



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería de Organización Industrial

**Comparativa de modelos y solvers para la
optimización de variantes de problemas
de rutas de vehículos**

Autor:

Alonso Alonso, Arturo

Tutor:

**Sáez Aguado, Jesús
Estadística e Investigación
Operativa**

Valladolid, junio 2016

*A mis padres, porque sin su esfuerzo
nada de esto habría sido posible.*

*A Alexandra y a mis amigos, por
aguantarme a lo largo de todos
estos años de duro trabajo.*

*'I conjecture that there is no good algorithm
for the traveling salesman problem'.*

–Jack Edmonds, 1967.

*'A surprise discovery by a 28-years-old mathematician has
rocked the world of mathematics and computer analysis'.*

–Malcom W.Browne, New York Times, 1979

*'Solving linear programming problems for
Industry is a multibillion-dollar-a-year business'.*

–Gina Kolata, New York Times, March 12, 1989.

*'The development of linear programming is - in my opinion - the most
important contribution of the mathematics of the 20th century to the solution
of practical problems arising in industry and commerce'.*

–Martin Grötschel, 2006.

Resumen

Los problemas de rutas de vehículos son uno de los casos más comunes en optimización, abarcando prácticamente cualquier tipo de transporte, en orden de mejorar la calidad del mismo.

En el presente Trabajo de Fin de Grado se evaluarán los principales tipos de problemas de rutas de vehículos (incluyendo problemas con ventanas de tiempo y con recogidas y distribuciones), para cada uno de los cuáles se revisará el estado del arte de los principales modelos de optimización existentes, formulando dichos modelos y resolviéndolos con varios solvers para una batería de problemas dada.

De esta forma, se pretende realizar una profunda comparación de los resultados obtenidos, consiguiendo identificar los mejores modelos y solvers para resolver cada uno de esos problemas de rutas.

Por último, para algunos casos, se proponen mejoras en los modelos que permiten la obtención de unos mejores resultados, reduciendo los costes de operación asociados.

Palabras clave

Rutas de vehículos, optimización, modelos, solvers, comparación.

Abstract

Vehicle routing problems are one of the most common optimization problem types, involving nearly each transportation activity, in order to improve its service quality.

In this document, we will evaluate the main vehicle routing problem types (including both time windows and pick up and delivery problems). For each one of them, we will review its current optimization models state of the art, programming these models and solving them by using several solvers for a huge variety of examples.

Therefore, we pretend to compare the amount of results worked out, which will provide us to identify the best models and solvers existent nowadays for solving each one of these vehicle routing problems.

Finally, for some of these problems, we propose little modifications on its models, which will let us to obtain better results, decreasing our transportation costs.

Key words

Vehicle routing problems, optimization, models, solvers, comparison.

ÍNDICE

ÍNDICE DE FIGURAS	- 17 -
ÍNDICE DE TABLAS	- 23 -
CAPÍTULO 1.INTRODUCCIÓN Y OBJETIVOS.....	- 27 -
1.1. Introducción.....	- 27 -
1.2. Objetivos.	- 28 -
1.3. Estructura.....	- 28 -
CAPÍTULO 2.PROBLEMAS DE RUTAS DE VEHÍCULOS.	- 31 -
2.1. Introducción.	- 31 -
2.2. Breve historia.	- 31 -
2.3. Tipología de problemas.	- 34 -
2.4. Criterios de comparación.	- 40 -
2.5. Aplicaciones.	- 40 -
2.6. Complejidad del problema.	- 43 -
CAPÍTULO 3.INVESTIGACIÓN OPERATIVA Y PROGRAMACIÓN LINEAL.....	- 45 -
3.1. Introducción.	- 45 -
3.2. La investigación operativa.	- 45 -
3.2.1. Modelización. Programación matemática.....	- 46 -
3.2.2. Optimización. Resolución de los modelos.....	- 48 -
3.3. La programación lineal entera.....	- 48 -
3.3.1. Historia. Origen en el TSP.	- 48 -
3.3.2. Métodos de resolución.	- 50 -
3.3.2.1. Introducción.	- 50 -
3.3.2.2. Métodos exactos.....	- 51 -
3.3.2.3. Métodos heurísticos y meta heurísticos.	- 53 -
3.3.3. Herramientas de implementación.	- 54 -
3.3.4. Solvers seleccionados y criterios de comparación.....	- 55 -

3.3.5. Diferencias entre Xpress-IVE y CPLEX.....	- 56 -
CAPÍTULO 4.EL PROBLEMA DEL VIAJANTE DE COMERCIO (TSP).....	- 59 -
4.1. Descripción del problema.	- 59 -
4.2. Teoría de grafos.	- 61 -
4.2.1. Tipos de grafos.	- 61 -
4.3. La distancia euclídea.....	- 62 -
4.4. Formulación.	- 63 -
4.4.1. Subtours.	- 64 -
4.4.2. Eliminación de subtours.	- 65 -
4.5. Datos empleados.....	- 69 -
4.6. Resultados obtenidos.....	- 69 -
4.6.1. Conclusiones obtenidas. Problema TSP.....	- 69 -
CAPÍTULO 5.PROBLEMAS DE RUTAS CON CAPACIDADES (CVRP).	- 71 -
5.1. Descripción del problema.	- 71 -
5.2. Modelos planteados. Caso homogéneo.....	- 73 -
a) Modelo basado en redes.....	- 73 -
b) Modelo de dos índices basado en Tucker – Miller – Zemlin.....	- 75 -
c) Modelo de tres índices basado en Tucker – Miller – Zemlin.....	- 76 -
d) Modelo de Christofides, Mingozzi y Toth.....	- 78 -
5.3. Modelos planteados. Caso heterogéneo.	- 80 -
a) Modelo de tres índices basado en Tucker – Miller - Zemlin.....	- 80 -
b) Modelo de Golden – Assad – Levys – Gheysens basado en TMZ.....	- 81 -
c) Modelo de Golden – Assad – Gheysens basado en redes.	- 83 -
d) Modelo de Christofides, Mingozzi y Toth.....	- 85 -
5.4. Datos empleados.....	- 86 -
5.5. Resultados obtenidos empleando Xpress-IVE.	- 87 -
5.5.1. Caso homogéneo.	- 87 -
5.5.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 88 -
5.5.3. Caso heterogéneo.	- 91 -
5.5.4. Resumen de resultados obtenidos. Caso heterogéneo.	- 93 -

5.6. Resultados obtenidos empleando CPLEX.....	94 -
5.6.1. Caso homogéneo.	94 -
5.6.2. Resumen de resultados. Caso homogéneo.	96 -
5.6.3. Caso heterogéneo.	97 -
5.6.3. Resumen de resultados obtenidos. Caso heterogéneo.	98 -
5.7. Comparativa Xpress-IVE vs CPLEX para el CVRP.	99 -
5.7.1. Caso homogéneo.	99 -
5.7.2. Resumen de resultados obtenidos. Caso homogéneo.....	102 -
5.7.3. Caso heterogéneo.	103 -
5.7.2. Resumen de resultados obtenidos. Caso heterogéneo.	108 -
5.8. Conclusiones.....	109 -

CAPÍTULO 6.PROBLEMAS DE RUTAS CON CAPACIDADES Y VENTANAS DE TIEMPO (CVRPTW)..... - 111 -

6.1. Descripción del problema.	111 -
6.2. Modelos planteados. Caso homogéneo.....	113 -
a) Modelo de Sopnamayee Acharya.	113 -
b) Modelo de Roberto Cordone y Roberto Wolfler de dos índices.-	115 -
-	
c) Modelo de Roberto Cordone y Roberto Wolfler de tres índices.-	118 -
-	
d) Modelo Time Index Formulation (TIF).....	120 -
e) Modelo de Paolo Toth y Daniele Vigo.	123 -
6.3. Modelos planteados. Caso heterogéneo.	126 -
a) Modelo de Sopnamayee Acharya.	126 -
b) Modelo de Roberto Cordone y Roberto Wolfler de tres índices.-	128 -
-	
c) Modelo Time Index Formulation (TIF).....	130 -
d) Modelo de Paolo Toth y Daniele Vigo.	132 -
e) Modelo de Manolis N.Kritikos y George Ioannou.	134 -
6.4. Datos empleados.....	138 -
6.5. Resultados obtenidos empleando Xpress-IVE.	138 -
6.5.1. Caso homogéneo.	138 -

6.5.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 140 -
6.5.3. Caso heterogéneo.	- 143 -
6.5.4. Resumen de resultados obtenidos. Caso heterogéneo.	- 145 -
6.6. Resultados obtenidos empleando CPLEX.....	- 146 -
6.6.1. Caso homogéneo.	- 146 -
6.6.2. Resumen de resultados. Caso homogéneo.	- 149 -
6.6.3. Caso heterogéneo.	- 150 -
6.6.3. Resumen de resultados obtenidos. Caso heterogéneo.	- 151 -
6.7. Comparativa Xpress-IVE vs CPLEX para el CVRPTW.....	- 153 -
6.7.1. Caso homogéneo.	- 153 -
6.7.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 157 -
6.7.3. Caso heterogéneo.	- 157 -
6.7.4. Resumen de resultados obtenidos. Caso heterogéneo.	- 161 -
6.8. Conclusiones.....	- 161 -

CAPÍTULO 7.PROBLEMAS DE RUTAS CON RECOGIDAS (PICK UP) Y ENTREGAS (DELIVERY) (CVRPPD). - 163 -

7.1. Descripción del problema.	- 163 -
7.2. Modelos planteados. Caso homogéneo.....	- 165 -
a) Modelo de Jan Dethloff.	- 165 -
b) Modelo de Quan Lu y Maged Dessouky.....	- 169 -
c) Modelo basado en redes multiproducto.	- 174 -
d) Modelo de M.W.P.Savelsbergh y M.Sol.....	- 178 -
e) Modelo de Paolo Toth y Daniele Vigo.	- 182 -
7.3. Modelos planteados. Caso heterogéneo.	- 186 -
a) Modelo de Jan Dethloff.	- 186 -
b) Modelo de Quan Lu y Maged Dessouky.....	- 188 -
c) Modelo basado en redes multiproducto.....	- 191 -
d) Modelo de M.W.P.Savelsbergh y M.Sol.....	- 193 -
e) Modelo de Paolo Toth y Daniele Vigo.	- 196 -
7.4. Datos empleados.....	- 198 -
7.5. Resultados obtenidos empleando Xpress-IVE.....	- 200 -
7.5.1. Caso homogéneo.	- 200 -

7.5.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 201 -
7.5.3. Caso heterogéneo.	- 204 -
7.5.4. Resumen de resultados obtenidos. Caso heterogéneo.	- 206 -
7.6. Resultados obtenidos empleando CPLEX.....	- 207 -
7.6.1. Caso homogéneo.	- 207 -
7.6.2. Resumen de resultados. Caso homogéneo.	- 209 -
7.6.3. Caso heterogéneo.	- 211 -
7.6.3. Resumen de resultados obtenidos. Caso heterogéneo.	- 212 -
7.7. Comparativa Xpress-IVE vs CPLEX para el CVRPPD.	- 214 -
7.7.1. Caso homogéneo.	- 214 -
7.7.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 217 -
7.7.3. Caso heterogéneo.	- 218 -
7.7.4. Resumen de resultados obtenidos. Caso heterogéneo.	- 223 -
7.8. Conclusiones.....	- 224 -

CAPÍTULO 8.PROBLEMAS DE RUTAS CON RECOGIDAS (PICK UP), ENTREGAS (DELIVERY) Y VENTANAS DE TIEMPO (CVRPPDTW). - 225 -

8.1. Descripción del problema.	- 225 -
8.2. Modelos planteados. Caso homogéneo.....	- 227 -
a) Modelo de Paolo Toth y Daniele Vigo.	- 227 -
b) Modelo de M.W.P.Savelsbergh y M.Sol.....	- 230 -
c) Modelo basado en redes multiproducto.	- 234 -
d) Modelo de Yvan Dumas, Jacques Desrosiers y François Soumis.	- 238 -
e) Modelo de Enrico Angelelli y Renata Mansini.....	- 242 -
8.3. Modelos planteados. Caso heterogéneo.	- 246 -
a) Modelo de Paolo Toth y Daniele Vigo.	- 246 -
b) Modelo de M.W.P.Savelsbergh y M.Sol.....	- 248 -
c) Modelo basado en redes multiproducto.	- 251 -
d) Modelo de Yvan Dumas, Jacques Desrosiers y François Soumis.	- 253 -
e) Modelo de Enrico Angelelli y Renata Mansini.....	- 256 -
8.4. Datos empleados.....	- 259 -

8.5. Resultados obtenidos empleando Xpress-IVE.....	- 260 -
8.5.1. Caso homogéneo.....	- 260 -
8.5.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 261 -
8.5.3. Caso heterogéneo.....	- 264 -
8.5.4. Resumen de resultados obtenidos. Caso heterogéneo.....	- 266 -
8.6. Resultados obtenidos empleando CPLEX.....	- 267 -
8.6.1. Caso homogéneo.....	- 267 -
8.6.2. Resumen de resultados. Caso homogéneo.....	- 269 -
8.6.3. Caso heterogéneo.....	- 270 -
8.6.3. Resumen de resultados obtenidos. Caso heterogéneo.....	- 271 -
8.7. Comparativa Xpress-IVE vs CPLEX para el CVRPPDTW.....	- 272 -
8.7.1. Caso homogéneo.....	- 272 -
8.7.2. Resumen de resultados obtenidos. Caso homogéneo.....	- 276 -
8.7.3. Caso heterogéneo.....	- 277 -
8.7.4. Resumen de resultados obtenidos. Caso heterogéneo.....	- 280 -
8.8. Conclusiones.....	- 281 -
CAPÍTULO 9.CONCLUSIONES FINALES Y FUTURAS LÍNEAS DE TRABAJO..	- 283 -
9.1. Conclusiones finales.....	- 283 -
9.2. Futuras líneas de trabajo.....	- 284 -
9.2.1. Heurísticas y meta heurísticas.....	- 284 -
9.2.2. Herramientas de tuning.....	- 288 -
9.2.3. Otros solvers.....	- 289 -
9.2.4. Otros modelos.....	- 289 -
9.2.5. Otros problemas.....	- 289 -
CAPÍTULO 10.AGRADECIMIENTOS.....	- 291 -
10.1. Agradecimientos.....	- 291 -
ANEXO.TRATANDO DE REDUCIR EL NÚMERO DE VEHÍCULOS EMPLEADO.-	293
-	
A.1. Motivación.....	- 293 -
A.2. Planteamiento.....	- 295 -

A.3. Resultados.....	- 296 -
A.3.1. Problema CVRP.....	- 296 -
A.3.2. Conclusiones obtenidas. Problema CVRP.	- 297 -
A.3.3. Problema CVRPTW.	- 298 -
A.3.4. Conclusiones obtenidas. Problema CVRPTW.	- 300 -
A.3.5. Problema CVRPPD.....	- 300 -
A.3.6. Conclusiones obtenidas. Problema CVRPPD.....	- 301 -
A.3.7. Problema CVRPPDTW.....	- 302 -
A.3.8. Conclusiones obtenidas. Problema CVRPPDTW.	- 303 -
 Bibliografía.....	 - 305 -

ÍNDICE DE FIGURAS

Figura 2.1: Puentes de Königsberg y simplificación como grafo.....	32 -
Figura 2.2: Solución obtenida por Euler para el desafío Knight's tour.	33 -
Figura 2.3: A la izquierda, un dodecaedro; A la derecha, The Icosian (1859) y la solución propuesta por Hamilton.....	33 -
Figura 2.4: ejemplo gráfico de un problema VRP.	34 -
Figura 2.5: One-to-one problem.	38 -
Figura 2.6: Many-to-many problem.....	38 -
Figura 2.7: One-to-many-to-one problem.....	39 -
Figura 2.8: subdivisiones básicas del VRP/CVRP y sus interconexiones.....	39 -
Figura 3.1: Concurso car 54.....	49 -
Figura 3.2: Solución de Dantzing et al. al problema TSP de 49 ciudades...-	50 -
Figura 3.3: taxonomía para los métodos de resolución exactos.	51 -
Figura 3.4: ejemplo práctico del algoritmo de Branch & Cut.....	52 -
Figura 3.5: ejemplo 1 con alguno de los parámetros de CPLEX.....	57 -
Figura 3.6: ejemplo 2 con alguno de los parámetros de CPLEX.....	58 -
Figura 4.1: ejemplo gráfico del problema TSP.....	60 -
Figura 4.2: ejemplo gráfico de la teoría de grafos propuesta por Euler.	61 -
Figura 4.3: grafo no dirigido.	62 -
Figura 4.4: grafo dirigido.	62 -
Figura 4.5: ejemplo de solución con subtours para el problema TSP.....	65 -
Figura 4.6: resultados obtenidos para el problema TSP.....	69 -
Figura 5.1: ejemplo gráfico del problema CVRP.	72 -
Figura 5.2: Distancias de Xpress-IVE para el problema CVRP homogéneo.-	87 -
Figura 5.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRP homogéneo.....	88 -
Figura 5.4: Distancias de Xpress-IVE. Problema CVRP heterogéneo.	91 -
Figura 5.5: Tiempos de Xpress-IVE. Problema CVRP heterogéneo.....	92 -
Figura 5.6: Distancias de CPLEX para el problema CVRP homogéneo.	94 -
Figura 5.7: Tiempos de CPLEX para el problema CVRP homogéneo.	95 -
Figura 5.8: Distancias de CPLEX para el CVRP heterogéneo.....	97 -
Figura 5.9: Tiempos de CPLEX para el CVRP heterogéneo.....	98 -

Figura 5.10: Comparativa de distancias modelo basado en redes.....	- 99 -
Figura 5.11: Comparativa de tiempos modelo basado en redes.	- 100 -
Figura 5.12: Resultados obtenidos con CPLEX para el CVRP heterogéneo.-	101
-	
Figura 5.13: Comparativa de distancias modelo de dos índices.	- 102 -
Figura 5.14: Comparativa de tiempos modelo de dos índices.	- 102 -
Figura 5.15: Comparativa de distancias modelo (a).	- 104 -
Figura 5.16: Comparativa de tiempos modelo (a).....	- 104 -
Figura 5.17: Comparativa de distancias modelo (b).	- 105 -
Figura 5.18: Comparativa de tiempos modelo (b).....	- 105 -
Figura 5.19: Comparativa de distancias modelo (c).....	- 106 -
Figura 5.20: Comparativa de tiempos modelo (c).	- 106 -
Figura 5.21: Comparativa de distancias modelo (d).	- 107 -
Figura 5.22: Comparativa de tiempos modelo (d).....	- 107 -
Figura 6.1: ejemplo gráfico del problema CVRPTW.....	- 112 -
Figura 6.2: Distancias de Xpress- IVE para el problema CVRPTW homogéneo..	- 139 -
Figura 6.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPTW homogéneo.	- 140 -
Figura 6.4: Distancias de Xpress-IVE. CVRPTW heterogéneo.	- 143 -
Figura 6.5: Tiempos de ejecución. CVRPTW heterogéneo.	- 143 -
Figura 6.6: Distancias de CPLEX para el problema CVRPTW homogéneo.-	147 -
Figura 6.7: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPTW homogéneo.	- 147 -
Figura 6.8: Distancias de CPLEX para el CVRPTW heterogéneo.	- 150 -
Figura 6.9: Tiempos de CPLEX para el CVRPTW heterogéneo.....	- 151 -
Figura 6.10: Comparativa de distancias modelo de Acharya.	- 154 -
Figura 6.11: Comparativa de tiempos modelo de Acharya.....	- 154 -
Figura 6.12: Comparativa de distancias modelo de Cordone – Wolfler de 2 índices.....	- 155 -
Figura 6.13: Comparativa de tiempos modelo de Cordone – Wolfler de 2 índices.....	- 155 -
Figura 6.14: Comparativa de distancias modelo de Cordone – Wolfler de 3 índices.....	- 156 -

Figura 6.15: Comparativa de tiempos modelo de Cordone – Wolfler de 3 índices.....	- 156 -
Figura 6.16: Comparativa de distancias modelo de Acharya.....	- 158 -
Figura 6.17: Comparativa de tiempos modelo de Acharya.....	- 158 -
Figura 6.18: Comparativa de distancias modelo de Cordone-Wolfler de 3 índices.....	- 159 -
Figura 6.19: Comparativa de tiempos modelo Cordone-Wolfler de 3 índices....	- 159 -
Figura 6.20: Comparativa de distancias modelo de Manolis N.Kritikos et al. ...	- 160 -
Figura 6.21: Comparativa de tiempos modelo de Manolis N.Kritikos et al.-	160 -
Figura 7.1: ejemplo gráfico del problema CVRPPD.	- 164 -
Figura 7.2: Distancias de Xpress- IVE para el problema CVRPPD homogéneo. .	- 201 -
Figura 7.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPPD homogéneo.....	- 201 -
Figura 7.5: Tiempos de ejecución Xpress-IVE. CVRPPD heterogéneo.....	- 204 -
Figura 7.6: Distancias de CPLEX para el problema CVRPPD homogéneo.-	208 -
Figura 7.7: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPPD homogéneo.....	- 208 -
Figura 7.8: Distancias de CPLEX para el CVRPPD heterogéneo.....	- 211 -
Figura 7.9: Tiempos de CPLEX para el CVRPPD heterogéneo.	- 212 -
Figura 7.10: Comparativa de distancias modelo de Quan Lu et al.	- 214 -
Figura 7.11: Comparativa de tiempos modelo de Quan Lu et al.....	- 215 -
Figura 7.12: Comparativa distancias modelo de Savelsbergh et al.....	- 215 -
Figura 7.13: Comparativa tiempos modelo de Savelsbergh et al.	- 216 -
Figura 7.14: Comparativa de distancias modelo de Toth y Vigo.	- 216 -
Figura 7.15: Comparativa de tiempos modelo de Toth y Vigo.	- 217 -
Figura 7.16: Comparativa de distancias modelo de Quan Lu et al.	- 219 -
Figura 7.17: Comparativa de tiempos modelo de Quan Lu et al.....	- 219 -
Figura 7.18: Comparativa de distancias modelo de redes multiproducto.-	220 -
Figura 7.19: Comparativa de tiempos modelo de redes multiproducto. ..	- 220 -
Figura 7.20: Comparativa distancias modelo de Savelsbergh et al.....	- 221 -

Figura 7.21: Comparativa de tiempos modelo de Savelsbergh et al.	- 221 -
Figura 7.22: Comparativa distancias modelo de Toth y Vigo.....	- 222 -
Figura 7.23: Comparativa de tiempos modelo de Toth y Vigo.....	- 222 -
Figura 8.1: tipos de problemas de rutas de vehículos.	- 225 -
Figura 8.2: ejemplo gráfico del problema CVRPPDTW.	- 226 -
Figura 8.3: Distancias de Xpress- IVE para el problema CVRPPDTW homogéneo.	- 260 -
Figura 8.4: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPPDTW homogéneo.....	- 261 -
Figura 8.5: Distancias Xpress-IVE. Problema CVRPPDTW heterog.....	- 264 -
Figura 8.6: Tiempos de ejecución Xpress-IVE. CVRPPDTW heterog.....	- 264 -
Figura 8.7: Distancias de CPLEX para el problema CVRPPDTW homogéneo.	- 268 -
Figura 8.8: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPPDTW homogéneo.....	- 268 -
Figura 8.9: Distancias de CPLEX para el CVRPPDTW heterogéneo.....	- 270 -
Figura 8.10: Tiempos de CPLEX para el CVRPPDTW heterogéneo.....	- 271 -
Figura 8.11: Comparativa distancias modelo de Savelsbergh et al.....	- 273 -
Figura 8.12: Comparativa tiempos modelo de Savelsbergh et al.	- 274 -
Figura 8.13: Comparativa distancias modelo de Dumas et al.	- 274 -
Figura 8.14: Comparativa tiempos modelo de Dumas et al.	- 274 -
Figura 8.15: Comparativa de distancias modelo de Enrico Angelelli et al.-	- 275 -
Figura 8.16: Comparativa de tiempos modelo de Enrico Angelelli et al. ..	- 275 -
Figura 8.17: Comparativa distancias modelo de Savelsbergh et al.....	- 277 -
Figura 8.18: Comparativa de tiempos modelo de Savelsbergh et al.	- 278 -
Figura 8.19: Comparativa de distancias modelo Dumas et al.	- 278 -
Figura 8.20: Comparativa de tiempos modelo de Dumas et al.	- 279 -
Figura 8.21: Comparativa distancias modelo de Enrico Angelelli et al.....	- 279 -
Figura 8.22: Comparativa de tiempos modelo de Enrico Angelelli et al. ..	- 280 -
Figura 9.1: % de problemas resueltos. Problema CVRP.....	- 285 -
Figura 9.2: % de problemas resueltos. Problema CVRPTW.	- 285 -
Figura 9.3: % de problemas resueltos. Problema CVRPPD.....	- 286 -
Figura 9.4: % de problemas resueltos. Problema CVRPPDTW.....	- 286 -

Figura 9.5: comparación método exacto vs heurística. Distancias obtenidas... -	287 -
Figura 9.6: comparación método exacto vs heurística. Tiempos de ejecución.. -	288 -
Figura A.1: Ejemplo fichero de datos para el problema CVRP homogéneo.-	294 -
Figura A.2: Restricción (3) sin modificar, modelo basado en redes para el problema CVRP homogéneo.....	294 -
Figura A.3: Restricción (3) modificada, modelo basado en redes para el problema CVRP homogéneo.....	295 -
Figura A.4: Distancias obtenidas. Modelo basado en TMZ de 2 índ.	296 -
Figura A.5: Tiempos de ejecución. Modelo basado en TMZ de 2 índ.....	297 -
Figura A.6: Distancias obtenidas. Modelo de Acharya.	299 -
Figura A.7: Tiempos de ejecución. Modelo de Acharya.....	299 -
Figura A.8: Distancias obtenidas. Modelo de Toth y Vigo.	300 -
Figura A.9: Tiempos de ejecución. Modelo de Toth y Vigo.	301 -
Figura A.10: Distancias obtenidas. Modelo de Savelsbergh et al.	302 -
Figura A.11: Tiempos de ejecución. Modelo de Savelsbergh et al.....	303 -

ÍNDICE DE TABLAS

Tabla 5.1: Resumen de resultados para el problema CVRP homogéneo empleando Xpress-IVE.	- 89 -
Tabla 5.2: Resultados de Xpress-IVE para el problema CVRP homogéneo.	- 90 -
Tabla 5.3: Resultados obtenidos con Xpress-IVE para el problema CVRP heterogéneo.	- 92 -
Tabla 5.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRP heterogéneo.	- 94 -
Tabla 5.5: Resultados obtenidos con CPLEX para el problema CVRP homogéneo.	- 95 -
Tabla 5.6: Resumen de resultados obtenidos con CPLEX para el problema CVRP homogéneo.	- 96 -
Tabla 5.7: Resumen de resultados obtenidos con CPLEX para el problema CVRP heterogéneo.	- 98 -
Tabla 5.8: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRP homogéneo.	- 103 -
Tabla 5.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRP heterogéneo.	- 109 -
Tabla 6.1: Resumen de resultados para el problema CVRP homogéneo empleando Xpress-IVE.	- 141 -
Tabla 6.2: Resultados de Xpress-IVE para el problema CVRPTW homogéneo.	- 142 -
Tabla 6.3: Resultados obtenidos con Xpress-IVE para el problema CVRPTW heterogéneo.	- 144 -
Tabla 6.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPTW heterogéneo.	- 146 -
Tabla 6.5: Resultados obtenidos con CPLEX para el problema CVRPTW homogéneo.	- 148 -
Tabla 6.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPTW homogéneo.	- 150 -
Tabla 6.7: Resultados obtenidos con CPLEX para el CVRPTW heterogéneo.	- 152 -
Tabla 6.8: Resumen de resultados obtenidos con CPLEX para el problema CVRPTW heterogéneo.	- 153 -

Tabla 6.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPTW homogéneo.	- 157 -
Tabla 6.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPTW heterogéneo.	- 161 -
Tabla 7.1: Resumen de resultados para el problema CVRPPD homogéneo empleando Xpress-IVE.	- 202 -
Tabla 7.2: Resultados de Xpress-IVE para el problema CVRPPD homogéneo. ..	203 -
Tabla 7.3: Resultados obtenidos con Xpress-IVE para el problema CVRPPD heterogéneo.	- 205 -
Tabla 7.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPPD heterogéneo.	- 207 -
Tabla 7.5: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD homogéneo.	- 209 -
Tabla 7.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD homogéneo.	- 210 -
Tabla 7.7: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD heterogéneo.	- 212 -
Tabla 7.8: Resultados obtenidos con CPLEX para el CVRPPD heterogéneo.	213 -
Tabla 7.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPD homogéneo.	- 218 -
Tabla 7.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPD heterogéneo.	- 224 -
Tabla 8.1: Resumen de resultados para el problema CVRPPDTW homogéneo empleando Xpress-IVE.	- 262 -
Tabla 8.2: Resultados de Xpress-IVE para el problema CVRPPDTW homogéneo.	- 263 -
Tabla 8.3: Resultados obtenidos con Xpress-IVE para el problema CVRPPDTW heterogéneo.	- 265 -
Tabla 8.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPPDTW heterogéneo.	- 267 -
Tabla 8.5: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW homogéneo.	- 269 -
Tabla 8.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW homogéneo.	- 270 -

Tabla 8.7: Resultados obtenidos con CPLEX para el CVRPPDTW heterogéneo. -	271 -
Tabla 8.8: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW heterogéneo.	- 272 -
Tabla 8.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPDTW homogéneo.....	- 276 -
Tabla 8.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPDTW heterogéneo.	- 281 -
Tabla 9.1: Resumen de resultados obtenidos en para cada problema de rutas.	- 284 -
Tabla A.1: Resumen de resultados obtenidos. Modelo basado en TMZ de 2 índices.....	- 297 -
Tabla A.2: Resumen de resultados obtenidos. Modelo de Acharya.....	- 299 -
Tabla A.3: Resumen de resultados obtenidos. Modelo de Toth y Vigo.	- 301 -
Tabla A.4: Resumen de resultados obtenidos. Modelo de Savelsbergh et al. ...	- 303 -

CAPÍTULO 1.

