



Implementation of the genetic algorithm for optimization of garbage collection considering the load-dependent cost

David De Santis Bermeo, M.Sc.¹ , Pedro Ramos De Santis, Ph.D.² 
Escuela Superior Politécnica del Litoral, Ecuador,
¹ dade@espol.edu.ec, ² pramos@espol.edu.ec.

Abstract— This paper proposes a variant of the Chinese postman problem with load-dependent costs (CPPLC), to be used in pick-up cases. The mathematical model is modified, as well as a genetic algorithm with a dynamic programming algorithm is implemented to solve a case study about a garbage collection problem in an urbanization in the city of Daule, Ecuador. The results obtained are compared with the current situation and the improvements found are analyzed, highlighting novel results found compared with the delivery case. Future lines of research within this type of problem are also highlighted.

Keywords—CPP, CPPLC, arc-routing, environmental logistics, route innovation.

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

Implementación de un algoritmo genético para la optimización del proceso de recolección de basura considerando costos dependientes de la carga

David De Santis Bermeo, M.Sc.¹ , Pedro Ramos De Santis, Ph.D.² .

Escuela Superior Politécnica del Litoral, Ecuador,

¹ dade@espol.edu.ec, ² pramos@espol.edu.ec.

Abstract— Este artículo plantea una variante del problema del cartero chino con costos dependientes de la carga (CPP-LC), para ser utilizado en casos de recogida. Se modifica el modelo matemático, así como se implementa un algoritmo genético con un algoritmo de programación dinámica para resolver un caso de estudio sobre un problema de recolección de basura, en una urbanización de la ciudad de Daule, Ecuador. Se comparan los resultados obtenidos con la situación actual y se analiza las mejoras encontradas, destacando resultados novedosos encontrados en comparación con el caso de entrega, así como se destaca futuras líneas de investigación dentro de este tipo de problemas.

Keywords—CPP, CPP-LC, enrutamiento de arcos, logística ambiental, innovación de rutas.

I. INTRODUCCIÓN

En los últimos años, el creciente interés por abordar los problemas ambientales ha generado un cambio con un enfoque de los problemas en la cadena de suministro proponiendo soluciones que no solo reduzcan los costos logísticos asociados, sino que, también enfrenten de manera efectiva la problemática ambiental específica relacionada con la contaminación generada en el proceso de transporte, almacenamiento entre otros.

En particular, la evolución de los problemas relacionados con el transporte, de manera particular con las rutas de vehículos ha transitado desde la consideración exclusiva de la optimización de la función objetivo que minimice la distancia hacia la inclusión de las emisiones de CO₂, marcando una tendencia destacada en la investigación operativa reciente [1, 2]. Este enfoque se ha expandido hacia modelos más complejos, centrándose en la optimización de la velocidad para minimizar el consumo de combustible [3, 4, 5]. El trabajo de [6] ha llevado esta línea de investigación al ámbito de las rutas de vehículos, abriendo así la puerta a contribuciones algorítmicas y de modelado [8, 9, 10, 11, 12]. Un elemento esencial en estos estudios es el modelo de consumo de combustible, el cual se basa en investigaciones previas como las de [13, 14].

Por otro lado, dentro de los problemas de optimización de rutas basados en aristas tenemos a [15, 16]. Una variante interesante sobre el problema del cartero chino lo realiza [17] en el problema de cartero chino con costos dependientes (CPP-LC) donde la distancia recorrida, y el peso del vehículo

(incluyendo la carga transportada) son las variables principales que influyen en el costo, en un caso de entregas.

Este artículo busca adaptar estos conceptos al contexto de la recolección de basura, es decir un CCP-LC, pero en problemas de recolección, considerando la carga de residuos como un factor determinante en el consumo generado, planteando la formulación de programación matemática basada en enrutamiento sobre aristas, que consideren no solo la minimización de la distancia sino también del peso. Adicional, se presenta la implementación de un algoritmo genético como metaheurística para la generación de una solución aproximada del problema y utilizándola en un caso de estudio aplicado a un problema de recolección de basura en áreas urbanas de la ciudad de Daule, obteniendo una solución que genere ahorros económicos, pero también ambientales, considerando la aplicabilidad y la extrapolación de los resultados a entornos urbanos similares.

