

Universidad Técnica Federico Santa María  
Departamento de Informática

# Instrucciones Entregable #2 G-VRP

Inteligencia Artificial

*Germán Treimun C.*  
*german.treimun@alumnos.usm.cl*  
Octubre 2020

## 1. Problema, objetivos y restricciones

El problema a resolver es el Green-Vehicle Routing Problem propuesto en [1]. El objetivo, por ende, es minimizar la distancia total recorrida por los vehículos al servir a cada uno de los consumidores. Las restricciones de problema son las mismas que en el paper original, ignorando la restricción de cantidad de vehículos máximo  $m$  tal como se ha hecho en la literatura.

## 2. Instancias de prueba

Las instancias de prueba a utilizar son un subconjunto de las propuestas en [1]. Se encuentran [en el siguiente enlace](#) (Moodle).

### 2.1. Formato de instancias de prueba

El contenido de cada instancia esta compuesto por dos secciones, en la primera sección se tienen 4 columnas: **ID**, **Type**, **Longitude**, **Latitude**, donde cada fila corresponde a la posición de los distintos puntos. Type en cada fila indica a que corresponde cada punto:

- Type **d** corresponde al almacén.
- Type **f** corresponde las estaciones de combustible.
- Type **c** corresponde a los consumidores.

Notar que las ubicaciones vienen dadas por longitud y latitud. Para trabajar con las distancias entre distintos puntos es necesario utilizar la **fórmula de Harvesine**. Luego, encontramos la segunda sección posterior a una línea vacía. Aquí se encuentran los parámetros de velocidad, consumo de combustible, capacidad, etc. En el caso de ustedes no deben considerar la cantidad de vehículos  $m$ .

Ejemplo de archivo de instancia:

ID	Type	Longitude	Latitude
D	d	-77.49439265	37.60851245
D2	f	-77.49439265	37.60851245
BD12	f	-76.338677	36.796046
BD20	f	-77.08760885	39.45787498
C1	c	-76.2725742	36.1481181
⋮	⋮	⋮	⋮
C20	c	-78.4659577	37.2447940

```
Q Vehicle fuel tank capacity /60/  
r fuel consumption rate /0.2/  
TL TourLength /11/  
v average Velocity /40/  
m numVeh /3/
```

Notar que en el ejemplo, existen D, y D2, que corresponden al mismo punto en coordenadas, pero aparece dos veces, dado que cuenta como almacén y estación de combustible a la vez.

### 3. Especificaciones del input del programa

El programa debe recibir como input (en línea de comandos) los siguientes parámetros:

1. La ruta del archivo de instancia, por ejemplo: `instances/20c3sU1.txt`
2. La máxima cantidad de iteraciones permitida en una ejecución de su programa (no aplica para técnicas completas).

Ejemplo de entrada: `./G_VRP-Method instances/20c3sU1.txt 1000`, donde `G_VRP-Method` es el nombre de su programa, `instances/20c3sU1.txt` la ruta completa de la instancia a resolver y 1000 la cantidad de iteraciones.

Es preciso mencionar que se pueden añadir otros parámetros adicionales en la entrada, si lo hace, **debe incorporarlos junto con una descripción en el archivo README.txt**, estableciendo al menos un ejemplo de cómo ejecutar su programa con estos parámetros.

### 4. Especificaciones del output del programa

Su programa debe mostrar por pantalla:

1. El valor de la función objetivo de la mejor solución (distancia del recorrido).
2. Recorrido realizado en la mejor solución (D -C1- C5 - BD1 - D - C3 ...)
3. Cantidad de iteraciones realizadas en el caso de técnica completa.
4. Iteración en la que se alcanzó la mejor solución en el caso de técnica incompleta.
5. Tiempo de ejecución.
6. Adicionalmente, si usted desea mostrar otros datos que sean importantes (y que puedan visualizarse fácilmente) puede mostrarlos al final del output, **especificando en el archivo README.txt qué quieren decir y qué sentido tiene añadirlos a la salida del programa**.

### 5. Recordatorios y recomendaciones

1. Recuerde que su programa debe ser realizado en C/C++ con un Makefile para poder compilarlo.
2. Recuerde que debe COMENTAR su código.
3. Recuerde incluir un README donde se especifique:
  - a) Instrucciones de ejecución
  - b) Parámetros necesarios para ejecutar el algoritmo, con una explicación de cada uno (objetivo del parámetro, significado, etc)
  - c) Ejemplo(s) de comando(s) para ejecutar su algoritmo
  - d) En caso de requerirlo, el detalle de datos adicionales incluidos en el output
  - e) Cualquier supuesto que considere importante.

4. Para realizar mediciones de tiempo de su programa hágalo usando “`time ./reco con- fig_W_H`”, donde el tiempo de ejecución total de su programa será “`user + sys`”. Se recomienda usar un archivo `bash` para automatizar la ejecución de su programa con distintos parámetros usando `time` y guardar los resultados en un archivo.
5. Si tiene problemas con algunas instancias específicas producto de, por ejemplo, cálculos muy grandes o demasiada memoria ram utilizada que saturó su pc, usted debería hacer lo siguiente:
  - a) Explicar en el informe cuáles instancias fueron las que originaron el problema
  - b) Explicar cuál fue el problema que usted detectó: tamaño de la instancia muy grande, error aleatorio, etc.

## Referencias

- [1] S. Erdogan and Elise Miller-Hooks. A green vehicle routing problem. *Transportation Research Part E-logistics and Transportation Review*, 48:100–114, 2012.