INTRODUCCIÓN Y OBJETIVOS.

1.1. Introducción.

En plena era de la customización (atención preferente a las necesidades del cliente), factores como la puntualidad, la flexibilidad y la calidad del transporte se han convertido en críticos a la hora de determinar la satisfacción de los clientes, y por tanto el nivel del servicio prestado por una empresa.

Con el fin de optimizar todos estos aspectos, en el último medio siglo se ha comenzado a delegar trabajos anteriormente manuales en softwares específicos, aliviando la carga soportada por los encargados de la gestión de una empresa, evitando así posibles errores debidos al “factor humano” y permitiendo resolver tareas complejas, imposibles de abarcar para una persona de cualquier otro modo.

Tras su uso, dichos softwares se han demostrado extraordinariamente efectivos ya que, con un pequeño conocimiento de lenguajes de programación y con el requerimiento de breves periodos de tiempo para su ejecución, son capaces de generar importantes ahorros, suponiendo un factor determinante de la calidad competitiva de una empresa.

En el caso concreto de actividades logísticas, día tras día se mueven en todo el mundo enormes cantidades de dinero en procesos relacionados con la recogida y distribución de mercancías. Por ello, a lo largo de los años, diversos autores han propuesto soluciones (modelos) que tratan de optimizar estos transportes empleando los citados softwares.

1.2. Objetivos.

En base a la justificación anterior, los tres Objetivos Principales (OP) del presente Trabajo Fin de Grado (TFG) son:

- OP 1: para cada tipo de problema de transporte considerado, analizar, en base a criterios específicos, los modelos propuestos por diversos autores para tratar de optimizar ese problema e intentar determinar cuál de ellos es el más eficaz.

Intrínseco a este objetivo está el hecho de realizar una revisión del estado del arte, analizando los principales modelos existentes actualmente.

- OP 2: en la mayoría de casos, el modelo propuesto por un autor se basa en emplear para el transporte un número de vehículos concreto, de manera que se limita la posibilidad de que el software determine el tamaño óptimo de vehículos a utilizar.

Por ello, para cada tipo de problema de transporte considerado, se desea determinar si el número de vehículos empleado puede optimizarse, sin que ello penalice a la ejecución del software.¹

- OP 3: como existen en el mercado multitud de softwares disponibles para llevar a cabo la optimización, se van a comparar dos de los más importantes a nivel mundial (CPLEX y Xpress-IVE). Esto nos permitirá, empleando criterios específicos, determinar para cada problema de transporte considerado, cuál de ellos es el que mejores resultados ofrece.

1.3. Estructura.

En lo relativo a su estructura, este TFG se estructura como sigue: primero, introduciremos los conceptos de problemas de rutas de vehículos, investigación operativa y programación lineal entera, pilares básicos sobre los que se apoya todo el desarrollo posterior; a continuación, para cada uno de los problemas de rutas de vehículos considerado, realizaremos una comparación de los modelos planteados y los solvers utilizados para resolverlos, evaluando

¹ Esta justificación puede verse en el Anexo, y según las conclusiones a las que ahí se llegue, decidiremos si merece la pena implementar esta optimización o no.

los resultados obtenidos; por último, el Capítulo 9 nos ofrece un resumen de las principales conclusiones a las que hemos llegado, indicando si hemos cumplido o no los objetivos del proyecto planteados en el apartado anterior.

Adicionalmente, y como ya se ha indicado en el apartado 1.2, en el Anexo encontramos la justificación en base a la cual modificaremos o no determinadas restricciones de cada tipo de problema, según lo visto en el OP 2.

CAPÍTULO 2

PROBLEMAS DE RUTAS DE VEHÍCULOS.

2.1. Introducción.

Desde que el ser humano comenzó a desarrollar interés por la innovación y la mejora continua, uno de los grandes objetivos perseguidos por las compañías ha sido el ahorro, bien sea de materiales, monetario, tiempo...

Los problemas de rutas de vehículos surgen en este contexto, movidos por la motivación de optimizar las actividades relacionadas con el transporte, tanto de mercancías como de personas, obteniendo así unos costes mínimos.

El objetivo es simple: visitar una colección de clientes o ciudades dados y, en base a las distancias entre ellos obtener la ruta que, partiendo de un punto inicial, visitando todos esos clientes o ciudades y regresando al punto de partida, recorra una distancia mínima. Esto es lo que se conoce como problema TSP (Traveling Salesman Problem) o problema del viajante de comercio.

Por simplicidad, a partir de ahora a los puntos a los que deben viajar los vehículos se les denominará o clientes o nodos (definición de nodo en el apartado 4.2), a no ser que se trate explícitamente de ciudades.

2.2. Breve historia.

Como materia, el TSP surgió mucho antes de ser considerado objeto de estudio por parte de determinadas áreas matemáticas (J.Cook, 2012). Podemos

encontrar ejemplos tanto en documentos de comerciantes de especias (base del nombre TSP) como en los circuitos que los abogados debían recorrer en el siglo XV para atender a sus clientes en los distritos que tenían asignados, entre otros, de manera que siempre buscaban minimizar la distancia total recorrida a lo largo de la ruta.

En el sentido matemático estricto, no fue hasta el siglo XVIII cuando Leonhard Euler escribió el más importante artículo hasta esa fecha describiendo problemas de ruteo de vehículos y aplicándolo al problema de los puentes de Königsberg (Euler, 1741). En este documento, además, se introdujo por vez primera el concepto de grafo, definiéndolo como un conjunto de vértices y arcos que unían dichos vértices, y que nos permitía representar la solución del problema de forma gráfica (ver apartado 4.2 y Figura 2.1, extraída de (Desconocido, 2016) .

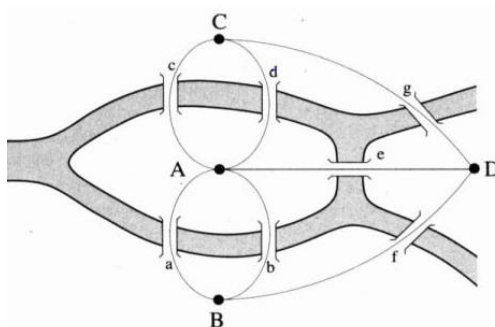


Figura 2.1: Puentes de Königsberg y simplificación como grafo.

Pocos años más tarde, el mismo Euler publicó un desafío (conocido popularmente como Knight’s tour) en el que proponía encontrar la secuencia que, con el menor número de movimientos, consiguiera hacer que un caballo recorriera todas las casillas de un tablero de ajedrez (movimiento en L) exactamente una sola vez, volviendo al punto de origen (Euler, 1766). Este problema podía ser enmarcado fácilmente empleando el lenguaje de grafos.

1	48	31	50	33	16	63	18
30	51	46	3	62	19	14	35
47	2	49	32	15	34	17	64
52	29	4	45	20	61	36	13
5	44	25	56	9	40	21	60
28	53	8	41	24	57	12	37
43	6	55	26	39	10	59	22
54	27	42	7	58	23	38	11

Figura 2.2: Solución obtenida por Euler para el desafío Knight's tour.

Ya en el siglo XIV, Sir William Rowan Hamilton, basándose en la teoría de grafos ya trazada por Euler, estudió formas de visitar por medio de arcos los veinte vértices de un dodecaedro, pasando por cada uno de ellos exactamente una vez. Para lograrlo, trasladó el problema a dos dimensiones, creando un elemento conocido como "The Icosian" y resolviéndolo (Laporte, "A Short History Of Travelling Salesman Problem", 2006).

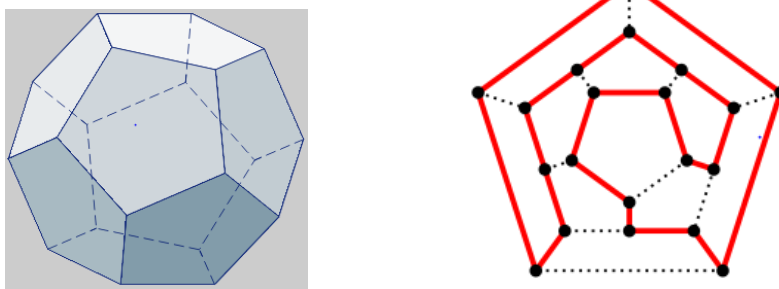


Figura 2.3: A la izquierda, un dodecaedro; A la derecha, The Icosian (1859) y la solución propuesta por Hamilton.

A raíz de estos avances, surgen dos tipologías de problemas, ambas basadas en la teoría de grafos: la primera, conocida con el nombre de circuito Euleriano, busca crear una ruta cerrada viajando por cada arco una sola vez, volviendo al punto de partida; la segunda, conocida con el nombre de circuito Hamiltoniano busca visitar cada vértice exactamente una vez, volviendo al punto de partida.

Sin embargo, tanto Euler como Hamilton sólo se dedicaron a estudiar la creación de rutas, pero no la optimización de las mismas. Si bien es cierto que en torno al año 1920 Karl Menger propuso un problema muy similar a lo que conocemos hoy en día como problema TSP (problema del mensajero), difiriendo de este en que no era necesario retornar al punto de partida, no fue hasta el año 1934 de la mano del matemático de Harvard Hassler Witney cuando se empezó a tener noción del problema como tal en un seminario impartido por este en la Universidad de Princeton en 1934 (J.Cook, 2012).

Si bien el propio Hassler reconoció no haber tenido intención de formular ningún problema, dicho seminario sirvió de inspiración para que en 1937 Merrill Flood (Universidad de Columbia) tratara de obtener soluciones óptimas para el ruteo de autobuses de transporte escolar, siendo este el principal responsable de publicitar el problema TSP ante la comunidad matemática de manera oficial.

2.3. Tipología de problemas.

A raíz de la formulación original planteada por Flood (1937), surgieron a lo largo de los años nuevos problemas relacionados con el ruteo de vehículos (Rocha, González, & Orijuela, 2011) (Toth & Vigo, 2002). Así, en 1959 de la mano de Dantzing y Ramser surgió el conocido como problema TSP generalizado, en el cual se modela la distribución de combustible a través de una flota de camiones a diferentes estaciones de servicio. Tan sólo un año después, en 1960, Tucker, Miller y Zemlin propusieron el problema TSP múltiple o m-TSP, el cual trata de visitar a un conjunto de clientes una sola vez por medio de m vehículos, trazando m rutas.

Tanto el TSP generalizado como el m-TSP sirvieron como marco para la creación de un nuevo tipo de problema, el VRP (Vehicle Routing Problem), en el cual ya no se tiene un solo vehículo como ocurría en el TSP original, sino múltiples vehículos, cada uno de los cuales debe partir de un origen, visitar una serie de clientes y volver al punto de origen, quedando cada cliente visitado una sola vez y minimizando la distancia total recorrida (y por consiguiente el coste asociado). La Figura 2.4 muestra un ejemplo de este problema (Universidad Pablo Olavide, 2016)

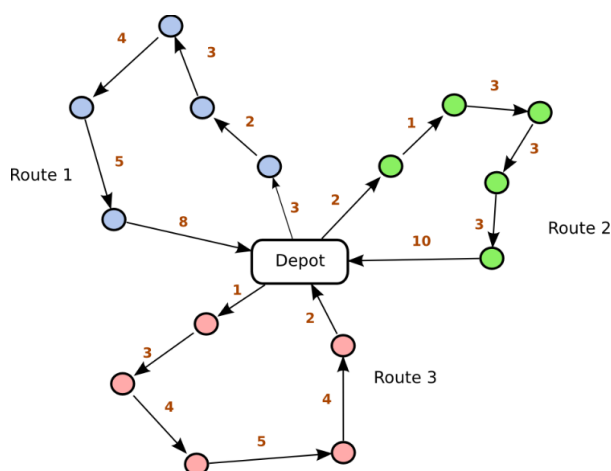


Figura 2.4: ejemplo gráfico de un problema VRP.

Es a partir del VRP cuando comienzan a desarrollarse una gran variedad de problemas, cada uno de los cuáles con unas características determinadas que

expondremos a continuación (Rocha, González , & Orijuela, 2011) (Toth & Vigo, 2002):

- CVRP (Capacitated Vehicle Routing Problem): una extensión del VRP, diferenciado de este en que ahora cada cliente tiene una demanda que debe ser satisfecha, de manera que la capacidad del vehículo se convierte en una restricción del problema. El objetivo es, atendiendo la demanda de cada cliente y sin sobrepasar la capacidad de cada vehículo, minimizar la distancia total recorrida en las rutas o el coste asociado a las mismas.

Dentro de este tipo podemos encontrar dos subtipos de problemas dependiendo de la relación existente entre las características de los clientes:

- SCVRP (Symmetric Capacitated Vehicle Routing Problem): el costo/distancia de ir de un cliente A al cliente B es igual que el de ir del cliente B al cliente A. Será el concepto que emplearemos durante todo el TFG.
- ACVRP (Asymmetric Capacitated Vehicle Routing Problem): cuando el coste/distancia de ir del cliente A al B es distinto que el de ir del cliente B al A.

Además, existe una segunda subdivisión en función de las características de la flota de vehículos:

- Flota homogénea: todos los vehículos tienen la misma capacidad máxima.
 - Flota heterogénea: cada vehículo tiene su propia capacidad máxima. Los problemas de flota homogénea son un caso particular de la flota heterogénea en donde todas las capacidades son iguales.
- DCVRP (Distance-constrained Capacitated Vehicle Routing Problem): variante del problema CVRP en donde, además de respetar la capacidad de cada vehículo, existen restricciones sobre la distancia máxima que puede recorrer dicho vehículo.
 - SDCVRP (Split Delivery Capacitated Vehicle Routing Problem): extensión del problema CVRP, en donde un cliente puede ser atendido varias veces por distintos vehículos, siempre y cuando se minimice la distancia/coste total. Este tipo de problemas es especialmente interesante cuando la demanda de un cliente es superior a la capacidad de los vehículos.

- PCVRP (Periodic Capacitated Vehicle Routing Problem): en los CVRP clásicos, el periodo que abarca la planificación es sólo de un día. En el caso de los PCVRP, el periodo de planificación se extiende a M días.
- MDCVRP (Multiple-Depot Capacitated Vehicle Routing Problem): variante del problema CVRP, en donde no se tiene un solo punto de origen, sino que se puede partir de varios puntos.

Dicho tipo de problema puede ser extendido al VRP, siempre y cuando eliminemos las restricciones de demandas y capacidades.

- CVRPTW (Capacitated Vehicle Routing Problem with Time Windows): una extensión del CVRP, en donde además de las restricciones de demandas y capacidades a cada cliente i se le asocia un intervalo de tiempo (ventana de tiempo) $[a_i, b_i]$ durante el cual tiene que ser atendido. Aquí entran en juego nuevos parámetros como el tiempo que dura el servicio en cada cliente o el tiempo empleado en ir de un cliente al siguiente en la ruta.

Dichas restricciones de ventanas de tiempo pueden ser igualmente formuladas tanto para los problemas TSP como VRP, eliminando las demandas de los clientes y en función de si empleamos uno o varios vehículos, respectivamente. Esto da lugar a variantes de esos problemas, el TSPTW (Traveling Salesman Problem with Time Windows) y el VRPTW (Vehicle Routing Problem with Time Windows), respectivamente

- CVRPB (Capacitated Vehicle Routing Problem with Backhauls): en este tipo de problemas, los clientes son divididos en dos grupos. En el primero de ellos, denominado “linehaul”, están aquellos clientes a los que hay que entregar una determinada cantidad de mercancía. En el segundo de ellos, denominado “backhaul”, están aquellos clientes de los que hay que recoger mercancía. La principal restricción es que, para cada ruta, las entregas deben hacerse antes que las recogidas.

No se modeliza este tipo de problema en el presente TFG, dada su similitud con el CVRPPD, tipo de problema que veremos a continuación y para el cuál sí que hemos planteado modelos.

- CVRPPD (Capacitated Vehicle Routing Problem with Pick up and Delivery): en este tipo de problema, cada cliente tiene una cantidad que debe ser recogida (pick up) en un punto y que debe ser entregada

(delivery) en otro punto, de manera que podríamos decir que cada cliente se “divide” en dos clientes, uno de recogida y otro de distribución. La idea es la siguiente: dentro de una misma ruta, el vehículo que atiende al cliente i debe pasar primero por el lugar de pick up y posteriormente (de manera directa o atendiendo a otros clientes entre medias) por el lugar de delivery.

Igual que ocurría en los casos anteriores, las capacidades de los vehículos deben ser respetadas, y se busca minimizar la distancia/coste asociado al conjunto de rutas realizadas.

Al tratarse de un problema de pick up y delivery, se transportan productos o materiales entre los distintos nodos. En caso de transportarse personas, el problema se denomina comúnmente Dial a Ride, siendo el planteamiento equivalente.

Además, típicamente se suele dividir este tipo de problema en tres subtipos (Cordeau, Laporte, & Ropke, 2007):

- One-to-one: cada nodo de pick up nos proporcionará un producto que debe de ser entregado en su correspondiente nodo de delivery, de manera que, si el nodo A es el nodo de pick up del nodo de delivery A' y B lo es de B' , no se puede recoger mercancía de A para entregarla en B' o de B para entregarla en A' , ya que las demandas A' y B' corresponden a productos diferentes.

Además, partimos de un depot o nodo origen, y tras atender todos los pedidos, finalizamos la ruta en otro depot o nodo destino, que puede o no coincidir con el nodo origen.

La Figura 2.5 ilustra este funcionamiento. En ella, los nodos 1 y 3 son nodos de pick up de los ítems a y b , respectivamente, en tanto que los nodos 2 y 4 son los nodos de delivery de los nodos 1 y 3, respectivamente.

La ruta trazada se interpretaría como sigue: partiendo del nodo origen, vamos al nodo pick up 1, en donde recogemos el producto a ; después, vamos al nodo pick up 3, donde recogemos el producto b ; con los productos a y b en el vehículo, llegamos al nodo delivery 2, en donde depositamos el producto b ; por último, en el nodo delivery 4, entregamos el producto a y llegamos al nodo destino.



Figura 2.5: One-to-one problem.

- Many-to-many: sería equivalente a un one-to-one en donde sólo se tiene un tipo de producto, de manera que la cantidad recogida en un nodo pick up puede ser repartida entre múltiples nodos delivery.

La Figura 2.6 ilustra este comportamiento. En ella vemos como, tras visitar los nodos de pick up 2 y 4, el vehículo transporta 8 unidades de un ítem (5 recogidas en el nodo 2 y 3 recogidas en el nodo 4). Cuando llega al nodo delivery 1, se depositan las 6 unidades que este nodo demanda, que consisten (por ejemplo) en 4 unidades del nodo 2 y 2 del nodo 4. Por último, el vehículo va al nodo delivery 3, en donde deposita las 2 unidades demandadas correspondientes, llegando al nodo destino con la unidad restante.



Figura 2.6: Many-to-many problem.

- One-to-many-to-one:

Aquí, cada nodo presenta dos indicadores: una demanda y una oferta, de manera que cuando un vehículo visita un nodo, entrega una cantidad correspondiente a la demanda de dicho nodo y recoge la cantidad ofertada por el nodo, con lo que vamos reponiendo la

carga del vehículo. Este funcionamiento implica que el vehículo debe salir cargado del nodo origen.

La Figura 2.7 representa este caso.



Figura 2.7: One-to-many-to-one problem.

- A partir de este punto, se suele producir una mezcla de los problemas anteriormente citados, como podemos observar en el siguiente gráfico (Toth & Vigo, 2002):

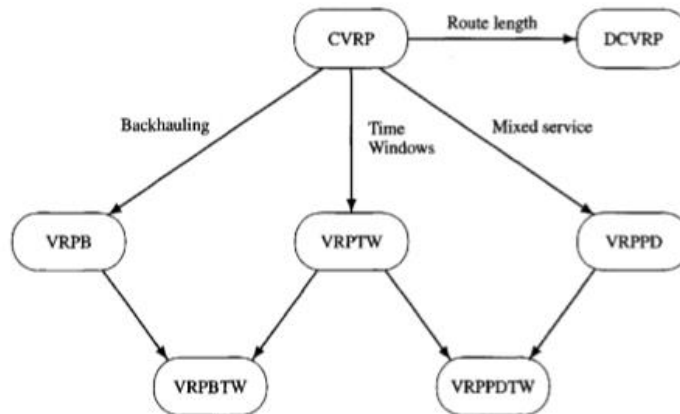


Figura 2.8: subdivisiones básicas del VRP/CVRP y sus interconexiones.

Así, por ejemplo, el problema CVRPPDTW queda definido como un problema con pick up y delivery, en donde cada cliente debe ser atendido en el marco de las ventanas de tiempo que posea, entrando en juego otros parámetros como el tiempo que dura el servicio en el cliente o el tiempo empleado en ir de un cliente al siguiente.

Es posible que existan algunas divisiones o variantes del problema que no se hayan citado en el presente apartado debido a que no son tan conocidas y no suelen utilizarse. No obstante, dentro de las citadas, cabe destacar CVRP, CVRPTW, CVRPPD y CVRPPDTW como unas de las más empleadas, que serán las que se plantearán en este TFG, junto con las formulaciones iniciales del problema TSP.

2.4. Criterios de comparación.

Dentro de cada tipo de problemas, a lo largo de los años han ido surgiendo múltiples formulaciones matemáticas que buscan obtener mejores resultados que sus antecesores. A cada una de estas formulaciones se le denomina comúnmente modelo.

Así, en el marco del objetivo principal OP 1 definido en la introducción, se identifican tres criterios para comparar los distintos modelos planteados, en orden de identificar el más eficiente de la forma más clara y objetiva posible:

- Distancias totales obtenidas.
- Tiempos de ejecución obtenidos.
- Número de problemas para los que se ha encontrado una solución factible en los 200 segundos de ejecución máxima marcados.

Además, se evaluará cada tipo de problemas concreto de una manera objetiva que nos permita obtener la conclusión más adecuada, de manera que si, por ejemplo, de darse el caso en que el Modelo 1 resuelva 6 de 7 problemas y el Modelo 2 resuelva 2 de 7 pero con unos resultados mejores que el Modelo 1, se seleccionará el Modelo 1 como modelo más adecuado, ya que es el que en líneas generales presenta unos resultados más aceptables.

2.5. Aplicaciones.

Sólo con su nomenclatura los problemas de ruteo de vehículos (TSP, VRP, CVRP...) dan una idea de la gran variedad de aplicaciones prácticas en las que pueden ser empleados. Sin embargo, existen muchas otras que a priori nos resultan desconocidas y para las que resolver problemas de rutas se ha convertido en un mecanismo fundamental de optimización de procesos.

En el presente apartado hablaremos sobre algunas de las principales aplicaciones que tienen los dichos problemas actualmente (J.Cook, 2012):

- Trazado de rutas (logística): posiblemente la aplicación más común de este tipo de problemas, en la cual tratamos de optimizar rutas de transporte, respetando ciertas restricciones e intentando minimizar la distancia/costes asociados. Veamos una serie de ejemplos:

- Viajante de comercio: desde sus orígenes, los problemas de ruteo de vehículos han sido empleados para facilitar la vida de comerciantes y transportistas, optimizando las rutas que estos debían recorrer.

Con la llegada de la era digital, los Global Positioning System (GPS) se han convertido en una herramienta ampliamente utilizada para esta función. Muy frecuentemente, los sistemas software de mapeo de dichos dispositivos incluyen algoritmos de resolución para problemas de rutas.

- Pick up y delivery: un uso muy común de los problemas de ruteo de vehículos es la planificación de las rutas de los autobuses escolares y de furgonetas de recogida/reparto. Un ejemplo práctico de estos casos sería cuando Merrill Flood introdujo de manera oficial el problema TSP (1937) para el primero y cuando Rapidis (firma especializada en transporte y logística) empleó la herramienta Concorde (apartado 3.2.3) para establecer las rutas de su cliente Forbruger-Kontakt.
 - Planificar inspecciones en lugares remotos. William Pulleyblank planificó su visita a 47 plataformas petrolíferas de Nigeria en base a los resultados obtenidos al resolver un problema TSP.
 - The Iron Butt Rally: en 2009, Maura Gatensby aplicó los algoritmos de Dantzing-Fulkerson-Johnson (Capítulo 3) para optimizar su ruta a través de 48 estados de los Estados Unidos en un recorrido que debía durar 10 días (desafío propuesto por la Asociación de motocicletas Iron Butt).
- Optimizar movimiento de telescopios: aunque normalmente asociamos los problemas de rutas con aplicaciones que requieren un desplazamiento físico a ciertas localizaciones, estos problemas son igualmente válidos cuando dichas localizaciones se encuentran a elevadísimas distancias. Un ejemplo típico es cuando las

observaciones de planetas, estrellas o galaxias tienen que ser hechas desde un telescopio.

Al proceso de rotación para conseguir realizar las observaciones con un telescopio se le denomina slewing. Para grandes telescopios, dicho proceso consume mucho tiempo y energía. Por tanto, un tour generado por un problema TSP que minimice el tiempo total de slewing puede suponer un ahorro considerable.

En un artículo publicado en Scientific American, Shawn Carlson describió cómo gracias a una heurística del TSP había conseguido que un telescopio recorriera 200 galaxias por noche, algo impensable hasta ese momento.

- **Búsqueda de planetas:** cuando el telescopio no se encuentra en un lugar fijo, sino que se encuentra en movimiento en el espacio (en un satélite o nave), una aplicación muy común suele ser la búsqueda de planetas, basada en los mismos principios que el punto anterior, en la que la resolución de un problema TSP determina la secuencia de las observaciones a realizar por el citado telescopio.
- **Guiado de máquinas industriales:** Wladimir Nickel, un ingeniero electrónico de Alemania, empleaba una máquina CNC (Controlada Numéricamente por Computador) para soldar en 256 localizaciones específicas. Por medio de la herramienta Concorde (apartado 3.2.3) resolvió el problema TSP asociado, lo que repercutió en un tiempo total de soldado mucho menor.
- **Organización de trabajos (scheduling):** la firma alemana BÖWE CARDTEC consiguió reducir en un 65% el tiempo total de set up de su proceso productivo resolviendo el problema TSP de scheduling asociado por medio de la herramienta Concorde.
- **Organización de datos:** el estilo de vida al que nos hemos acostumbrado en el siglo XXI hace que diariamente surjan nuevas commodities que requieren del tratamiento de una gran cantidad de datos. Uno de los ejemplos más típicos es la industria de los videojuegos, que ha experimentado un crecimiento exponencial en sus prestaciones en sus últimos años. En este contexto, la resolución de problemas TSP permite introducir de manera más rápida, por ejemplo, los datos sobre las texturas necesarias, garantizando unas

transiciones de escena a escena en el videojuego mucho más suaves.