En la sección 2, se presenta el caso de estudio, así como los conceptos matemáticos necesarios para resolver el mismo. En la sección 3 se muestran las variaciones al modelo matemático CPP-LC, para problemas con recolección. En la sección 4 se presenta la estructura del algoritmo genético, los tipos de operadores genéticos utilizados, así como una adaptación al algoritmo de programación dinámica propuesta por [17], utilizado dentro del algoritmo genético, pero aplicándolo a casos de recolección. En la sección 5 se presentan los resultados obtenidos dentro del caso de estudio y se realiza una discusión de los resultados encontrados comparándolos con la situación actual. Finalmente, en la sección 6 se exponen las conclusiones del estudio realizado, así como también se proponen futuras líneas de investigación para este tipo de problemas.

II. EL PROBLEMA

La problemática es una variante del conocido Problema del Cartero Chino con Costos Dependientes de la Carga (CPP-LC) [17], adaptado para problemas de recolección. Este problema aborda la optimización de rutas, teniendo en cuenta tanto la distancia física como la carga transportada en vehículos de recolección. En términos generales, se puede definir de la siguiente manera: se considera un grafo no dirigido conexo $G = (V, E)$, donde $V = \{1, 2, \dots, n\}$ representa el conjunto de vértices y $E = \{e_1, e_2, \dots, e_m\}$ denota el conjunto de aristas.

Cada arista $e \in E$ tiene 2 parámetros: $d_e \geq 0$ que representa la distancia y $q_e \geq 0$ la demanda en la arista e , representando, en este caso de estudio, metros a recorrer y kilogramos de basura a ser recolectados a lo largo de una arista, respectivamente.

Se considerará en este caso de estudio, camión recolector de basura con peso W , en vacío y $W + Q_{max}$ completamente cargado, donde Q_{max} representa la carga máxima que puede ser transportado. El camión inicia su recorrido en el nodo inicial sin carga, atravesando todas las aristas del grafo recolectando desechos y regresando al vértice inicial al finalizar la ruta. Durante el recorrido de una arista e , se realiza el servicio, que implica la recolección de una cantidad q_e de basura por parte del vehículo. Antes o después del servicio, la arista e puede ser atravesada nuevamente, en modo de rumbo muerto (sin recoger basura al recorrer la arista), en múltiples ocasiones.

En cada instancia en que el vehículo atraviesa una arista e , ya sea sirviéndola o desplazándose por ella, se genera un costo proporcional a la longitud de la arista multiplicada por el peso actual del vehículo. Esta función objetivo es similar a la que propuso [18] con la diferencia que Kara la propuso para un problema de rutas basado en nodos y Coberán en aristas, pero ambos tenían el mismo concepto, la suma producto de la distancia recorrida por el peso transportado en cada arista. Entonces, para estimar el costo incurrido al atravesar cada arista en este problema, hay que hacer una diferencia entre aristas donde se recoge producto, representada por la ecuación 1

$$d_e \left(W + Q_e + \frac{q_e}{2} \right) \quad (1)$$

y las aristas que se transitan sin recoger desechos, representada por la ecuación 2

$$d_e(W + Q_e) \quad (2)$$

Donde Q_e representa la carga recolectada hasta antes de recorrer la arista e , $\frac{q_e}{2}$ indica que la carga a recolectar sobre la arista e , es recogida de manera proporcional a lo largo de la misma. Si definimos un subconjunto $E_s \in E$, que representa todas las aristas que se visitarán más de una vez, sin recoger producto, repitiendo las aristas, dentro de este conjunto tantas veces como se las visite. El costo total de para el CPP-LC con recolección viene dado por la ecuación 3:

$$\sum_{e \in E} d_e \left(W + Q_e + \frac{q_e}{2} \right) + \sum_{e \in E_s} d_e(W + Q_e) \quad (3)$$

Por lo tanto, el CPP-LC para recolección, consiste en visitar todas las aristas del grafo al menos una vez, recogiendo los

desechos, saliendo del nodo inicial sin carga y regresando al mismo nodo con los desechos de todas las aristas, minimizando el costo dado por la ecuación 3.

A. Caso de Estudio

La urbanización seleccionada en la ciudad de Daule para el estudio se muestra en la Figura 1, con un grafo representando su topología que consiste en 8 nodos y 10 aristas representado en la Figura 2. Mediante un trabajo de campo, se estimó el peso de la basura recolectada en kilogramos para cada arista del grafo, además se calculó la distancia en metros de cada arista, utilizando Google Maps. Esta información puede ser observada en la tabla 1.

Como se discutió previamente, el peso a recolectar añade una complejidad adicional al problema, ya que la variabilidad en la cantidad de basura recolectada influye directamente en la ruta tomada por el vehículo de recolección.



