

HEURISTICAS PARA LA RESOLUCIÓN DE UN PROBLEMA DE RUTEO DE VEHÍCULOS PERIÓDICO REAL

Alejandra Méndez¹, Marisa Pontin¹, Maria Ziletti¹ y Luis Chávez²

¹ Facultad de Ingeniería, Universidad Nacional de Río Cuarto,
Campus Universitario, (5800) Río Cuarto, Argentina
e-mail: {mpontin; amendez; mziletti} [@ing.unrc.edu.ar](mailto:ing.unrc.edu.ar)

² Facultad de Ciencias Exactas, Físico Químicas y Naturales, Universidad Nacional de Río Cuarto,
Campus Universitario, (5800) Río Cuarto, Argentina
e-mail: lchavez@dc.exa.unrc.edu.ar

Palabras Clave: Problemas de ruteo, Búsqueda tabú, Meta-heurísticas.

Resumen: *El problema de la recolección de residuos infecciosos puede modelarse como un Problema de Ruteo de Vehículos Periódico. Este es un problema multinivel que congrega dos problemas de optimización clásicos: un problema de planificación y uno de ruteo de vehículos. Un diseño óptimo consiste en asignar días de recolección a cada institución generadora y diseñar un conjunto de rutas para cada día de modo tal que el costo total de recolección sea mínimo sin exceder la capacidad o tiempo de viaje de los vehículos. Para la resolución de este problema de optimización combinatoria multinivel se propone una técnica híbrida que involucra dos heurísticas bien conocidas. Un Algoritmo Evolutivo es utilizado para obtener la planificación óptima mientras que la determinación del valor de aptitud de cada solución es realizada a través de un algoritmo de Búsqueda Tabu; ésta involucra la resolución de un problema de ruteo para cada planificación. Se muestran los resultados obtenidos por el esquema propuesto aplicado a la recolección de residuos patógenos en la Ciudad de Río Cuarto, Córdoba*

1 INTRODUCCION

El tratamiento y recolección de residuos infecciosos presenta algunas particularidades, dentro de los servicios de recolección de sólidos de una zona urbana. Una de las características a tener en cuenta es el aspecto económico y la otra es lo vinculado a la salud pública.

Por las razones expuestas existen reglamentaciones que regulan lo referido a residuos infecciosos. En la República Argentina la ley N° 24051 regula la recolección, transporte y procesamiento o disposición final de los residuos infecciosos.

Se entiende por residuos infecciosos, aquellos capaces de inducir infecciones, representando un riesgo potencial para la salud de la población. Así, este tipo de residuos generados en instituciones como hospitales y clínicas, laboratorios, veterinarias, casas de sepelios, etc. necesitan un tratamiento diferenciado.

En la ciudad de Río Cuarto, si bien algunos centros hospitalarios cuentan con incineradores para reducir sus residuos infecciosos, actualmente resultan de capacidad insuficiente; además existe un gran número de ellos que no disponen de medios adecuados para su tratamiento y es por esto que el Municipio debe afrontar la tarea.

En la ciudad el servicio de recolección de residuos infecciosos consiste en la recolección propiamente dicha y el traslado hasta un centro para su incineración. La recolección se realiza diariamente, atendiendo a un número de centros que varía con el transcurso de los días de la semana de acuerdo a las diferentes frecuencias de visitas que requieren cada uno de ellos.

La literatura se refiere a esta clase de problema como un *Periodic Vehicle Routing Problem* (PVRP)¹. El PVRP puede pensarse como una generalización de un *Vehicle Routing Problem* (VRP) convencional que intenta determinar un conjunto óptimo de rutas diarias para un período de tiempo dado. Así, resolver un PVRP lleva implícito la resolución de dos problemas básicos: un problema de planeamiento cuyo objetivo es la determinación de un conjunto de días de visita para cada centro dentro del período de tiempo que se considera y un problema de ruteo de vehículos para cada día a los fines de determinar las mejores rutas diarias.

El presente trabajo modela el problema de recolección de residuos infecciosos como un PVRP. Para abordar la planificación se utiliza un algoritmo evolutivo y el problema para la determinación del conjunto de rutas óptimas en cada día se resuelve con un algoritmo de búsqueda local.

Este trabajo considera la aplicación de técnicas de computación evolutiva (EC) para resolver la planificación semanal, presentando un algoritmo genético (GA) mejorado, desde el punto de vista de la representación de los individuos que conforman la población y un mecanismo de mutación que permite explotar las posibilidades de hallar buenas soluciones dentro de regiones factibles.