2.6. Complejidad del problema.

Dependiendo de la complejidad computacional que un problema conlleve, existen diversas clases en las que dicho problema puede ser enmarcado (J.Cook, 2012) (Araúzo Araúzo, 2016):

- Problemas tipo P (Polynomial): cuando se conoce al menos un algoritmo polinómico para resolver el problema.²
- Problemas tipo NP (Non-deterministic Polynomial): cuando puede aplicarse un algoritmo polinómico para comprobar si una posible solución dada es válida o no.
- Problemas tipo NP-Complejos: un caso particular de los NP, de extrema complejidad, en donde se cree que no existen algoritmos polinómicos para resolver estos problemas, pero no se puede demostrar.

Dentro de esta clasificación, los problemas de rutas de vehículos están clasificados como NP, de manera que nosotros podemos comprobar si una solución es correcta por medio de un algoritmo polinómico.

Partiendo de esto, diversas herramientas han sido desarrolladas para resolver problemas de rutas, de manera que nosotros podemos ir evaluando cada posible solución y comprobando, en base a unas restricciones dadas, si dicha solución es posible o no. Sobre este tema versará el capítulo siguiente.

² Un algoritmo polinómico es aquel cuya función de complejidad es $O(n^a)$ o menor, siendo problemas eficientes. Varios ejemplos de algoritmos polinómicos podrían ser el constante ($O(1)$), el lineal ($O(n)$), el logarítmico ($O(\log(n))$)...

CAPÍTULO 3

INVESTIGACIÓN OPERATIVA Y PROGRAMACIÓN LINEAL.

3.1. Introducción.

La investigación operativa en general y la programación lineal en particular han impulsado en las últimas décadas las actividades de optimización, haciendo posible la resolución de problemas que hasta hace poco se antojaban irresolubles.

A continuación, describiremos estos dos conceptos, hablaremos de sus orígenes y de las principales herramientas o softwares en las que hoy en día se apoyan para resolver problemas (solvers).

3.2. La investigación operativa.

También conocida como Investigación de operaciones u Operations Research (nombre en inglés), se trata de una metodología surgida en torno a los años 50, y que se aplica a multitud de sectores de producción y manufactura, distribución, transporte, telecomunicaciones y salud, entre otros muchos otros (Sáez Aguado, 2014).

Esencialmente consta de dos partes: la modelización, o construcción de modelos matemáticos y la optimización, o resolución de estos modelos. Dichas

partes se encuentran fuertemente relacionadas y son definidas brevemente a continuación.

3.2.1. Modelización. Programación matemática.

La modelización o programación matemática es un área de la Matemática que trata de la resolución de modelos de optimización. Típicamente, un modelo de optimización tiene la siguiente forma:

$$\begin{array}{ll}
 \text{Maximizar o minimizar} & f(x_1, x_2, \dots, x_n) \\
 \text{Sujeto a} & g_1(x_1, x_2, \dots, x_n) \leq b_1 \\
 & g_2(x_1, x_2, \dots, x_n) \leq b_2 \\
 & \dots \\
 & g_m(x_1, x_2, \dots, x_n) \leq b_m \\
 & x_1, x_2, \dots, x_n \geq 0
 \end{array}$$

Como vemos, consta de varios elementos, los cuales podemos clasificar de la siguiente manera:

1. Variables de decisión (x_1, x_2, \dots, x_n): son las incógnitas cuyo valor se quiere determinar. Dependen del tipo de problema, con lo cual es muy común encontrarse variables de decisión de diferentes tipos (enteras si sólo pueden tomar valores enteros, binarias si pueden valer 1 ó 0...).
2. Función objetivo ($f(x_1, x_2, \dots, x_n)$): es una función matemática que representa aquello que deseamos optimizar, ya sea maximizándola o minimizándola. Para que nos hagamos una idea, un ejemplo tipo de función objetivo que se desee minimizar sería un coste o una distancia recorrida, en tanto que una que se desee maximizar sería un beneficio.
3. Restricciones ($g_1(x_1, x_2, \dots, x_n) \leq b_1$): son las condiciones que debe cumplir la solución del problema. El \leq es el tipo de restricción, y puede ser igualmente $<$, $>$, $=$, \geq . Cuanto más fielmente representemos las restricciones del sistema real, mejor se adecuará nuestra solución a la realidad.

En función de las características que posean estos tres elementos, existen diferentes tipos de problemas de programación matemática, pudiendo clasificarse los más importantes en:

- a) Problemas de programación lineal: tanto la función objetivo como las restricciones deben de ser lineales, es decir, de la forma:

$$\text{Función objetivo: } f(x) = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

$$\text{Restricciones: } a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq, =, \geq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq, =, \geq b_2$$

...

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq, =, \geq b_m$$

Siendo c_1, \dots, c_n y a_{11}, \dots, a_{mn} los coeficientes de la función objetivo y de las restricciones, respectivamente, no pudiendo existir ninguna variable no lineal (sinusoidal, exponencial...) ni ningún producto o cociente entre variables.

El problema de programación lineal consiste en determinar los valores de las variables x_1, x_2, \dots, x_n continuas que proporcionen el mejor valor para la función objetivo $f(x)$, cumpliendo todas las restricciones propuestas.

- b) Problemas de flujo en redes: caso particular de la programación lineal, en donde las variables corresponden a los flujos definidos sobre un grafo o una red. Como vimos en el apartado 2.2, el concepto de grafo fue introducido por Euler para resolver el problema de los puentes de Königsberg (Euler, 1741). Este concepto será definido con mayor profundidad en el apartado 4.2.
- c) Problemas de programación entera: otro caso particular de la programación lineal, en donde se restringe el formato de las variables continuas que teníamos en los problemas de programación lineal al considerarlas números enteros. Distinguimos dos tipos de variables, enteras generales (≥ 0) y enteras binarias ($\in [0,1]$).
- d) Programación no lineal: en contraposición con los problemas de programación lineal, en este caso la función objetivo y/o las restricciones pueden ser funciones no lineales (puede existir producto/cociente entre variables, funciones seno, coseno, exponencial...).

Debe prestarse mucha atención a la definición del problema, ya que, por ejemplo, la selección de unas variables erróneas o una elevada cantidad de las mismas nos puede condicionar de manera negativa el modelo. Por el contrario, la correcta definición de la función objetivo y las restricciones nos permiten

representar mejor el sistema real, haciendo que los resultados obtenidos sean más fiables.

3.2.2. Optimización. Resolución de los modelos.

Una vez planteado el modelo de forma adecuada, se puede aplicar alguno de los métodos disponibles para resolverlo, obteniéndose como resultado unos valores de las variables y de la función objetivo planteada.

Si la modelización ha sido correcta, dichos resultados constituirán una solución real del problema.

3.3. La programación lineal entera.

Muchos problemas de optimización pueden modelizarse de forma lineal, por lo que esta metodología es la más empleada dentro de la investigación operativa. Los problemas de ruteo de vehículos propuestos en este TFG se corresponden con problemas de programación lineal entera pura o mixta, siendo una mezcla de los anteriormente definidos problemas de flujo en redes y programación entera.

3.3.1. Historia. Origen en el TSP.

En la primavera de 1962, una campaña publicitaria patrocinada por la multinacional estadounidense Procter & Gamble (J.Cook, 2012) planteó un desafío basado en el problema TSP que otorgaría un premio de \$10,000 a quien consiguiera proponer la ruta más corta, que partiendo de Chicago (Illinois) volviera a Chicago (Illinois) pasando por otras 32 poblaciones de los Estados Unidos.

Dicha campaña ocasionó un gran revuelo entre la comunidad matemática, no sólo porque en ella aparecieran Toody y Muldoon, dos personajes de la popular serie Car 54, sino porque el planteamiento a primera vista simple del problema era contrarrestado por la complejidad de su resolución.



Figura 3.1: Concurso car 54.

La propuesta inicial de los matemáticos de la época de probar cada posible combinación de rutas y elegir entre ellas la más corta fue rápidamente desestimada debido a que, suponiendo que nos diera igual ir de una ciudad A a otra ciudad B que de B a A, el número de combinaciones posibles sería de $32!/2 = 1,31 * 10^{35}$, totalmente imposible de resolver manualmente.

Es más, incluso hoy en día, realizar semejante cantidad de iteraciones con uno de los computadores más potentes del mundo, el IBM Roadrunner Cluster del Departamento de Energía de los Estados Unidos llevaría la friolera de 28 trillones de años.

¿Qué hacer? La respuesta la encontramos pocos años antes cuando, en 1954, un grupo de matemáticos de RAND Corporation consiguieron resolver por programación lineal un TSP de 49 ciudades (10^{27} veces mayor número de iteraciones que en el concurso car 54). Estos matemáticos fueron George Dantzig, Ray Fulkerson y Selmer Johnson.

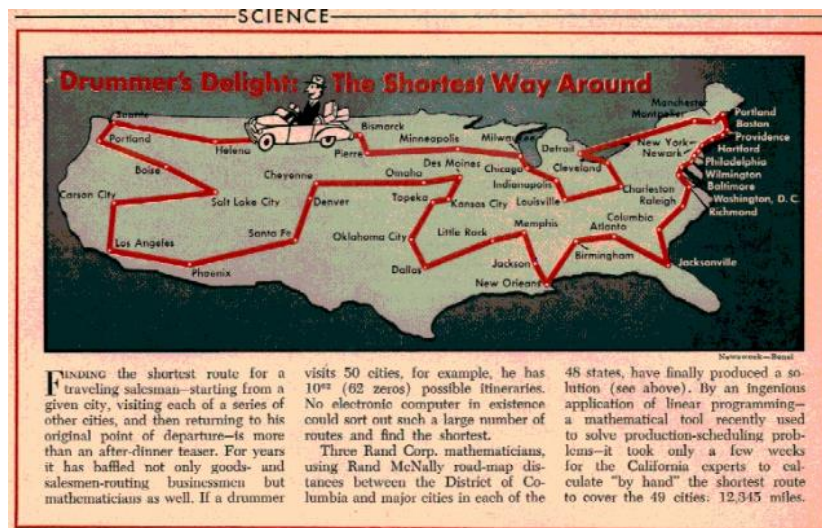


Figura 3.2: Solución de Dantzing et al. al problema TSP de 49 ciudades.

Aunque sus planteamientos no fueron utilizados en el desafío de Procter & Gamble de 1962, sí lo fueron en los años siguientes, en donde rutas optimizadas en problemas TSP de 80 ciudades (Panagiotis Miliotis en 1975), 120 ciudades (Martin Grötschel en 1997), 318 ciudades (Harlan Crowder) y así sucesivamente, llegando hasta el mayor problema TSP resuelto hasta la fecha, que consta de 85900 ciudades, resuelto empleado el código Concorde (apartado 3.2.3) (J.Cook, 2012).

3.3.2. Métodos de resolución.

3.3.2.1. Introducción.

En lo que respecta a los métodos de resolución de los problemas de ruteo de vehículos, estos se suelen agrupar en tres grandes categorías: métodos exactos, heurísticas y meta heurísticas (Rocha, González, & Orijuela, 2011). Los primeros buscan soluciones óptimas por medio de modelos matemáticos resueltos mediante el empleo de solvers específicos, lo cual suele acarrear tiempos de computación bastante elevados. Las heurísticas y meta heurísticas, por su parte, buscan dar soluciones aproximadas suficientemente buenas con tiempos de computación cortos.

Otra diferencia entre estos tipos de métodos de resolución es que los métodos exactos precisan de un software específico para obtener alguna solución, mientras que las heurísticas y meta heurísticas no, pudiendo implementarse en cualquier lenguaje de programación y ejecutarse sin necesidad de tener algún algoritmo implantado.

Como ya se ha mencionado en el Capítulo 1, uno de los objetivos de este TFG se centra en la comparación para algunas de las variantes del VRP de los principales modelos exactos conocidos, intentando establecer una jerarquía en cuanto a su efectividad. Las heurísticas y meta heurísticas se citarán a continuación, pero no forman parte de dichos objetivos, quedando como posibles líneas futuras de trabajo.

3.3.2.2. Métodos exactos

Debido a que el tiempo de computación necesario para resolver un problema de ruteo de vehículos por medio de métodos exactos crece exponencialmente con el número de clientes a considerar, estos métodos presentan limitaciones cuando se trata de resolver problemas de gran tamaño.

Típicamente se suele clasificar estos métodos en tres grupos (Rocha, González , & Orijuela, 2011) (Laporte, 1991):



Figura 3.3: taxonomía para los métodos de resolución exactos.

Como el número de algoritmos existentes dentro de cada uno de estos métodos es muy elevado, hablaremos sólo de los algoritmos de Branch & Bound y de Branch & Cut para búsqueda directa de árbol y del método simplex para la programación lineal. La programación dinámica, por su parte, es menos empleada, por lo que sólo se mencionará.

- Métodos de búsqueda directa de árbol: la búsqueda se realiza sobre todos los nodos de un árbol de acuerdo con criterios específicos de cada método. La dificultad reside en que hay que reprogramarlos para cada problema tratado, de manera que no resultan muy flexibles en ese sentido.
- Algoritmo de Branch & Bound: basados en programación lineal, este algoritmo de enumeración parcial está basado en realizar una “poda” en los distintos nodos del árbol, en base a planos de corte y cotas dadas por determinadas restricciones. Estas podas permiten, resolviendo la relajación lineal del problema entero, determinar si una solución no es factible o es peor que la mejor solución factible encontrada hasta el momento, y detener la ejecución ahí, pasando a evaluar otras ramas del árbol que sí puedan mejorar la solución actual.
- Algoritmo de Branch & Cut: similar al anterior, pero en vez de resolver una relajación lineal sencilla, emplea planos de corte para “podar” muchos más nodos al tratarse de una relajación muy fuerte.

La Figura 2.10 muestra un ejemplo de este algoritmo, empleado por muchos solvers para resolver problemas de programación lineal entera. Ahí se pueden apreciar con más claridad conceptos como árbol (definido por las ramas que van surgiendo) y poda (si obtenemos una solución no factible, de ese punto no surgirán ramas).

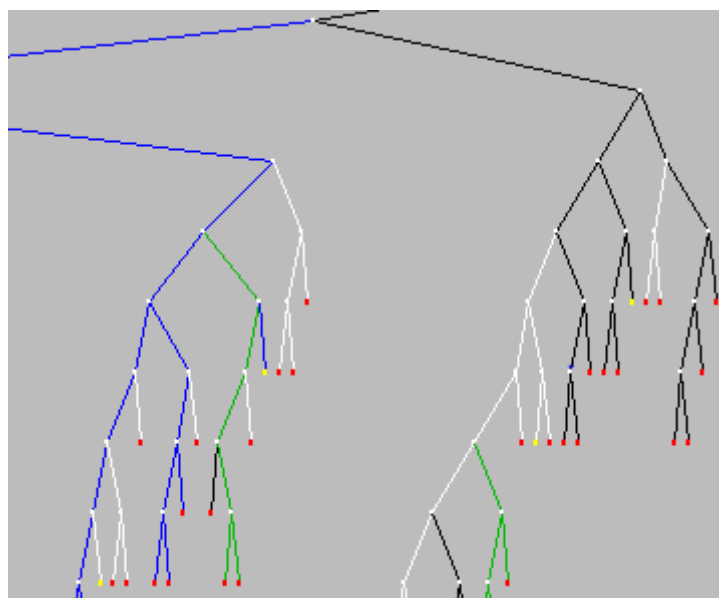


Figura 3.4: ejemplo práctico del algoritmo de Branch & Cut.

- Programación dinámica: al igual que en el caso de los métodos de búsqueda directa de árbol, hay que reprogramarlos para cada problema tratado, por lo que no resultan muy utilizados y no se hablará más de ellos.
- Programación lineal: al contrario que en los casos anteriores, al programar empleando este método se logra una gran flexibilidad, de manera que una única programación vale para cualquier ejemplo de un mismo problema.

Su algoritmo más conocido es el método simplex, desarrollado en torno al año 1950 por George Dantzing, el cual nos permite, dándole como dato un modelo formulado en programación lineal continua, obtener la solución óptima del mismo.

Este, en conjunción con algoritmos de Branch & Cut, es empleado para resolver problemas de programación lineal entera por los buenos resultados que generan.

3.3.2.3. Métodos heurísticos y meta heurísticos.

Como ya se mencionó anteriormente, la finalidad de estos métodos es encontrar soluciones lo mejores posibles (que no tienen por qué ser óptimas, como ocurría en los métodos exactos) con tiempos de ejecución muy reducidos, aportando una mayor flexibilidad y haciendo que su ejecución resulte más cómoda.

La diferencia principal entre los métodos heurísticos y los meta heurísticos reside en que los primeros suelen quedarse atascados en óptimos locales, de manera que podemos creer que una solución es muy buena, pero resulta que sólo lo es en el entorno local, mientras que los segundos tratan de superar estos óptimos locales, evaluando todo el rango de soluciones posibles (entorno global).

En lo relativo a su clasificación, tanto de los métodos heurísticos como de los meta heurísticos existen multitud de implementaciones posibles, más numerosas a medida que pasan los años. Como no son objeto de estudio por parte de este TFG, simplemente nombraremos algunas de las más importantes, dejando para futuros trabajos su análisis.

- Métodos heurísticos.
 - De construcción: buscan construir una solución factible. Aproximadamente todas están basadas en el método Greedy.
 - Algoritmo de ahorros de Clark & Wright.

- Inserción secuencial.
- De mejora: a partir de una solución factible dada por los métodos de construcción, tratan de mejorarla.
 - 2 opt.
 - 3 opt.
 - Algoritmo de intercambios.
- Métodos meta heurísticos.
 - Recocido simulado (Simulated Annealing).
 - Búsqueda tabú.
 - Algoritmos genéticos.
 - Colonia de hormigas.

3.3.3. Herramientas de implementación.

Desde su creación, el método simplex, junto con algoritmos de Branch & Cut han sido incluidos en una gran variedad de programas (cada uno de ellos incorporando alguna modificación en el algoritmo de resolución), a los cuáles se les denomina comúnmente solvers, dado que se encargan de resolver problemas de programación lineal.

Podríamos dividir los principales solvers existentes en dos grupos (Meindl & Templ, 2012):

- Solvers comerciales (código cerrado):
 - CPLEX: desarrollado y comercializado por IBM (IBM, 2012), se caracteriza por poder soportar problemas con un gran número de variables. Incorpora además varias interfaces para conectar el solver con diferentes programas y lenguajes.
 - Xpress-IVE: desarrollado íntegramente para resolver problemas lineales, incorpora también interfaces y librerías para ampliar sus prestaciones (Fico, 2012). Las herramientas de tuning y mapping son muy útiles y frecuentemente empleadas para evaluar los valores de los parámetros del sistema y representar las soluciones, respectivamente.

- Gurobi: solver empleado tanto para resolver problemas lineales como no lineales. Está escrito en C y es accesible para diversos lenguajes de programación (Gurobi, s.f.).
- Concorde: basado en el código que lleva el mismo nombre y desarrollado por William Cook, permite resolver problemas TSP simétricos, ostentando el record de problema TSP con mayor número de nodos optimizado (85900). Se encuentra disponible a través de internet (Cook W. , s.f.)
- Solvers gratuitos (código abierto):
 - LP_SOLVE.
 - GLPK.
 - CLP.
 - SCIP.
 - SoPlex.

3.3.4. Solvers seleccionados y criterios de comparación.

Como se vio en el apartado 1.2, el segundo objetivo de este TFG es el de realizar la comparación de dos de los más importantes solvers existentes, CPLEX y Xpress-IVE, para ver si alguno de los dos ofrece en líneas generales mejores resultados que el otro para cada tipo de problema planteado, pudiendo seleccionarse como el solver más adecuado para resolver los problemas de ese tipo.

Esta comparación se realiza en base a unos criterios cuidadosamente seleccionados, que son los siguientes:

- Tres criterios principales:
 - Distancias totales obtenidas.
 - Tiempos de ejecución obtenidos.
 - Número de problemas para los que se ha encontrado una solución factible en los 200 segundos de ejecución máxima marcados.

Así, se empleará un razonamiento lógico deductivo para identificar el solver más adecuado en base a estos criterios. Esto quiere decir que se evaluará cada caso concreto de una manera objetiva que nos permita obtener la conclusión más adecuada.

Por ejemplo, de darse el caso en que el Xpress-IVE resuelva 8 de 10 problemas y CPLEX 2 de 10 pero con unos resultados mejores que el primero, se seleccionará Xpress-IVE como solver más adecuado, ya que es el que en líneas generales presenta una mejor eficiencia.

- Un criterio secundario:
 - El coste de adquirir una licencia para emplear estos solvers, que en base a (GAMS Development Corporation, 2016) se estima como:
 - Coste de una licencia de Xpress-IVE: 9600 \$.
 - Coste de una licencia de CPLEX: 9600 \$.

En base a este criterio secundario, y debido a que ambos solvers presentan el mismo coste por licencia, si para un tipo de problemas no se pudiera discernir qué solver es el más adecuado, se descarta el criterio económico como método de diferenciación, y se consideraría que ambos solvers son igualmente aptos para resolver problemas de ese tipo.

Por último, indicar que para Xpress-IVE, se ha empleado la versión del programa 7.7 de 64 bits, mientras que para CPLEX se ha empleado la versión 12.5. Ambas versiones son relativamente nuevas (a fecha de 2016 se emplea la versión 12.6 de CPLEX, en tanto que la versión 7.7 de Xpress-IVE data de 2014-2015), por lo que su estudio puede considerarse representativo de la situación actual.

3.3.5. Diferencias entre Xpress-IVE y CPLEX.

:

De concluirse que, para un tipo de problema, uno de estos dos solvers es más completo y ofrece mejores resultados que el otro, ¿podríamos decir que existe una diferencia en los algoritmos de resolución que haga que sea mejor?

Al contrario de lo que puede parecer, la respuesta a esta pregunta es no. Tanto Xpress-IVE como CPLEX utilizan algoritmos de resolución basados en Branch and Cut y el método simplex, empleando para su implementación diversas heurísticas que, si bien no son las mismas en uno y otro solver, son

prácticamente idénticas. Por tanto, la hipótesis algorítmica no es un razonamiento válido.

Entonces, ¿cómo podríamos justificar que uno de los dos solvers sea más adecuado que el otro? La respuesta es en base a los parámetros de fábrica. A continuación, explicamos este razonamiento.

Tanto CPLEX como Xpress-IVE poseen alrededor de unos 200 parámetros que definen cómo deben de resolver cada problema. Estos parámetros engloban aspectos tan diversos como la tolerancia mínima a partir de la cual se considera una solución como óptima, la memoria máxima asignada antes de saturar el programa para almacenar los planos de corte realizados, el tiempo de ejecución máximo para encontrar una solución óptima, máximo número de nodos evaluados, máximo número de soluciones factibles encontradas, entre otros muchos otros.

Si bien es cierto que ambos solvers han sido ejecutados sin modificar los valores de los parámetros que vienen por defecto (salvo el tiempo máximo de ejecución, que se ha fijado en 200 segundos como medida estándar), no se han evaluado todos los parámetros de cada solver para ver si coinciden, por lo que concluimos que las diferencias que pudieran existir entre un solver y otro se deberán a los valores por defecto de dichos parámetros.

Programación matemática / General	
Inicio avanzado	Inicio avanzado estándar
Informe de tiempo de computación	Hora de reloj de pared
Algoritmo para problemas continuos	Automático
Algoritmo para optimización cuadrática continua	Automático
Recuento global de subprocesos predeterminado	0
Límite de tiempo global	200.0
Límite de tiempo determinista	1.0E75
Directorio para archivos de trabajo	. <input type="button" value="Examinar..."/>
Memoria disponible para almacenamiento de trabajo	128.0
Parámetro de modo paralelo	Automático
Solución destino	Automático

Figura 3.5: ejemplo 1 con alguno de los parámetros de CPLEX.

Programación matemática / Programación de enteros mixtos / Límites	
Límite de suma de restricciones para generación de cortes	3
Subprocesos raíz auxiliares	0
Número de pases de planos de corte	0
Factor multiplicador de fila para los cortes	4.0
Límite de candidatos para generar cortes de Gomory fraccionales	200
Límite de pases para generar cortes de Gomory fraccionales	0
Límite de nodos MIP	9223372036800000000
Tiempo dedicado a pulir una solución	0.0
Tiempo dedicado a sondear	1.0E75
Frecuencia para intentar reparar un inicio MIP infactible	0
Límite de soluciones enteras MIP	9223372036800000000
Límite de lista de candidatos MIP	10
Límite de iteraciones de simplex MIP	0

Figura 3.6: ejemplo 2 con alguno de los parámetros de CPLEX.

CAPÍTULO 4

EL PROBLEMA DEL VIAJANTE DE COMERCIO (TSP).

4.1. Descripción del problema.

Previamente, en el Capítulo 2 se ha realizado una introducción a los problemas de ruteo de vehículos, identificando como origen de los mismos el caso más sencillo, el problema del viajante de comercio o TSP (Traveling Salesman Problem).

Recordando lo anteriormente mencionado, el TSP se plantea como un problema en el que nosotros tenemos un conjunto de clientes, ciudades (denominados nodos comúnmente) separados por una cierta distancia. Nuestro objetivo consistirá en determinar la ruta que deberá trazar un vehículo³ y que, partiendo de un nodo inicial y pasando por todos los nodos, vuelva al nodo inicial y recorra una distancia mínima. (E.Miller, W.Tucker, & A.Zemlin) (Sáez Aguado, "El problema TSP", 2016).

La Figura 4.1 muestra un caso de este problema. Se trata del ejemplo burma14, de la librería (Librería TSP, s.f.), de 14 nodos. El punto de origen para el trazado de la ruta es el nodo 1.

³ En el problema TSP, la ruta será trazada por un vehículo. Sin embargo, en problemas derivados de este, podemos encontrarnos con el caso de trazar rutas empleando varios vehículos (ver capítulos posteriores).

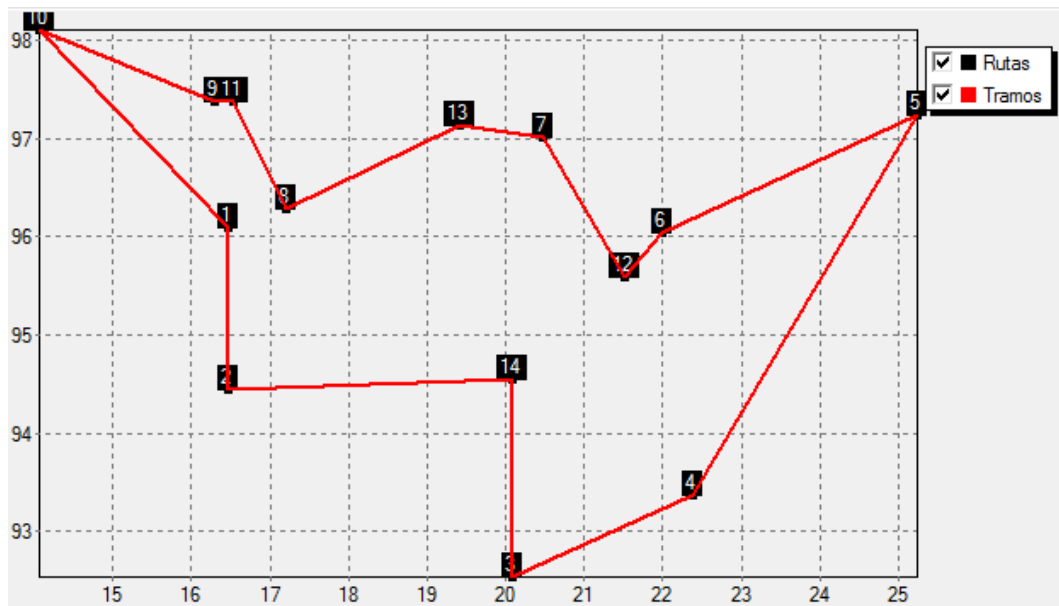


Figura 4.1: ejemplo gráfico del problema TSP.

Dado que el TSP fue también la motivación base para el desarrollo de la investigación operativa (Capítulo 3), pilar fundamental del presente TFG, consideramos de vital importancia su total comprensión, de manera que sirva como precedente para los problemas posteriores, variantes del mismo y más complejas.

El capítulo se estructura como sigue: primero, aclararemos los conceptos de la teoría de grafos, muy empleada como nomenclatura en los capítulos siguientes; segundo, comentaremos brevemente la estructura común de los ficheros de datos y definiremos el concepto de distancia euclídea; tercero, hablaremos acerca de las formulaciones empleadas, los subtours y su eliminación por medio de restricciones específicas; por último, para los modelos planteados, expondremos brevemente los resultados obtenidos, intentando discernir cuál de todos es el que mejores resultados ofrece.

