

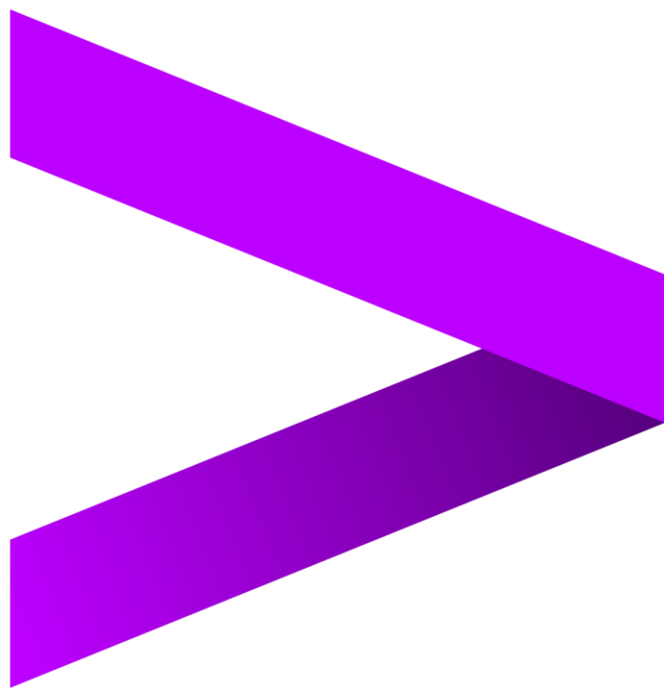


RETO

ACCENTURE

Algoritmos de optimización con vehículo eléctrico

Octubre 2022



1. EL RETO

Diez años han pasado desde que el fabricante Tesla comenzara a entregar las primeras unidades del Model S, la primera berlina eléctrica de lujo del mundo. Desde entonces, el sector del automóvil está viviendo una revolución para dejar en el olvido los motores de combustión. Una revolución que es imparable, a pesar de que sigue teniendo muchos obstáculos.

El cambio climático es real y es por ello que grandes firmas están destinando importantes sumas de dinero a las inversiones durante el periodo 2020-2025 para el desarrollo y producción de vehículos eléctricos.

Desde Accenture participamos activamente en ser también parte de la solución. Por eso proponemos un cambio radical en la estrategia de nuestros clientes del sector logístico: asegurar que el 100% de sus vehículos sean furgonetas que funcionen con baterías eléctricas, con todos los retos que ello conlleva.

2. INTRODUCCIÓN A LOS ALGORITMOS DE OPTIMIZACIÓN

Un problema de optimización consiste en maximizar o minimizar una función real eligiendo sistemáticamente valores de entrada (tomados de un conjunto permitido) y computando el valor de la función. La generalización de la teoría de la optimización y técnicas para otras formulaciones comprende un área grande de las matemáticas aplicadas. Los puntos fundamentales que debe cumplir un proceso para ser considerado un verdadero algoritmo son los siguientes:

- Debe ser preciso: es por ello que se debe indicar el orden exacto de ejecución de cada paso implicado en el proceso.
- Debe estar perfectamente definido. Esto significa que en el caso de ejecutarse el mismo más de dos veces, siempre se debe obtener el mismo resultado independientemente de la cantidad de veces que se siga.
- Debe ser finito, es decir que el algoritmo debe culminar en algún momento de su ejecución, expresado en otras palabras, debe tener un número de pasos bien determinados hasta concluir con su tarea.
- Debe ser legible. Esto significa que el texto que describe debe ser claro y conciso, de una manera tal que permita su comprensión inmediata, es decir sin procedimientos rebuscados o poco claros.
- Debe estar definido en tres partes fundamentales, las cuales son: Entrada, Proceso y Salida.

Un ejemplo es el algoritmo de Dijkstra para la determinación del camino más corto, publicado por primera vez en 1959. La idea de dicho algoritmo es explorar todos los caminos más cortos que parten del vértice inicial y llevan a todos los demás vértices en una red de nodos interconectados. Este algoritmo ha sido muy importante en el campo de la telemática para el encaminamiento de paquetes en la red.

En el caso de la optimización de rutas de transporte el objetivo es encontrar la ruta más eficiente. No se trata únicamente de encontrar el camino más corto, otros factores relevantes (como las paradas requeridas, peajes o los puntos de recarga) deberían tenerse en cuenta para encontrar una solución óptima.

3. MECÁNICA DEL PROBLEMA

3.1 Planteamiento

Una empresa de paquetería con repartos dentro de la Zona30 de Madrid quiere realizar la entrega de los paquetes con nuevas furgonetas eléctricas optimizando los tiempos de entrega al máximo. Se han

adquirido un total de 4 furgonetas del mismo modelo con una capacidad máxima de 500 kg, que pueden recorrer 300 km cuando circulan sin carga y 210 km cuando el vehículo circula al máximo de su capacidad. La relación entre el peso transportado y la distancia que puede recorrer la furgoneta es lineal.