Fig. 1. Toma aérea de la urbanización, utilizando Google Maps.

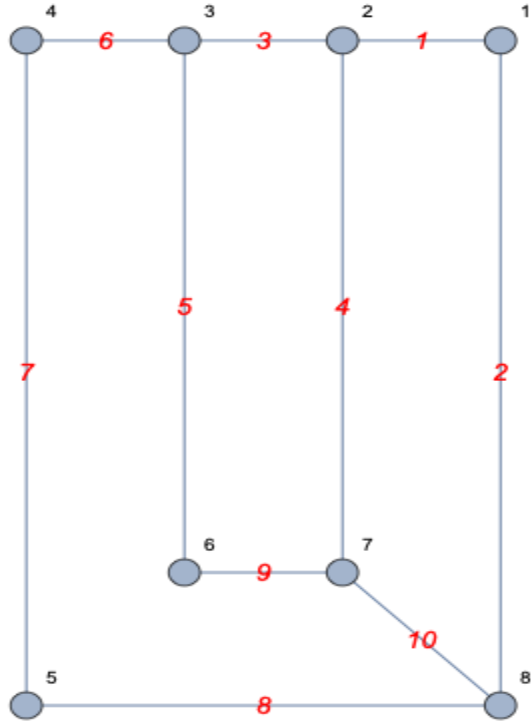


Fig. 2. Grafo representativo de la instancia de estudio

TABLA I
DATOS DE LAS ARISTAS PARA LA PROBLEMÁTICA

| Arista | Nodos Inicial | Nodos Final | d(m) | q(kg) |
|--------|---------------|-------------|-------|-------|
| 1 | 1 | 2 | 43 | 42 |
| 2 | 1 | 8 | 242.6 | 300 |
| 3 | 2 | 3 | 40.4 | 60 |
| 4 | 2 | 7 | 189.8 | 240 |
| 5 | 3 | 6 | 189.8 | 228 |
| 6 | 3 | 4 | 44.6 | 60 |
| 7 | 4 | 5 | 203.7 | 318 |
| 8 | 5 | 8 | 135.3 | 240 |
| 9 | 6 | 7 | 40.4 | 54 |
| 10 | 7 | 8 | 98.7 | 54 |

En la tabla 1 , la columna arista, representa el número asignado a cada arista, también se puede apreciar el nodo inicial y final, así como la distancia de las aristas en metros: d y la carga de basura a recolectar en kilogramos: q .

III. EL MODELO MATEMÁTICO

El modelo matemático utilizado es el propuesto por [17], para resolver el CPP-LC, el cual se lo ha modificado para poderlo utilizar en casos de entrega.

Los índices $i, j \in E$, representan los nodos, el índice $e \in E$ las aristas y el índice $k=1,2,\dots,m$, el orden o momento de visita de cada arista. La cantidad de momentos irán desde 1 hasta m . Por ejemplo, la arista 23, puede ser visitada en segundo lugar es decir en el momento 2 y la arista 12 en primer lugar, es decir el momento 1.

Sea $\delta^+(i) = \{j: (i,j) \in E, j \in V \setminus \{i\}\}$ el conjunto de nodos de llegada de los arcos que salen del nodo i y $\delta^-(i) = \{j: (j,i) \in E, j \in V \setminus \{i\}\}$ el conjunto de nodos de partida de los arcos que llegan al nodo i

Hay 3 grupos de variables:

y_{ij}^k variable binaria, será 1 si visito la arista ij , en el momento k , para recolectar los desechos, 0 si no.

x_{ij}^k variable binaria, será 1 si visito la arista ij , entre el momento k y el $k+1$, sin recolectar desechos, 0 si no.

f_k variable positiva que representa la cantidad de desechos recolectados en el momento k , esta variable será 0 en el momento $k=1$ (al salir del depósito) y será $Q = \sum_{k=1}^m q_k$ en el momento $k=m$ (al regresar al depósito).