Cabe denotar que no se va a realizar una comparación tan exhaustiva como en los capítulos posteriores, ya que sólo vamos a formular el problema TSP para el solver Xpress-IVE. Esto es debido a que se trata del problema más básico y sólo queremos que este capítulo sienta las bases de lo que viene a continuación. Sin embargo, no queremos desperdiciar la oportunidad de realizar la comparación entre los principales modelos propuestos, por lo que esta se realizará esta de la forma más breve y concisa posible.

4.2. Teoría de grafos.

Introducida Leonhard Euler (Euler, 1741) en el problema de los puentes de Königsberg (ver Capítulo 2), la teoría de grafos pronto se convirtió en una forma intuitiva y visual de plantear problemas de ruteo de vehículos. Tal fue su repercusión, que la nomenclatura que plantea sigue siendo empleada frecuentemente en la actualidad (si recordamos, alguna vez hemos empleado previamente los conceptos de nodo o arco).

Según esta teoría, todo problema de rutas de vehículos puede ser planteado con un grafo $G=(N,A)$ consistente en un conjunto de nodos (N) y de arcos que unan dichos nodos (A). Los nodos representarían los clientes o ciudades que debemos visitar. Los arcos, por su parte, representarían el hecho de viajar de un nodo a otro.

Cada arco tendrá asociado una distancia⁴ entre los nodos que une, de manera que si ese arco es utilizado, recorreremos esa distancia. Esto, junto con que nuestra función objetivo busca minimizar distancia total, provocará que generemos la ruta óptima.

La Figura 4.2 muestra estos conceptos. En ella, podemos observar un conjunto de cuatro nodos (3, 10, 4 y 7) y cuatro arcos (uno que une los nodos 7 y 3; otro para los nodos 3 y 10 y así sucesivamente).

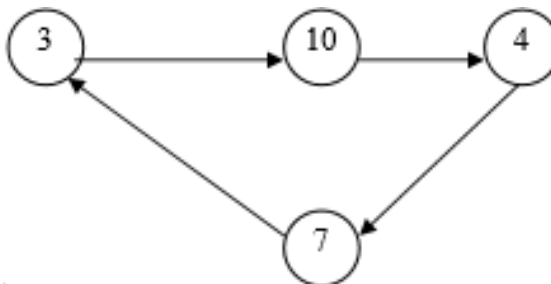


Figura 4.2: ejemplo gráfico de la teoría de grafos propuesta por Euler.

4.2.1. Tipos de grafos.

⁴ A veces puede representarse también como un coste monetario por emplear ese arco, siendo el planteamiento equivalente sólo que con una función objetivo que tenderá a minimizar el coste total.

Dependiendo de la naturaleza del grafo, podemos distinguir (Sáez Aguado, "Métodos matemáticos en ingeniería de organización I", 2014):

- Grafos no dirigidos: equivalente a la definición del SCVRP (ver apartado 2.3), en la cual el coste/distancia en que incurrimos al ir del nodo A al nodo B es igual que el coste/distancia de ir del nodo B al nodo A. Por tanto, el flujo a través de cada arco se permite en ambas direcciones. La Figura 4.3 ilustra este caso (Laporte, "A Short History Of Travelling Salesman Problem", 2006):

Este será el tipo de grafo al que nos referiremos durante todo el proyecto.



Figura 4.3: grafo no dirigido.

- Grafo dirigido: su definición es equivalente a la del ACVRP (ver apartado 2.3), que difiere de la anterior en que ahora el coste/distancia incurrido al ir del nodo A al nodo B no es el mismo que el de ir del nodo B al nodo A, por lo que el flujo a través de los arcos queda limitado a una sola dirección. La Figura 4.4 ilustra este caso (Laporte, "A Short History Of Travelling Salesman Problem", 2006):

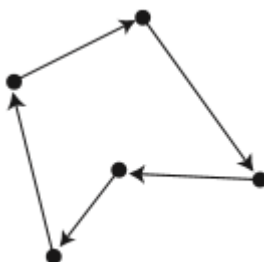


Figura 4.4: grafo dirigido.

4.3. La distancia euclídea.

En el presente TFG se ha optado por minimizar la distancia total recorrida por el/los vehículos en la ruta. Por ello, hay que establecer un procedimiento para determinar la distancia entre cada par de nodos existentes. Esto puede conseguirse de muy diversas formas:

- Poseer ficheros de datos en los que ya venga definida una matriz con la distancia entre cada par de nodos.
- Emplear herramientas como Google Maps para determinar la distancia entre dos puntos, en el caso en que nuestro problema sea un caso real.
- Calcular la distancia durante la programación.

Este último es nuestro caso, ya que los ficheros de datos de que disponemos no cumplen el primer requisito, y calcular las distancias empleando Google Maps para 7 nodos podría estar bien, pero cuando el número de nodos aumenta considerablemente esta tarea se vuelve ardua y tediosa.⁵

Para calcular la distancia durante la programación, partimos de que en nuestros ficheros de datos aparecen las coordenadas de cada nodo, que se corresponderán con las posiciones de dicho nodo en un espacio euclídeo. En base a esto, empleando el Teorema de Pitágoras para cada par de puntos, podemos calcular la distancia entre ellos (la línea recta que los une).

A modo de ejemplo, si las coordenadas del nodo 1 fueran (x_1, y_1) y las del nodo 2 fueran (x_2, y_2) , la distancia entre los nodos 1 y 2 sería:

$$d_{12} = \sqrt{((x_1 - x_2)^2 + (y_1 - y_2)^2)}$$

Este procedimiento será empleado en todos los modelos desarrollados, implementando el cálculo en la programación de cada uno.

4.4. Formulación.

En su forma más básica, un problema TSP podría definirse como sigue (Sáez Aguado, "El problema TSP", 2016):

Variables:

- x_{ij} : variable binaria para indicar si está activo el arco entre el nodo i y el nodo j .

⁵ Aunque existen aplicaciones de Google Maps que permiten realizar este procedimiento de una forma más cómoda y rápida. Aun así, no se ha optado por esta solución ya que la relación tiempo empleado/beneficio no nos pareció aceptable.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .

$$\text{Min} \quad \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1..n \quad (1)$$

(1) De cada nodo debe salir un y sólo un arco.

$$\sum_{j=1}^n x_{ji} = 1 \quad \forall i = 1..n \quad (2)$$

(2) A cada nodo debe llegar un y sólo un arco.

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1..n$$

Sin embargo, se trata de una formulación no válida, ya que en su solución se generan subtours, que definimos a continuación y que provocan que los resultados obtenidos sean erróneos.

4.4.1. Subtours.

Un subtour se define como una unión entre distintos nodos que, pese a no violar ninguna restricción del modelo, genera una solución no válida. Suelen ser debidos generalmente a errores de programación, a la inexistencia de restricciones para eliminarlos y sobre todo a que los solvers empleados son inteligentes, de manera que si no limitas la existencia de subtours y trazándolos se consigue un coste/distancia menor, será seleccionada una ruta con ellos.

Para el problema TSP anteriormente expuesto, un ejemplo con subtours podría ser:

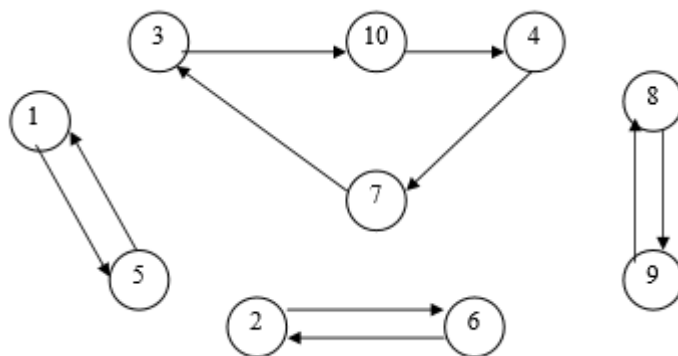


Figura 4.5: ejemplo de solución con subtours para el problema TSP.

Como vemos, no violamos ninguna restricción del modelo, ya que de cada nodo parte y a cada nodo llega un solo arco. Sin embargo, vemos cómo hemos generado 4 rutas distintas, cuando lo que estamos buscando es una única ruta que pase por todos los nodos propuestos.

4.4.2. Eliminación de subtours.

Para evitar los problemas que la generación de subtours ocasiona en las soluciones obtenidas, se definen restricciones que tienen como función exclusiva la eliminación de los mismos, y que imposibilitan la existencia de estas rutas secundarias.

Conforme vamos formulando problemas más complejos, estas restricciones varían, existiendo restricciones de eliminación de subtours específicas como pueden ser las basadas en ventanas de tiempo (Capítulo 6). Excluyendo este caso, casi cualquiera de estas restricciones son variaciones de las propuestas para el problema TSP, que exponemos a continuación (Sáez Aguado, "El problema TSP", 2016):

a) Restricciones basadas en conjuntos.

Se trata de que, al menos, haya un arco saliente de cualquier subtour S:

$$\sum_{\substack{i \in S \\ j \notin S}} x_{ij} \geq 1, \quad \forall S \subset N/S \neq \emptyset, S \neq N$$

El problema es que se generan demasiadas restricciones. Por ejemplo, si $|N|=n$, se necesitarían $2^n - 2$ restricciones del tipo anterior. Por este motivo, y dado que su implementación resulta más compleja que en los casos siguientes, no la hemos empleado.

- b) Restricciones de Tucker – Miller – Zemlin (E.Miller, W.Tucker, & A.Zemlin).

En este caso, seleccionamos un nodo inicial para el recorrido, como el nodo 1 por ejemplo, y añadimos $n-1$ variables auxiliares continuas y sin restricción de signo u_i , $i=2, \dots, n$, junto a las $(n-1)^2$ restricciones:

$$u_i - u_j + n \cdot x_{ij} \leq n - 1^6$$

Se puede demostrar que este conjunto de restricciones impide que se formen subtours (Sáez Aguado, "El problema TSP", 2016). Por reducción al absurdo, si se formara un subtour, éste no contendría al nodo 1, por ejemplo $S=\{u_1, u_2, \dots, u_k\}$ con k nodos. Escribiendo las restricciones anteriores, quedaría: $u_i - u_{i+1} \leq n-1$, $i=1, \dots, k$. Y ahora, sumando estas k desigualdades, se llega a $kn \leq k(n-1)$, lo cual es imposible. Por lo tanto, cualquier subtour debe contener al nodo 1, lo que es equivalente a decir que sólo puede existir un subtour.

La formulación anteriormente planteada para el problema TSP añadiendo estas restricciones de eliminación de subtours quedaría como sigue:

Variables:

- x_{ij} : variable binaria para indicar si existe el arco ij .
- u_i : variable auxiliar de Tucker – Miller – Zemlin (TMZ) para eliminación de subtours. Representan la posición del nodo i en la ruta.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}$$

⁶ Existen algunas variantes para este tipo de restricciones, pero todas son muy similares.

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1..n \quad (1)$$

(1) De cada nodo debe salir un y sólo un arco.

$$\sum_{j=1}^n x_{ji} = 1 \quad \forall i = 1..n \quad (2)$$

(2) A cada nodo debe llegar un y sólo un arco.

$$u_i - u_j + n \cdot x_{ij} \leq n - 1 \quad \forall i \neq \text{origen}; i \neq j; j \leq n \quad (3)$$

(3) Restricciones de Tucker-Miller-Zemlin para la eliminación de subtours.

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1..n$$

$$u_i \geq 0 \quad \forall i = 1..n$$

c) Problema auxiliar de redes.

En este caso, consideramos unas variables de flujo, $y_{ij} \geq 0$ y planteamos un problema de flujo en redes, con la siguiente formulación (Sáez Aguado, "El problema TSP", 2016):

Variables:

- x_{ij} : variable binaria para indicar si existe el arco ij .
- y_{ij} : variable entera positiva que indica la cantidad transportada del nodo i al nodo j .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- b_i : demanda u oferta de cada nodo. La oferta (cantidad positiva) del nodo origen elegido será $n-1$ y la demanda (cantidad negativa) del resto de nodos será -1 . De esta forma aseguramos que los nodos estén conectados entre sí, porque las unidades de flujo tienen que llegar a cada nodo concreto.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1..n \quad (1)$$

(1) De cada nodo debe salir un y sólo un arco.

$$\sum_{j=1}^n x_{ji} = 1 \quad \forall i = 1..n \quad (2)$$

(2) A cada nodo debe llegar un y sólo un arco.

$$\sum_{j=1}^n y_{ij} - \sum_{j=1}^n y_{ji} = b_i \quad \forall i = 1..n \quad (3)$$

(3) Ecuaciones de Kirchhoff para la eliminación de subtours: el flujo que sale de un nodo menos el flujo que entra en un nodo debe de ser igual a la demanda/oferta de dicho nodo.

$$y_{ij} \leq (n - 1)x_{ij} \quad \forall i, j = 1..n \quad (4)$$

(4) Si no existe arco ij no puede existir flujo entre el nodo i y el nodo j . De existir el arco, el flujo no puede ser mayor que la cantidad ofertada por el nodo inicial. Esta restricción se añade para garantizar la relación entre las variables binarias y las variables de flujo.

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1..n$$

$$y_{ij} \geq 0 \quad \forall i, j = 1..n$$

4.5. Datos empleados.

Para la implementación de los modelos con restricciones de TMZ y flujo en redes para el problema TSP, hemos empleado la librería de datos (Librería TSP, s.f.). Dentro de esta librería, se han seleccionado los ficheros de datos siguientes: pr8, burma14, dantzig42, att48, eil51, st70, eil76, kroA100, a280 y ali535.

4.6. Resultados obtenidos.

Ejecutando los modelos de TMZ y de flujo en redes para con el solver Xpress-IVE y fijando un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la ejecución, quedándose con la mejor solución factible calculada hasta el momento, obtenemos lo siguiente:

Datos empleados		Modelo basado en redes		Modelo basado en TMZ	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
pr8	8	1251,12	0,035	1251,12	0,06
burma14	14	30,8785	0,066	30,8785	0,169
dantzig42	42	679,202	1,81	682,689	199,69
att48	48	33523,7	26,683	37544,2	199,868
eil51	51	428,872	6,418	428,872	22,015
st70	70	677,11	199,851	832,32	200,467
eil76	76	544,369	199,898	582,919	199,855
kroA100	100	24941,5	200,293	25205,3	199,577
a280	280	#N/A	200	#N/A	200
ali535	535	#N/A	200	#N/A	200

Figura 4.6: resultados obtenidos para el problema TSP.

4.6.1. Conclusiones obtenidas. Problema TSP.

Como podemos observar, tanto el modelo de TMZ (a), como el basado en redes (b) resuelven 8 de los 10 problemas planteados.

Para estos 8 problemas, el modelo (b) obtiene mejores distancias en 5 de ellos, obteniéndose el mismo resultado en los 3 restantes. Fijándonos en los tiempos de ejecución, vemos que el modelo (b) consigue unos mejores resultados en 5 casos, por otros 5 en los que se llega al máximo tiempo de ejecución marcado.

En base a esto, queda claro que el modelo basado en redes se antoja más adecuado para resolver problemas TSP, ya que sus resultados son mejores según los criterios de distancias y tiempos de ejecución obtenidos.

CAPÍTULO 5

PROBLEMAS DE RUTAS CON CAPACIDADES (CVRP).

5.1. Descripción del problema.

Como ya se explicó en el apartado 2.3, el CVRP (Capacitated Vehicle Routing Problem) es un problema de rutas de vehículos en el cual debemos optimizar las rutas recorridas por un conjunto de vehículos dado, los cuales visitarán a una serie de clientes con unas demandas asociadas (cantidades a entregar o recoger, en función del criterio de signos empleado⁷), de manera que se deben de respetar las capacidades de cada uno de estos vehículos.

La figura 5.1 muestra gráficamente un caso de este problema, empleando la opción de trazado de gráficos de Xpress-IVE. En ella podemos observar la solución al problema E016-03m dada por el modelo basado en Tucker – Miller – Zemlin de 2 índices (ver apartado 5.2 (b)). En negro aparece el índice de cada cliente, en verde la demanda de cada nodo y las rutas trazadas para los tres vehículos considerados aparecen en rojo.

Obsérvese que para el cliente inicial o nodo origen, la oferta (cantidad a entregar) se corresponde con la suma de las demandas (cantidad requerida) del resto de nodos. Se encuentra con signo negativo ya que, debido a la estructura de los ficheros de datos manejados, las demandas vienen como cantidades positivas por lo que se ha representado de esta forma para que tengan signos opuestos en todo momento.

⁷ Generalmente las ofertas (cantidades a entregar) tienen signo positivo, en tanto que las demandas (cantidades requeridas) tienen signo negativo. Sin embargo, puede darse el caso opuesto, siendo un planteamiento equivalente.

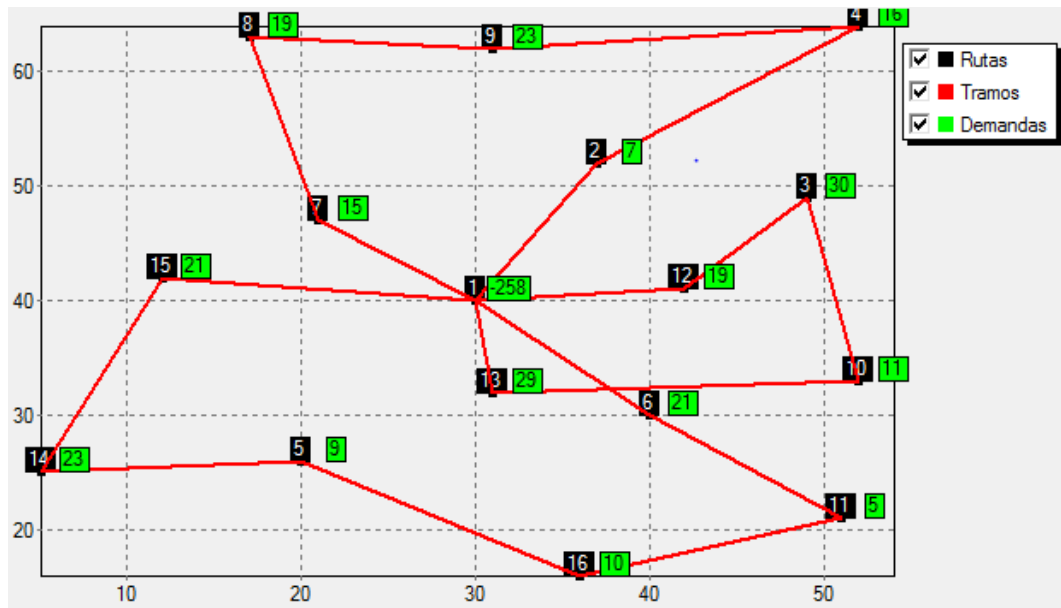


Figura 5.1: ejemplo gráfico del problema CVRP.

El primer objetivo que hemos perseguido será averiguar cuál de los modelos planteados ofrece mejores resultados desde el punto de vista de los criterios planteados en el apartado 2.4. Además, y según los resultados que se pueden ver en el Anexo para el problema CVRP, como segundo objetivo se identifica el ya mencionado de intentar reducir el número de vehículos empleado, siempre y cuando esto suponga obtener una menor distancia total recorrida (minimizando por tanto los costes asociados al transporte).

Para conseguir este segundo objetivo ha sido necesario modificar algunas restricciones respecto a sus formulaciones originales. Dichas restricciones han sido referenciadas a pie de página.

Como tercer y último objetivo, está el ya citado anteriormente de ver qué solver (Xpress-IVE o CPLEX) es el más adecuado para resolver problemas CVRP.

La distancia entre dos nodos o clientes viene dada por la fórmula de distancia euclídea⁸, definida ya en el apartado 4.3. Asimismo, y como se introdujo en el apartado 2.3, consideraremos el caso SCVRP, es decir, que la distancia entre el cliente A y el B es igual a la que existe entre el cliente B y el A.

Por último, y en consonancia con lo visto en el apartado 2.3, se han considerado dos tipos de problemas: homogéneo, en el cual todos los vehículos

⁸ Principalmente por la estructura de los ficheros de datos empleados, en los que los nodos aparecen identificados por sus coordenadas. De igual forma podríamos emplear unas distancias dadas, por ejemplo, por Google Maps o por cualquier otra herramienta.

tienen la misma capacidad y heterogéneo, en el que cada vehículo puede tener una capacidad distinta al resto.

5.2. Modelos planteados. Caso homogéneo.

a) Modelo basado en redes.

Este modelo es una extensión del propuesto en (Sáez Aguado, "Modelo de flujo en redes que no precisa conocer de antemano el número de vehículos", 2016), y se define como sigue:

Variables:

- x_{ij} : variable binaria para indicar si existe el arco ij .
- y_{ij} : variable entera positiva que indica la cantidad transportada del nodo i al nodo j .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- b_i : demanda u oferta de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.

Formulación:

$$\text{Min} \quad \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

(1) De cada nodo distinto del nodo origen debe salir un y sólo un arco.

$$\sum_{j=1}^n x_{ji} = 1 \quad \forall i \neq \text{origen} \quad (2)$$

(2) A cada nodo distinto del nodo origen debe llegar un y sólo un arco.

$$\sum_{j=1}^n x_{origen j} \leq K \quad (3)$$

(3) Del nodo origen deben salir a lo sumo K arcos, 1 por vehículo empleado.⁹

$$\sum_{j=1}^n x_{ij} - \sum_{j=1}^n x_{ji} = 0 \quad \forall i = 1..n \quad (4)$$

(4) El número de arcos que salen de un nodo debe ser igual al número de arcos que llegan a dicho nodo.

$$\sum_{j=1}^n y_{ij} - \sum_{j=1}^n y_{ji} = -b_i \quad \forall i = 1..n \quad (5)$$

(5) Ecuaciones de Kirchhoff para eliminación de subtours (ver apartado 4.4.2 (c)). Representan que el flujo que sale de un nodo menos el flujo que entra en un nodo debe de ser igual a la demanda/oferta de dicho nodo.

El signo negativo de b_j es debido a que, en los datos, las demandas son cantidades positivas, y según el criterio de signos empleado deben de ser negativas.

$$y_{ij} \leq cap \cdot x_{ij} \quad \forall i, j = 1..n \quad (6)$$

(6) No puede existir flujo de i a j si no existe el arco ij . De existir el arco ij , se tiene una cota superior para la cantidad transportada de i a j . Esta restricción sirve para relacionar las variables binarias y las variables de flujo.

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1..n$$

$$y_{ij} \geq 0 \quad \forall i, j = 1..n$$

⁹ Restricción modificada respecto de la formulación original.

b) Modelo de dos índices basado en Tucker – Miller – Zemlin.

Modelo propuesto en (Sáez Aguado, "Modelo de dos índices para el problema de rutas con capacidades", 2016), y que se basa en una extensión de las restricciones de eliminación de subtours propuestas por Tucker – Miller – Zemlin.

Variables:

- x_{ij} : variable binaria para indicar si existe el arco ij .
- u_i : variable auxiliar para eliminación de subtours propuesta por Tucker Miller Zemlin. Son una generalización de las propuestas para el problema TSP (apartado 4.4.2 (b)).

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n x_{ij} d_{ij}$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

(1) De cada nodo distinto del nodo origen debe salir un y sólo un arco.

$$\sum_{j=1}^n x_{ji} = 1 \quad \forall i \neq \text{origen} \quad (2)$$

(2) A cada nodo distinto del nodo destino debe llegar un y sólo un arco.

$$\sum_{j=1}^n x_{origen j} \leq K \quad (3)$$

(3) Del nodo origen deben salir a lo sumo K arcos, 1 por vehículo empleado.¹⁰

$$\sum_{j=1}^n x_{ij} - \sum_{j=1}^n x_{ji} = 0 \quad \forall i = 1..n \quad (4)$$

(4) El número de arcos que salen de un nodo debe ser igual al número de arcos que llegan a dicho nodo.

$$u_i - u_j + (cap - d_j - d_i) \cdot x_{ji} + cap \cdot x_{ij} \leq cap - d_j \\ \forall i \neq origen; i \neq j; j \neq origen; j \neq i; d_i + d_j \leq cap \quad (5)$$

(5) Restricciones de TMZ para eliminación de subtours. Son una generalización de las propuestas para el problema TSP (apartado 4.4.2 (b)).

$$d_i \leq u_i \leq cap \quad \forall i \neq origen \quad (6_1 \text{ y } 6_2)$$

(6_1 y 6_2) Cotas superiores e inferiores para las variables de TMZ.

$$u_i \leq cap + (d_i - cap) * x_{origen i} \quad \forall i \neq origen \quad (7)$$

(7) Otras cotas superiores para las variables de TMZ.

$$x_{ij} \in \{0,1\} \quad \forall i, j = 1..n$$

$$u_i \geq 0 \quad \forall i = 1..n$$

c) Modelo de tres índices basado en Tucker – Miller – Zemlin.

Este modelo ha sido obtenido de (Sáez Aguado, "Problemas de rutas de vehículos", 2016), y está basado en una generalización de las restricciones de

¹⁰ Restricción modificada respecto de la formulación inicial

Tucker – Miller – Zemplin para eliminación de subtours. Queda formulado como sigue:

Variables:

- x_{ijk} : variable binaria para indicar si existe el arco ij , y este se recorre con el vehículo k .
- u_{ik} : variable auxiliar para eliminación de subtours propuesta por Tucker – Miller - Zemplin. Son una generalización de las propuestas para el problema TSP.
- y_{ij} : variable binaria que nos indica si en el nodo i actúa el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_j : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.

Formulación:

$$\text{Min} \quad \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

(1) En cada nodo distinto del nodo origen sólo puede actuar un vehículo.

$$\sum_{k=1}^K y_{\text{origen } k} \leq K \quad (1')$$

(1') En el nodo origen actúan K vehículos.¹¹

¹¹ Restricción modificada respecto de la formulación original.

$$\sum_{j=1}^n x_{ijk} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (2)$$

(2) Si en un nodo no actúa un vehículo, no puede existir un arco recorrido por ese vehículo y que parta de ese nodo.

$$\sum_{j=1}^n x_{jik} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (3)$$

(3) Si en un nodo no actúa un vehículo, no puede existir un arco recorrido por ese vehículo y que llegue a ese nodo.

$$u_{ik} - u_{jk} + cap \cdot x_{ijk} \leq cap - d_j$$

$$\forall i \neq origen; i \neq j; j \neq origen; k = 1..K; d_i + d_j \leq cap \quad (4)$$

(4) Restricciones de TMZ para eliminación de subtours. Se trata de una generalización de las restricciones propuestas para el problema TSP.

$$d_i \leq u_{ik} \leq cap \quad \forall i \neq origen; k = 1..K \quad (5 \text{ y } 6)$$

(5 y 6) Cotas superiores e inferiores para las variables de TMZ.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$y_{ik} \in \{0,1\} \quad \forall i = 1..n; k = 1..K$$

$$u_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

d) Modelo de Christofides, Mingozzi y Toth.

Este modelo aparece formulado en (N.Christofides, A.Mingozzi, & P.Toth, 1979) como sigue:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .

- u_i : variable auxiliar para eliminación de subtours propuesta por Tucker Miller Zemlin. Son una generalización de las propuestas para el problema TSP.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_j : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{i=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen} \quad (1)$$

(1) A todo nodo salvo el nodo origen debe llegar un arco.

$$\sum_{i=1}^n x_{ijk} - \sum_{i=1}^n x_{jik} = 0 \quad \forall j = 1..n; k = 1..K \quad (2)$$

(2) El número de arcos que llegan a un nodo debe de ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^n x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (3)^{12}$$

(3) Para cada vehículo, del nodo origen puede salir un arco.

¹² En este caso, esta restricción no ha sido modificada respecto de la formulación original.

$$\sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq \text{origen}}}^n (x_{ijk} \cdot d_j) \leq \text{cap} \quad \forall k = 1..K; \quad (4)$$

(4) La suma de las demandas atendidas por un vehículo no puede ser mayor que la capacidad del mismo. Se trata de una cota superior para la carga del vehículo.

$$u_i - u_j + n \cdot \sum_{k=1}^K x_{ijk} \leq n - 1 \quad \forall i = 1..n; j \neq \text{origen} \quad (5)$$

(5) Restricciones de eliminación de subtours muy similares a las propuestas por TMZ.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$u_i \geq 0 \quad \forall i = 1..n$$

5.3. Modelos planteados. Caso heterogéneo.

a) Modelo de tres índices basado en Tucker – Miller - Zemlin.

Idéntico al visto para el caso homogéneo, solo que con capacidades distintas para cada uno de los vehículos.

