Heurística híbrida GRASP+VND para ruteo con pedidos dinámicos

Problema: VRP con pedidos dinámicos (DVRP)

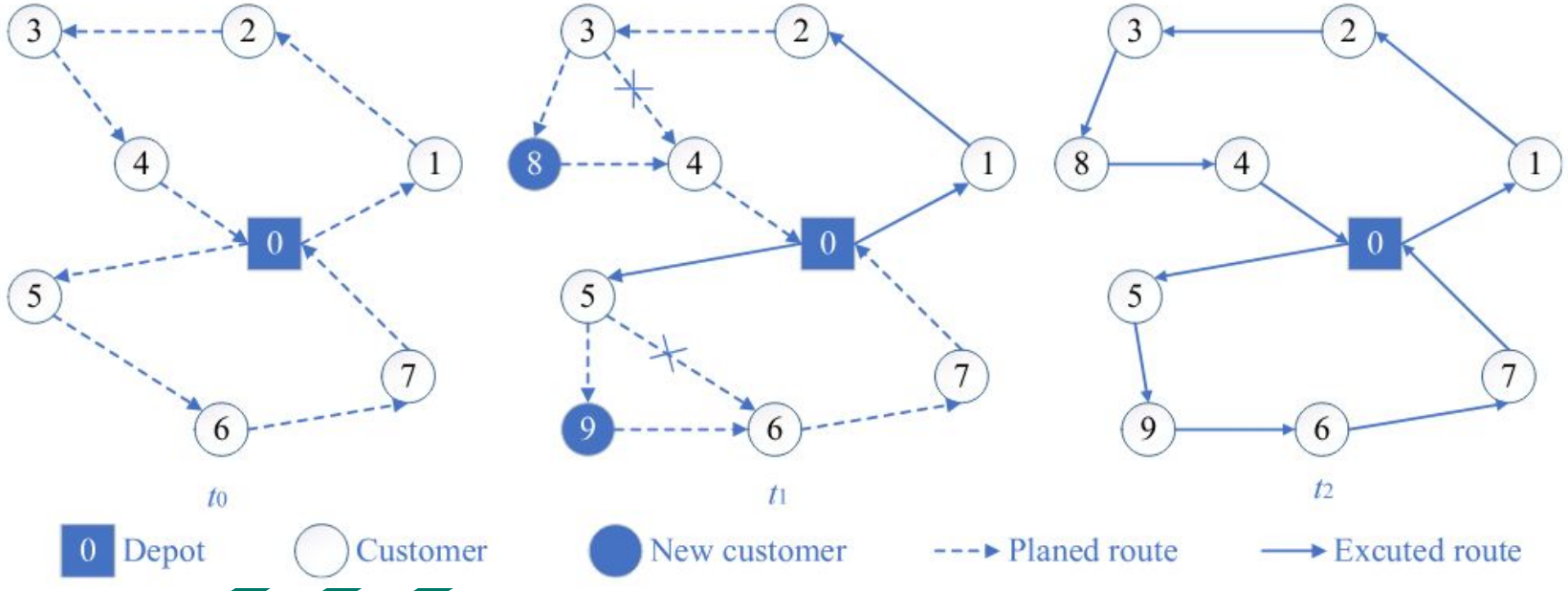
•VRP: planificar rutas para una flota minimizando distancia/tiempo y respetando capacidad, ventanas, etc.

•Dinámico:

–Nuevas solicitudes aparecen durante la operación.

–Se requiere optimizar/ajustar rutas en tiempo de ejecución.

•Objetivo del trabajo: minimizar la distancia total manteniendo la factibilidad al llegar pedidos.


3
2
4
1
0
5
7
6
t_0
3
2
8
4
1
0
5
7
9
6
t_1
3
2
8
4
1
0
5
7
9
6
t_2
0 Depot
Customer
New customer
Planed route
Excuted route

Metodología

•GRASP (Greedy Randomized Adaptive Search Procedure):

–Construye soluciones iniciales con criterio greedy y aleatoriedad controlada (RCL).

•VND (Variable Neighborhood Descent):

–Mejora sistemática con distintos vecindarios si hay ganancia; si no, cambia de vecindario.

•Iterar: diversidad (GRASP) + intensificación (VND).