Hay 3 centros logísticos (A, B y C), localizados dentro de la Zona30 de Madrid. En el centro logístico A hay 2 cargadores eléctricos: 1 cargador rápido y 1 cargador lento. En los centros logísticos B y C se tienen 2 cargadores eléctricos lentos. Se ha comprobado que una de las furgonetas eléctricas tarda 1 hora en recargar sus baterías en un cargador rápido, mientras que en un cargador lento tarda 3 horas (si se recargan las baterías un tiempo inferior, la carga será proporcional; si se hacen cargas intermedias, la carga también es proporcional).

Los centros logísticos se encuentran en las siguientes coordenadas:

- **Centro logístico A** (id_centro=1): 40.406215, -3.690616
- **Centro logístico B** (id_centro=2): 40.473556, -3.681687
- **Centro logístico C** (id_centro=3): 40.434069, -3.718529

El tiempo de entrega una vez el repartidor ha llegado a su destino depende del tamaño del paquete: un paquete con un peso de hasta 5kg se entrega en 5 minutos, un paquete de hasta 20kg en 7 minutos y un paquete con un peso mayor de 20kg en 15 minutos.

El horario de reparto comienza a las 8h y termina a las 19h, debiendo comenzar en uno de los centros y dejando las furgonetas en uno de los centros logísticos no más tarde de la hora de cierre. La velocidad media de las furgonetas durante el reparto es de 50km/h.

NOTA: Para simplificar el problema se considera la distancia en línea recta entre las coordenadas, sabiendo que la distancia del **Centro logístico A** al **Centro logístico B** es: 7.54 km

3.2 Input del script

Se facilita la localización de los centros logísticos, una lista de los paquetes que hay en cada centro, con sus pesos y dónde han de entregarse (la distribución de paquetes en cada furgoneta queda a cargo del equipo). Se proporciona aparte las coordenadas exactas de cada centro logístico y las coordenadas de las direcciones de entrega de los paquetes. Los centros logísticos son el punto de partida y llegada de las furgonetas.

El formato de los datos es el siguiente:

Cabecera fichero 1:

id_centro;coord_x;coord_y

Donde:

- **id_centro** es el id del centro logístico donde se encuentra almacenado el paquete.
- **coord_x** es la longitud del centro logístico.
- **coord_y** es la latitud del centro logístico.

Cabecera fichero 2:

id_centro;id_pos;id_paquete;peso

Donde:

- **id_centro** es el id del centro logístico donde se encuentra almacenado el paquete.
- **id_pos** es el id de la posición en la que el paquete ha de ser entregado.
- **id_paquete** es el id del paquete.

- **peso** es el peso del paquete en kg.

Cabecera fichero 3:

id_pos; coord_x; coord_y

Donde:

- **id_pos** es id_pos en el fichero 2.
- **coord_x** es la longitud.
- **coord_y** es la latitud.

Por ejemplo, para la posición: Castellana 85: 40.4483737;-3.6941491 que corresponde a 40°26'54.2"N 3°41'38.9"W.

3.3 Ejecución y características del script

El algoritmo deberá ser desarrollado en lenguaje de programación Python (versión 3.8), con extensión “.py” y cuyo nombre será SCRIPT_NombreEquipo (sin espacios). Este script SCRIPT_NombreEquipo.py deberá ser autoinstalable en caso de disponer de alguna dependencia, pues deberá contener todo lo necesario para que el programa funcione en cualquier ordenador con conexión a internet. Deberán de indicarse las librerías de Python utilizadas al igual que sus versiones en un fichero *requirements.txt*, para evitar problemas de incompatibilidad. El código deberá estar debidamente comentado para facilitar su comprensión.

Las características del ordenador son las siguientes:

- OS Name Microsoft Windows 10 Enterprise
- System: Dell Inc. Precision 3520. x64-based PC
- Processor Intel(R) Core(TM) i7-7820HQ CPU @ 2.90GHz, 2901 Mhz, 4 Core(s), 8 Logical Processor(s)
- Installed Physical Memory (RAM) 16,0 GB

