir el bloque al mismo tiempo que se intercambia, entendiendo que invertir en este caso significa cambiar el orden establecido. Esto produce una secuencia que resulta equivalente al bloque que le dio origen recorrido en el sentido opuesto, semejante a espejar un objeto.

4 EJEMPLO DE APLICACION

En esta sección se muestra un ejemplo de aplicación del CVRP para la determinación de un conjunto de rutas óptimas en el caso de la recolección de residuos infecciosos en la ciudad de Rio Cuarto.

En dicha ciudad están registradas como instituciones generadoras de residuos infecciosos aproximadamente 260 establecimientos. Estos presentan muy diferentes volúmenes de residuos producidos y por lo tanto las frecuencias de visitas requeridas por cada uno de ellos son variables.

En este trabajo se consideró un subgrupo de 58 instituciones, las más importantes desde el punto de vista de cantidad de residuos que generan, que deben ser atendidas diariamente y para las cuales se contó con los datos de localización geográfica, y demandas diarias promedio.

Se realizó un preprocesamiento de los datos de localización, para calcular las distancias mínimas de cada uno de los centros entre sí, utilizando un algoritmo de camino mínimo entre nodos. De este modo se pudo obtener una matriz de distancias, asimétrica, cuyos elementos conforman el vector de costos de la función objetivo, basando, por lo tanto, el criterio de optimización en la distancia total recorrida por cada vehículo en la ruta propuesta. Esto implica disminuir tanto el costo de combustible asociado como el desgaste de los camiones utilizados.

Los datos correspondientes a los diferentes generadores considerados se muestran en la tabla 1

Clientes	**Nº de nodo**	**Demanda**
Clínica Sta Cecilia	1	22
Hospital Central	2	178
Maternidad Kowald	3	26
Centro Municipal de Salud	4	6
Clínica Reg del Sud	5	88
Instituto de Neonatología	6	32
Instituto Médico	7	120
Instituto de Urología y U Cardiológica	8	62
Policlinico Privado SAn Lucas	9	74
Clínica de Ojos Privada	10	18
Clínica Philippe Pinel	11	8
Dispensario N° 1	12	8
Dispensario N° 2	13	8
Dispensario N° 3	14	12
Dispensario N° 4	15	12
Dispensario N° 5	16	6
Dispensario N° 6	17	4
Dispensario N° 7	18	6
Dispensario N° 8	19	8
Dispensario N° 9	20	10
Dispensario N° 10	21	6
Dispensario N° 11	22	4
Dispensario N° 12	23	4
Ni idea!! Dispensario N° 13	24	2
Instituto Oncohematológico	25	8
Unidad de Nefrologia y Diálisis	26	46
Centro Otorringologia y Fonoaudilogía	27	4
Servicio Penintenciario	28	6
CEMYD	29	10
Servicio Enfermería Sociedad Francesa	30	14
Laboratorio Ficco y Vettorazi	31	6
Hogar de Ancianos San José	32	106
Tomografía Computada	33	4
Centro Clínico de Hígado	34	20
Mater Vita	35	38
Laboratorio Bosch	36	8

Tabla 1: Datos para 58 generadores

Clientes	Nº de nodo	Demanda
Centro Privado de Traumatología y Artroscopía	37	18
Dr Jorge Alazino	38	8
Dr Justo Magnasco	39	8
Clínica del Niño	40	6
Odontologi Servicios Sociales Grassi	41	12
Dres Granero	42	4
Centro Medico de la Visión Privado	43	14
Peqeño Cottolengo Don Orione	44	20
Dr J C Daita	45	10
Centro Integral de Alergia	46	4
Clínica y Maternidad Suiza	47	18
Residencia Geriátrica Amanecer	48	10
Hogar Sol de Otoño	49	6
Diagnósticos SRL	50	4
Amor y Paz SRL	51	6
Centro Médico RC	52	8
Prestaciones Integrales Médicas	53	8
Dra Audisio Claudia	54	4
Centro de Diagnóstico Médico	55	6
HUMANE	56	4
Asilo San Carlos Borromeo	57	8
Vasquetto Industrial y Comercial	58	4

Tabla 1. Continuación: Datos para 58 generadores

5 EXPERIMENTOS Y RESULTADOS

El conjunto de problemas test seleccionados corresponden a diferentes instancias del CVRP y se encuentran disponibles en la siguiente dirección http://www.branchandcut.org/.