Una vez realizada la planificación semanal, es necesario para cada día resolver un problema de ruta óptima del tipo VRP. Este es el nombre genérico dado a una amplia serie de problemas de optimización en los cuales se debe determinar el mejor conjunto de rutas que debe seguir una flota de vehículos para servir un número de clientes geográficamente dispersos. El objetivo del VRP es entonces, encontrar el conjunto de rutas, que se originan y

terminan en un depósito, de modo tal de servir a todos los clientes con el mínimo costo.

El VRP es un problema de optimización combinatoria complejo. Como ha sido demostrado en la literatura, es NP-Hard², y por lo tanto el tiempo de resolución basado en métodos exactos de enumeración, crece exponencialmente con el tamaño del problema.

Para esta clase de problemas se utilizan a menudo métodos heurísticos de resolución^{3,4}. La mayoría de los métodos desarrollados emplean heurísticas tales como Búsqueda Tabú⁵, *Simulated Annealing*⁶ o Colonia de Hormigas⁷. En este trabajo se utiliza la técnica de Búsqueda Tabú, que se ofrece como una alternativa válida para la optimización diaria.

2 FORMULACIÓN DEL PROBLEMA

Considérese el grafo $G=(N,A)$ y el período de np días, en el cual N es el conjunto de nodos que representa las instituciones generadoras o clientes a atender y A el conjunto de arcos que conectan los nodos.

Cada cliente requiere que la tarea de recolección sea realizada con una frecuencia $k(j)$ que mide la cantidad de veces que el cliente j debe ser visitado, con $1 \leq k(j) \leq np$.

El PVRP básico consiste en seleccionar $k(j)$ días distintos de visitas para el nodo j y resolver los np problemas VRP que resultan, de modo de minimizar el costo total de recorrido. Es obviamente un problema de tipo NP-hard ya que incluye al VRP de período simple como caso particular.

3 ENFOQUE HEURÍSTICO

Para abordar el problema de optimización que resuelve simultáneamente el problema de planificación y las rutas de recolección se ha diseñado un algoritmo evolutivo híbrido que utiliza un método de búsqueda tabú en la etapa de evaluación de aptitud de las soluciones.

Los componentes fundamentales de un algoritmo evolutivo son: la representación de los individuos, el método de selección de padres, los operadores genéticos y el diseño de la función de aptitud. En las siguientes subsecciones se describe la implementación de la estrategia seleccionada.

3.1 Representación de las soluciones

Considerando el período de np días de planificación y que cada institución generadora, j , tiene una frecuencia $k(j)$ asociada, es posible representar su régimen de visitas a través de un conjunto P de vectores binarios \mathbf{x} . Cada componente x_i toma el valor 1 si dicho cliente es visitado el día i o 0 en caso contrario. Si, por ejemplo, para $np=7$, un cliente debe recibir una visita semanal, el conjunto de vectores binarios es $P=\{0000001, 0000010, 0000100, 0001000, 0010000, 0100000, 1000000\}$.

La cardinalidad del conjunto P , esto es el número de alternativas de visitas válidas, está dada por el número de combinaciones de np tomados de $k(j)$

Si N es el número total de clientes a atender, entonces una planificación semanal completa puede ser representada por un vector binario de dimensión $np.N$.

Para reducir el número de símbolos empleados, se propone utilizar una notación de números enteros simplemente asignando al vector \mathbf{x} el valor entero que resulta de la conversión de sus componentes tomadas como un número binario, cuya expresión está dada por:

$$y_j = \sum_{i=1}^7 2^{x_i} \quad j = 1 \dots N \quad (1)$$

De este modo cada individuo es representado por un vector \mathbf{y} de N números enteros, cada uno de los cuales representa el régimen de visitas para cada cliente. Cada componente del vector debe contener un número que se corresponda con la cantidad de visitas requeridas $k(j)$. En la tabla 1 se muestran el conjunto de alternativas válidas para el caso de $np=7$.

k	Alternativas Válidas
1	1 2 4 8 16 32 64
2	3 5 6 9 10 12 17 18 20 24 33 34 36 40 48 65 66 68 72 80 96
3	7 11 13 14 19 21 22 25 26 28 35 37 38 41 42 44 49 50 52 56 67 69 70 73 74 76 81 82 84 88 97 98 100 104 112
4	15 23 27 29 30 39 43 45 46 51 53 54 57 58 60 71 75 77 78 83 85 86 89 90 92 99 101 102 105 106 108 113 114 116 120
5	31 47 55 59 61 62 79 87 91 93 94 103 107 109 110 115 117 118 121 122 124
6	63 95 111 119 123 125 126
7	127

Tabla 1. Alternativas de visitas válidas de acuerdo a la frecuencia requerida

Una propuesta de planificación estará formada por un vector de números enteros de n componentes. Por ejemplo, si se consideran 5 clientes, el vector $\mathbf{p}=[1, 47, 20, 5, 126]$ propone visitar una vez al cliente 1, 5 veces al cliente 2, 2 a los clientes 3 y 4 y seis veces a la semana al cliente 5.