3.4 Output del script

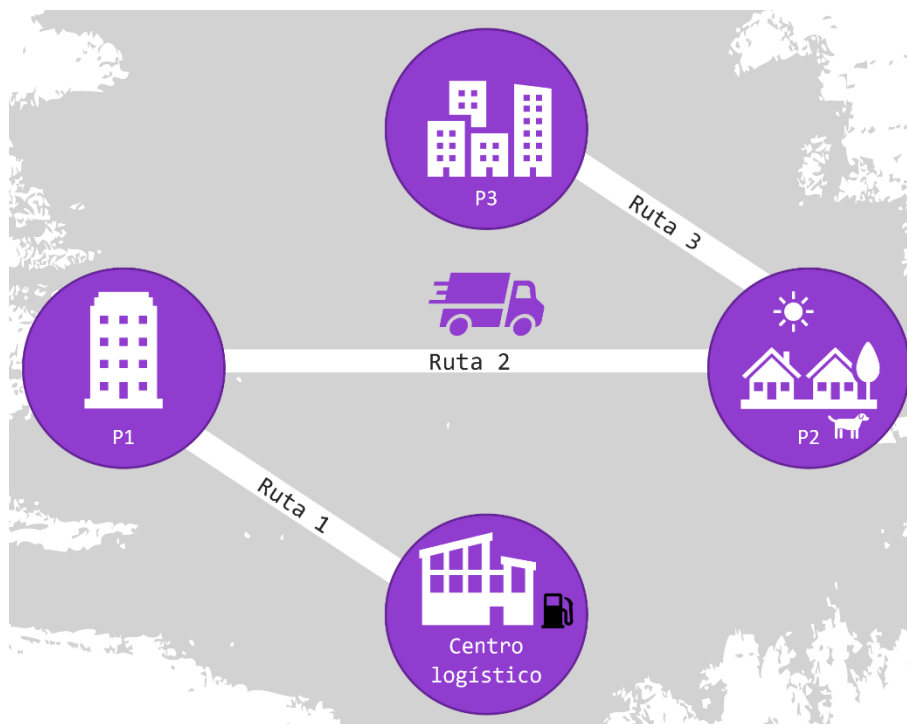
Se espera que la salida del script sea un archivo csv con la siguiente cabecera:

id_furgo;id_pos;hora_llegada;hora_salida;ids_paquetes

Donde:

- **id_furgo** es el id de la furgoneta. Puede tomar los valores {1,2,3,4}
- **id_pos** es el id_pos dado en el fichero de entrada.
- **hora_llegada** es la hora de llegada a la posición id_pos, en formato hh:mm:ss
- **hora_salida** es la hora de salida de la posición id_pos, en formato hh:mm:ss
- **ids_paquetes** es la lista de id_paquete entregados en la posición id_pos, separados por coma ‘,’.

A continuación, proporcionamos a modo de ejemplo una de nuestras antiguas rutas de reparto:



```
id_furgo;id_pos;hora_llegada;hora_salida;ids_paquetes
1;P1;08:05:00;08:10:00;1,4
1;P2;08:20:00;08:35:00;2
1;P3;09:05:00;09:12:00;3,7
```

3.5 Entregables

Cada equipo deberá hacer entrega de una memoria explicativa de lo realizado y un archivo zip de hasta 1GB con el nombre del equipo en el que se incluyan:

- el fichero SCRIPT_NombreEquipo.py
- un requirements.txt
- todos los archivos que el script necesite y/o documentos que el equipo considere necesarios.

El fichero deberá poder ejecutarse desde un terminal con el siguiente comando:

```
python SCRIPT_NombreEquipo.py <ruta_input_csv> <ruta_output_csv>
```

La memoria podrá ser una presentación o documento de texto. Deberá contener, al menos:

- Metodología y estrategia desarrollada para la resolución del problema
- Principales dificultades encontradas
- Conocimientos adquiridos
- Fuentes principales de consulta (webs de consulta, libros...)

3.6 Criterios de valoración

La valoración se realizará a través de los siguientes elementos:

- Resultados obtenidos para 10 casos diferentes.
- Se valora un 60% la media de los resultados obtenidos entre los 10 casos. Puntúa en un intervalo 0-10 puntos:

- 10 puntos a la mejor solución encontrada (será aquella que consiga entregar todos los paquetes en menor tiempo).
- 0 puntos para las soluciones que superen el máximo de 11h de reparto.
- El resto de las notas se calcularán de forma proporcional respecto a la mejor solución.
- Se valora un 15 % la memoria sobre el trabajo realizado, valorando la claridad, calidad, innovación, originalidad y metodología propuestas entre otros. Puntúa en un intervalo 0-10 puntos.
- Se valora un 20% el tiempo de ejecución medio del algoritmo hasta llegar a una solución. Puntúa en un intervalo 0-10 puntos:
 - 10 puntos al menor tiempo de ejecución del algoritmo.
 - 0 puntos para la solución que tarde más de 15min (para un caso).
 - El resto de las notas se calcularán de forma proporcional respecto a la mejor solución.
- Se valora un 5% la claridad y calidad del código entregado. Comentarios en el código.

El jurado se reserva el derecho a desestimar las soluciones presentadas de aquellos participantes que no cumplan los requisitos exigidos en este documento.