

# Ruteo de entrega de productos para una empresa de e-commerce

Verónica Victoria García De la Fuente, Emily Rebeca Méndez Cruz,  
Carolina Longoria Lozano, Juan Pablo Echeagaray González

Optimización determinista

MA2001B.200

Dr. Fernando Elizalde Ramírez

Dr. Jaime Eduardo Martínez Sánchez

**Resumen**—Una empresa de e-commerce ha solicitado que alumnos del Tecnológico de Monterrey diseñen un programa de rutas para la entrega de los productos adquiridos por sus clientes en su tienda *online*. Con las bases de datos recibidas se formuló un modelo de CVRP, este modelo fue embebido y optimizado para minimizar las distancias recorridas por los camiones de entrega. El producto final despliega un reporte de las rutas así como una visualización de estas.

**Index Terms**—CVRP, Programación Entera, Coppel, Rutas

## I. INTRODUCCIÓN

### I-A. Introducción sobre Coppel

Coppel es una empresa multiproducto que lleva años volviéndose la empresa que es el día de hoy. Fundada en 1941, Coppel empezó como una tienda de radios y relojes, y tras la Segunda Guerra Mundial tuvieron la idea de comenzar a vender a crédito. 80 años después, Coppel es una empresa reconocida nacional e internacionalmente, con un enfoque comercial. Crecieron, no sólo su propia marca y nombre, sino también la cantidad de marcas que se pueden encontrar en las tiendas [1]. De acuerdo a LinkedIn, Coppel tiene 14,682 colaboradores y 1,228 sucursales, en más de 450 ciudades del país. Una gran parte de su funcionamiento se da por medio de e-commerce, dando la opción a sus clientes de volverse “Clientes Digitales” [2].

### I-B. Introducción al E-Commerce

El comercio electrónico, e-commerce, consiste en realizar transacciones por medios digitales, e incluye distribución, venta, compra, marketing y suministro de información de productos o servicios [3]. Al buscar las empresas más exitosas, hay muy pocas que no integran el e-commerce como parte integral de su modelo. Le da a las empresas la posibilidad de crear una economía de plataforma [4] en la que pueden mantener un número relativamente bajo de empleados, cosa que es conveniente para las empresas. Es un mercado que no muestra indicaciones de frenar su crecimiento. De acuerdo a los datos de Stephanie Chevalier en Statista, del 2014 al 2021, las ventas de e-commerce han subido de 1,336 a 4,938 billones de dólares. Uno de los culpables de este crecimiento es lo cómodo que es para el cliente dar un click y recibir lo que quiere en la puerta de su casa. Por medio del e-comercio,

el cliente puede observar todos los productos disponibles, ordenar lo que quiera desde la comodidad de su hogar, y simplemente esperar el tiempo de llegada. Para que esto se lleve a cabo de la mejor manera, se debe diseñar una política de reparto eficiente.

### I-C. Trabajo relacionado

El *Travelling Salesman Problem*, conocido como *Problema del Agente Viajero* en español, es un problema que se encarga de buscar la ruta más corta y eficiente para llegar a un destino. Esta se basa en que existen múltiples opciones de llegar a un mismo destino, pero enfocándose en reducir costos de transporte, por lo que obtenemos de solución la ruta más corta [5].

Para ésta problemática se tiene contemplado que entre más destinos haya automáticamente el nivel de complejidad para el cálculo de la solución óptima aumenta. Por esta razón TSP es clasificado como un problema de NP-Hard, de acuerdo con la teoría de la complejidad computacional [5].

Para la solución de este problema existen diferentes métodos debido a la popularidad y complejidad que tiene, estos son algunos de ellos:

*I-C1. Vecino más cercano:* También conocido como KNN, este es uno de los algoritmos más simples para resolver este problema. Se trata de un método de aprendizaje automático básico aplicado para la logística de transporte, en este caso, el conductor -o agente viajero- siempre comienza su recorrido con el destino más cercano. Aunque para este método la solución encontrada no siempre logra la optimización [5].

*I-C2. Ramificación y atadura:* El método Branch and Bound es un algoritmo complejo diseñado para solucionar problemas que cuentan con variables de decisión enteras; para este método, el problema se divide en múltiples subproblemas donde cada uno de estos tiene varias soluciones posibles. Se debe destacar que al seleccionar una solución, esta puede afectar en las posibles soluciones de subproblemas posteriores, ya que como su nombre lo indica, actúa de manera ramificada. Este método se aplica como solución del problema del vendedor viajero [5].