Formulación:

$$\text{Min} \quad \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

$$\sum_{k=1}^K y_{origen\ k} \leq K \quad (1')^{13}$$

$$\sum_{j=1}^n x_{ijk} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (2)$$

$$\sum_{j=1}^n x_{jik} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (3)$$

$$u_{ik} - u_{jk} + cap_k \cdot x_{ijk} \leq cap_k - d_j \\ \forall i \neq origen; i \neq j; j \neq origen; k = 1..K; d_i + d_j \leq cap_k \quad (4)$$

$$d_i \leq u_{ik} \leq cap_k \quad \forall i \neq origen; k = 1..K \quad (5\ y\ 6)$$

$$\sum_{i=1}^n (d_i \cdot y_{ik}) \leq cap_k \quad \forall k = 1..K \quad (7)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$y_{ik} \in \{0,1\} \quad \forall i = 1..n; k = 1..K$$

$$u_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

b) Modelo de Golden – Assad – Levys – Gheysens basado en TMZ.

Modelo propuesto en (Golden, Assad, Levy, & Gheysens, 1984) y que queda formulado como sigue:

¹³ Restricción modificada respecto de la formulación original.

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- u_i : variable auxiliar para eliminación de subtours propuesta por Tucker - Miller - Zemlin. Son una generalización de las propuestas para el problema TSP (apartado 4.4.2 (b)).

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_j : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- M : valor auxiliar empleado como cota superior en las restricciones de TMZ. Suele ser la suma de las capacidades de todos los vehículos.

Formulación:

$$Min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{i=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq origen \quad (1)$$

(1) A cada nodo distinto del nodo origen debe llegar un arco.

$$\sum_{i=1}^n x_{ijk} - \sum_{i=1}^n x_{jik} = 0 \quad \forall j = 1..n; k = 1..K \quad (2)$$

(2) El número de arcos que entra en un nodo debe ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j=1 \\ j \neq origen}}^n x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (3)$$

(3) Del nodo origen, para cada vehículo, puede salir a lo sumo un arco. ¹⁴

$$u_{origen} = 0 \quad (4)$$

$$u_j - u_i \geq (d_j + M) \cdot \sum_{k=1}^K x_{ijk} - M \quad \forall i = 1..n; j \neq origen \quad (5)$$

$$u_j \leq \sum_{i=1}^n \sum_{k=1}^K (x_{ijk} \cdot cap_k) \quad \forall j = 1..n \quad (6)$$

(4), (5), (6) Restricciones de Tucker-Miller-Zemlin para eliminación de subtours.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$u_i \geq 0 \quad \forall i = 1..n$$

c) Modelo de Golden – Assad – Gheysens basado en redes.

Modelo propuesto en (Gheysens, Golden, & Assad, 1984) y que es formulado como sigue:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ij} : variables enteras para las ecuaciones de Kirchhoff, empleadas para eliminación de subtours en problemas de flujo en redes (apartado 4.4.2 (c)).

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- M : valor auxiliar empleado como cota superior en las restricciones de flujo en redes. Suele ser $\max(cap_k) - cap_k$, aunque también puede calcularse como en el modelo anterior, haciendo suma de las capacidades de todos los vehículos.

¹⁴ Restricción modificada respecto de la formulación original.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{i=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen} \quad (1)$$

(1) A cada nodo distinto del nodo origen debe llegar un arco.

$$\sum_{i=1}^n x_{ijk} - \sum_{i=1}^n x_{jik} = 0 \quad \forall j = 1..n; k = 1..K \quad (2)$$

(2) El número de arcos que llegan a un nodo debe de ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^n x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (3)$$

(3) Para cada vehículo, del nodo origen puede salir un arco. ¹⁵

$$\sum_{j=1}^n y_{ij} - \sum_{j=1}^n y_{ji} = -d_i \quad \forall i = 1..n \quad (4)$$

(4) Restricciones de Kirchhoff para eliminación de subtours. Representan que el flujo que entra en un nodo menos el flujo que sale del mismo debe de ser igual a la demanda de dicho nodo.

El lado derecho de la restricción aparece como $-d_i$ debido a la estructura del fichero de datos, que identifica las demandas como positivas. Sin embargo, para las restricciones de flujo, las demandas suelen considerarse cantidades negativas, de ahí el cambio de signo.

¹⁵ Restricción modificada respecto de la formulación original.

$$y_{ij} \leq \sum_{k=1}^K (x_{ijk} \cdot cap_k) \quad \forall i, j = 1..n \quad (5)$$

(5) Cota superior para la cantidad transportada del nodo i al nodo j , suponiendo que exista el arco ij para algún vehículo k . Esta restricción garantiza la relación entre las variables binarias y las variables de flujo.

$$y_{origen j} - cap_k \leq M(1 - x_{origen j k}) \quad \forall j \neq origen; k = 1..K \quad (6)$$

(6) Otra cota superior. Viene formulada en el artículo, pero se trata de una restricción redundante.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$y_{ij} \geq 0 \quad \forall i, j = 1..n$$

d) Modelo de Christofides, Mingozzi y Toth.

Idéntico al visto para el caso homogéneo, solo que con capacidades distintas para cada uno de los vehículos.

Formulación:

$$Min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{i=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq origen \quad (1)$$

$$\sum_{i=1}^n x_{ijk} - \sum_{i=1}^n x_{jik} = 0 \quad \forall j = 1..n; k = 1..K \quad (2)$$

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^n x_{\text{origen } j k} \leq 1 \quad \forall k = 1..K \quad (3)^{16}$$

$$\sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq \text{origen}}}^n (x_{ijk} \cdot d_j) \leq \text{cap}_k \quad \forall k = 1..K; \quad (4)$$

$$u_i - u_j + n \cdot \sum_{k=1}^K x_{ijk} \leq n - 1 \quad \forall i = 1..n; j \neq \text{origen} \quad (5)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..n; k = 1..K$$

$$u_i \geq 0 \quad \forall i = 1..n$$

5.4. Datos empleados.

Para ejecutar las simulaciones, se ha empleado la librería (Librería CVRP homogéneo, s.f.) para obtener los ficheros de datos del caso homogéneo. Para el caso heterogéneo, se han asignado las capacidades de cada uno de los vehículos a mano.

Dentro de la citada librería, se han ejecutado los siguientes ficheros de datos para el caso homogéneo: E010-03m, E016-03m, E021-04m, E026-08m, E030-04s, E036-11h, E041-14h, E045-04f, E072-04f, E076-07s y E101-08e.

Para el caso heterogéneo, se han empleado: E010-03m_heterog, E016-03m_heterog, E021-04m_heterog, E026-08m_heterog, E030-03g_heterog, E036-11h_heterog, E045-04f_heterog, E051-05e_heterog, E072-04f_heterog, E076-07u_heterog, E101-08e_heterog.

¹⁶ En este caso, esta restricción venía ya formulada así en el artículo original, por lo que no ha sido modificada.

5.5. Resultados obtenidos empleando Xpress-IVE.

Se ha establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización, quedándose con la mejor solución factible obtenida hasta ese momento.

5.5.1. Caso homogéneo.

Los resultados obtenidos han sido:

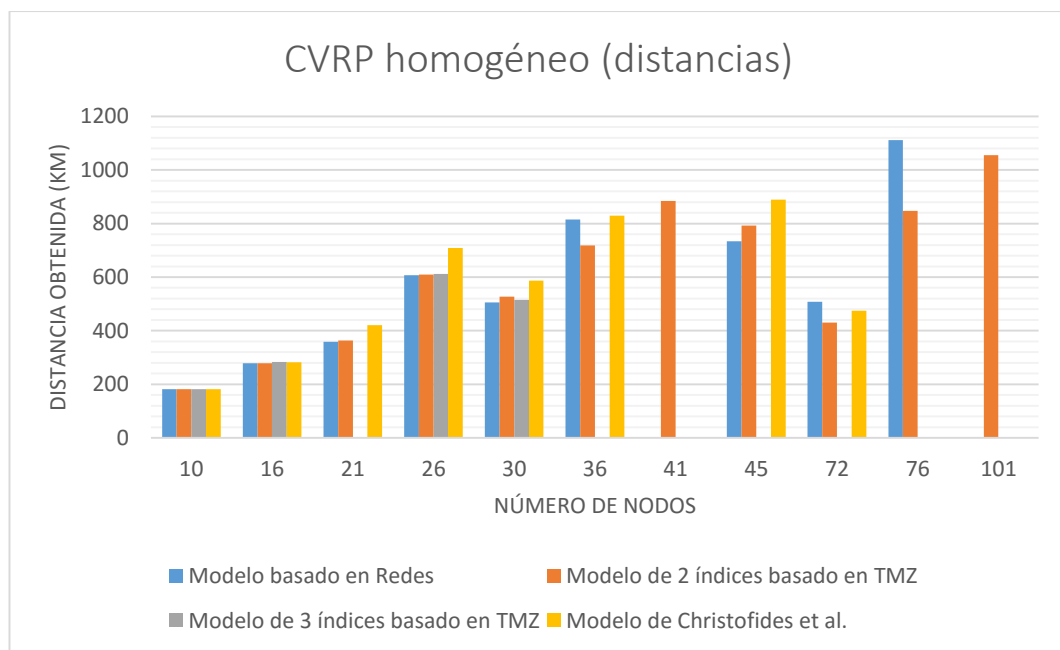


Figura 5.2: Distancias de Xpress- IVE para el problema CVRP homogéneo.

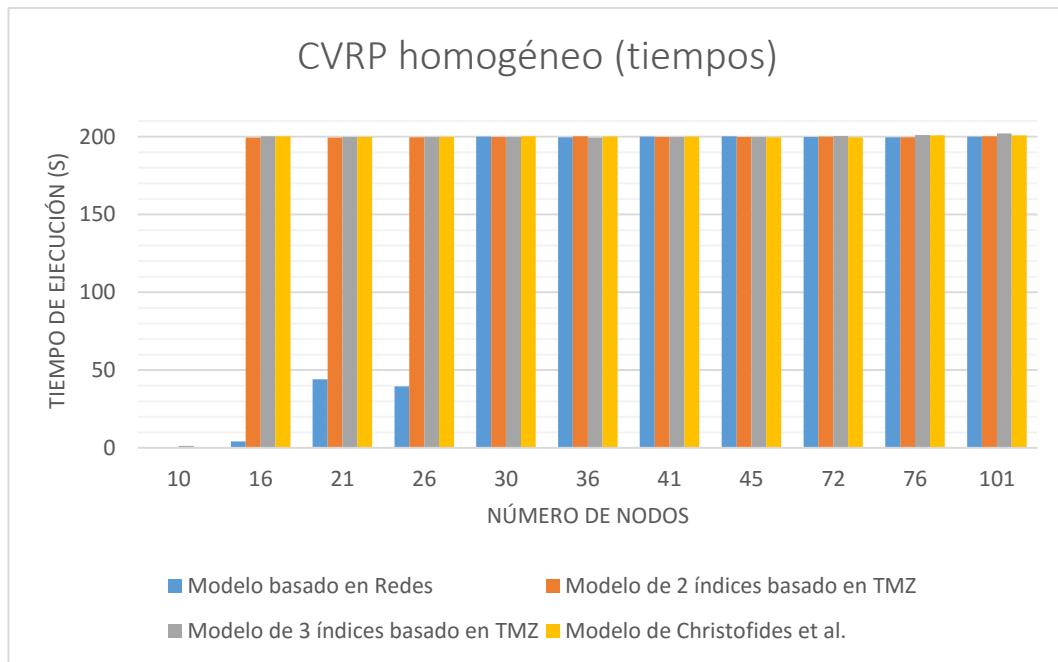


Figura 5.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRP homogéneo.

5.5.2. Resumen de resultados obtenidos. Caso homogéneo.

Como podemos observar, el modelo de 3 índices basado en TMZ (c) sólo consigue resolver 4 de los 11 problemas propuestos, obteniendo siempre unos resultados iguales o peores que el modelo basado en redes (a), por lo que queda rápidamente descartado. Por su parte, el modelo de Christofides et al. (d) pese a resolver 8 de los 11 problemas, presenta soluciones peores a las de los modelos (a) y (b) de dos índices basado en TMZ desde el punto de vista de las distancias obtenidas, e iguales desde el de tiempos de ejecución, por lo que también es descartado.

Comparando los modelos (a) y (b) sin embargo la situación es más compleja: el modelo (b) resuelve los 11 problemas, pero el modelo (a) presenta unos tiempos de ejecución menores para los 4 primeros ficheros de datos, obteniendo soluciones para 9 de los 11 problemas propuestos.

Para los 9 problemas que resuelven tanto (a) como (b), el modelo (a) obtiene un mejor resultado en la distancia obtenida en 4 de ellos. Por su parte, el modelo (b) supera al (a) en 3, quedando igualados en los 2 restantes. Desde el punto de vista de los tiempos de ejecución, el modelo (a) obtiene mejores resultados en 5 de estos 9 problemas, obteniéndose resultados similares en el resto

Por tanto, los modelos (a) y (b) serán los que se implementarán en CPLEX, para ver si este otro solver ofrece una visión más clara acerca de cuál de los dos es más efectivo.

En la Tabla 5.2 se puede observar el resumen de resultados obtenidos para el CVRP homogéneo empleando Xpress-IVE según los criterios de comparación establecidos. Los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor solución, se cuenta igualmente ya que han llegado a la mejor solución)¹⁷. El criterio número de problemas resueltos es el número de los problemas planteados para los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Basado en redes	6 de 11	4 de 11	9 de 11	CPLEX
Basado en TMZ de 2 ind.	7 de 11	0 de 11	11 de 11	CPLEX
Basado en TMZ de 3 ind.	1 de 11	0 de 11	4 de 11	Descartado
Christofides et al.	1 de 11	0 de 11	8 de 11	Descartado

Tabla 5.1: Resumen de resultados para el problema CVRP homogéneo empleando Xpress-IVE.

¹⁷ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Datos empleados		Modelo basado en redes		Modelo de 2 índices basado en TMZ		Modelo de 3 índices basado en TMZ		Modelo de Christofides et al.	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
E010-03m	10	181,588	0,119	181,588	0,2	181,588	1,316	181,588	0,35
E016-03m	16	278,726	4,116	278,73	199,41	282,729	200,157	282,334	200,23
E021-04m	21	358,402	44,087	363,45	199,46	#N/A	199,862	420,161	199,76
E026-08m	26	607,651	39,654	609,76	199,57	611,782	199,719	709,352	199,85
E030-04s	30	505,011	200,03	527,54	199,72	514,779	199,818	587,41	200,26
E036-11h	36	815,543	199,55	718,48	200,2	#N/A	199,447	829,111	200,01
E041-14h	41	#N/A	200	884,75	199,87	#N/A	199,844	#N/A	200,1
E045-04f	45	733,478	200,17	791,86	199,76	#N/A	199,766	889,151	199,61
E072-04f	72	508,202	199,859	430,01	199,94	#N/A	200,37	474,852	199,66
E076-07s	76	1112,12	199,652	847,32	199,65	#N/A	201,001	#N/A	200,74
E101-08e	101	#N/A	200	1054,91	200,3	#N/A	202,118	#N/A	200,81

Tabla 5.2: Resultados de Xpress-IVE para el problema CVRP homogéneo.

5.5.3. Caso heterogéneo.

Los resultados obtenidos para el caso heterogéneo han sido:

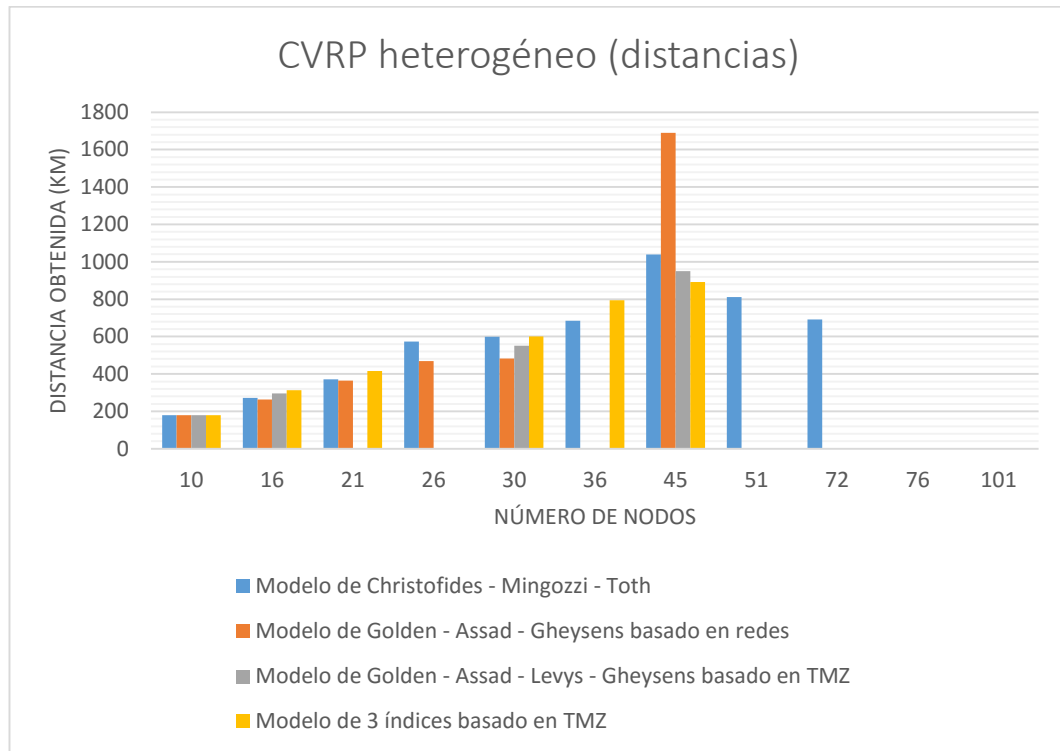


Figura 5.4: Distancias de Xpress-IVE. Problema CVRP heterogéneo.

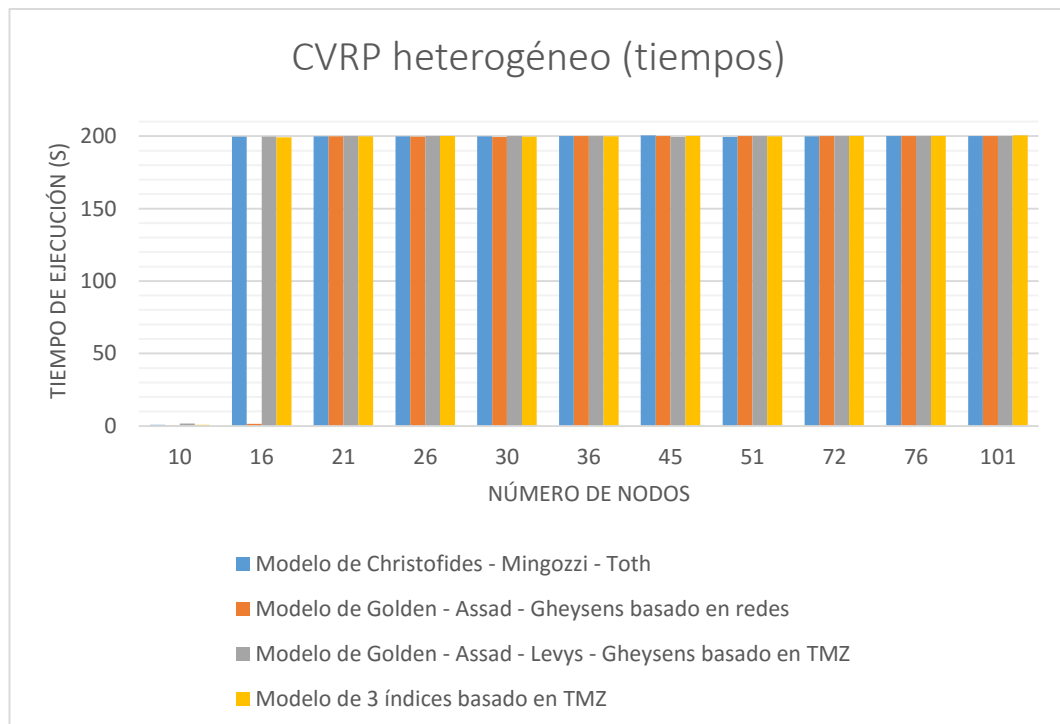


Figura 5.5: Tiempos de Xpress-IVE. Problema CVRP heterogéneo.

Datos empleados		Modelo de Christofides - Mingozzi - Toth		Modelo de Golden - Assad - Gheysens basado en redes		Modelo de Golden - Assad - Levys - Gheysens basado en TMZ		Modelo de tres índices basado en TMZ	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
E010-03m_heterog	10	178,6	0,837	178,6	0,461	178,6	1,715	178,6	0,69
E016-03m_heterog	16	272,338	199,693	263,754	1,468	295,24	199,57	312,656	199,232
E021-04m_heterog	21	370,716	199,884	365,084	199,919	#N/A	200	416,227	199,733
E026-08m_heterog	26	572,979	199,9	469,515	199,514	#N/A	200	#N/A	200
E030-03g_heterog	30	599,217	199,876	481,986	199,331	551,359	200	599,863	199,603
E036-11h_heterog	36	685,166	200,02	#N/A	200	#N/A	200	794,915	199,787
E045-04f_heterog	45	1039,61	200,547	1689,61	200,08	950,764	199,304	891,176	199,988
E051-05e_heterog	51	811,705	199,386	#N/A	200	#N/A	200	#N/A	199,892
E072-04f_heterog	72	692,132	199,825	#N/A	200	#N/A	200	#N/A	200
E076-07u_heterog	76	#N/A	200	#N/A	200	#N/A	200	#N/A	200,073
E101-08e_heterog	101	#N/A	200	#N/A	200	#N/A	200	#N/A	200,606

Tabla 5.3: Resultados obtenidos con Xpress-IVE para el problema CVRP heterogéneo.

5.5.4. Resumen de resultados obtenidos. Caso heterogéneo.

En teoría, resolver problemas heterogéneos suele dar peores resultados que resolver su equivalente homogéneo, debido a que son más complejos.

En nuestro caso concreto, podemos observar cómo pese a que el modelo de Golden et al. basado en TMZ (b) tan solo resuelve 4 de los problemas propuestos, sus resultados se enmarcan dentro de los mejores calculados. Esta distorsión a la hora de decidir se traslada también al resto de modelos, resolviendo 6 de los 11 problemas tanto el modelo (a) de tres índices basado en TMZ como el modelo (c) Golden et al. basado en redes, de manera que ninguno de estos tres llega a destacar realmente.

Si bien es cierto que el modelo de Christofides et al. (d) resuelve gran parte de los problemas (9 de 11) con buenos resultados, no queremos aventurarnos a tomar una decisión precipitada que resulte ser errónea.

Por todo ello, se ha decidido implementar los cuatro modelos en CPLEX, para ver si teniendo las comparaciones generadas por ambos se puede llegar a alguna conclusión.

En la Tabla 5.4 se puede observar el resumen de resultados obtenidos para el CVRP heterogéneo empleando Xpress-IVE según los criterios de comparación establecidos. Los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)¹⁸. El criterio número de problemas resueltos es el número de los problemas planteados para los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Christofides et al.	4 de 11	0 de 11	9 de 11	CPLEX
Golden et al. redes	5 de 11	2 de 11	6 de 11	CPLEX
Golden et al. TMZ	1 de 11	0 de 11	4 de 11	CPLEX
Basado en TMZ de 3 ind.	2 de 11	0 de 11	6 de 11	CPLEX

¹⁸ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Tabla 5.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRP heterogéneo.

5.6. Resultados obtenidos empleando CPLEX.

Al igual que ocurría en el caso de Xpress-IVE, y para garantizar la equidad en la ejecución de ambos modelos, se ha establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización, quedándose con la mejor solución obtenida hasta ese momento.

5.6.1. Caso homogéneo.

Los resultados obtenidos han sido:

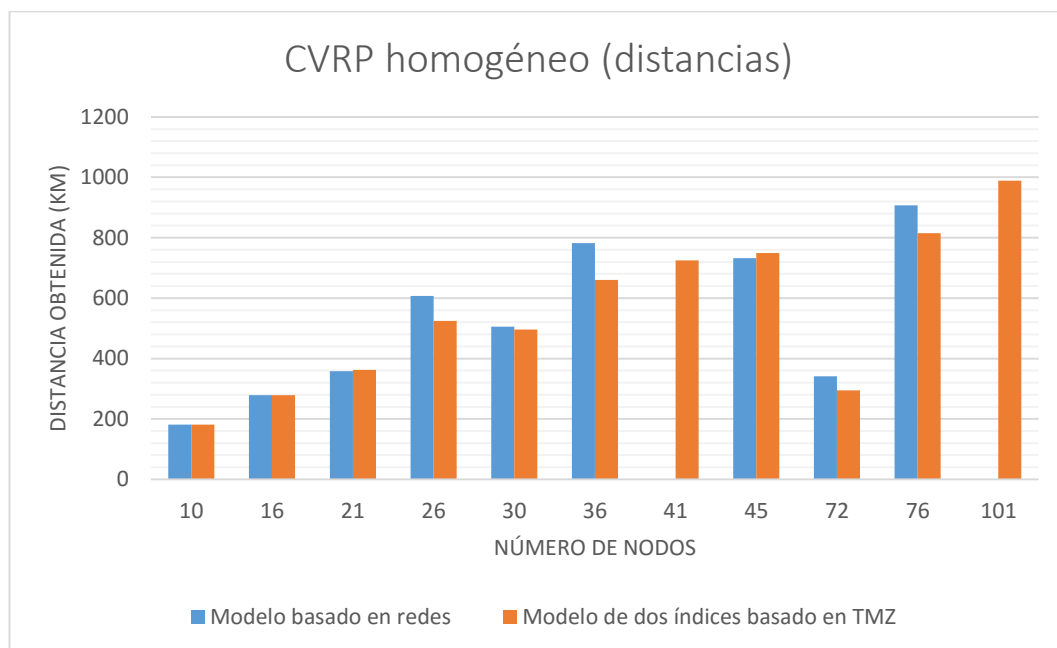


Figura 5.6: Distancias de CPLEX para el problema CVRP homogéneo.

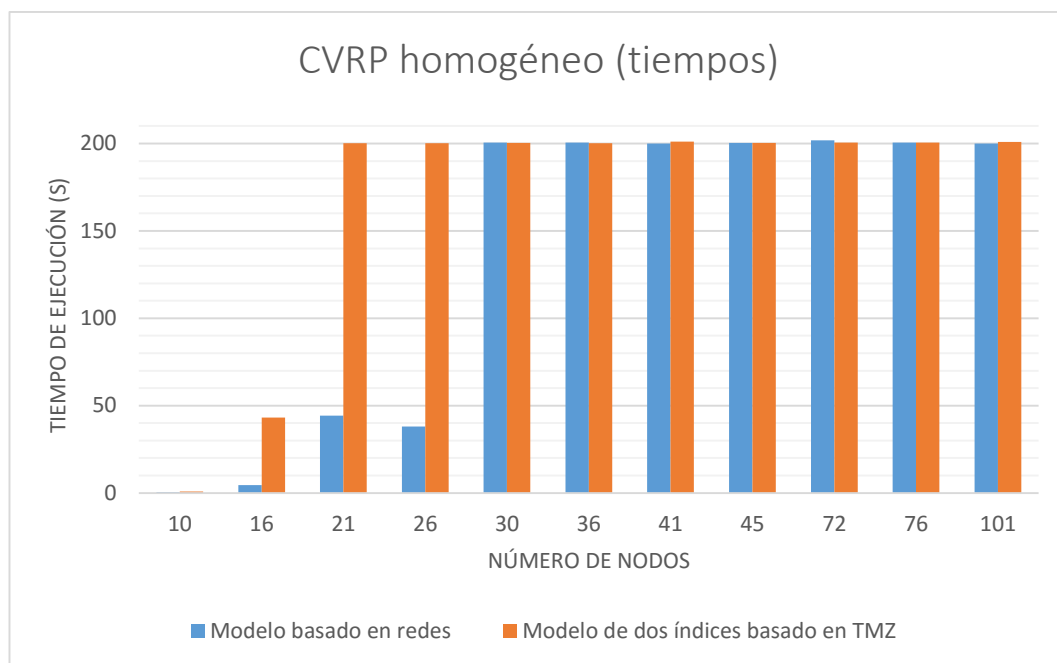


Figura 5.7: Tiempos de CPLEX para el problema CVRP homogéneo.

