

# Técnicas metaheurísticas basadas en file systems distribuidos para problemas de ruteo de vehículos que involucran grandes volúmenes de datos

Ricardo Di Pasquale<sup>1</sup>      Javier Marengo<sup>2</sup>

<sup>1</sup> Universidad Católica Argentina, Argentina  
rdipasquale@uca.edu.ar

<sup>2</sup> Instituto de Ciencias, Universidad Nacional de General Sarmiento, Argentina  
jmarengo@dc.uba.ar

En este trabajo estamos interesados en la resolución de problemas de optimización combinatoria que involucran grandes volúmenes de datos de entrada. En particular, trabajamos en una variante del problema clásico de ruteo de vehículos con ventanas de tiempo (VRPTW) sobre datos reales de tránsito, con el objetivo de obtener soluciones robustas. La característica más importante de esta variante del VRPTW es que contamos con una cantidad extremadamente grande de datos de entrada referidos a velocidades de desplazamiento de los vehículos por cada punto de la ciudad a lo largo del tiempo, y en este contexto no es aceptable simplificar estos datos para obtener una matriz de tiempos de viaje aproximados.

Los datos de entrada de este problema incluyen un grafo dirigido  $G = (V, A)$  que representa el mapa de la ciudad, un vértice especial  $d \in V$  que representa el depósito de origen de los vehículos, vértices  $v_1, \dots, v_k \in V$  que representan a los clientes a visitar, y una cantidad  $c \in \mathbb{Z}_+$  que especifica la cantidad máxima de clientes que puede visitar cada vehículo. Se particiona el intervalo  $[0, 24]$  – que representa las 24 horas del día – en intervalos  $I_1, \dots, I_t$  y, para cada arco  $e \in A$  y cada intervalo  $j \in \{1, \dots, t\}$ , se tiene el tiempo de viaje  $t_{ej} \in \mathbb{R}_+$  de atravesar el arco  $e$  en el intervalo  $I_j$  del día. Estos datos se obtienen a partir de valores reales de flujo vehicular. Finalmente, tenemos una ventana de tiempo  $(e_v, l_v) \in [0, 24]^2$  y un tiempo de servicio  $q_v \in \mathbb{R}_+$  para cada cliente  $v \in V$  que determinan en qué momento del día puede ser visitado y cuánto tiempo se tarda en despachar la visita, respectivamente. El problema consiste en hallar una ruta para cada vehículo que comience y termine en el depósito  $d$  y que pase por no más de  $c$  clientes, de modo tal que cada cliente sea visitado dentro de su ventana de tiempo por la ruta de exactamente un vehículo y que el tiempo total de viaje sea mínimo. Para evaluar el tiempo de viaje de una ruta, se toma en cuenta el horario de salida de los vehículos y los tiempos estimados de viaje de cada arco en cada momento del día.

Un desafío importante que presenta este problema está dado por el almacenamiento de los datos de entrada. Dependiendo de la discretización seleccionada, los datos de algunas instancias sobre la Ciudad de Buenos Aires pueden requerir

de varias decenas de GB de almacenamiento. Esto plantea restricciones a la tecnología a usar en la implementación de los algoritmos, dado que no es factible ubicar estos datos en memoria RAM y es muy lento recurrir al disco para consultarlos cada vez que se debe evaluar la función objetivo de una solución.

Para atacar estos problemas, presentamos una plataforma basada en el file system distribuido HDFS de Apache Hadoop y la tecnología Apache Spark, que permite manejar estos volúmenes de datos de manera adecuada. Sobre esta plataforma, implementamos un framework que da soporte a la ejecución de algoritmos genéticos distribuidos. En particular, desarrollamos un algoritmo genético para este problema por medio de operaciones Spark sobre estructuras RDD (*resilient distributed data*), distribuyendo así la carga de trabajo sobre varias computadoras. Presentamos experimentos computacionales que muestran que este enfoque es adecuado incluso para el problema clásico de ruteo de vehículos, y mostramos resultados sobre la variante del VRPTW considerada en este trabajo. Finalmente, presentamos algunas consideraciones sobre cómo se podría usar este enfoque para trabajar directamente con datos obtenidos desde dispositivos GPS (que implican volúmenes de almacenamiento mayores que los mencionados arriba) en lugar de considerar los datos consolidados de tiempo de viaje por arco y momento del día.