



# Presentación técnicas multiarranque, GRASP.

Antonio Jose Morano Moriña

Modelos Bioinspirados y Heurísticas de Búsqueda

# 1-Problema a resolver

- En el trabajo seleccionado, se resuelve el **Problema de Enrutamiento de Vehículos con Recogida y Entrega Simultánea (VRPSPD)** pertenece a la familia de problemas de enrutamiento de vehículos con restricciones de capacidad, donde tenemos:
  - Un **depósito central** y un **conjunto de clientes con demandas conocidas de recogida y entrega** de mercancía. El objetivo es encontrar un sistema de rutas que permita satisfacer la demanda de todos los clientes al menor costo posible. Teniendo en cuenta las siguientes restricciones:
    - Todos los recorridos se inician y terminan en el depósito.
    - Se dispone de una flota de vehículos de diferentes capacidades.
    - Al diseñar las rutas, la carga del vehículo es una mezcla entre la mercancía recogida en los clientes ya visitados y la que todavía falta por entregar, y en ningún punto del recorrido la carga puede exceder la capacidad del vehículo.

## 2-Representación del problema

➡ Un depósito central (**0**)

➡ Un conjunto de clientes  $I = \{1, 2, \dots, n\}$

- ➡ Las demandas de entrega (**di**) y de recogida (**pi**).
- ➡ El coste de viaje entre cada par de clientes (**cij**)

➡ Una ruta es una secuencia ordenada de clientes asociadas a un vehículo en específico, se denota por:

$$[R, k] = [(r_0 = 0, r_1, \dots, r_l, r_{l+1} = 0), k].$$

Donde el costo de viaje es la suma de los costos de todos los arcos presentes en R.

➡ Una solución es un conjunto de rutas denotado por:

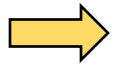
$$S = \{[R_1, k_1], [R_2, k_2], \dots, [R_m, k_m]\}.$$

El costo de S es la suma de los costos de todas las rutas, se denota por C (S). El objetivo es:

$$\min C (S)$$

## 2-Representacion del problema

Al asumir una flota limitada, encontrar una solución factible para este problema es NP – duro, de ahí que se tenga que crear una condición de factibilidad.



Una ruta  $[R, k]$  es débil factible si el total de mercancía para entregar, así como el total de mercancía por recoger no exceden la capacidad del vehículo  $k$ , es decir, si se verifican simultáneamente las expresiones:

$$\sum_{i=1}^l d_{r_i} \leq Q^k$$

$$\sum_{i=1}^l p_{r_i} \leq Q^k$$

Si se cumplen ambas existe un ordenamiento de los clientes que garantizan la condición de factibilidad

# 3-Características de las tecnicas GRASP.

## Búsqueda local

Será realizada por los algoritmos **VND** , **RVND** y **Enfriamiento simulado**.

Tanto **VND** como **RVND** son ideales para la explotación en zonas cercanas al óptimo, mientras que **enfriamiento simulado** es ideal para escapar de los óptimos locales permitiendo una mayor exploración del espacio de búsqueda.

Para el funcionamiento de estos algoritmos, se usan 6 funciones de vecindad que son:

### Para Inter Ruta

1. Cruzar dos rutas diferentes
2. Mover un cliente de una ruta hacia otra diferente
3. Intercambiar dos clientes de dos rutas diferentes

### Para Intra Ruta

1. Mover un cliente dentro de una misma ruta
2. Intercambiar dos clientes de de la misma ruta
3. Eliminar cruzamientos de 2 aristas (2-Opt)

# 3- Características de las técnicas GRASP

## Estrategia de solución

Como función objetivo se emplea la expresión:

$$\min \sum_{s \in S} C([R_s, k_s]) + \mu P([R_s, k_s])$$

Donde  $\mu$  es un factor de peso para penalidad. En la medida que se aumenta el valor de  $\mu$  se favorece la obtención de soluciones factibles.

Y siendo el valor de  $P([R, k])$  sacado de la formula:

$$P([R, k]) = \sum_{j=0}^l \max \{0, L([R, k], j) - Q^k\}$$

Y a su vez el de  $L([R, k], j)$  de:

$$L([R, k], j) = \sum_{i=0}^j p_{r_i} + \sum_{i=j+1}^l d_{r_i}$$

# 3- Características de las técnicas GRASP

La característica fundamental de GRASP es que es una metaheurística de múltiples reinicios que en cada paso realiza una búsqueda local que toma como punto de partida una solución greedy-aleatoria.

## Creacion de la solucion Greedy-Aleatoria

En cada iteración se construye una lista restringida de candidatos (RCL) que contiene los clientes que están relativamente cerca del último cliente adicionado a la solución.

Un cliente se incluye en la RCL si al añadirlo en la ruta actual no se afecta la factibilidad débil y tal que su costo de inserción se encuentre en el intervalo  $[c_{\min}, c_{\min} + \alpha (c_{\max} - c_{\min})]$  donde  $c_{\min}$ ,  $c_{\max}$  son, respectivamente, el menor y el mayor costo de inserción.