Datos empleados		Modelo basado en redes		Modelo de dos índices basado en TMZ	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
E010-03m	10	181,5877592	0,561	181,5877592	0,53
E016-03m	16	278,7262997	4,493	278,7262997	36,442
E021-04m	21	358,4022742	44,242	358,4022742	200,117
E026-08m	26	607,6509456	38,127	524,140038	200,133
E030-04s	30	505,0110791	200,507	496,5477681	200,258
E036-11h	36	781,8636886	200,585	662,6750299	200,148
E041-14h	41	#N/A	200	745,1628097	200,196
E045-04f	45	732,5996458	200,414	744,86063	200,866
E072-04f	72	340,9780555	201,849	300,8211824	201,334
E076-07s	76	907,1771905	200,508	817,8105159	200,757
E101-08e	101	#N/A	200	1059,158368	200,991

Tabla 5.5: Resultados obtenidos con CPLEX para el problema CVRP homogéneo.

5.6.2. Resumen de resultados. Caso homogéneo.

En este caso, y como ocurría con los resultados obtenidos con Xpress-IVE, el modelo basado en redes (a) resuelve 9 de los 11 problemas planteados, en tanto que el modelo de dos índices basado en TMZ (b) resuelve los 11.

Además, de los 9 problemas que resuelven ambos modelos, el modelo (b) obtiene mejores resultados desde el punto de vista de las distancias obtenidas en 5 de ellos, con resultados prácticamente iguales para otros 3 casos y 1 resultado en el que el modelo (a) es mejor.

Si bien es cierto que el modelo (a) obtiene unos tiempos mejores en los 4 primeros problemas, esta brevedad no hace que los resultados obtenidos sean mejores (son iguales o incluso peores que en el modelo (b)).

Por tanto, y en base a lo anteriormente citado, se concluye que el modelo de dos índices basado en TMZ es el que mejores resultados obtiene desde el punto de vista de distancias calculadas, pudiendo escogerse el modelo basado en redes para resolver problemas pequeños en aquellos casos en que el tiempo de ejecución sea crítico.

En la Tabla 5.6 se puede observar el resumen de resultados obtenidos para el CVRP homogéneo empleando CPLEX según los criterios de comparación establecidos. Los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)¹⁹. El criterio número de problemas resueltos es el número de los problemas planteados para los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Basado en redes	4 de 11	4 de 11	9 de 11	Para tiempos de ejecución críticos en problemas pequeños
Basado en TMZ de 2 ind.	10 de 11	0 de 11	11 de 11	SELECCIONADO

Tabla 5.6: Resumen de resultados obtenidos con CPLEX para el problema CVRP homogéneo.

¹⁹ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

5.6.3. Caso heterogéneo.

Los resultados obtenidos han sido los siguientes:

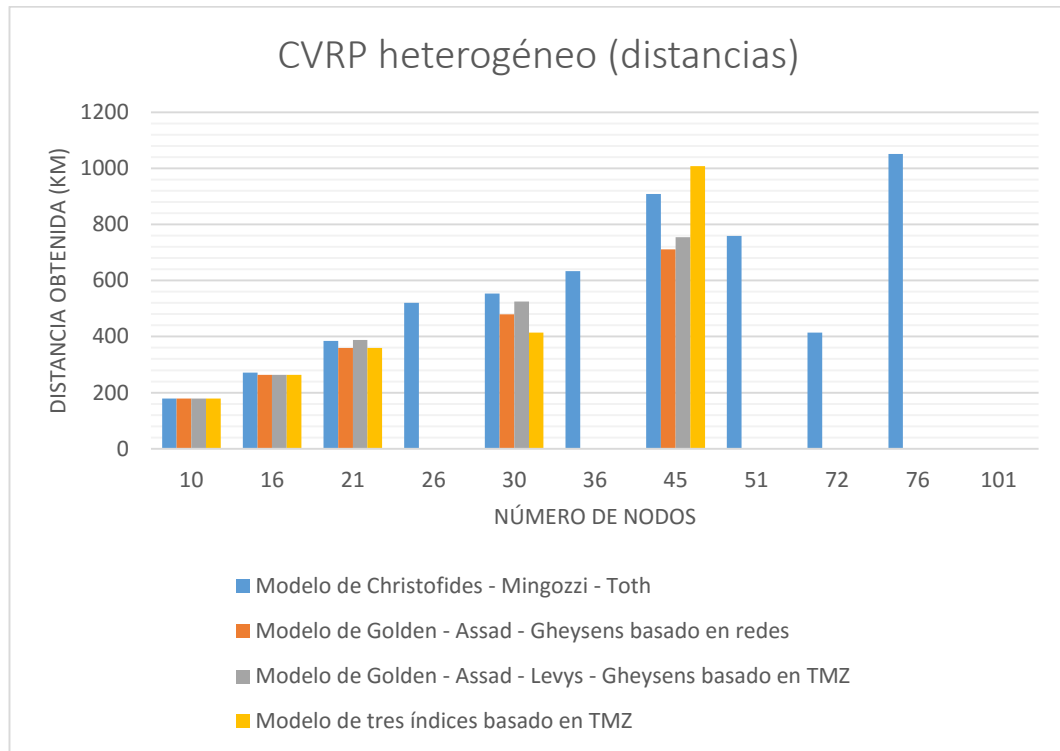


Figura 5.8: Distancias de CPLEX para el CVRP heterogéneo.

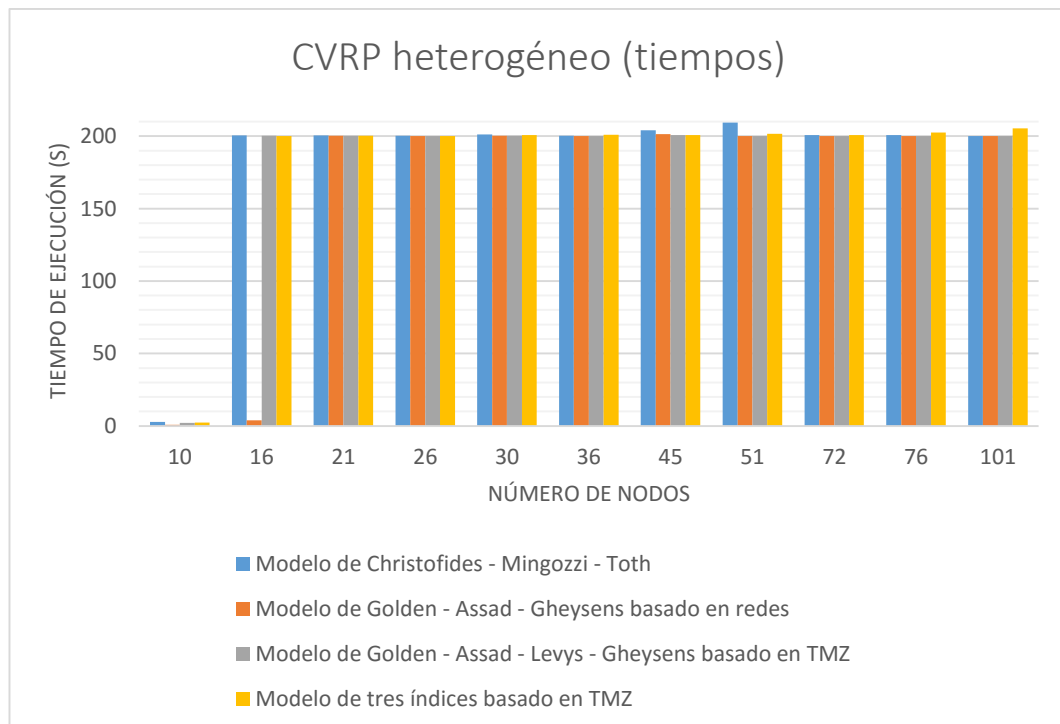


Figura 5.9: Tiempos de CPLEX para el CVRP heterogéneo.

5.6.3. Resumen de resultados obtenidos. Caso heterogéneo.

La idea que teníamos al ver los resultados de Xpress-IVE de que el modelo de Golden et al. basado en TMZ (b) podría ser una solución adecuada si conseguía resolver mayor número de problemas no se ha visto respaldada con esta nueva simulación. Por tanto, se considera un modelo no válido que sólo consigue resolver 5 de los 11 problemas propuestos.

Igualmente ocurre con todos los modelos salvo el de Christofides et al. (d), ya que ninguno de ellos ha conseguido elevar el número de problemas resueltos con respecto a la simulación realizada con Xpress-IVE (incluso se resuelve una menor cantidad: 6 con Xpress-IVE vs 5 con CPLEX).

Por tanto, y dado que ha sido el que más problemas ha resuelto con ambos solvers (9 con Xpress-IVE y 10 con CPLEX), obteniendo no los mejores resultados, pero sí muy aceptables, se selecciona el modelo (d) como el más adecuado para resolver problemas CVRP de flota heterogénea.

En la Tabla 5.7 se puede observar el resumen de resultados obtenidos para el CVRP heterogéneo según los criterios de comparación empleando CPLEX²⁰.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Christofides et al.	6 de 11	0 de 11	10 de 11	SELECCIONADO
Golden et al. redes	4 de 11	2 de 11	5 de 11	Descartado
Golden et al. TMZ	1 de 11	1 de 11	5 de 11	Descartado
Basado en TMZ de 3 ind.	4 de 11	1 de 11	5 de 11	Descartado

Tabla 5.7: Resumen de resultados obtenidos con CPLEX para el problema CVRP heterogéneo.

²⁰ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

5.7. Comparativa Xpress-IVE vs CPLEX para el CVRP.

5.7.1. Caso homogéneo.

Para los dos modelos planteados tanto en Xpress-IVE como en CPLEX (modelo basado en redes (a) y modelo de dos índices basado en TMZ (b)), comparando ambos resultados, obtenemos lo siguiente:

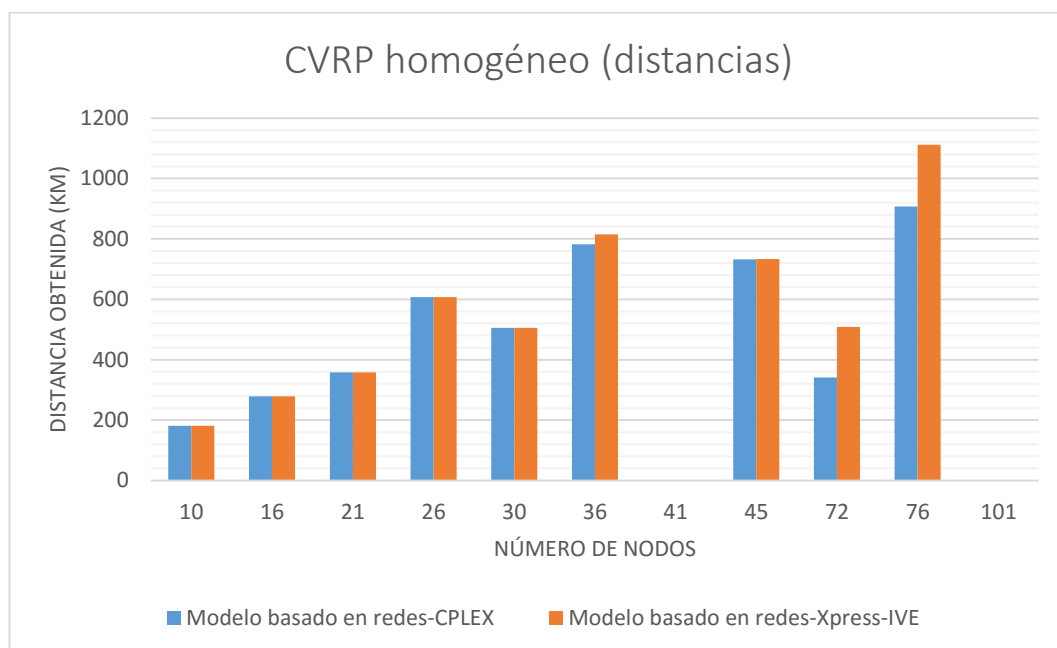


Figura 5.10: Comparativa de distancias modelo basado en redes.

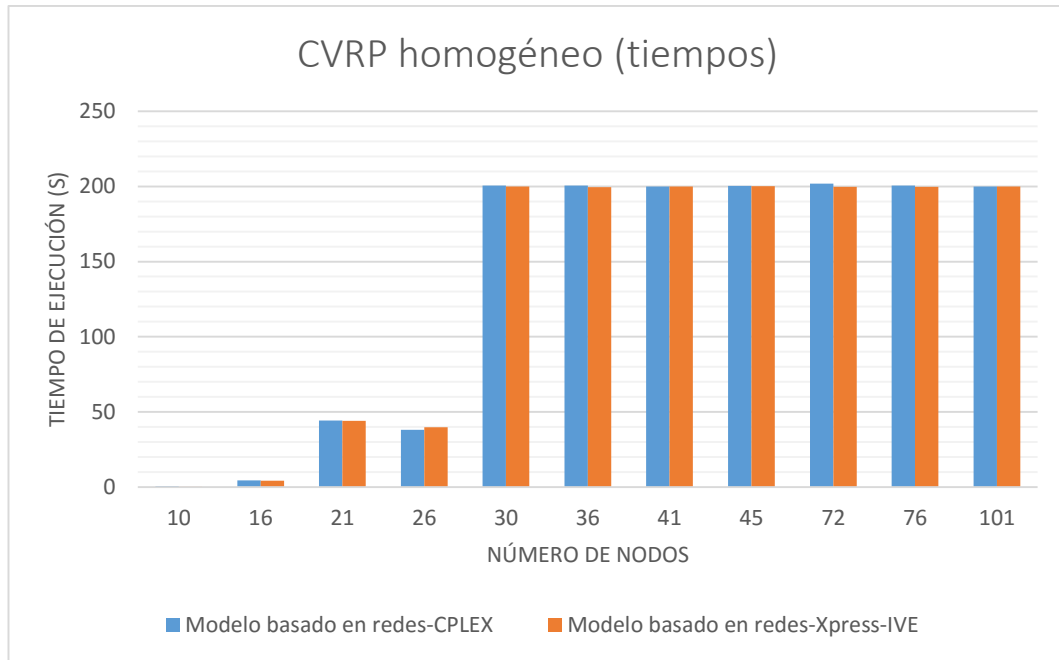


Figura 5.11: Comparativa de tiempos modelo basado en redes.

Datos empleados		Modelo de Christoffides - Mingozzi - Toth		Modelo de Golden - Assad - Gheysens basado en redes		Modelo de Golden - Assad - Levys - Gheysens basado en TMZ		Modelo de tres índices basado en TMZ	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
E010-03m_heterog	10	178,6003025	2,745	178,6003025	0,874	178,6003025	2,153	178,6003025	2,449
E016-03m_heterog	16	271,0320783	200,461	263,7536495	3,963	263,7536495	200,238	263,7536495	200,164
E021-04m_heterog	21	384,9754502	200,569	359,6958908	200,392	388,3271413	200,32	359,1869194	200,226
E026-08m_heterog	26	520,2286662	200,18	#N/A	200	#N/A	200	#N/A	200
E030-03g_heterog	30	552,9670068	201,209	479,6088464	200,195	524,874335	200,304	413,7057728	200,788
E036-11h_heterog	36	633,2037676	200,382	#N/A	200	#N/A	200	#N/A	200,912
E045-04f_heterog	45	908,5836799	204,05	711,0918715	201,474	753,839904	200,648	1007,396766	200,678
E051-05e_heterog	51	759,5066261	209,305	#N/A	200	#N/A	200	#N/A	201,615
E072-04f_heterog	72	414,3861913	200,616	#N/A	200	#N/A	200	#N/A	200,616
E076-07u_heterog	76	1050,962497	200,835	#N/A	200	#N/A	200	#N/A	202,395
E101-08e_heterog	101	#N/A	200	#N/A	200	#N/A	200	#N/A	205,359

Figura 5.12: Resultados obtenidos con CPLEX para el CVRP heterogéneo.

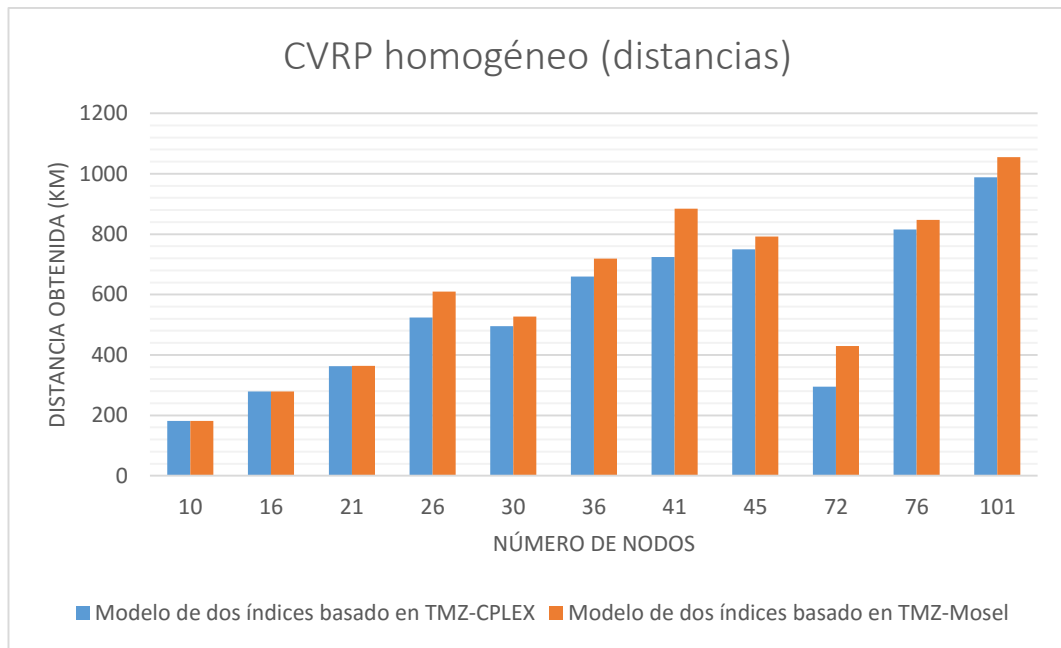


Figura 5.13: Comparativa de distancias modelo de dos índices.

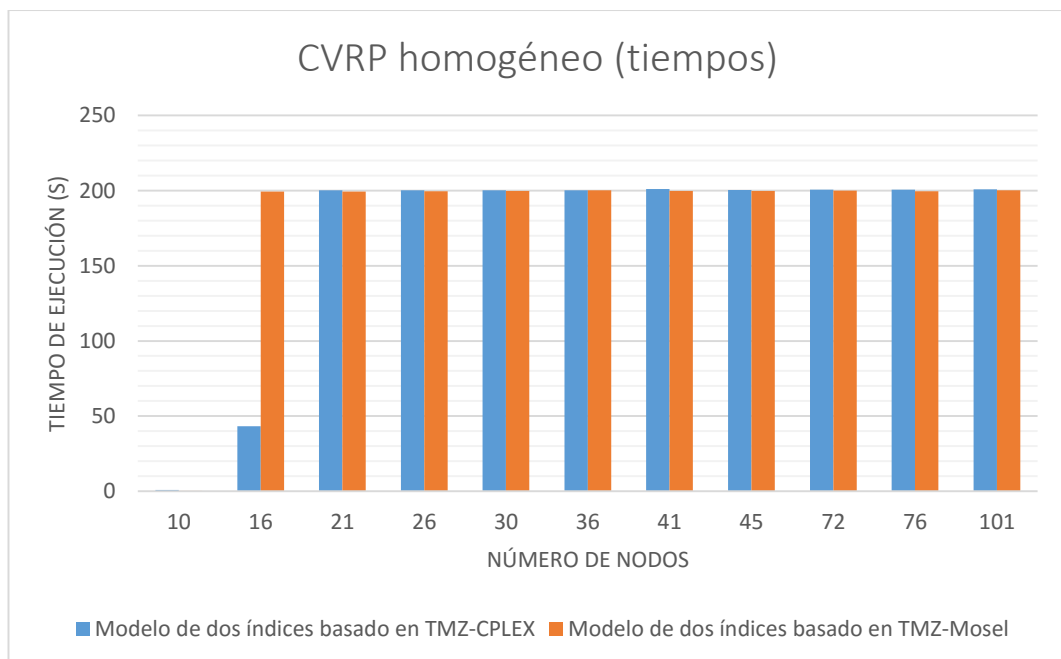


Figura 5.14: Comparativa de tiempos modelo de dos índices.

5.7.2. Resumen de resultados obtenidos. Caso homogéneo.

Para el modelo (a), ambos solvers resuelven los mismos problemas, obteniéndose siempre unos resultados iguales o mejores con CPLEX tanto desde el punto de vista de distancias como de tiempos.

Para el modelo (b), ocurre exactamente lo mismo. Ambos solvers resuelven los mismos problemas y CPLEX se demuestra superior tanto en distancias como en tiempos.

Por tanto, podemos concluir que CPLEX se antoja un solver más efectivo que Xpress-IVE para resolver problemas CVRP homogéneos.

En la Tabla 5.9 se puede observar el resumen de resultados obtenidos para el CVRP homogéneo comparando según los criterios de comparación Xpress-IVE con CPLEX²¹.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	9 de 11	1 de 22	20 de 22	Descartado
CPLEX	20 de 22	0 de 22	20 de 22	SELECCIONADO

Tabla 5.8: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRP homogéneo.

5.7.3. Caso heterogéneo.

Para los dos modelos planteados tanto en Xpress-IVE como en CPLEX (modelo de tres índices basado en TMZ (a), modelo de Golden et al. basado en TMZ (b), modelo de Golden et al. basado en reces (c) y modelo de Christofides et al. (d)), comparando ambos resultados, obtenemos lo siguiente:

²¹ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

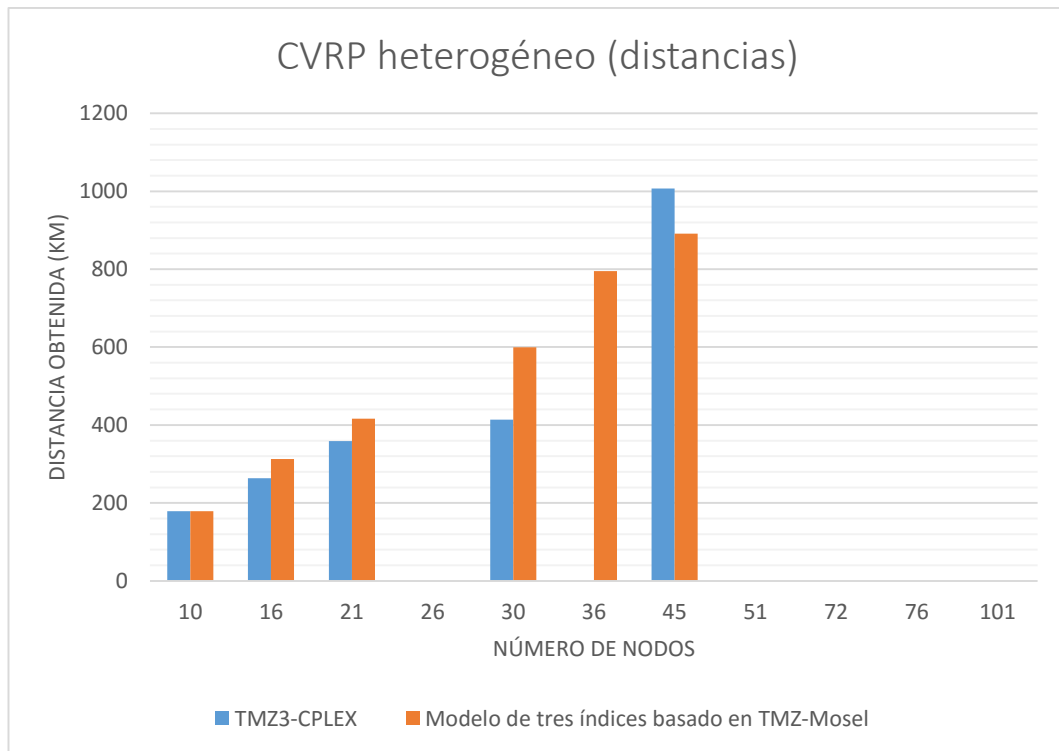


Figura 5.15: Comparativa de distancias modelo (a).

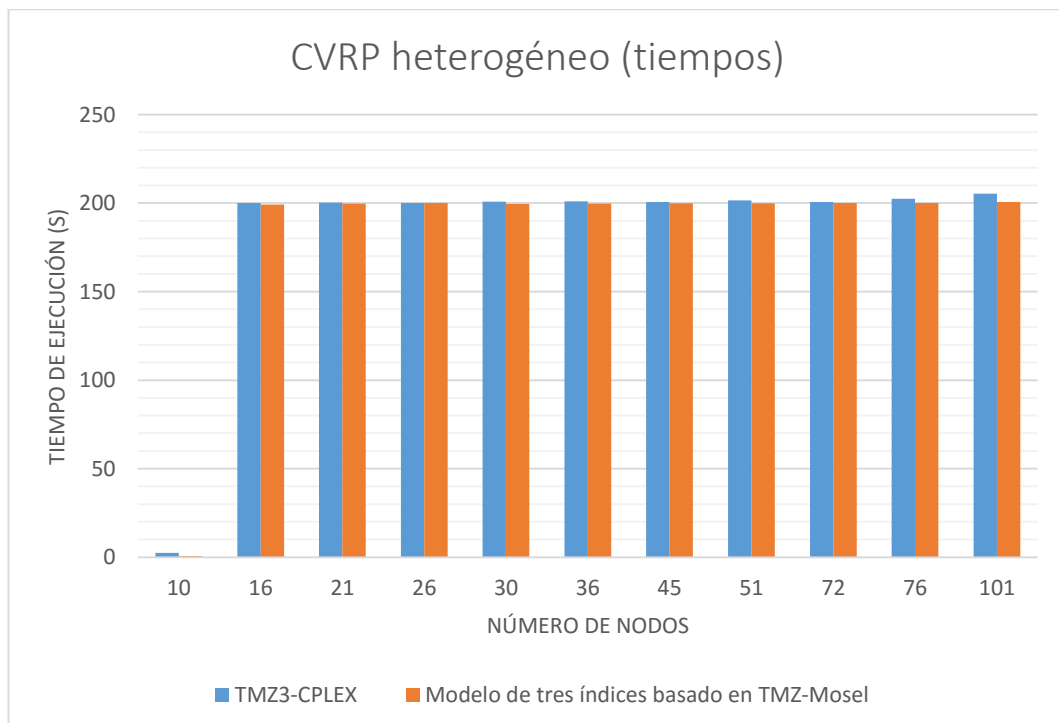


Figura 5.16: Comparativa de tiempos modelo (a).

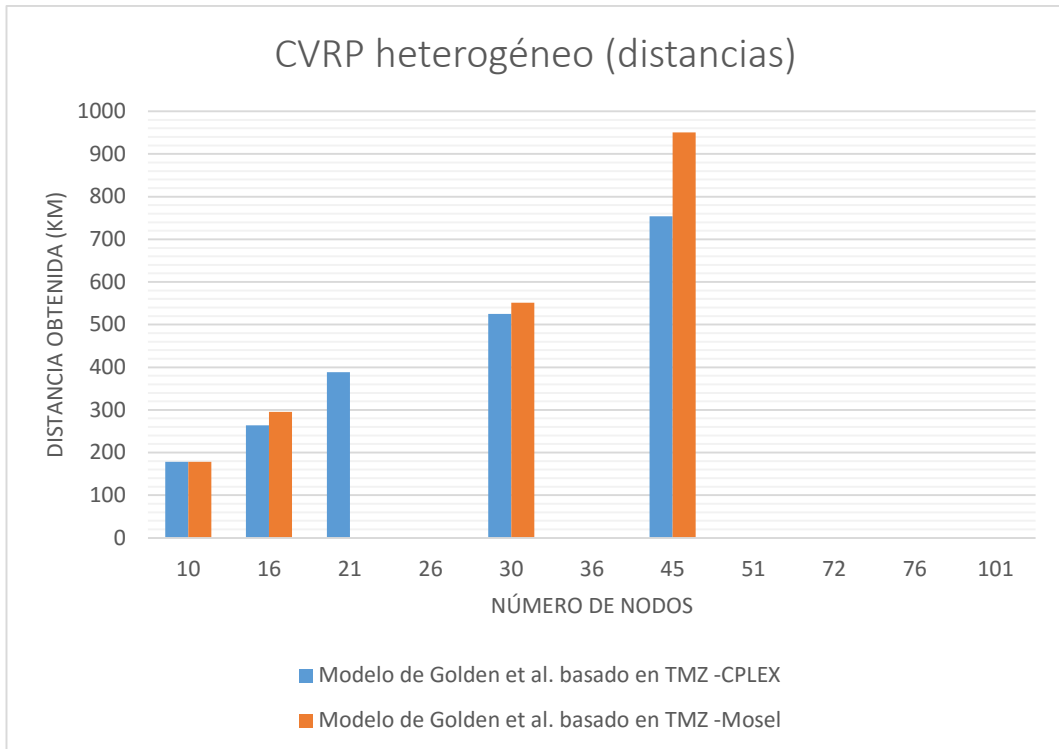


Figura 5.17: Comparativa de distancias modelo (b).