El modelo matemático con los cambios realizados, es el siguiente

$$\text{Minimize } \sum_{k=1}^m \sum_{e=(i,j) \in E} d_e \left((W + f_k + \frac{1}{2}q_e)(y_{ij}^k + y_{ji}^k) + (W + f_{k+1})(x_{ij}^k + x_{ji}^k) \right) \quad (4)$$

$$\text{subject to } \sum_{k=1}^K (y_{ij}^k + y_{ji}^k) = 1, \forall e \in E, \quad (5)$$

$$\sum_{(i,j) \in E} (y_{ij}^k + y_{ji}^k) = 1, \forall k = 1, \dots, K, \quad (6)$$

$$f_{k+1} = f_k + \sum_{e=(i,j) \in E} q_e (y_{ij}^k + y_{ji}^k), \forall k = 1, \dots, K-1, \quad (7)$$

$$f_1 = 0, \quad (8), \quad f_{K+1} = Q, \quad (9)$$

$$y^1(\delta^-(i)) + x^1(\delta^-(i)) = y^2(\delta^+(i)) + x^1(\delta^+(i)), \quad \forall i \in V \setminus \{1\}, \quad (10)$$

$$y^k(\delta^-(i)) + x^k(\delta^-(i)) = y^{k+1}(\delta^+(i)) + x^k(\delta^+(i)), \quad \forall i \in V, \forall k = 2, \dots, K-1, \quad (11)$$

$$y^K(\delta^-(i)) + x^K(\delta^-(i)) = x^K(\delta^+(i)), \forall i \in V \setminus \{1\}, \quad (12)$$

$$y^K(\delta^-(1)) + x^K(\delta^-(1)) = y^1(\delta^+(1)) = 1, \quad (13)$$

$$x_{ij}^k, x_{ji}^k, y_{ij}^k, y_{ji}^k \in \{0,1\}, \forall (i,j) \in E, \forall k = 2, \dots, m, \quad (14)$$

$$L_k \leq f_k \leq U_k, \forall k = 2, \dots, m. \quad (15)$$

Donde la ecuación 4, es la función objetivo, que representa el costo de atravesar todas las arista del grafo V , incluidas las que se pasa sin recoger carga, considerando la distancia de las misma y la carga transportada al momento de pasar por cada una de ellas, expresada por la ecuación 3. La restricción 5 y 6 nos dicen que todas las aristas de V deben ser visitadas, y solo una por cada momento k . La restricción 7 nos dice que la carga que llevo en el momento $k+1$, será igual a la carga que tenía en el momento k más lo que se recoge en la arista ij o ji . Las restricciones 8 y 9 nos garantizan que el vehículo recolector saldrá del depósito sin carga y regresará al mismo con la carga de todas las aristas del grafo representada en Q . Las restricciones 10,11,12 y 13 nos garantizan la conservación de flujo de las aristas, es decir, si se pasa por la arista 12 en el momento 1, entonces en el momento 2 se debe empezar por una arista que empiece en 2, ya sea para recoger carga o para atravesarla hasta llegar a la arista donde se recogerá la carga y así hasta regresar al depósito. La restricción 14, son las restricciones de las variables binarias y_{ij}^k , x_{ij}^k y la restricción 15 son las cotas inferior y superior para la variable f_k , que en el caso más relajado $U_k = Q$ y $L_k = 0$.

Para adaptar el problema a un caso de recolección se modificó la función objetivo 4 y las restricciones 7,8 y 9 y se eliminó la restricción que nos decía que para hacer deadheading por una arista la debía haber servido primero.

IV. ALGORITMO GENÉTICO

A. Estructura

El algoritmo genético tiene la siguiente estructura:

1. Generación de una población aleatoria de tamaño n .
2. Selección los m individuos más aptos considerando la aptitud, es decir, el costo de la solución considerando la distancia recorrida y la carga transportada, dada por la ecuación 3., Donde $m = [n \cdot p_c]$.
3. Elaboración del cruce de los individuos basado en la técnica de 2 puntos de corte, propuesta por [19].
4. Los hijos generados pasan por un algoritmo de búsqueda local, basado en el movimiento 2 OPT [20].
5. Los hijos generados en 4, pueden sufrir un proceso de mutación con una probabilidad p_m . El proceso de mutación está basado en el movimiento 1 OPT o regla de intercambio [20].
6. El mejor individuo de la población actual, es siempre seleccionado, por el proceso elitista, para que pase a la siguiente generación.
7. Se regresa al paso 2, hasta que se cumpla el tiempo máximo de ejecución T_{max} .

B. Representación de la solución

Basados en el grafo propuesto en la figura 2 para la problemática en estudio, y con la información para cada arista, dada por la tabla 1. Una solución factible es la siguiente:

12,23,36,67,78,85,54,43, [32],27, [78],82.