Todos ellos están disponibles con datos dados en forma explícita a través de la matriz de costo o distancias y el vector de demandas. Las características de cada una de las instancias resueltas se muestran en la tabla 2. El problema de recolección de residuos infecciosos aparece en la última fila de dicha tabla como rio59.

Problema	*n*	*K*	Demanda	Optimos de referencia
eil13	12	4	variable	247
gr17	16	3	uniforme	2685
gr21	20	3	uniforme	3704
gr24	23	4	uniforme	2053
fri26	25	3	uniforme	1353
bayg29	28	4	uniforme	2050
bays29	28	5	uniforme	2963
eil31	30	7	variable	379
swiss42	41	5	uniforme	1668
rio59	58	2	variable	----

Tabla 2: Datos de los problemas resueltos

Los valores de los parámetros utilizados en la heurística se describen en la tabla 3. Ambas metodologías de resolución fueron implementadas en una computadora basada en procesador Intel Core2duo.

Parámetros	**Valores**
Probabilidad de cruzamiento	0.7
Probabilidad de mutación	0.025
Porción de la población a ser modificada	0.5
Numero de individuos de la población	200-500
Probabilidad de aplicación búsqueda local	0.4 -0.6

Tabla 3: Valores de los parámetros correspondientes al AM

Las soluciones óptimas han sido alcanzadas en todos los problemas test, cuando se utilizó el algoritmo heurístico como método de resolución, con diferentes esfuerzos de cómputos. En la tabla 4 se muestran los tiempos, en segundos, relevados para cada caso. Dichos tiempos reportados son tiempos mínimos requeridos para lograr el óptimo.

Problema	**Algoritmo Memético**
eil13	1.15
gr17	12.6
gr21	27
gr24	55.8
fri26	94.2
bays29	430.2
eil31	656.6
swiss42	2364

Tabla 4: Tiempos de cómputo asociados a cada instancia utilizando el AM

En el caso de la metaheurística implementada los esfuerzos computacionales están conectados fuertemente con el tamaño del problema (cantidad de clientes a atender) y se traducen en la cantidad de iteraciones requeridas y en el tiempo insumido en la búsqueda local. Si bien el algoritmo es eficiente en los problemas de hasta 25 clientes, reflejado en buenos tiempos para alcanzar las soluciones optimas, muestra cierta dispersión en la calidad de las soluciones encontradas para los problemas de mayor dimensión.

La resolución con GAMS se realizó utilizando el módulo CPLEX 9.0.2. Si bien para los problemas más pequeños, las opciones por defecto fueron suficientes para obtener los óptimos de referencia, a medida que crecía el tamaño de la instancia a resolver, fue necesario realizar diferentes ajustes respecto a la tolerancia o gap, tiempo máximo permitido para la corrida, número de iteraciones máximo y selección del método de ramificación. Los resultados obtenidos, tiempos de cómputo y tolerancias fijadas para cada caso se muestran en la tabla 5.

Problema	Óptimo obtenido	Gap fijado(%)	Tiempos(seg)	Optimos de referencia
eil13	247	0.001	0.218	247
gr17	2685	0.001	155	2685
gr21	3704	0.001	1376	3704
gr24	2053	10	3962	2053
fri26	1353	5	2502	1353
bays29	3049	13	10540	2963
eil31	379	10	66	379
swiss42	1713	10	3244	1668

Tabla 5: Resultados asociados a los diferentes problemas utilizando GAMS

En el caso del problema de aplicación, rio59, el mejor valor encontrado de función objetivo, con el AM, fue de 71.04, en un tiempo de 7416 seg correspondiente a las siguientes rutas:

Ruta # 1: 22-20-12-53-33-27-48-15-19-21-49-13-32-57-11-24-44-16-2-17-38-14-45-18 23-1
Ruta # 2: 3-29-35-7-36-10-41-54-30-42-56-37-31-8-55-26-9-40-46-43-34-39-5-6-47-58 52-25-4-50-51-28

El valor obtenido con GAMS, fue ligeramente superior correspondiendo a 73.12 con un gap de 28.2 % en un tiempo de 3292 seg. El conjunto de rutas obtenido con GAMS es:

Ruta #1: 28-51-50-4-52-58-47-6-5-39-35-25-7-27-33-53-3-29-12-20-22
Ruta #2: 34-9-55-38-37-56-42-30-54-17-2-16-44-24-11-57-32-13-49-21-19-15-48-41-10-36-43-46-40-26-8-31-45-14-18-23-1

6 CONCLUSIONES

En este trabajo se comparó desempeño de un algoritmo memético frente a la resolución exacta del CVRP con un algoritmo comercial, de modo tal de establecer la calidad de las soluciones obtenidas por la heurística, sus posibilidades de escalamiento y el esfuerzo computacional asociado a cada método.