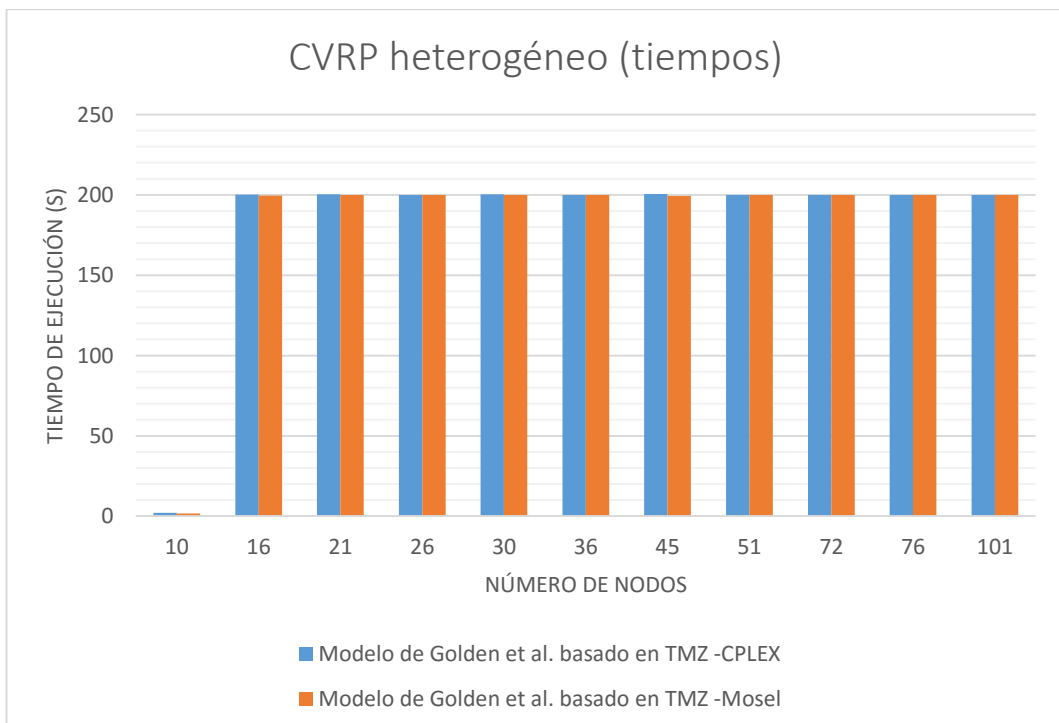


Figura 5.18: Comparativa de tiempos modelo (b).

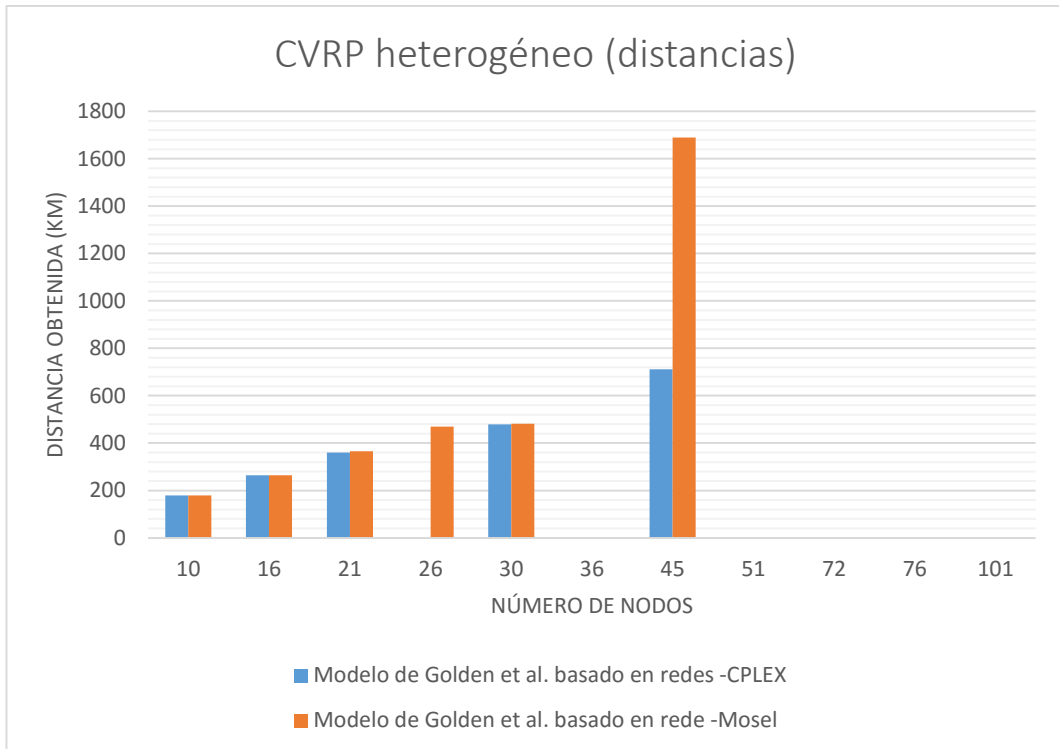


Figura 5.19: Comparativa de distancias modelo (c).

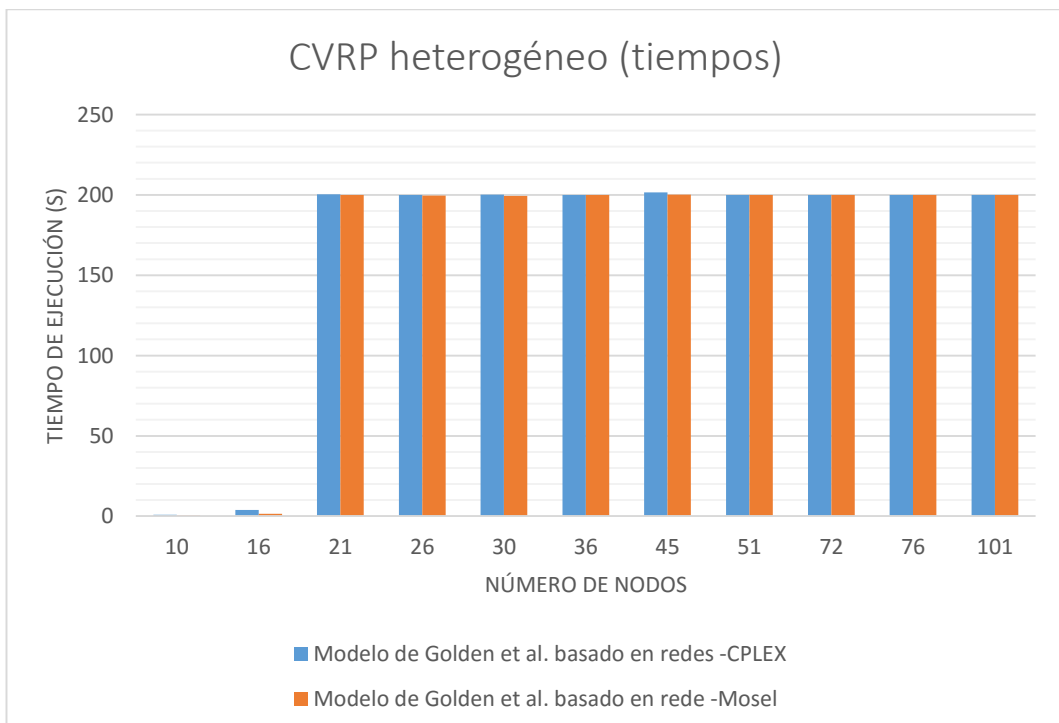


Figura 5.20: Comparativa de tiempos modelo (c).

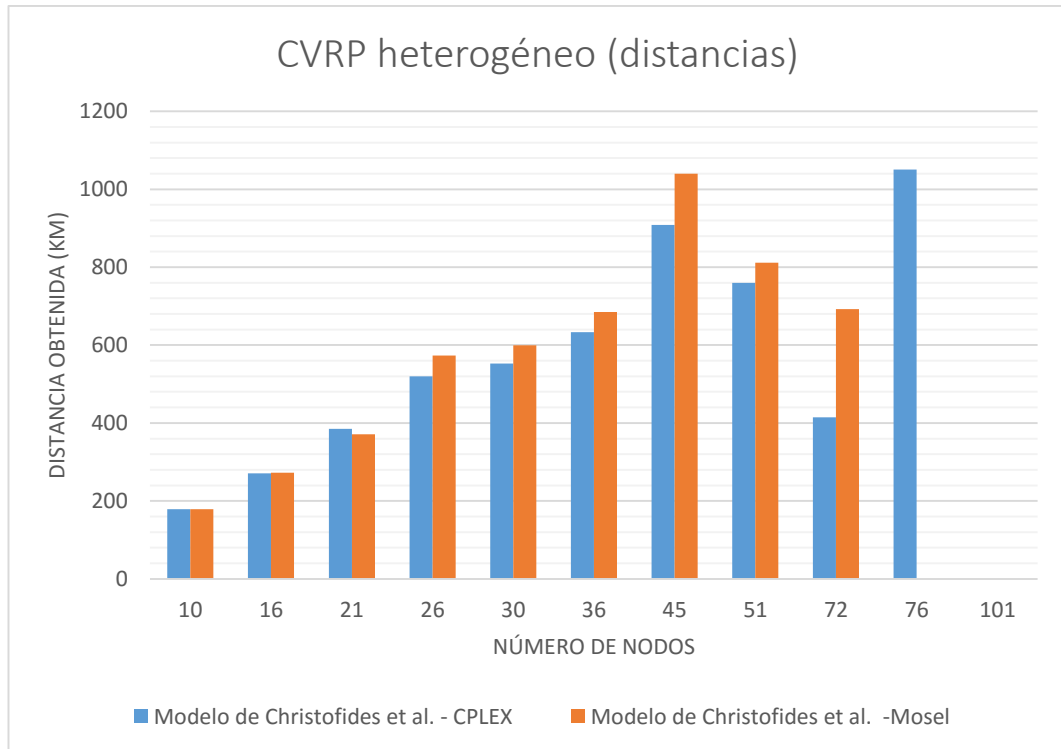


Figura 5.21: Comparativa de distancias modelo (d).

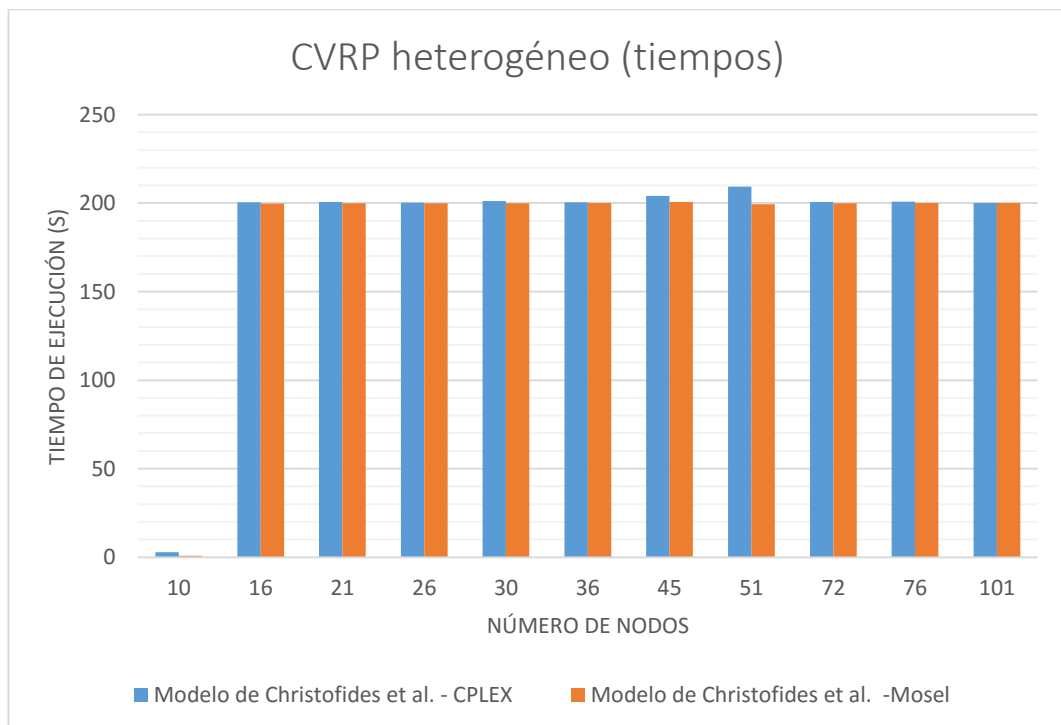


Figura 5.22: Comparativa de tiempos modelo (d).

5.7.2. Resumen de resultados obtenidos. Caso heterogéneo.

Para el modelo (a), CPLEX resuelve 5 de los 11 problemas, por 6 que resuelve Xpress-IVE. De los 5 problemas resueltos por ambos solvers, en 4 obtiene mejores distancias CPLEX y en 1 Xpress-IVE, obteniéndose prácticamente iguales resultados desde el punto de vista de tiempos.

Para el modelo (b), CPLEX resuelve 5 de los 11 problemas, por 4 que resuelve Xpress-IVE. De los 4 problemas resueltos por ambos solvers, en 3 obtiene mejores distancias CPLEX, con iguales resultados para el problema restante. Desde el punto de vista de tiempos, los resultados son prácticamente idénticos.

Para el modelo (c), CPLEX resuelve 5 de los 11 problemas, por 6 que resuelve Xpress-IVE. De los 5 problemas resueltos por ambos solvers, en 1 obtiene mejores distancias CPLEX, obteniéndose prácticamente iguales resultados para los 4 problemas restantes. Los tiempos prácticamente son idénticos, a lo sumo Xpress-IVE obtiene mejores tiempos en 1 caso.

Para el modelo (d), que recordemos ha sido seleccionado como mejor modelo para resolver problemas CVRP heterogéneos, CPLEX resuelve 10 de los 11 problemas, por 9 que resuelve Xpress-IVE. De los 9 problemas resueltos por ambos solvers, en 6 obtiene mejores distancias CPLEX, en 1 Xpress-IVE, y prácticamente iguales en los 2 restantes. Los tiempos son prácticamente idénticos.

En total, de 44 problemas, tanto CPLEX como Xpress-IVE han resuelto 25. De los 23 problemas resueltos por ambos solvers, CPLEX ha obtenido mejores distancias en 14 de ellos y Xpress-IVE en 2, obteniéndose prácticamente iguales resultados en los 7 restantes. Los tiempos prácticamente han sido los mismos para todos los problemas.

Por lo tanto, y dado que CPLEX obtiene iguales o mejores resultados que Xpress-IVE en un 91,3 % de los casos (% sobre el número de problemas para los que se ha obtenido solución factible), se identifica como el solver más adecuado para resolver problemas CVRP heterogéneos.

En la Tabla 5.10 se puede observar el resumen de resultados obtenidos para el CVRP heterogéneo según los criterios de comparación comparando Xpress-IVE con CPLEX²².

²² Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	12 de 44	1 de 44	25 de 44	Descartado
CPLEX	23 de 44	0 de 44	25 de 44	SELECCIONADO

Tabla 5.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRP heterogéneo.

5.8. Conclusiones.

A modo de resumen de todo lo anteriormente citado, en base a los experimentos realizados, y a los criterios establecidos en los apartados 2.4 y 3.2.4., concluimos que:

- Para problemas CVRP homogéneos, el modelo de dos índices basado en TMZ ha demostrado ser el más adecuado, pudiendo escogerse el modelo basado en redes para resolver problemas pequeños en los que el tiempo de ejecución sea crítico.
- Para resolver problemas CVRP heterogéneos, el modelo de Christofides et al. se ha seleccionado como el más adecuado, sobre todo debido a la falta de efectividad del resto de modelos formulados.
- En la comparativa de solvers, se ha seleccionado CPLEX como el que mejores resultados genera para resolver problemas CVRP.

Si bien cuestionarse el por qué un modelo obtiene mejores resultados que otro no tendría mucho sentido debido a que la respuesta reside en la formulación matemática empleada, sí lo tiene en el caso de los solvers. Para responder a esta pregunta, rogamos leer el apartado 3.2.5, en donde se ha tratado este tema con anterioridad.

CAPÍTULO 6

PROBLEMAS DE RUTAS CON CAPACIDADES Y VENTANAS DE TIEMPO (CVRPTW).

6.1. Descripción del problema.

En el apartado 2.3 ya se definió el problema CVRPTW (Capacitated Vehicle Routing Problem with Time Windows) como una extensión del CVRP en donde además de las restricciones de demandas y capacidades, a cada cliente/nodo i se le asocia un intervalo de tiempo (ventana de tiempo) $[a_i, b_i]$ durante el cual tiene que ser atendido, siendo a_i la fecha más temprana en que puede comenzar el servicio en dicho nodo y b_i la fecha más tardía.

Además de las ventanas de tiempo, entran en juego nuevos parámetros como el tiempo que dura el servicio en cada cliente (tiempo de servicio, suele denotarse como s_i , aunque depende de artículos) o el tiempo empleado en ir de un cliente al siguiente en la ruta (suele aparecer como t_{ij} o como θ_{ij} , aunque igualmente depende de artículos).

Como ejemplo práctico de este tipo de problemas sólo hay que imaginar una fábrica en la cual los horarios estén fijados, estableciéndose un periodo de tiempo durante el cual cargar o descargar cada vehículo. Si el vehículo llega antes de poder comenzar el servicio, tendrá que esperar. Por el contrario, si llega más tarde de la hora máxima establecida, no podrá realizar el servicio.

La figura 6.1 muestra gráficamente un caso de este problema, empleando la opción de mapeado de soluciones de Xpress-IVE. En ella podemos observar la solución al problema TW10 dada por el modelo de Cordone – Wolfler de 2

índices (ver apartado 6.2 (b)). En negro aparece el índice de cada cliente; en verde la demanda de cada nodo; las rutas trazadas para los tres vehículos considerados aparecen en rojo; el azul refleja la ventana de tiempo de cada nodo ($[a_i, b_i]$).

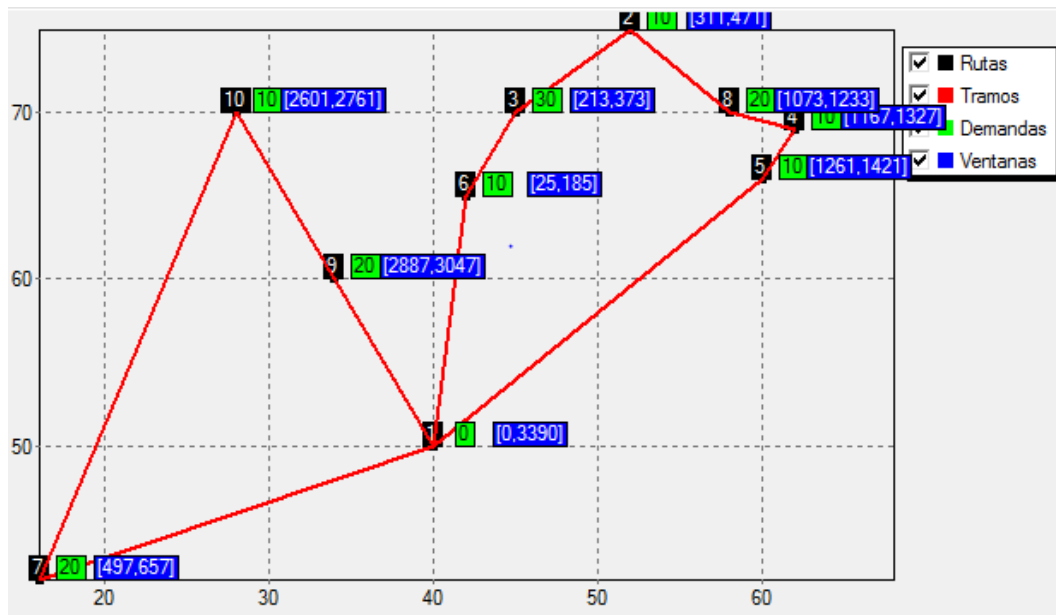


Figura 6.1: ejemplo gráfico del problema CVRPTW.

El nodo origen tiene una demanda nula, lo que indica que no tiene necesidades, sino que de él partirán los vehículos descargados, visitarán a los clientes dentro de las ventanas de tiempo establecidas, recogerán la mercancía de dichos nodos y retornarán al nodo origen para terminar la ruta.

Al igual que ocurría en el capítulo anterior, el primer objetivo que hemos perseguido es discernir qué modelo es el más adecuado para resolver problemas de este tipo, según los criterios establecidos en el apartado 2.4. Asimismo, y en base a los resultados obtenidos en el Anexo para el problema CVRPTW, queda justificada la elección como segundo objetivo el intentar reducir el número de vehículos empleado, siempre y cuando esto suponga obtener una menor distancia total recorrida. Para ello, ha sido necesario modificar algunas restricciones respecto a sus formulaciones originales, las cuáles han sido indicadas a pie de página.

Como tercer objetivo, nos planteamos llegar a una solución justificada de qué solver (Xpress-IVE o CPLEX) es el más adecuado para resolver problemas CVRPTW.

Por último, y en consonancia con lo visto en el apartado 2.3, se han considerado dos tipos de problemas: homogéneo, en el cual todos los vehículos tienen la misma capacidad y heterogéneo, en el que cada vehículo puede tener una capacidad distinta al resto. También consideraremos el caso SCVRP (ver apartado 2,3), es decir, que la distancia entre el cliente A y el B es igual a la que existe entre el cliente B y el A.

6.2. Modelos planteados. Caso homogéneo.

a) Modelo de Sopnamayee Acharya.

El primero de los modelos seleccionados aparece en (Acharya, 2013) y queda formulado de la siguiente manera:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ik} : variable binaria para indicar si el cliente i es atendido por el vehículo k .
- t_{ik} : tiempo de llegada al nodo i del vehículo k . Se ha añadido un índice respecto del artículo original para facilitar la comprensión de los resultados de salida.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_j : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.
- s_i : tiempo de servicio en el nodo i .
- θ_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .
- T : constante de valor elevado para servir como cota superior a algunas restricciones (en este caso se calcula como $1000 \cdot b_{origen}$).

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{i=1 \\ i \neq \text{origen}}}^n (y_{ik} \cdot d_i) \leq \text{cap} \quad \forall k = 1..K \quad (2)$$

(2) La suma de las demandas de los nodos atendidos por un vehículo no puede ser mayor que la capacidad del mismo.

$$\sum_{k=1}^K y_{\text{origen } k} \leq K \quad (3)$$

(3) El nodo origen es atendido a lo sumo por todos los vehículos, existiendo por consiguiente a lo sumo K arcos salientes de dicho nodo, siendo K el número de vehículos que aparece en el fichero de datos.²³

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall j \neq \text{origen} \quad (3')$$

(3') Todo nodo salvo el nodo origen debe ser atendido por un solo vehículo.

$$\sum_{j=1}^n x_{ijk} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (5)$$

(5) En caso de que un nodo sea atendido por un determinado vehículo, debe existir un solo arco saliente de dicho nodo. En caso contrario, no existirán arcos salientes de dicho nodo empleando ese vehículo.

²³ Esta restricción ha sido modificada respecto de la formulación original.

$$\sum_{j=1}^n x_{jik} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (6)$$

(6) En caso de que un nodo sea atendido por un vehículo, la suma de los arcos entrantes a dicho nodo empleando ese vehículo será 1. En caso contrario, será 0.

$$t_{jk} \geq t_{ik} + s_i + \theta_{ij} - T(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (8)$$

(8) Restricciones de tiempo. Representan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo igual al tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \sum_{j=1}^n x_{ijk} \leq t_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i = 1..n; k = 1..K \quad (9_1 \text{ y } 9_2)$$

(9_1 y 9_2) En el contexto de las ventanas de tiempo, los tiempos de comienzo de servicio en cada nodo estarán acotados por un límite superior y un límite inferior.

$$x_{ijk} \in \{0,1\} \quad \forall i,j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K$$

$$y_{ik} \in \{0,1\}$$

$$t_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

b) Modelo de Roberto Cordone y Roberto Wolfler de dos índices.²⁴

En el artículo (Cordone & Wolfler Calvo, 2001), los autores proponen una nueva heurística para el CVRPTW. No obstante, previamente realizan una formulación

²⁴ En este caso no ha sido necesario modificar ninguna restricción respecto de la formulación original. La propia formulación consigue el segundo objetivo fijado.

del problema CVRPTW muy interesante desde el punto de vista de los resultados obtenidos. Dicha formulación aparece como sigue:

Variables:

- x_{ij} : variable binaria para indicar si existe el arco ij .
- y_i : carga del vehículo al salir del nodo i .
- p_i : especifica el tiempo de comienzo de servicio en el nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.
- s_i : tiempo de servicio en el nodo i .
- t_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .
- M : constante de valor elevado para servir como cota superior a algunas restricciones.

Formulación:

$$Min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ij} d_{ij}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1 \quad \forall i \neq origen \quad (2)$$

(2) De cualquier nodo distinto del nodo origen debe salir un arco.

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1 \quad \forall j \neq \text{origen} \quad (3)$$

(3) A cualquier nodo distinto del nodo origen debe llegar un arco.

$$p_i + s_i + t_{ij} - p_j \leq M(1 - x_{ij}) \quad \forall i \neq j; j \neq \text{origen} \quad (4)$$

(4) Restricciones de tiempo. Representan que, de existir el arco ij , el tiempo de comienzo de servicio en el nodo j deberá ser, como mínimo, el tiempo de comienzo de servicio del nodo i , más el tiempo de servicio en el nodo i , más el tiempo empleado en ir del nodo i al nodo j . En caso de no existir el arco ij , se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \leq p_i \leq b_i \quad \forall i = 1..n \quad (5_1 \text{ y } 5_2)$$

(5_1 y 5_2) En el contexto de las ventanas de tiempo, los tiempos de comienzo de servicio estarán acotados por un límite superior y un límite inferior.

$$y_i + d_j - y_j \leq M(1 - x_{ij}) \quad \forall i \neq j; j \neq \text{origen} \quad (6)$$

(6) De existir el arco ij , la carga transportada por el vehículo al salir del nodo j será como mínimo la carga transportada al salir del nodo i , más la demanda del nodo j . En caso de no existir el arco ij , se tiene una cota superior para la carga del vehículo.

$$d_i \leq y_i \leq \text{cap} \quad \forall i = 1..n \quad (7_1 \text{ y } 7_2)$$

(7_1 y 7_2) Cotas superior e inferior para la carga transportada por los vehículos.

$$x_{ij} \in \{0,1\} \quad \forall i,j/a_i + s_i + t_{ij} \leq b_j$$

$$y_i \geq 0 \quad \forall i = 1..n$$

$$p_i \geq 0 \quad \forall i = 1..n$$

c) **Modelo de Roberto Cordone y Roberto Wolfler de tres índices.**

En base a la formulación del modelo anterior (Cordone & Wolfler Calvo, 2001), hemos desarrollado un modelo propio semejante, pero añadiendo un índice para denotar el vehículo empleado, a fin de comprobar si este modelo ofrece mejores resultados respecto al modelo de dos índices.

VARIABLES:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ik} : carga del vehículo k al salir del nodo i .
- p_{ik} : especifica el comienzo del servicio en el nodo i con el vehículo k .

PARÁMETROS:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_j : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.
- s_i : tiempo de servicio en el nodo i .
- t_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .
- M : constante de valor elevado para servir como cota superior a algunas restricciones.