Donde las aristas 32 y 78, se las recorre 2 veces. Cuando las aristas están entre corchetes, significa que durante el recorrido no se recoge basura, ya que previamente se lo había realizado. Esta solución puede ser representada por 2 vectores:

- x : vector de sucesión de aristas, considerando el número de arista asignado en la tabla 1.
- x_d : vector que indica cómo se atraviesa la arista. Por ejemplo, la arista 3 puede ser atravesada 1 de 23 o de 32. En el primer caso x_d será 1 y el segundo 2. Siempre que la arista sea atravesada en la forma definida en la tabla 1, se le asignará a x_d el valor de 1.

Por lo tanto, la solución factible puede ser representada de la siguiente forma:

$$x = [1 3 5 9 10 8 7 6 4 2]$$

$$x_d = [1 1 1 1 1 2 2 2 1 2]$$

Nótese que, dentro de esta representación, solo se escriben las aristas cuando se pasa recogiendo desechos, ya que las aristas que se atraviesan sin recolectar desechos, se pueden obtener mediante el algoritmo del camino más corto entre el nodo final de la arista k y el nodo inicial de la arista $k + 1$. Durante este recorrido, la cantidad transportada permanece constante.

C. Modificación al Algoritmo de Programación (PD)

El principal aporte dentro de esta implementación, fue la modificación del algoritmo de programación dinámica propuesta por [17], la cual garantiza que dada una secuencia de aristas x , para un problema de entrega, se puede generar un vector de direcciones x_d , que minimice la distancia recorrida.

Esta implementación fue modificada para ser utilizada en problemas de recolección.

El algoritmo de PD, necesita los siguientes parámetros:

- d_{ij} la matriz de distancia más corta entre todos los nodos del grafo, la cual puede ser calculada utilizando el algoritmo de Floyd–Warshall [21]

- (i_k, j_k) representan las aristas donde i_k es el nodo inicial y j_k es el nodo final de la arista en la posición k .
- $m = |E|$ es la cantidad de aristas del grafo V .
- $f_k(d)$ representa el costo acumulado de atravesar las aristas desde k hasta m .
- W es el peso en vacío del camión de reparto.
- Q_k es la carga recogida hasta justo antes de recorrer la arista k .

El algoritmo de PD queda de la siguiente forma:

$$f_1(1) = \left(W + \frac{q_1}{2}\right) d_{i_1, j_1} + \min \begin{cases} (W) d_{1, i_1} + f_2(1), \\ (W) d_{1, j_1} + f_2(2), \end{cases} \quad (20)$$

$$f_k(1) = \left(W + Q_k + \frac{q_k}{2}\right) d_{i_k, j_k} + \min \begin{cases} (W + Q_k) d_{j_{k-1}, i_k} + f_{k+1}(1), \\ (W + Q_k) d_{j_{k-1}, j_k} + f_{k+1}(2), \\ k = 2, \dots, m-1, \end{cases} \quad (21)$$

$$f_k(2) = \left(W + Q_k + \frac{q_k}{2}\right) d_{i_k, j_k} + \min \begin{cases} (W + Q_k) d_{i_{k-1}, i_k} + f_{k+1}(1), \\ (W + Q_k) d_{i_{k-1}, j_k} + f_{k+1}(2), \\ k = 2, \dots, m-1, \end{cases} \quad (22)$$

$$f_m(1) = \left(W + Q_k + \frac{q_m}{2}\right) d_{i_m, j_m} + \min \begin{cases} (W + Q_k) d_{j_{m-1}, i_m} + (W + Q_k + q_m) d_{j_m, 1}, \\ (W + Q_k) d_{j_{m-1}, j_m} + (W + Q_k + q_m) d_{i_m, 1}, \end{cases} \quad (23)$$

$$f_m(2) = \left(W + Q_k + \frac{q_m}{2}\right) d_{i_m, j_m} + \min \begin{cases} (W + Q_k) d_{i_{m-1}, i_m} + (W + Q_k + q_m) d_{j_m, 1}, \\ (W + Q_k) d_{i_{m-1}, j_m} + (W + Q_k + q_m) d_{i_m, 1}. \end{cases} \quad (24)$$

En cada arista se toma la decisión sobre cómo será atravesada ($x_d = 1$ o $x_d = 0$), empezando en la arista final m hasta la arista inicial 1. La decisión en cada arista k dependerá de la decisión de la arista siguiente $k+1$; solo la arista final no dependerá de otra arista.

Los cambios fueron los siguientes:

1. En la ecuación 20, la carga del vehículo cuando sale del nodo de inicio es W , ya no $W+Q$ como en el caso de entregas, donde $Q = \sum_{k=1}^m q_k$.
2. En la ecuación 20, el peso al recorrer la primera arista es $W + \frac{q_1}{2}$, y no $W + Q - \frac{q_1}{2}$.
3. En las ecuaciones 21 y 22, el peso al recorrer la arista k es $W + Q_k + \frac{q_k}{2}$, ya no $W + Q_k - \frac{q_k}{2}$.