Una población de r soluciones candidatas se almacenan en una matriz de r filas por N columnas.

3.2 Generación de la población inicial

La población inicial se genera de manera tal de asegurar que todos los individuos son inicialmente factibles en el sentido de que cada cliente tiene asociado un número de visitas igual al requerido. Esto se consigue eligiendo el régimen al azar pero dentro de las alternativas válidas para cada caso, extraídas de la tabla 1.

3.3 Operadores genéticos

Un algoritmo genético (AG) tradicional utiliza esquemas de selección y apareamiento panmíticos, esto es cada individuo compite globalmente con todos los otros presentes en la población. Durante el proceso de selección, un individuo seleccionado puede potencialmente

aparearse con cualquier otro individuo en la población. Aunque estos esquemas, tales como selección estocástica con reemplazo, son ampliamente utilizados, ellos no están equipados para realizar búsquedas de genotipos “exitosos” en paisajes adaptivos de gran tamaño y multimodales, ya que la población es incapaz de mantener radicalmente diferentes genotipos con buen *fitness* debido a la convergencia. El problema de la convergencia prematura, o mejor aún el balance entre explotación y exploración, ha sido un problema constante en los AGs tradicionales. En la literatura pueden encontrarse muchas variantes sobre el AG clásico que intentan tratar con este problema ^{8,9}

En este trabajo se utilizó un AE de población estructurada donde el intercambio de información entre los individuos está restringida a una estructura de vecindad local. Con el objetivo de simular una topología del espacio de búsqueda determinada, los individuos se distribuyen en una grilla unidimensional (anillo) o bidimensional (toroide)¹⁰; el solapamiento de las vecindades permite una difusión uniforme de las buenas soluciones a través de la grilla.

El método define, en primer lugar, un conjunto de individuos que son seleccionados de alguna manera para ser "padres". La selección de las "madres" se realiza, para cada padre entre los individuos de su vecindad. Para evitar diferencias desproporcionadas en los valores de *fitness*, se somete a la población a un *ranking* lineal antes de comenzar el proceso. De esta manera los cromosomas son seleccionados proporcionalmente a su rango mas que a los valores reales de la función de aptitud, de forma de evitar que superindividuos eliminen material genético deseable y hagan que la evolución converja prematuramente¹¹.

Se implementa cruzamiento de un punto de cruce, dado que la representación utilizada lo admite. Después de determinar aleatoriamente un punto de cruce, p_c , se genera descendencia de dos padres mediante el intercambio de los genes a la derecha de p_c .

El mecanismo mutación se realiza de forma tal de garantizar la factibilidad de la solución. Por esta razón se concibe la mutación de un gen j como un cambio de propuesta semanal para el cliente j , pero sólo entre las posibilidades que se encuentran disponibles para dicho cliente. En otras palabras el gen tomará otro valor elegido dentro de las alternativas posibles de acuerdo con el número de visitas que requiere.

3.4 Función de aptitud

Cada gen del cromosoma representa una propuesta de visitas para el cliente j , en el período de np días, con una frecuencia $k(j)$. El valor de aptitud asociado a dicho cromosoma resulta de determinar el costo del conjunto de rutas óptimas para cada uno de los días (C_i), en el período np . Esto involucra resolver np problemas del tipo VRP. La expresión de la función de aptitud esta dada por:

$$Costo = \sum_{i=1}^{np} C_i \quad (2)$$

Donde C_i es el costo del conjunto de rutas óptimas para el día i .