Para el conjunto de problemas seleccionados el comportamiento de la heurística ha sido satisfactorio si se comparan los tiempos de cómputo insumidos así como la calidad de las soluciones obtenidas respecto de aquellas logradas con el método exacto. Sin embargo, si se tiene en cuenta las posibilidades de escalamiento que ofrecen ambos enfoques, las técnicas heurísticas son más robustas, ya que el software comercial con el que se cuenta exhibe marcadas dificultades en la disponibilidad de memoria siendo necesario aumentar el gap con el consecuente deterioro de la calidad de la solución.

Por otra parte las posibilidades de abordar problemas de mayor envergadura con el algoritmo memético diseñado, con tiempos de procesamiento razonables y buenas soluciones, dependen de poder diseñar un muestreo apropiado del espacio de búsqueda que permita diversificar la población e implementar la búsqueda local sólo en unas pocas regiones promisorias.

REFERENCIAS

Ai, J; Kachitvichyanukul, V. Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem. *Computers & Industrial Engineering* 56 pp.380–387, 2009.

Alba, E.; Dorronsoro, B. Computing nine new best-so-far solutions for Capacitated VRP with a cellular Genetic Algorithm. *Information Processing Letters* 98 225–230, 2006.

Baldacci, R., Hadjiconstantinou, E., Mingozzi, A. An exact algorithm for the capacitated Vehicle routing problem based on a two-commodity network flow formulation. *Operations Research* 52, pp. 723–738, 2004.

Chung-Ho Wang, Jiu-Zhang Lu. A hybrid genetic algorithm that optimizes capacitated vehicle routing problems. *Expert Systems with Applications* 36; pp. 2921–2936, 2009.

Fukasawa R., Longo H., Lysgaard J., Poggi D. M., Reis M., Uchoa E. & Werneck R. F. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Math. Program. Ser.* 106, pp. 491–511, 2006.

Fukasawa R., Lysgaard J., Poggi D. M., Reis M., Uchoa E. & Werneck R. F. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Proceedings of the X IPCO, Lecture Notes in Computer Science* 3064, pp. 1–15, 2004.

Holland, J. *"Adaptation in Natural and Artificial Systems"*. University of Michigan Press, 1975.

Juan, A.; Faulin, J.; Ruiz, R.; Barrios, B.; Caballe,S. The SR-GCWS hybrid algorithm for solving the capacitated vehicle routing problem. *Applied Soft Computing* 10, pp.215–224, 2010.

Kara, İ., Bektaş, T. Integer Linear Programming Formulation of the Generalized Vehicle Routing Problem, EURO/INFORMS Joint International Meeting, July 06-10, Istanbul, Turkiye, 2003.

Lacomme, P.; Prins, C. y Ramdane-Cherif W. *"A genetic algoritm for the CARP and its extensions"*. In: E.J.W. Boers et al (eds.), Applications of evolutionary computing. Lecture Notes in Computer Science 2037, Springer, pp.473-483. 2001.

Lacomme, P.; Prins, C.; Ramdane-Cherif, W. Competitive Memetic Algorithms for Arc Routing Problems. *Annals of Operations Research*, Vol.131, pp. 159-185, 2004.

Lysgaard J, LetchfordAN, EgleseRW. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming, Series* A;100, pp. 423–45, 2004.

Mendez, A.; Palumbo, D.; Carnero, M.; Hernández, J. Algoritmos Meméticos Aplicados a la Resolución de un Problema de Ruteo de Vehículos Periódico. *Mecánica Computacional*, pp.2675-2685, Vol.XXVIII, 2009.

Moscato, P. *"Memetic Algorithms: a short introduction"*. In: D. Corne, M. Dorigo and F. Glover (eds.) New Ideas in optimization. McGraw-Hill 1999

Pisinger, D; Ropke. S. A general heuristic for vehicle routing problems. *Computers & Operations Research* 34; pp. 2403 – 2435, 2007.

Toth, P., & Vigo, D. *An overview of vehicle routing problems*. In P. Toth & D. Vigo (Eds.). Philadelphia: Siam, 2002.

Ulusoy, G. The Fleet size end Mixed Problem for Capacitated Arc Routing. *European Journal Operational Research*, 22, pp. 329-337, 1985.