4. En las ecuaciones 23 y 24, el peso al recorrer la arista m es $W + Q_k + \frac{q_m}{2}$, ya no $W + \frac{q_m}{2}$.
5. En las ecuaciones 23 y 24, el peso antes de llegar a la arista m es $W + Q_k$, ya no $W + q_m$.
6. En las ecuaciones 23 y 24, el peso cuando el vehículo se dirige al nodo de inicio es $W + Q_k + q_m$, ya no W , como en el caso de entrega.

El algoritmo de programación dinámica y la regla relativa a que las aristas que visito sin recolectar producto, se obtienen mediante el algoritmo del camino más corto, tienen un rol protagónico dentro de la metaheurística, ya que reduce el problema a encontrar una secuencia óptima de aristas sin repetir, pudiendo aplicar todas las técnicas existentes para el algoritmo genético, cuando la representación de la solución es una secuencia de valores, como lo es el TSP (Problema del agente viajero).

La metaheurística y el algoritmo de programación dinámica fueron implementados en Matlab versión R2016B y corrida en una computadora con sistema operativo Windows 11 de 64 bits, con procesador 12th Gen Intel(R) Core(TM) i7-1260P 2.10 GHz, con una memoria RAM de 16 GB.

Los parámetros utilizados para la ejecución del algoritmo genético fueron:

$$n: 20, p_c: 0.7, p_m: 0.1, T_{max}: 10 \text{ seg}$$

IV. RESULTADOS DEL CASO DE ESTUDIO

El caso de estudio presentado en la sección 2, fue resuelto utilizando la metaheurística del algoritmo genético, obteniendo los siguientes resultados:

A. Resultados

La situación actual es que el carro recolector de basura hace la siguiente ruta de recolección : 12, 23, 36, 67, 78, 85, 54, 43, [32], 27, [78], 81, la cual se aprecia en la Figura 3, obteniendo una distancia total recorrida de 1.367,2 metros y un costo total de 24.378.528,8 $m \cdot kg$.

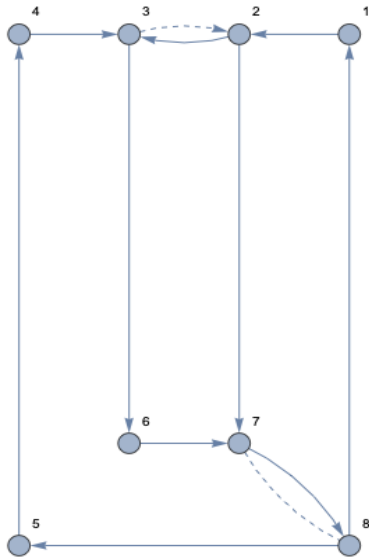


Fig. 3. Ruta actual realizada por el carro recolector

La distancia recorrida en la situación actual y la propuesta son las mismas, sin embargo, el costo de la situación actual es mayor, y esto se debe al orden en que se van recorriendo las aristas para la recolección de la basura.

Esto se explica porque en nuestra propuesta, se deja las calles con un mayor volumen de basura para el final, como se aprecia en la figura 5, siendo estas las 3 aristas: 7,8 y 2, cuya suma resulta en un peso a recolectar de 858 kg, que representa cerca del 54% de la basura total a recoger. Mientras que, en la situación actual, las últimas 3 calles por donde se recoge la basura son las aristas: 6, 4 y 2, resultando una suma de 600 kg por recoger, que representa el 38% del total de basura.

La solución propuesta, encontrada por el algoritmo genético fue la siguiente: 12, [23], 36, 67, [78], 87, 72, 23, 34, 45, 58, 81. Dando una distancia total recorrida de 1.367,2 metros y un costo total de 24.198.133,4 $m \cdot kg$.