La calidad de la solución greedy – aleatoria depende del valor del parámetro  $\alpha$  seleccionado, si  $\alpha = 0$  la solución es completamente greedy y si  $\alpha = 1$  la solución es completamente aleatoria

# 3-Características de las técnicas GRASP.

Algoritmo 1 – GRASP clásico

Consta de dos etapas: generación de las soluciones iniciales y búsqueda local.

---

```
1: Construir un conjunto de soluciones golosas-aleatorias  $\rightarrow P_0, O^* \leftarrow \emptyset$ 
2: for all  $S_t \in P_0$  do
3:   Realizar una búsqueda local a partir de  $S_t \rightarrow O_t$ 
4:   if  $C(O_t) < C(O^*)$  then
5:      $O_t \rightarrow O^*$ 
6:   end if
7: end for
8: return  $O^*$ 
```



# 3-Características de las técnicas GRASP.

Algoritmo 2 – GRASP Binivel

Se separa la búsqueda local en dos etapas

---

```
1: Construir un conjunto de soluciones golosas-aleatorias  $\rightarrow P_0, O^* \leftarrow \emptyset$ 
2: for all  $S_t \in P_0$  do
3:   Realizar una búsqueda local considerando solamente movimientos Inter Ruta  $S_t \rightarrow Z_t$ 
4:   Realizar una búsqueda local considerando solamente movimientos Intra Ruta  $Z_t \rightarrow O_t$ 
5:   if  $C(O_t) < C(O^*)$  then
6:      $O_t \rightarrow O^*$ 
7:   end if
8: end for
9: return  $O^*$ 
```

# 3-Características de las técnicas GRASP.

- Algoritmo 3 – GRASP con perturbacion

```
1: Construir un conjunto de soluciones golosas-aleatorias  $\rightarrow P_0, O^* \leftarrow \emptyset$ 
2: for all  $S_t \in P_0$  do
3:    $O_{t0} \leftarrow S_t$ 
4:   for  $i = 1$  to  $N$  do
5:     Seleccionar aleatoriamente un mecanismo de ruptura  $u_i$ 
6:     Modificar el óptimo local previamente obtenido empleando  $u_i \rightarrow S_{ti}$ 
7:     Realizar una búsqueda local a partir de  $S_{ti} \rightarrow O_{ti}$ 
8:     if  $C(O_{ti}) < C(O^*)$  then
9:        $O_{ti} \rightarrow O^*$ 
10:    end if
11:  end for
12: end for
13: return  $O^*$ 
```

La desventaja fundamental de los Algoritmos 1 y 2 es que no se aprovechan, en las iteraciones sucesivas, información del óptimo local previamente obtenido.

En este trabajo se emplean dos mecanismos de perturbación: intercambiar  $m$  clientes de la ruta  $a$  con  $n$  clientes de la ruta  $b$  y eliminar 20 por ciento de los clientes y reinsertarlos nuevamente en posiciones aleatorias

# 3-Características de las técnicas GRASP.

- Algoritmo 4 – Grasp con perturbacion Extendido

---

```
1: Construir un conjunto de soluciones golosas-aleatorias  $\rightarrow P_0, O^* \leftarrow \emptyset$ 
2: for all  $S_t \in P_0$  do
3:    $O_{t0} \leftarrow S_t$ 
4:   for  $i = 1$  to  $N$  do
5:     Seleccionar aleatoriamente un mecanismo de ruptura  $u_i$ 
6:     Modificar el óptimo local previamente obtenido empleando  $u_i \rightarrow S_{ti}$ 
7:     Realizar una búsqueda local a partir de  $S_{ti} \rightarrow O_{ti}$ 
8:     if  $i = N \wedge O_{ti}$  es débil factible then
9:       Realizar un ordenamiento exacto de las rutas de  $O_{ti} \rightarrow O_{ti}^e$ 
10:    else
11:       $O_{ti} \rightarrow O_{ti}^e$ 
12:    end if
13:    if  $C(O_{ti}^e) < C(O^*)$  then
14:       $O_{ti}^e \rightarrow O^*$ 
15:    end if
16:  end for
17: end for
18: return  $O^*$ 
```

Teniendo en cuenta el costo computacional de encontrar la solución exacta se sugiere realizar una única ejecución del modelo en la última iteración del bloque de perturbación.

# 4-Resultados

Atendiendo a las características de la distribución geográfica de los clientes y la capacidad de los vehículos, este conjunto puede separarse en cuatro grupos:

**SCA3x, Clientes dispersos con pocas rutas.** Estos son las instancias más simples del conjunto.

Problema	Literatura	A.1	A.1 <sub>G</sub>	A.2	A.3	A.3 <sub>G</sub>	A.4
SCA30	<b>635.62</b>	636.37		640.56	636.09		<u>636.06</u>
SCA31	<b>697.84</b>	<b>697.83</b>		<b>697.83</b>	<b>697.83</b>		<b>697.83</b>
SCA32	<b>659.34</b>	<b>659.33</b>		664.19	<b>659.33</b>		<b>659.33</b>
SCA33	<b>680.04</b>	<b>680.03</b>		680.58	<b>680.03</b>		<b>680.03</b>
SCA34	<b>690.50</b>	<b>690.48</b>		<b>690.48</b>	<b>690.48</b>		<b>690.48</b>
SCA35	<b>659.90</b>	661.08	<b>659.91</b>	664.69	<b>659.91</b>		<b>659.91</b>
SCA36	<b>651.09</b>	652.95		653.69	<b>651.11</b>		<b>651.11</b>
SCA37	<b>659.17</b>	666.17		660.80	<b>659.18</b>		<b>659.18</b>
SCA38	<b>719.47</b>	<b>719.50</b>		<b>719.50</b>	<b>719.50</b>		<b>719.50</b>
SCA39	<b>681.00</b>	<b>681.02</b>		681.25	<b>681.02</b>		<b>681.02</b>

**SCA8x, Clientes dispersos con muchas rutas.**

SCA80	<b>961.50</b>	983.94		997.47	972.97	<u>970.67</u>	<u>970.67</u>
SCA81	<b>1049.65</b>	1068.26		1090.75	1057.09	<u>1054.93</u>	1067.24
SCA82	<b>1039.64</b>	1052.92		1060.24	1047.57		<u>1044.97</u>
SCA83	<b>983.34</b>	1008.46		1021.15	1005.19	<u>994.77</u>	1002.83
SCA84	<b>1065.49</b>	1067.74	<b>1065.49</b>	1086.69	<b>1065.49</b>		1067.66
SCA85	<b>1027.08</b>	1058.17		1063.55	<u>1039.87</u>		1043.58
SCA86	<b>971.82</b>	977.89		987.69	973.26		<b>971.84</b>
SCA87	<b>1051.28</b>	1076.38		1071.55	<u>1067.03</u>		1067.12
SCA88	<b>1071.18</b>	1086.80	1082.91	1080.59	<b>1071.20</b>		<b>1071.20</b>
SCA89	<b>1060.50</b>	1090.03		1078.86	<u>1065.60</u>		1073.96

# 4-Resultados

## CON3x, Problemas urbanos con pocas rutas.

CON30	<b>616.52</b>	<b>616.50</b>	623.36	<b>616.50</b>		<b>616.50</b>
CON31	<b>554.47</b>	557.18	560.30	<b>554.45</b>		<b>554.45</b>
CON32	<b>518.00</b>	521.39	522.09	<u>519.13</u>		<u>519.13</u>
CON33	<b>591.19</b>	<b>591.20</b>	591.21	<b>591.20</b>		<b>591.20</b>
CON34	<b>588.79</b>	591.41	589.88	<b>588.78</b>		<b>588.78</b>
CON35	<b>563.70</b>	564.86	565.10	<b>563.67</b>		<b>563.67</b>
CON36	<b>499.05</b>	502.17	502.17	501.28	<u>499.38</u>	500.36
CON37	<b>576.48</b>	582.10	581.37	<u>578.18</u>		578.41
CON38	<b>523.05</b>	<b>523.08</b>	<b>523.08</b>	<b>523.08</b>		<b>523.08</b>
CON39	<b>578.25</b>	582.69	579.65	582.69		<b>578.25</b>

## CON8x, Problemas urbanos con muchas rutas.

CON80	<b>857.17</b>	880.60	867.41	867.33	864.40	<u>860.60</u>
CON81	<b>740.85</b>	742.26	755.79	742.93		<b>740.83</b>
CON82	<b>712.89</b>	716.34	717.59	<b>712.89</b>		713.44
CON83	<b>811.07</b>	828.16	832.24	818.43		<u>816.27</u>
CON84	<b>772.25</b>	787.24	782.27	773.59	<u>772.41</u>	773.73
CON85	<b>754.88</b>	759.83	772.16	<u>758.11</u>		<u>758.11</u>
CON86	<b>678.92</b>	690.59	686.80	<u>683.08</u>		687.96
CON87	<b>811.96</b>	816.38	817.86	<u>813.02</u>		814.50
CON88	<b>767.53</b>	783.16	783.74	778.36	770.76	<u>769.66</u>
CON89	<b>809.00</b>	821.26	816.75	<u>811.15</u>		812.04

# 5-Conclusiones

Al analizar los resultados obtenidos, se puede apreciar un bajo rendimiento de los métodos desarrollados en los problemas con un mayor número de rutas (SCA8x y CON8x), lo que sugiere un aumento de la exploración Inter Ruta.

Aunque no se observan diferencias significativas entre los resultados, vale resaltar que el Algoritmo de Perturbación tuvo un mayor rendimiento logrando una efectividad de aproximadamente el 50% para el conjunto de prueba.

# 6-Bibliografia

El trabajo científico sobre el cual se ha llevado a cabo la presentación puede ser encontrado en el siguiente enlace:

[\(PDF\) ESTRATEGIA GRASP PARA EL PROBLEMA DE ENRUTAMIENTO DE VEHÍCULOS CON RECOGIDA Y ENTREGA SIMULTÁNEA \(researchgate.net\)](#)

También se encuentra el PDF adjunto en la misma carpeta.