GRASP: construcción diversificada

- Evalúa inserciones factibles y su costo incremental.
- Arma una lista restringida de candidatos (RCL) con las mejores opciones.
- Selecciona aleatoriamente dentro de la RCL → diversidad controlada.

VND: refinamiento mediante vecindarios

- Secuencia de vecindarios (p. ej., relocate, swap, 2-opt).
- Si mejora, se mantiene; si no, cambia al siguiente vecindario.
- Termina cuando ningún vecindario produce mejora.

Consideraciones del problema

- Una solución esta dada de la forma S = {r0, r1, ..., rk}, donde ri = {v0, v1, ..., vp, vp+1} representa una ruta que empieza y termina en el mismo lugar (por ejemplo un depósito), es decir v0 = vp+1.
- El costo entre dos puntos puede estar dado por la distancia recorrida, o el tiempo empleado.
- El depósito tendrá una apertura (e0) y un cierre (l0) en donde los pedidos estáticos estarán definidos antes de la apertura y los dinámicos se incluirán en un instante ti tal que e0 < ti < l0.

Algunas restricciones del problema

- Un cliente debe ser visitado una única vez por un único vehículo.
- Tener en cuenta la capacidad de cada vehículo.
- Las visitas deben ser en un instante ti tal que e0 < ti < l0

Pseudocódigo

Algorithm 1: Hybrid_GRASP_VND

Input: A maximum number of iterations *maxIter*
Output: best solution found S^*

```
1   S* ←∅; f(S*) ← ∞
2   iter ← 1
3   while iter ≤ maxIter do
4     S_init ← GreedyRandomGenerator()
5     S_impr ← VND(S_init)
6     if f(S_impr) < f(S*) then
7       S* ← S_impr
8     end if
9   end while
10   return S*
```

GreedyRandomGenerator

Se definen 4 heurísticas con parámetros S (solución parcial) y custs (clientes no visitados) a utilizar:

Closest: Inserta cada cliente al final de cada ruta con costo D(u,c)

Earliest: Inserta cada cliente en cada posición posible de cada ruta con costo ti’-ti

Farthest: Inserta cada cliente en cada posición posible de cada ruta con costo D(u,v) - D(u,c) - D(c,v)

Nearest: Inserta cada cliente en cada posición posible de cada ruta con costo D(u,c) + D(c,v) - D(u,v)

GreedyRandomGenerator

Algorithm 6: GreedyRandomGenerator()

Input: a set of unvisited customers *custs*,
a random number $\alpha \in (0,1)$
Output: solution S

```
1   S ← ∅
2   CL ← ∅
3     while custs ≠ ∅ do
4     if CL = ∅ then
5       c ← SelectRandom(custs)
6       S ← S ∪ {0, c, 0}
7       custs ← custs\{c}
8     end if
9     M ← Select at random from {Closest, Nearest, Farthest, Earliest}
10    CL ← M(S, custs)
11    if CL = ∅ then
12      continue
13    end if
14    g_min ← min{ g | move(r, i, c, g) ∈ CL}
15    g_max ← max{ g | move(r, i, c, g) ∈ CL }
16    threshold ← g_min + α (g_max − g_min)
17    RCL ← { move(r, i, c, g) ∈ CL | g ≤ threshold }
18    move(r*, i*, c*, g*) ← select at random from RCL
19    r* ← r* ∪{c*}
20    custs ← custs\{c*}
21  end while
22  return S
```

VND: Vecindarios

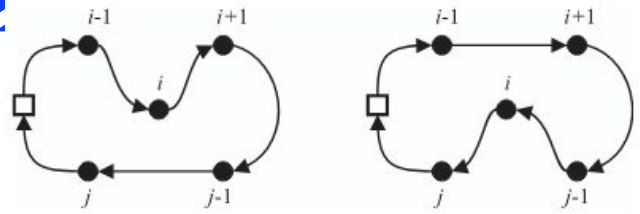


Fig. 3. Relocate.

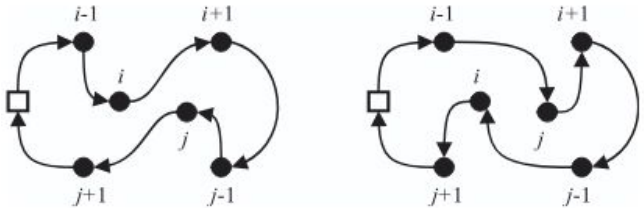


Fig. 4. 3 Swap.