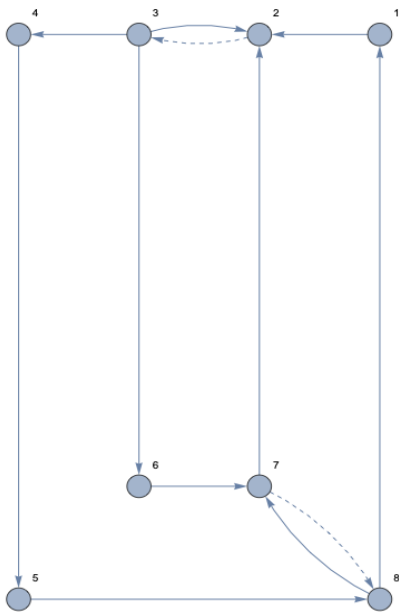


Fig. 4. Ruta propuesta a realizada por el carro recolector

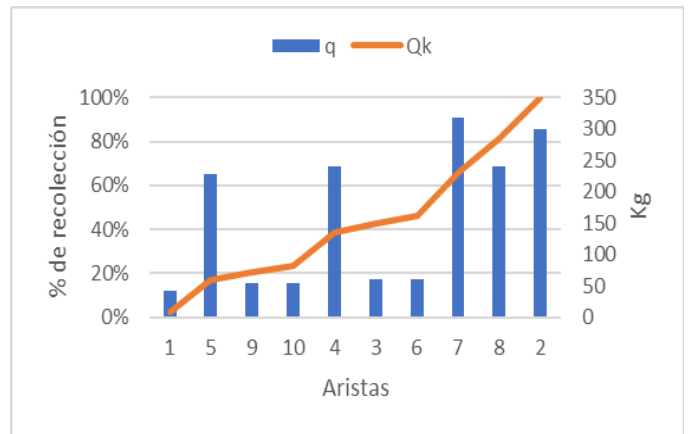


Fig. 5. Demanda por arista (q) vs. demanda recolectada (Qk) para la solución propuesta.

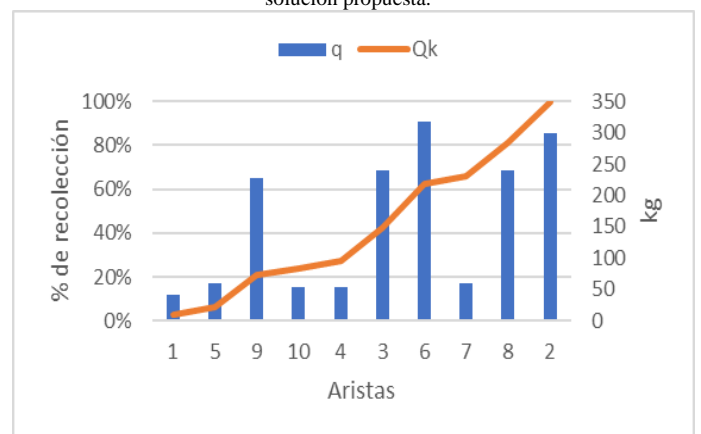


Fig. 6. Demanda por arista (q) vs. demanda recolectada (Qk) para la situación actual.

B. Discusión de los Resultados

La diferencia en el costo total es de 180.395,4 $m \cdot kg$, lo que representa cerca del 1% de disminución del costo total. Si se mantiene este porcentaje de ahorro, sobre el recorrido realizan los carros recolectores de toda la ciudad, todo el año, el ahorro tanto en lo económico, como en la contaminación ambiental será considerable.

Otra observación bastante interesante es que, en los problemas de recolección, si se debe pasar por una arista más de una vez, entonces se recoge la basura por esa arista la última vez que se pasa por ahí, por ejemplo, en la solución propuesta se pasa por la arista 23 sin recoger basura y luego se vuelve a pasar por esta arista, recogiendo la basura, que es lo contrario a lo que ocurre en los problemas de entrega, donde se pasa entregando los productos la primera vez que se pasa por la arista. Esto se debe, es más conveniente viajar la mayor cantidad de distancia con la menor cantidad de peso posible.

VI. CONCLUSIONES Y FUTURAS INVESTIGACIONES

Se realizó una variante al modelo matemático del CPP-LC, así como al algoritmo de programación dinámica, para que puedan ser utilizados en el caso de recolección. Se comprobó el ahorro generado al utilizar los modelos de optimización para la planificación de rutas, en un pequeño caso de estudio. También, se destaca la ida que, en los problemas de recolección, si se debe pasar por una arista más de una vez, lo óptimo es recoger lo solicitado la última vez que se pase por esta.

Finalmente, se propone modificar el problema para que la función de costos no solo considere la multiplicación de distancia por carga, sino que esté relacionada directamente con el consumo de combustible (lt. diésel / km.), y en consecuencia con kilogramos de CO₂ emitidos al ambiente, como ya se ha realizado en trabajos de optimización de rutas, pero basados en nodos. Esto implicaría que este trabajo podría ser extendido a la recolección y entrega de mercancías, personal entre otros, teniendo como objetivo la reducción de kg de CO₂ emitidos al ambiente considerando factores como la distancia, la carga, velocidad.