*I-C3. Fuerza bruta:* Consiste en la enumeración sistemática de todas las posibles rutas de distribución, se calcula y compara todas las posibles soluciones revisando cuál o cuáles de ellas cumplen mejor a los objetivos de la empresa. Esto con el fin de establecer una única solución, que para el caso de este problema, sería la más corta y por ende la óptima [5].

## II. DEFINICIÓN DEL PROBLEMA

Sean los conjuntos:

$$\begin{aligned} i &\in \{1, \dots, n\} \\ j &\in \{1, \dots, n\} \\ k &\in \{1, \dots, V\} \end{aligned}$$

Sean los parámetros:

$V$  : Número de vehículos disponibles  
 $Q$  : Capacidad de carga máxima del vehículo en  $m^3$   
 $n$  : Número de clientes a visitar  
 $c_{ij}$  : Distancia geodésica entre el punto  $i$  y el punto  $j$   
 $d_j$  : Demanda en  $m^3$  del cliente  $j$

Sean las variables:

$x_{ijk}$  : Variable binaria que toma el valor de 1 si el arco del punto  $i$  al punto  $j$  forma parte de la ruta óptima recorrida por el vehículo  $k$   
 $x_{ijk} \in \{0, 1\}$

Modelamos el problema de optimización como:

$$\begin{aligned} \min \quad & z = \sum_{k=1}^V \sum_{j=1}^n \sum_{i=1}^n c_{ij} x_{ijk} \\ & \sum_{i=1}^n x_{ijk} = \sum_{i=1}^n x_{jik} \\ & \sum_{k=1}^V \sum_{i=1}^n x_{ijk} = 1 \\ & \sum_{j=2}^n x_{1jk} = 1 \\ \text{sujeto a:} \quad & \sum_{i=1}^n \sum_{j=2}^n d_j x_{ijk} \leq Q \\ & u_j - u_i \geq d_j - Q(1 - x_{ijk}) \\ & d_i \leq u_i \leq Q \\ & x_{ijk} \in \{0, 1\} \end{aligned}$$

Las restricciones anteriores hacen referencia a:

1. Número de veces que un camión entra por un punto debe de ser igual que el número de veces que las que sale
2. Cada camión solamente puede entrar 1 vez a un punto
3. Todos los vehículos salen del CEDIS
4. Límite a la capacidad de carga máxima del vehículo
5. Las últimas 2 restricciones eliminan los *subtours*

Para la problemática actual, estamos planeando realizar 192 entregas ( $n$ ), se dispondrán de 9 vehículos ( $V$ ), cada uno de ellos tendrá una capacidad máxima de carga de  $18 m^3$  ( $Q$ ). Los valores numéricos para las demandas y distancias son calculados dentro de la aplicación.

El heurístico que usamos en nuestro modelo es el costo asociado de ir de un punto de la ciudad a otro. No estamos modelando la distancia que hay entre cada punto como la distancia manejando, sino que estamos usando la distancia geodésica que hay entre cada punto, una gran desventaja del

uso de este heurístico es que puede generar rutas aparentemente óptimas pero que no son factibles cuando se agregan las posibles maneras de conectar los puntos de entrega. Hemos decidido seguir adelante con este heurístico ya que hay varios proyectos reconocidos que usan este heurístico para resolver problemas de ruteo de vehículos [6].

## III. PUNTOS DE ENTREGA

Haciendo uso de Python y de la librería *Folium* hemos visualizado una muestra de los últimos pedidos registrados en la base de datos, Fig 1; cada punto azul representa un cliente, y el punto negro dentro del mapa es donde se encuentra el CEDIS.

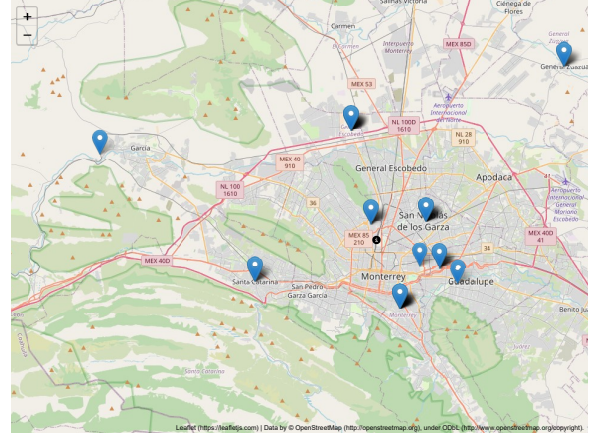


Figura 1. Puntos de entrega

Gracias a la nueva base de datos proporcionada por Coppel, hemos logrado incrementar la precisión del proceso de geocodificación de los domicilios.