La resolución del conjunto de problemas VRP resultantes, se realiza utilizando una

estrategia basada en Búsqueda Tabú (TS)³, técnica metaheurística que utiliza un procedimiento de búsqueda local para explorar el espacio de soluciones, junto con mecanismos de memoria adaptiva diseñados para evitar el estancamiento en mínimos locales y la visita cíclica de las mismas soluciones. La información histórica acerca del proceso de búsqueda de soluciones es almacenada en las llamadas listas tabú. La implementación del procedimiento de búsqueda tabú comprende los siguientes aspectos:

- a) A partir de una solución inicial (\mathbf{q}_0), se define en cada iteración una vecindad de posibles soluciones, $N(\mathbf{q})$, modificando la solución corriente a través de una secuencia de movimientos. La solución inicial se obtiene extrayendo del cromosoma el conjunto de n clientes, con $n \leq N$, que serán visitados el día i . Dada la solución $\mathbf{q}=(q_1, q_2, \dots, q_n)$, la vecindad de \mathbf{q} se define como el conjunto de todas las soluciones \mathbf{q}' que se obtienen de intercambiar, por pares, clientes de la solución actual \mathbf{q} . Así, el entorno completo de una solución \mathbf{q} con n clientes está constituida por ${}_n C_2 = n(n-1)$ soluciones adyacentes que pueden ser obtenidas a partir de estos intercambios.
- b) Los elementos de dicha vecindad son examinados para determinar el mejor de ellos que está ausente de la lista tabú. Un elemento de la vecindad es evaluado utilizando una función que tiene en cuenta el costo del recorrido completo, de un día, partiendo del depósito visitando todos los clientes de la ruta y retornando al depósito. Dicha función puede escribirse como :

$$C_i = D_{deposito, q_1} + \sum_{j=1}^{n-1} D_{q_j, q_{j+1}} + D_{q_n, deposito} \quad (3)$$

Donde \mathbf{D} es la matriz de distancias mínimas entre clientes y n es la cantidad de clientes que se visitan en el día i . Esta solución es seleccionada como punto de partida para una nueva iteración aún cuando no mejore la solución anterior \mathbf{q} , a la vez que se guarda la mejor solución encontrada hasta el momento (\mathbf{q}^*).

- c) El método lleva un registro de los movimientos efectuados para alcanzar \mathbf{q}' desde \mathbf{q} en cada iteración. Esta *memoria* del proceso modifica la generación de la vecindad determinando que soluciones pueden ser alcanzadas por un movimiento a partir de la solución corriente. Normalmente esto se lleva a cabo a través de una lista de movimientos temporalmente prohibidos llamada lista tabú basada en lo reciente, la cual se actualiza en cada iteración. Pasado un cierto número de iteraciones, llamado período tabú (pt), el movimiento es nuevamente permitido. La lista tabú representa la memoria a corto plazo del proceso. El estado tabú asociado a un movimiento puede eventualmente ser revocado si se satisface algún criterio de aspiración como por ejemplo la generación de una solución de mejor calidad que cualquier otra visitada previamente durante la búsqueda
- d) En contraste, la lista tabú basada en la frecuencia con que ciertos movimientos son realizados, representa la memoria a largo plazo de la técnica y permite la diversificación de la búsqueda al seleccionar movimientos que han sido menos

El régimen de visitas que requiere cada institución para un período, np , de siete días se muestra en la tabla 3.

Frecuencia de Visitas	Clientes
1	14,15,16,17,19,20,21,22,23,24,25,26,27,28
2	13
4	1
6	7
7	2,3,4,5,6,8,9,10,11,12

Tabla 3. Cantidad de visitas de los clientes

Se ejecutó un algoritmo evolutivo con una población de 50 individuos, una probabilidad de cruzamiento de 0,7 y una probabilidad de mutación de 0,001. Se realizaron 25 ejecuciones con 200 iteraciones en cada una de ellas.

En relación a TS el periodo tabú es función de la cantidad de clientes que se visitan en el día, y esta dado por $pt = \lfloor \sqrt{n(n-1)/2} \rfloor$. El horizonte de frecuencia se fijó en 5

En la tabla 4 se muestra la planificación de costo mínimo encontrada. El costo asociado a esta planificación es de 1387.50

Lunes	4 3 2 1 16 6 7 8 9 21 5 11 10 12 13
Martes	4 3 2 14 1 6 7 8 9 5 11 10 12
Miércoles	4 3 2 1 9 6 17 24 10 12 11 8 27 22 5 18
Jueves	4 3 2 19 6 7 8 9 21 5 11 10 12
Viernes	4 3 2 9 6 7 8 21 10 12 11 23 5 26 25 28 13
Sábado	4 3 15 2 1 6 7 8 9 5 11 10 12
Domingo	4 3 2 20 6 7 8 9 21 5 11 10 12

Tabla 4. Planificación de mínimo costo.

Los resultados encontrados por las distintas ejecuciones se muestran en la figura 1. El valor medio obtenido es de 1408.6, con un desvío estándar de 8.35. Las diferencias de la planificación que muestra la tabla 4, respecto de las obtenidas en otras corridas no son importantes y la máxima diferencia en el costo asociado es del 2.2%.