REFERENCIAS

[1] K. Sar, and P. Ghadimi, "A systematic literature review of the vehicle routing problem in reverse logistics operations". *Computers & Industrial Engineering*, 177, 2023.

[2] R. Moghdani, K. Salimifard, R. Demir and A. Benyettou, "The green vehicle routing problem: A systematic literature review". *Journal of Cleaner Production*, 279, 2021.

[3] M. Su, Z. Su, S. Cao, K.S. Park, and S.H. Bae, "Fuel Consumption Prediction and Optimization Model for Pure Car/Truck Transport Ships". *Journal of Marine Science and Engineering*, 11(6), 2023.

[4] I. Su and P. Nielsen, "Speed optimization algorithm with routing to minimize fuel consumption under time-dependent travel conditions". *Production and Manufacturing Research*, 8(1), 2020.

[5] Y. Niu, Z. Yang, P. Chen and J. Xiao, "A Hybrid Tabu Search Algorithm for a Real-World Open Vehicle Routing Problem Involving Fuel Consumption Constraints". *Complexity*, 2018.

[6] T. Bektaş and G. Laporte., "The pollution-routing problem". *Transportation Res. Part B* 45:1232–1250, 2011.

[7] Kumar, R., Kondapaneni, K., Dixit, V., Goswami, A., Thakur, L.S., Tiwari, M.K. (2016). Multi-objective modeling of production and pollution routing problem with time window: A self-learning particle swarm optimization approach. *Computers & Industrial Engineering*, 99:29-40.

[8] Ç. Koç, T. Bektaş, O. Jabali and G. Laporte, "The fleet size and mix pollution-routing problem". *Transportation Res. Part B*, 70:239–254, 2014.

[9] Ç. Koç, T. Bektaş, O. Jabali and G. Laporte G., "The impact of location, fleet composition and routing on emissions in urban logistics". *Transportation Res. Part B*, 84:81–102, 2016a

[10] Ç. Koç, T. Bektaş, O. Jabali and G. Laporte, "A comparison of three idling options in long-haul truck scheduling". *Transportation Res. Part B*, 93:631–647, 2016b.

[11] R. Kramer, A. Subramanian and T. Vidal, "A metaheuristic approach for the pollution-routing problem". *Eur. J. Oper. Res.*, 243:523–539, 2015.

[12] S. Dabia, E. Demir and T. Van Woensel, "An exact approach for a variant of the pollution-routing problem". *Transportation Sci.*, 51:607–628, 2017.

[13] R. Tolouei and H. Titheridge, "Vehicle mass as a determinant of fuel consumption and secondary safety performance. *Transportation Research Part D: Transport and Environment*, 14(6), 2009.

[14] M. Barth and K. Boriboonsomsin, "Real-world impacts of traffic congestion". *Transportation Res. Record: J. Transportation Res. Board*, 2058:163–171, 2008.

[15] J. Hua and K. Li-Shan, K., "Genetic algorithm for Chinese postman problems". *Wuhan University Journal of Natural Sciences*, 8(1):316–318, 2003.

[16] A. Shafagi and A. Haghani, A., "Generalized Maximum Benefit Multiple Chinese Postman Problem". *Transportation Research Part C: Emerging Technologies*, 55:261–272, 2015.

[17] A. Corberán, G. Erdoğan, G. Laporte, I. Plana and J.M. Sanchis, "The Chinese postman problem with load-dependent costs". *Transportation Science*, 52(2):370–385, 2018.

[18] I. Kara, B.Y. Kara and M.K. Yetis MK, "Energy Minimizing Vehicle Routing Problem". *Combinatorial Optimization and Applications*, 4616, 2007.

[19] L. Davis, "Genetic algorithms and the optimal ordering of chromosomes". *Journal of the Association for Computing Machinery*, 32(4):692–717, 1985.

[20] X. Yiyong, Z. Qiuhong, K. Ikou and X. Yuchun, "Development of a fuel consumption optimization model for the capacitated vehicle routing problem". *Computers & Operations Research*, 39:1419–1431. 2012.

[21] Floyd, R.W. Algorithm 97—Shortest path. *Comm. ACM* 5:345. Hertz A, Laporte G, Nanchen-Hugo P (1999) Improvement procedures for the undirected rural postman problem. *INFORMS J. Comput.* 11:53–62, 1992.