## IV. METODOLOGÍA

Para la solución de esta problemática hemos usado el lenguaje de programación Python; para los procesos de lectura, limpieza, exploración y agregación de datos hemos usado la librería *pandas* en conjunto con *matplotlib* y *seaborn*, para la solución del problema de programación lineal usamos la librería *ortools* de Google, y para la visualización de los rutas generadas hemos usado *folium*. Una liga al código fuente que hemos desarrollado se encuentra en el apéndice A.

El dispositivo que usamos para desarrollar y correr la aplicación disponía de un CPU Intel i5 con 8 GB de memoria RAM, como sistema operativo se ha usado Ubuntu 20.04 LTS.

### IV-A. Limpieza de la base de datos

Hemos recibido 4 bases de datos. Una conteniendo las direcciones de los centros de distribución de Coppel en México, un archivo que contiene un listado de todos los productos que vende la empresa más algunas de sus propiedades, en particular nos interesa el volumen de cada artículo. Disponemos también de un archivo que contiene todos los tipos de vehículos que Coppel usa para realizar las entregas, y una base de datos

que contiene un listado de las entregas que Coppel debe de realizar.

#### IV-B. Procesamiento de domicilios

Para la geocodificación de los domicilios hemos usado la librería *Photon*, desde el paso anterior hemos creado una columna que contiene todos los datos de referencias de los domicilios de los compradores. El proceso de geocodificación toma un tiempo promedio de 15 minutos en la máquina que usamos, en promedio se logran codificar 25 % de los domicilios que recibe. Al final de este proceso tenemos las coordenadas asociadas a cada uno de los domicilios

#### IV-C. Cálculo de matriz de distancias

De las coordenadas de cada domicilio calculamos una matriz de distancia todos contra todos, la distancia a usar es la geodésica.

#### IV-D. Generación de rutas

Para la solución del problema hemos usado la librería *ortools* de Google. Hacemos uso de la matriz de distancias calculada en el paso anterior, un listado de cuánto volumen debe de ser entregado a cada uno de los clientes, un número de vehículos a mandar, y su capacidad máxima de carga en metros cúbicos. La solución generada consiste de un listado de las rutas a seguir por cada uno de los vehículos, el volumen total que llevarán, cuál fue la distancia total recorrida por todos los vehículos y cuánta carga llevaron.

#### IV-E. Visualización de rutas

La visualización de rutas se realizó con la API de la librería *folium*, con ella hemos generado un mapa de Monterrey en la que se visualizan las rutas que siguen todos los vehículos; el mapa también despliega información del domicilio al que se realiza la entrega, y el volumen total de los productos que debe de recibir.

### V. EXPERIMENTACIÓN Y RESULTADOS

La red con la que tratamos consta de 193 nodos, 192 referentes a los clientes, y 1 nodo más para el Centro de Distribución. El número de conexiones (aristas) dentro del grafo es  $\binom{193}{2} = 18528$

Después de correr la aplicación, se desplegará el navegador el conjunto de rutas de todos los vehículos, en la figura 2 podemos ver 2 de las rutas encontradas.

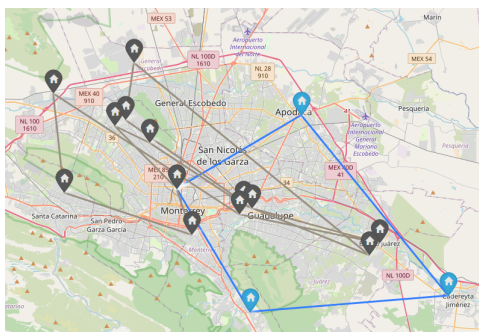


Figura 2. Ejemplo de rutas generadas

Como se puede apreciar en la tabla I, para esta simulación, la aplicación encontró un conjunto de rutas que minimizan la distancia que recorren los camiones en total hasta 483.49 km. En las mejores corridas la ruta más corta que realiza un camión es de 32 km, y la más larga de 97.02 km.