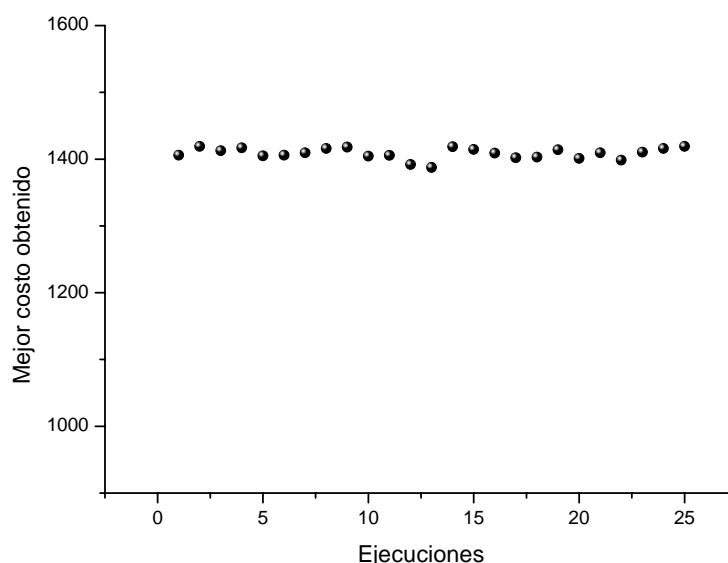


Figura 1. Costo de las planificaciones obtenidas en las ejecuciones del AE.

5 CONCLUSIONES

Se ha desarrollado una estrategia para resolver el problema de la recolección de residuos infecciosos, el cual ha sido modelado como un Problema de Ruteo de Vehículos Periódico, a través de un algoritmo híbrido que combina técnicas que provienen de la computación evolutiva y técnicas de búsqueda local. Soluciones potenciales de una planificación semanal evolucionan por acción de un AE en el problema principal, mientras que, al momento de la determinación de la función de evaluación se resuelve otro problema de optimización (VRP) a través de una técnica basada en TS:

Los resultados obtenidos muestran que la eficiencia del método utilizado es satisfactoria por cuanto no se observan diferencias importantes en las distintas ejecuciones que fueron llevadas a cabo y se obtuvieron buenas soluciones en todos los casos.

Se observó además que la combinación de las técnicas mencionadas contribuye a mejorar el desempeño por cuanto se conjugan la capacidad exploratoria que poseen los algoritmos poblacionales con la capacidad de explotación de regiones atractivas del espacio de búsqueda que son inherentes a los métodos basados en búsqueda local.

6 REFERENCIAS

- [1] Shih, L Chang, H. A routing and scheduling system for infectious waste collection. *Environmental Modeling and Assessment Vol 6*. pp.261–269, Kluwer Academic Publishers. 2001.
- [2] Garey, M. ; Johnson, D. “Computers and Intractability. A Guide to the theory of NP-Completeness”, W.H. Freeman and Company, New York (1999).
- [3] Lacomme, F. ; Prins, C. et al "Evolutionary algorithms for periodic arc routing problems", *European Journal of Operational Research*, 165 pp 535-553. 2005

- [4] Prins, C. A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*. Vol.. 31, No12, pp 1985-2002. 2004
- [5] Glover, F.; Melián B. "Tabu Search" *Revista Iberoamericana de Inteligencia Artificial*. Vol 19. pp 29-48. 2003
- [6] Bent, R. and Hentenryck, P. V. (2001). A Two-Stage Hybrid Local Search for the Vehicle Routing Problem with Time Windows. Technical Report, CS-01-06, Brown University.
- [7] Gambardella, L. M., Taillard, E. and Agazzi, G. (1999) MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows, In D. Corne, M. Dorigo and F. Glover (eds.), *New Ideas in Optimization*. McGraw-Hill, London, UK, pp. 63-76.
- [8] Collins, J. "Studies in Artificial Evolution". PhD Thesis, University of California, 1992.
- [9] Alba, E.; Tomassini, M., "Parallelism and Evolutionary Algorithms", *IEEE Transactions of Evolutionary Computation*. 6, 443-462, 2002.
- [10] Sarma, J. De Jong, K., "An Analysis of Local Selection Algorithms in a Spatially Structured Algorithms, Lansing, MI, July 1997. Evolutionary Algorithm", *Proceedings of ICGA-97, the 7th International Conference on Genetic*
- [11] Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)