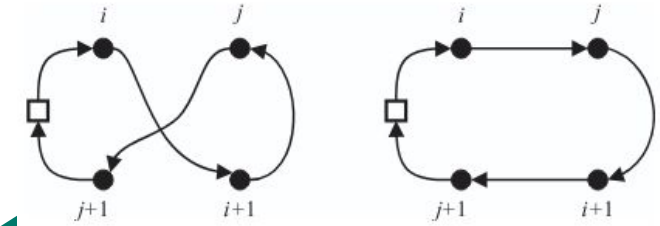


Fig. 5. 2-opt.

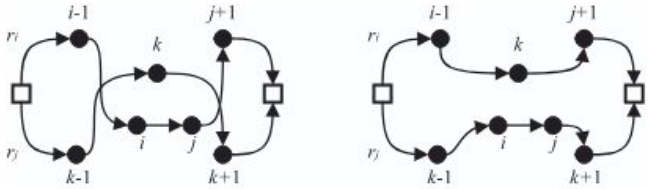


Fig. 6. (1,λ) interchange.

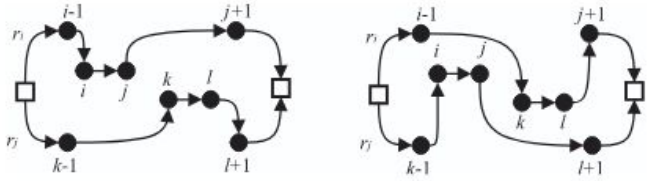


Fig. 7. String-Exchange.

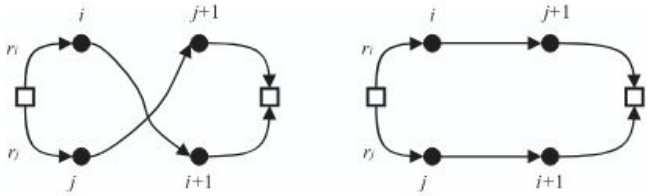


Fig. 8. Crossover.

VND

Algorithm 7: VND

Input: Solution S, neighborhoods N
Output: Improved solution S

```
1   h ← 1
2   Randomly shuffle N
3   while h ≤ |N| do
4     S' ← N_h(S)
5     if f(S') < f(S) then
6       S ← S'
7       h ← 1
8       Randomly shuffle N
9     else
10      h ← h + 1
11    end if
12  end while
13  return S
```

Table 4. Average results of GRASP/VND, Closest, Nearest, Farthest, and Earliest.

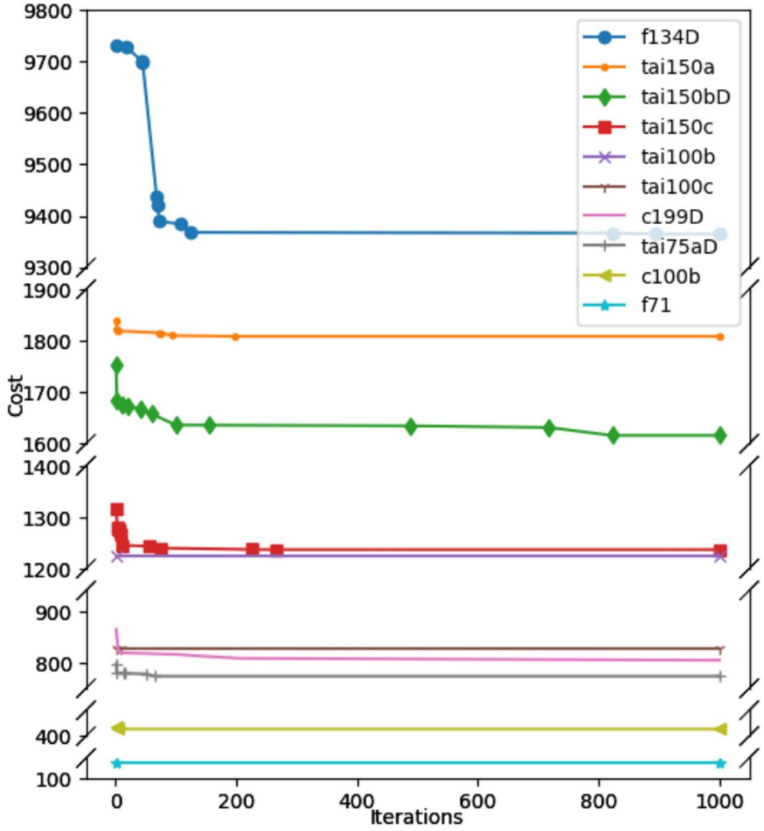
Instance	GRASP/VND	Closest	Nearest	Farthest	Earliest
c100b	**833.55**	871.31	854.35	878.56	847.39
c199D	**1515.36**	1591.41	1626.81	1626.62	1592.76
f134D	**12481.80**	15621.42	13808.26	15529.22	13538.66
f71	**253.64**	314.29	302.25	300.62	310.07
tai100b	**2066.47**	2269.96	2141.29	2199.73	2158.48
tai100c	**1459.31**	1624.54	1519.75	1589.38	1521.07
tai150a	**3187.69**	3334.59	3283.72	3339.14	3302.82
tai150bD	**2829.76**	2977.02	3010.28	2997.50	2956.94
tai150c	**2441.06**	2682.59	2537.22	2596.19	2518.44
tai75aD	**1713.52**	1786.80	1775.85	1787.13	1764.12