Número de ruta	Clientes a visitar	Distancia recorrida (km)	Consumo gasolina (l)	Volumen libre (m <sup>3</sup> )
1	30	97.02	10.19	5.72
2	16	57.26	6.01	3.09
3	4	32.00	3.36	0.13
4	19	92.00	9.66	4.07
5	28	44.40	4.66	0
6	23	37.49	3.94	0
7	26	40.04	4.20	0
8	29	42.88	4.50	1.10
9	26	40.40	4.24	3.44

Cuadro I

RUTAS GENERADAS POR EL PROGRAMA

### VI. DISCUSIÓN

A pesar de que el producto que hemos desarrollado ya es funcional, hemos encontrado algunas áreas de oportunidad que de ser resueltas incrementarían aún más el valor de la aplicación:

#### VI-A. Geo-codificación de los domicilios

Actualmente la aplicación hace uso de una librería de libre uso llamada *Photon* para geo-codificar los domicilios disponibles en la base de datos; consideramos que la adquisición de una licencia comercial podría aumentar aún más el número de domicilios que son geo-codificados con éxito.

#### VI-B. Cálculo de distancia entre 2 puntos

La aplicación utiliza como métrica de distancia la distancia geodésica, que está relativamente lejos de ser una aproximación rigurosa para determinar la distancia entre 2 puntos. Sugerimos también que se adquiriera una licencia comercial que calcule esta distancia. Su uso le daría una mayor validez a los resultados que encontramos.

### VII. ODS

Igual que encontrar la ruta más eficiente para Coppel, buscamos tener un impacto social por esta razón reconocemos la importancia que tiene que la innovación trascienda los objetivos personales, así como a los objetivos colectivos. Por eso mismo, tenemos la intención de tener un impacto positivo en los Objetivos de Desarrollo Sostenible propuestos por la ONU. Una de las ODS implementadas en el proyecto es el objetivo número 9, Industria, Innovación e Infraestructura, ya que los valores de esta ODS se alinean perfectamente con los de nuestro proyecto. Este objetivo tiene como enfoque la construcción de infraestructuras resilientes en forma de rutas adaptables que se basan en las necesidades, la industrialización sostenible en forma de la minimización de kilómetros recorridos y por lo tanto energía ocupada, e innovación de un sistema ya existente. Otro de los objetivos en los que impactamos positivamente es el número 11, Ciudades y Comunidades Sustentables, ya que reconocemos que el mundo está cada vez más urbanizado, y la sostenibilidad no se puede quedar atrás. El e-commerce está cada vez más presente en las vidas de la gente, y al minimizar los efectos negativos de esta industria es el mejor paso que podemos tomar para un futuro verde.

## VIII. CONCLUSIONES

El ruteo de vehículos sigue siendo uno de los retos más relevantes para la Investigación de Operaciones, el diseño de algoritmos eficientes que logren encontrar rutas de entrega óptimas con cantidades cada vez más grandes de puntos de entrega es un imperativo para cualquier departamento de logística.

Dado el gran número de puntos de entrega y de la enorme cantidad de posibles rutas a explorar, el uso de algoritmos heurísticos que logren encontrar rutas sub-óptimas es una alternativa a considerar para cualquier empresa; el que estos métodos puedan encontrar soluciones buenas en tiempos razonables hace que sean mucho más sencillos de implementar en producción.

El proceso de determinar cuál es el mejor sub-conjunto de rutas tendrá tanto repercusiones económicas para la empresa como ecológicas para el ambiente, el proyecto que hemos desarrollado busca reducir la mayor cantidad de distancia recorrida posible por parte de las rutas, lo cual trae consigo una disminución del gasto realizado en el transporte de sus productos, mientras que se reducen en general las emisiones de carbono de sus vehículos.

## APÉNDICE A CÓDIGO FUENTE

El código generado puede ser consultado aquí

## REFERENCIAS

- [1] “Acerca de nosotros,” 2022. [Online]. Available: <https://www.coppel.com/acerca-de-nosotros>
- [2] “Coppel - linkedin,” 2022. [Online]. Available: <https://www.linkedin.com/company/coppel-sa-de-cv>
- [3] “Qué es e-commerce o comercio electrónico,” Abril 2014. [Online]. Available: <https://www.visa.com.co/dirija-su-negocio/pymes/notas-y-recursos/tecnologia/que-es-ecommerce-o-comercio-electronico.html>
- [4] B. Dekker and M. Okano-Heijmans, “Europe’s digital decade?: Navigating the global battle for digital supremacy,” Clingendael Institute, Tech. Rep., 2020. [Online]. Available: <http://www.jstor.org/stable/resrep26543.6>
- [5] M. M. Flood, “The traveling-salesman problem,” *Operations Research*, vol. 4, no. 1, pp. 61–75, 1956. [Online]. Available: <http://0-www-jstor-org.biblioteca-ils.tec.mx/stable/167517>
- [6] G. Erdoğan, “An open source spreadsheet solver for vehicle routing problems,” *Computers & Operations Research*, vol. 84, pp. 62–72, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0305054817300552>