Instance	GRASP/VND	N_1	N_2	N_3	N_4	N_5	N_6
c100b	**833.55**	1224.47	1190.58	1138.40	1133.62	913.80	1179.76
c199D	**1515.36**	2133.42	2340.63	2262.79	2086.55	1729.40	2173.53
f134D	**12481.80**	20529.15	21023.37	19888.97	20409.65	15152.19	21192.46
f71	**253.64**	366.91	415.18	355.08	379.08	332.73	404.09
tai100b	**2066.47**	2708.29	3010.38	2707.06	2770.10	2420.15	2899.11
tai100c	**1459.31**	1916.48	2193.87	1921.94	2010.11	1651.38	2303.33
tai150a	**3187.69**	3799.51	4565.30	3906.83	4053.15	3606.68	4542.18
tai150bD	**2829.76**	3692.07	4151.74	3669.08	3889.77	3206.94	4141.38
tai150c	**2441.06**	3367.08	4248.81	3319.36	3731.17	2997.41	4145.02
tai75aD	**1713.52**	2062.28	2361.81	2107.81	2208.47	1907.01	2314.37

Resultados

- La heurística GRASP+VND logra rutas de alta calidad en escenarios dinámicos.
- Competitiva frente a enfoques comparados en los benchmarks probados.
- El método propuesto junta los puntos fuertes de GRASP+VND, mejorando la calidad de los resultados de una manera óptima.

Convergencia a una solución



Comparación con otras heurísticas

Instance	GRASP/VND	E-ACO		BSO		AAC	
		Best	Gap(%)	Best	Gap(%)	Best	Gap(%)
c50	**534.54**	607.21	−13.59	597.18	−11.72	551.95	−3.26
c75	931.51	**924.71**	0.73	938.79	−0.78	1156.83	−24.19
c100	**906.61**	973.40	−7.37	912.83	−0.69	1311.72	−44.68
c100b	833.55	869.22	−4.28	900.48	−8.03	**800.93**	3.91
c120	1307.59	1108.15	15.25	1173.10	10.29	**1049.47**	19.74
c150	**1211.50**	1378.63	−13.80	1254.47	−3.55	2188.33	−80.63
c199	**1519.62**	1561.12	−2.73	1631.23	−7.34	1650.85	−8.64
f71	**253.64**	259.71	−2.39	270.28	−6.56	301.79	−18.98
f134	**12821.86**	–	–	14831.29	−15.67	13015.56	−1.51
tai75a	**1710.31**	1690.91	1.13	1713.46	−0.18	1755.33	−2.63
tai75b	1373.51	1509.56	−9.91	1426.93	−3.89	**1306.47**	4.88
tai75c	1404.66	**1329.42**	5.36	1392.96	0.83	1424.76	−1.43
tai75d	1389.65	1409.14	−1.40	1503.79	−8.21	**1334.67**	3.96
tai100a	**2128.47**	2281.7	−7.20	2237.90	−5.14	2194.93	−3.12
tai100b	**2066.47**	2255.83	−9.16	2068.12	−0.08	2126.09	−2.89
tai100c	1459.31	**1442.45**	1.16	1479.48	−1.38	1544.50	−5.84
tai100d	1747.55	**1581.36**	9.51	1855.26	−6.16	1909.55	−9.27
tai150a	3187.69	3307.63	−3.76	3391.39	−6.39	**2999.27**	5.91
tai150b	**2799.55**	3128.00	−11.73	2899.56	−3.57	2846.28	−1.67
tai150c	**2441.06**	2583.36	−5.83	2596.67	−6.37	2718.36	−11.36
tai150d	**2757.04**	2808.99	−1.88	3073.54	−11.48	3230.67	−17.18
			−3.10		−4.58		−9.47

Instance	GRASP/VND	E-ACO		ACO-CD		AAC	
		Avg	Gap(%)	Avg	Gap(%)	Avg	Gap(%)
c50	577.70	647.21	−12.03	593.32	−2.70	**570.89**	1.18
c75	**955.52**	1045.44	−9.41	944.18	1.19	1213.45	−26.99
c100	**955.87**	1044.96	−9.32	963.80	−0.83	1380.25	−44.40
c100b	855.90	950.17	−11.01	948.56	−10.83	**841.44**	1.69
c120	1359.79	1197.68	11.92	1235.30	9.16	**1153.29**	15.19
c150	**1265.82**	1472.40	−16.32	1268.16	−0.18	2386.93	−88.57
c199	**1562.66**	1836.86	−17.55	1566.37	−0.24	1758.51	−12.53
f71	294.61	297.08	−0.84	**289.24**	1.82	309.94	−5.20
f134	**13563.43**	–	–	15827.69	−16.69	15528.81	−14.49
tai75a	**1765.18**	1983.92	−12.39	1964.59	−11.30	1782.91	−1.00
tai75b	**1395.83**	1647.78	−18.05	1527.33	−9.42	1452.26	−4.04
tai75c	1479.00	1470.6	0.57	1617.84	−9.39	**1441.91**	2.51
tai75d	**1407.26**	1661.73	−18.08	1522.00	−8.15	1422.27	−1.07
tai100a	**2192.67**	2550.64	−16.33	2213.04	−0.93	2232.71	−1.83
tai100b	**2142.63**	2500.72	−16.71	2391.59	−11.62	2182.61	−1.87
tai100c	**1493.52**	1743.07	−16.71	1584.20	−6.07	1562.66	−4.63
tai100d	**1829.41**	1843.82	−0.79	2275.15	−24.37	1912.43	−4.54
tai150a	3244.14	3684.03	−13.56	3473.58	−7.07	**3185.73**	1.80
tai150b	**2871.59**	3439.38	−19.77	3347.30	−16.57	2880.57	−0.31
tai150c	**2505.24**	2729.15	−8.94	2851.53	−13.82	2743.55	−9.51
tai150d	**2817.62**	3186.08	−13.08	3191.28	−13.26	3345.16	−18.72
			−10.92		−7.20		−10.35

Tiempo computacional

Table 8
Computational time of GRASP/VND compared with AAC.

Instance	GRASP/VND	AAC
c50	**39.92**	525.68
c75	**81.05**	1142.41
c100	**251.54**	1356.50
c100b	**158.37**	2086.25
c120	**420.41**	2471.25
c150	**812.37**	3861.15
c199	**1565.87**	3001.35
f71	**103.25**	1143.72
f134	**283.28**	3417.7
tai75a	**64.30**	942.42
tai75b	**59.96**	906.60
tai75c	**76.24**	906.20
tai75d	**65.33**	1091.89
tai100a	**233.45**	1917.05
tai100b	**175.45**	1887.13
tai100c	**147.30**	1929.09
tai100d	**191.10**	1991.84
tai150a	**826.17**	3323.27
tai150b	**705.41**	3687.81
tai150c	**559.37**	3867.34
tai150d	**459.76**	3496.64
Avg	**346.66**	2140.63

¡Gracias!

•Chen, Yin, Sang, Deng (2025). Egyptian Informatics Journal. DOI: 10.1016/j.eij.2025.100638. Link