Formulación:

$$Min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{j=1}^n x_{ijk} - \sum_{j=1}^n x_{jik} = 0 \quad \forall i = 1..n; k = 1..K \quad (1)$$

(1) El número de arcos que llegan a un nodo empleando un vehículo debe ser igual al número de arcos que salen de dicho nodo empleando ese mismo vehículo.

$$\sum_{\substack{j=1 \\ j \neq i}}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen} \quad (2)$$

(2) Para todo nodo distinto del nodo origen, el número de arcos que salen de dicho nodo debe ser igual a 1.

$$\sum_{j=1}^n x_{\text{origen } j k} \leq 1 \quad \forall k = 1..K \quad (3)$$

(3) Para el nodo origen, para cada vehículo, puede existir un arco.²⁵

$$\sum_{\substack{i=1 \\ i \neq j}}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen} \quad (4)$$

(4) Para todo nodo distinto del nodo origen, debe existir un arco entrante a dicho nodo.

$$p_{ik} + s_i + t_{ij} - p_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (5)$$

(5) Restricciones de tiempo. Representan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo igual al tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij con el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \sum_{j=1}^n x_{ijk} \leq p_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i \neq \text{origen}; k = 1..K \quad (6_1 \text{ y } 6_2)$$

(6_1 y 6_2) Cotas superior e inferior para los tiempos de comienzo de servicio en cada nodo para cada vehículo.

²⁵ Esta restricción ha sido modificada respecto de la formulación original.

$$y_{ik} + d_j - y_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (7)$$

(7) Restricción para la carga transportada por el vehículo k.

$$d_i \sum_{j=1}^n x_{ijk} \leq y_{ik} \leq cap \sum_{j=1}^n x_{ijk} \quad \forall i = 1..n; k = 1..K \quad (8_1 \text{ y } 8_2)$$

(8_1 y 8_2) Cotas superior e inferior para la carga transportada por el vehículo k al salir del nodo i.

$$x_{ijk} \in \{0,1\} \quad \forall i, j/a_i + s_i + t_{ij} \leq b_j; k = 1..K$$

$$y_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

$$p_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

d) Modelo Time Index Formulation (TIF).

En (Dash, Günlük, Lodi, & Tramontani, 2009), aparece un interesante concepto de cubos de tiempo que se emplea para resolver problemas CVRPTW. Se trata de un artículo bastante interesante por su contenido innovador, de manera que recomendamos encarecidamente su lectura. En lo referente al modelado, queda formulado como sigue:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{itk} : variable binaria que vale 1 si el tiempo de servicio en el nodo i para ir de i a j con el vehículo k es t , y 0 en caso contrario. Se ha identificado t como un índice que genera un número de particiones temporales igual al máximo valor b_i que tengamos (ver definición de b_i a continuación en el apartado parámetros).
- z_{itk} : variable binaria que indica si en el nodo i el tiempo de servicio comienza en t con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .

- d_i : demanda de cada nodo.
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.
- s_i : tiempo de servicio en el nodo i .
- θ_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .

Conjuntos:

- W_t : para cada nodo i , W_t hace referencia a que se debe respetar la ventana de tiempo de ese nodo. Esto es, por ejemplo, que si para el nodo i aparece $t \in W_t$, t deberá cumplir las restricciones $t \geq a_i$ y $t \leq b_i$, siendo t el tiempo de comienzo del servicio para el nodo i .
- V^+ : se emplea para denotar el conjunto de nodos j tales que puede existir arco ij . Esto es, se debe cumplir que $a_i + s_i + \theta_{ij} \leq b_j$ y, además, que $t + s_i + \theta_{ij} \leq b_j$, siendo t el tiempo de comienzo del servicio para el nodo i .
- V^- : este conjunto sirve para denotar el conjunto de nodos j tales que puede existir arco entre el nodo j y el nodo i . Esto es, se debe cumplir que $a_j + s_j + \theta_{ji} \leq b_i$ y, además, que $t + s_j + \theta_{ji} \leq b_i$, siendo t el tiempo de comienzo del servicio para el nodo j .
- $I_k(i,t)$: con este conjunto denotamos a la colección de posibles tiempos de comienzo de servicio para el nodo k , asumiendo que el tiempo de comienzo de servicio en el nodo i es t y que el arco (k,i) está seleccionado. Esto se traduce en que $I_k(i,t) = \{\tau \in W_k: \max\{\tau + \theta_{ki}, a_i\} = t\}$. Dicho de otra manera, si el tiempo de comienzo de servicio t en el nodo i es a_i , entonces el tiempo de comienzo de servicio en el nodo k es algún $\tau \in W_k$ tal que $\tau \leq a_i - \theta_{ki}$, y si es cualquier otro t , el tiempo de comienzo de servicio en el nodo k es exactamente $t - \theta_{ki}$, siempre y cuando pertenezca a W_k .

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{t \in W_t} \sum_{k=1}^K z_{itk} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

(1) Para cada nodo distinto del nodo origen, sólo puede haber un tiempo de comienzo de servicio, empleando un único vehículo.

$$\sum_{t \in W_t} \sum_{k=1}^K z_{\text{origen} tk} \leq K \quad (1')$$

(1') Para el nodo origen habrá a lo sumo K tiempos de comienzo de servicio, siendo K el número de vehículos máximo a emplear, que viene en el fichero de datos.²⁶

$$\sum_{j \in V^+_i} y_{ijtk} = z_{itk} \quad \forall i = 1..n; t \in W_t; k = 1..K \quad (2)$$

(2) De ser el tiempo de comienzo de servicio del nodo i t empleando el vehículo k, tendremos que el número de arcos salientes de dicho nodo i será 1. En caso contrario, del nodo i no podrán salir arcos empleando el vehículo k en dicho instante t.

En el artículo original se excluye el nodo destino al poner $i = 1..n$, pero esto da lugar a que se generen subtours ya que en nuestro caso los nodos origen y destino son el mismo nodo.

$$\sum_{k \in V^-_i} \sum_{\tau \in I_k(i,t)} y_{k i \tau v} = z_{itv} \quad \forall i \neq \text{origen}; t \in W_t; v = 1..K \quad (3)$$

²⁶ Esta restricción ha sido modificada respecto de la formulación original.

(3) De ser el nodo i (distinto del nodo origen) atendido por el vehículo v en un tiempo de servicio t , existirá un nodo k atendido por el vehículo v tal que el arco ki exista y cuyo tiempo de comienzo de servicio τ pertenecerá a $I_k(i,t)$.

$$\sum_{t \in W_t} y_{ijtk} = x_{ijk} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K \quad (4)$$

(4) De existir el arco ij empleando el vehículo k , el nodo i será atendido en un tiempo de servicio t empleando el vehículo k .

En el artículo se excluye el nodo origen en i , pero esto da lugar a que se generen subtours ya que en nuestro caso el nodo origen y destino coinciden.

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \cdot d_i \leq cap \quad \forall k = 1..K \quad (5)$$

(5) Restricción que nos indica que la carga transportada por un vehículo no puede ser mayor que la capacidad máxima de dicho vehículo.

En el artículo se excluye el nodo origen en i , pero esto da lugar a que se generen subtours ya que en nuestro caso el nodo origen y destino coinciden.

$$\begin{aligned} x_{ijk} &\in \{0,1\} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K \\ y_{ijtk} &\in \{0,1\} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; t \geq a_i; t \leq b_i; k = 1..K \\ z_{itk} &\in \{0,1\} \quad \forall i/t \geq a_i; t \leq b_i; k = 1..K \end{aligned}$$

e) Modelo de Paolo Toth y Daniele Vigo.

En el libro (Toth & Vigo, "The Vehicle Routing Problem", 2002) aparece formulado el siguiente modelo:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- w_{ik} : variable para indicar el tiempo de comienzo del servicio en el nodo i con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como estamos en el caso homogéneo, todos los vehículos tienen la misma capacidad.
- s_i : tiempo de servicio en el nodo i .
- t_{ij} : tiempo en ir del nodo i al nodo j .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M_{ij} : valor grande para servir de cota superior. En el libro se define con una fórmula para calcularlo, pero da problemas, motivo por el que simplemente se le asigna un valor elevado.

Formulación:

$$Min \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{j=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq origen \quad (2)$$

(2) Para todo nodo distinto del nodo origen debe existir un arco saliente de dicho nodo, empleando un solo vehículo.

$$\sum_{j=1}^n x_{origen jk} \leq 1 \quad \forall k = 1..K \quad (3)$$

(3) Para el nodo origen, para cada vehículo puede existir un arco saliente de dicho nodo. ²⁷

$$\sum_{j=1}^n x_{ijk} - \sum_{j=1}^n x_{jik} = 0 \quad \forall i = 1..n; k = 1..K \quad (4)$$

²⁷ Esta restricción ha sido modificada respecto de la formulación original.

(4) El número de arcos que salen de un nodo debe ser igual al número de arcos que entran en dicho nodo.

$$\sum_{i=1}^n x_{i \text{ origen } k} \leq 1 \quad \forall k = 1..K \quad (5)$$

(5) Para el nodo origen, para cada vehículo puede existir un arco entrante en dicho nodo.²⁸

$$w_{ik} + s_i + t_{ij} - w_{jk} \leq (1 - x_{ijk})M_{ij}$$

$$\forall i = 1..n; j \neq \text{origen}; i \neq j; k = 1..K \quad (6)$$

(6) Restricciones de tiempo. Representan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo igual al tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij con el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \sum_{j=1}^n x_{ijk} \leq w_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i \neq \text{origen}; k = 1..K \quad (7_1 \text{ y } 7_2)$$

(7_1 y 7_2) Cotas superior e inferior para los tiempos de comienzo de servicio de todos los nodos salvo el nodo origen.

$$a_{\text{origen}} \leq w_{\text{origen } k} \leq b_{\text{origen}} \quad \forall k = 1..K \quad (8_1 \text{ y } 8_2)$$

(8_1 y 8_2) Cotas superior e inferior para el tiempo de comienzo de servicio del nodo origen.

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \cdot d_j \leq \text{cap} \quad \forall k = 1..K \quad (9)$$

²⁸ Esta restricción ha sido modificada respecto de la formulación original.

(8_1 y 8_2) Representa que la carga transportada por un vehículo en cada ruta no puede superar la capacidad máxima de dicho vehículo.

$$x_{ijk} \in \{0,1\} \quad \forall i,j/a_i + s_i + t_{ij} \leq b_j; k = 1..K$$

$$w_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

6.3. Modelos planteados. Caso heterogéneo.

a) Modelo de Sopnamayee Acharya.

Modelo equivalente al propuesto en el caso homogéneo (Acharya, 2013) solo que se han adaptado algunas restricciones al carácter heterogéneo de los vehículos existentes.

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ik} : variable binaria para indicar si el cliente i es servido por el vehículo k .
- t_{ik} : tiempo de llegada al nodo i del vehículo k . Se ha añadido un índice respecto del artículo original para facilitar la comprensión.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- s_i : tiempo de servicio en el nodo i .
- θ_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .

- T: constante de valor elevado para servir como cota superior a algunas restricciones (en este caso se calcula como $1000 \cdot b_{\text{origen}}$.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{i=1 \\ i \neq \text{origen}}}^n (y_{ik} \cdot d_i) \leq \text{cap}_k \quad \forall k = 1..K \quad (2)$$

$$\sum_{k=1}^K y_{\text{origen } k} \leq K \quad (3) \quad ^{29}$$

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall j \neq \text{origen} \quad (3')$$

$$\sum_{j=1}^n x_{ijk} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (5)$$

$$\sum_{j=1}^n x_{jik} = y_{ik} \quad \forall i = 1..n; k = 1..K \quad (6)$$

$$t_{jk} \geq t_{ik} + s_i + \theta_{ij} - T(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (8)$$

$$a_i \sum_{j=1}^n x_{ijk} \leq t_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i = 1..n; k = 1..K \quad (9_1 \text{ y } 9_2)$$

²⁹ Restricción modificada respecto de la formulación original.

$$x_{ijk} \in \{0,1\} \quad \forall i,j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K$$

$$y_{ik} \in \{0,1\}$$

$$t_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

b) Modelo de Roberto Cordone y Roberto Wolfler de tres índices.

Modelo equivalente al propuesto en el caso homogéneo (variante de (Cordone & Wolfler Calvo, 2001)), solo que empleando vehículos heterogéneos.

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ik} : carga del vehículo k al salir del nodo i .
- p_{ik} : especifica el comienzo del servicio en el nodo i con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- s_i : tiempo de servicio en el nodo i .
- t_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .
- M : constante de valor elevado para servir como cota superior a algunas restricciones.

Formulación:

$$\text{Min} \quad \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{j=1}^n x_{ijk} - \sum_{j=1}^n x_{jik} = 0 \quad \forall i = 1..n; k = 1..K \quad (1)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen} \quad (2)$$

$$\sum_{j=1}^n x_{\text{origen} j k} \leq 1 \quad \forall k = 1..K \quad (3)^{30}$$

$$\sum_{\substack{i=1 \\ i \neq j}}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen} \quad (4)$$

$$p_{ik} + s_i + t_{ij} - p_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (5)$$

$$a_i \sum_{j=1}^n x_{ijk} \leq p_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i \neq \text{origen}; k = 1..K \quad (6_1 \text{ y } 6_2)$$

$$y_{ik} + d_j - y_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq j; j \neq \text{origen}; k = 1..K \quad (7)$$

$$d_i \sum_{j=1}^n x_{ijk} \leq y_{ik} \leq cap_k \sum_{j=1}^n x_{ijk} \quad \forall i = 1..n; k = 1..K \quad (8_1 \text{ y } 8_2)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j/a_i + s_i + t_{ij} \leq b_j; k = 1..K$$

$$y_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

³⁰ Esta restricción ha sido modificada respecto de la formulación original.

$$p_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

c) Modelo Time Index Formulation (TIF).

Modelo equivalente al propuesto anteriormente para el caso homogéneo (Dash, Günlük, Lodi, & Tramontani, 2009), sólo que adaptado para el caso heterogéneo. Su formulación aparece a continuación:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ijt} : variable binaria que indica si el tiempo de servicio en el nodo i para ir de i a j con el vehículo k es t . Se ha identificado t como un índice que genera un número de particiones temporales igual al máximo valor b_i que tengamos (ver definición de b_i a continuación en el apartado parámetros).
- z_{itk} : variable binaria que indica si en el nodo i el tiempo de servicio comienza en t con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- s_i : tiempo de servicio en el nodo i .
- θ_{ij} : tiempo que se tarda en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- a_i : cuándo puede comenzar como pronto el servicio en el nodo i .
- b_i : cuándo puede comenzar como tarde el servicio en el nodo i .

Conjuntos:

- W_t : para cada nodo i , W_t hace referencia a que se debe respetar la ventana de tiempo de ese nodo. Esto es, por ejemplo, que si para el nodo i aparece $t \in W_t$, t deberá cumplir las restricciones $t \geq a_i$ y $t \leq b_i$, siendo t el tiempo de comienzo del servicio para el nodo i .
- V^+ : se emplea para denotar el conjunto de nodos j tales que puede existir arco ij . Esto es, se debe cumplir que $a_i + s_i + \theta_{ij} \leq b_j$ y además,

que $t + s_i + \theta_{ij} \leq b_j$, siendo t el tiempo de comienzo del servicio para el nodo i .

- V^- : se emplea para denotar el conjunto de nodos j tales que puede existir arco entre el nodo j y el nodo i . Esto es, se debe cumplir que $a_j + s_j + \theta_{ji} \leq b_i$ y además, que $t + s_j + \theta_{ji} \leq b_i$, siendo t el tiempo de comienzo del servicio para el nodo j .
- $l_k(i,t)$: se emplea para denotar la colección de posibles tiempos de comienzo de servicio para el nodo k , asumiendo que el tiempo de comienzo de servicio en el nodo i es t y que el arco (k,i) está seleccionado. Esto se traduce en que $l_k(i,t) = \{\tau \in W_k: \max\{\tau + \theta_{ki}, a_i\} = t\}$. Dicho de otra manera, si el tiempo de comienzo de servicio t en el nodo i es a_i , entonces el tiempo de comienzo de servicio en el nodo k es algún $\tau \in W_k$ tal que $\tau \leq a_i - \theta_{ki}$, y si es cualquier otro t , el tiempo de comienzo de servicio en el nodo k es exactamente $t - \theta_{ki}$, siempre y cuando pertenezca a W_k .

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{t \in W_t} \sum_{k=1}^K z_{itk} = 1 \quad \forall i \neq \text{origen} \quad (1)$$

$$\sum_{t \in W_t} \sum_{k=1}^K z_{\text{origen} tk} \leq K \quad (1')^{31}$$

$$\sum_{j \in V^+_i} y_{ijtk} = z_{itk} \quad \forall i = 1..n; t \in W_t; k = 1..K \quad (2)$$

³¹ Restricción modificada respecto de la formulación original.

$$\sum_{k \in V-i} \sum_{\tau \in I_k(i,t)} y_{ki\tau v} = z_{itv} \quad \forall i \neq \text{origen}; t \in W_t; v = 1..K \quad (3)$$

$$\sum_{t \in W_t} y_{ijtk} = x_{ijk} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K \quad (4)$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \cdot d_i \leq \text{cap}_k \quad \forall k = 1..K \quad (5)$$

$$\begin{aligned} x_{ijk} &\in \{0,1\} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; k = 1..K \\ y_{ijtk} &\in \{0,1\} \quad \forall i, j/a_i + s_i + \theta_{ij} \leq b_j; t \geq a_i; t \leq b_j; k = 1..K \\ z_{itk} &\in \{0,1\} \quad \forall i/t \geq a_i; t \leq b_j; k = 1..K \end{aligned}$$

d) Modelo de Paolo Toth y Daniele Vigo.

Adaptación al caso heterogéneo del modelo formulado en (Toth & Vigo, "The Vehicle Routing Problem", 2002). Queda formulado como sigue:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- w_{ik} : variable para indicar el comienzo del servicio en el nodo i con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- s_i : tiempo de servicio en el nodo i .
- t_{ij} : tiempo en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).

- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M_{ij} : valor grande para servir de cota superior. En el libro se define con una fórmula para calcularlo, pero da problemas, motivo por el que simplemente se le asigna un valor elevado.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{j=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen} \quad (2)$$

$$\sum_{j=1}^n x_{\text{origen } jk} \leq 1 \quad \forall k = 1..K \quad (3)^{32}$$

$$\sum_{j=1}^n x_{ijk} - \sum_{j=1}^n x_{jik} = 0 \quad \forall i = 1..n; k = 1..K \quad (4)$$

$$\sum_{i=1}^n x_{i \text{ origen } k} \leq 1 \quad \forall k = 1..K \quad (5)^{33}$$

$$w_{ik} + s_i + t_{ij} - w_{jk} \leq (1 - x_{ijk})M_{ij} \quad \forall i = 1..n; j \neq \text{origen}; i \neq j; k = 1..K \quad (6)$$

³² Esta restricción ha sido modificada respecto de la formulación original.

³³ Esta restricción ha sido modificada respecto de la formulación original.

$$a_i \sum_{j=1}^n x_{ijk} \leq w_{ik} \leq b_i \sum_{j=1}^n x_{ijk} \quad \forall i \neq \text{origen}; k = 1..K \quad (7_1 \text{ y } 7_2)$$

$$a_{\text{origen}} \leq w_{\text{origen } k} \leq b_{\text{origen}} \quad \forall k = 1..K \quad (8_1 \text{ y } 8_2)$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \cdot d_j \leq \text{cap}_k \quad \forall k = 1..K \quad (9)$$

$$x_{ijk} \in \{0,1\} \quad \forall i,j/a_i + s_i + t_{ij} \leq b_j; k = 1..K$$

$$w_{ik} \geq 0 \quad \forall i = 1..n; k = 1..K$$

e) Modelo de Manolis N.Kritikos y George Ioannou.

Modelo específico para el caso de flotas heterogéneas propuesto en (N.Kritikos & Ioannou, 2013), que queda formulado como sigue a continuación:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre empleando el vehículo k .
- z_k : variable binaria para indicar si el vehículo k se utiliza o no.
- a_i : variable que indica el momento de llegada al nodo i .
- p_i : variable que indica el momento de salida del nodo i .
- Q_k : variable entera que indica la carga total del vehículo k al realizar una ruta.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- d_i : demanda de cada nodo.
- cap_k : capacidad de cada vehículo.
- s_i : tiempo de servicio en el nodo i .

- t_{ij} : tiempo en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- e_i : fecha más temprana de comienzo del servicio en el nodo i .
- l_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : valor grande para servir de cota superior para algunas restricciones.

Formulación:

$$\text{Min} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{i=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen} \quad (4)$$

(4) Para todo nodo distinto del nodo origen, debe existir un arco entrante a dicho nodo.

$$\sum_{j=1}^n \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen} \quad (5)$$

(5) Para todo nodo distinto del nodo origen, debe existir un arco saliente de dicho nodo.

$$x_{ijk} \leq z_k \quad \forall i, j = 1..n; k = 1..K \quad (6)$$

(6) Si el vehículo k no se usa, no puede existir ningún arco ij que emplee el vehículo k .

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^n x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (7)^{34}$$

³⁴ Esta restricción no ha sido modificada, viene así formulada en el artículo y nos ayuda a cumplir con nuestro segundo objetivo.

$$\sum_{\substack{i=1 \\ i \neq \text{origen}}}^n x_{i \text{ origen } k} \leq 1 \quad \forall k = 1..K \quad (8)^{35}$$

(7), (8) Para el nodo origen, para cada vehículo, puede existir un arco entrante a dicho nodo y un arco saliente de dicho nodo.

$$\sum_{j=1}^n x_{ijk} - \sum_{j=1}^n x_{jik} = 0 \quad \forall i = 1..n; k = 1..K \quad (9)$$

(9) El número de arcos que entran en un nodo debe de ser igual al número de arcos que salen de dicho nodo.

$$a_j \geq (p_i + t_{ij}) - (1 - x_{ijk})M \quad \forall i = 1..n; j \neq \text{origen}; k = 1..K \quad (10)$$

(10) De existir el arco ij y que este sea recorrido por el vehículo k, el tiempo de llegada al nodo j será como mínimo el tiempo de salida del nodo i más el tiempo empleado en ir del nodo i al nodo j. De no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de llegada y salida.

Junto a esta última restricción, se plantea en el artículo otra idéntica, pero empleando el signo \leq . Sin embargo, al plantear estas dos ecuaciones juntas, se dan problemas de no factibilidad del modelo. Por tanto, sólo se ha planteado una de ellas.

$$a_i \leq p_i - s_i \quad \forall i = 1..n \quad (12)$$

(12) Restricción que nos indica que el tiempo de salida de un nodo debe ser como mínimo el tiempo de llegada a dicho nodo más el tiempo de servicio en dicho nodo.

$$e_i \leq p_i - s_i \leq l_i \quad \forall i = 1..n \quad (13_1 \text{ y } 13_2)$$

(13_1 y 13_2) Se deben respetar las restricciones impuestas por las ventanas de tiempo.

³⁵ Esta restricción no ha sido modificada, viene así formulada en el artículo y nos ayuda a cumplir con nuestro segundo objetivo.

Se formula de esta forma debido a que aquí lo que nos va a interesar va a ser $p_i - s_i$, ya que p_i (tiempo de salida del nodo i) puede ser mayor que l_i (tiempo máximo para que comience el servicio en el nodo i), de manera que mientras que $p_i - s_i$ sea $\leq l_i$, no tendremos ningún problema.

$$a_{origen} = 0 \quad (14)$$

(14) El tiempo de llegada al nodo origen debe ser 0 (comenzamos el servicio en dicho nodo lo antes posible).

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \cdot d_j = Q_k \quad \forall k = 1..K \quad (15)$$

$$Q_k \leq cap_k \quad \forall k = 1..K \quad (16)$$

(15), (16) Restricciones que nos aseguran que la carga transportada por un vehículo al realizar la ruta no sea superior a la capacidad de carga máxima de dicho vehículo.

La restricción (16) ha sido modificada respecto a la formulación original, ya que en el artículo toleran que exista una cierta sobrecarga. Aquí esto no va a suceder, y la carga total transportada por un vehículo va a ser como mucho su capacidad.

$$x_{ijk} \in \{0,1\} \quad \forall i,j/e_i + s_i + t_{ij} \leq l_j; k = 1..K$$

$$z_k \in \{0,1\} \quad \forall k = 1..K$$

$$a_i \geq 0 \quad \forall i = 1..n$$

$$p_i \geq 0 \quad \forall i = 1..n$$

$$Q_k \geq 0 \quad \forall k = 1..K$$

6.4. Datos empleados.

En este tipo de problema, se ha empleado como librería de datos para realizar las simulaciones del caso homogéneo (Librería CVRPTW homogéneo, s.f.). De dicha librería, se ha seleccionado el ejemplo C201 como base.

Para el caso heterogéneo, se han empleado los mismos datos, solo que se han asignado las capacidades de cada uno de los vehículos a mano.

Este fichero de datos (C201) fue resuelto inicialmente por W.Cook y J.L.Rich (Cook & Rich, 1999), J.Larsen (Larsen, 1999), entre otros para algún número concreto de nodos (25, 50 y 100). Dichos autores emplearon heurísticas con las que consiguieron obtener las mejores soluciones conocidas hasta la fecha, debido en parte a que emplear un solver para resolver problemas CVRPTW hasta hace unos años era algo inviable.

No obstante, en los últimos años se ha vuelto posible resolver estos problemas empleando solvers, debido la mejora del algoritmo Branch and Cut que han tenido los mismos. Así, vamos a aprovechar la oportunidad y comparar los resultados obtenidos empleando heurísticas con los resultados de nuestros modelos.

Volviendo a los datos empleados, dentro de la citada librería, empleando como base el problema C201, se han generado ficheros con distinto número de nodos: TW7, TW10, TW20, TW30, TW40, TW50, TW60, TW70, TW80, TW90, TW100 para el caso homogéneo y TW7_heterog, TW10_heterog, TW20_heterog, TW30_heterog, TW40_heterog, TW50_heterog, TW60_heterog, TW70_heterog, TW80_heterog, TW90_heterog, TW100_heterog para el heterogéneo. El número que tiene asociado cada problema representa el número de nodos a recorrer.

6.5. Resultados obtenidos empleando Xpress-IVE.

Se ha establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización, quedándose con la mejor solución factible obtenida hasta ese momento.

6.5.1. Caso homogéneo.

Los resultados obtenidos han sido:

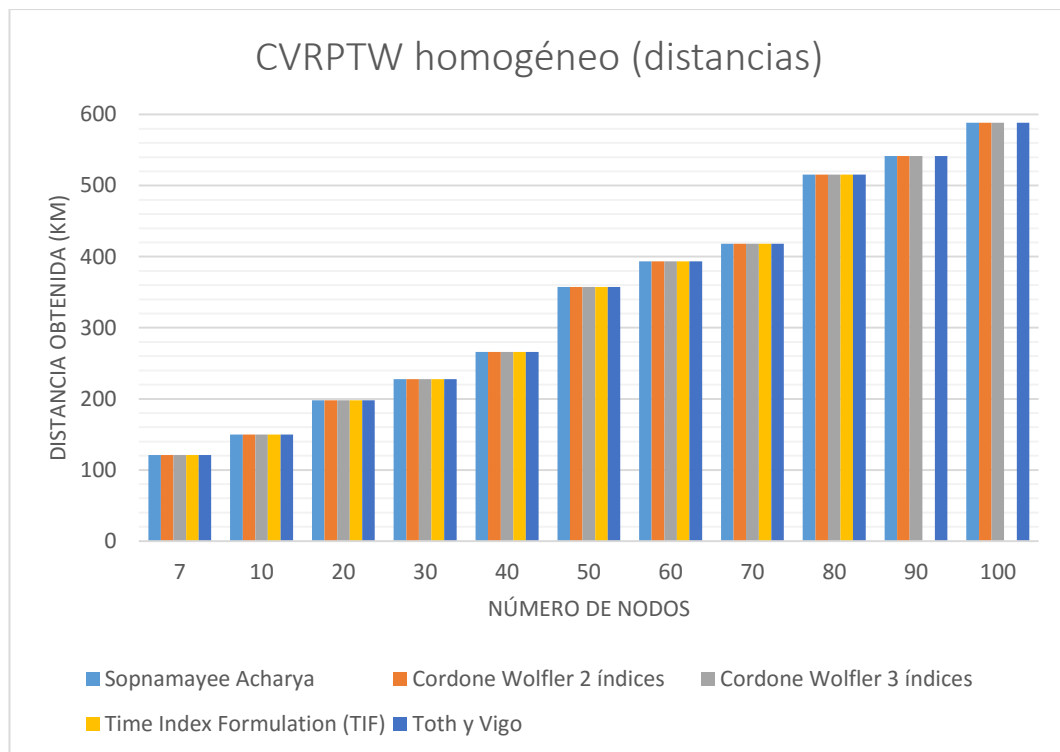


Figura 6.2: Distancias de Xpress-IVE para el problema CVRPTW homogéneo.

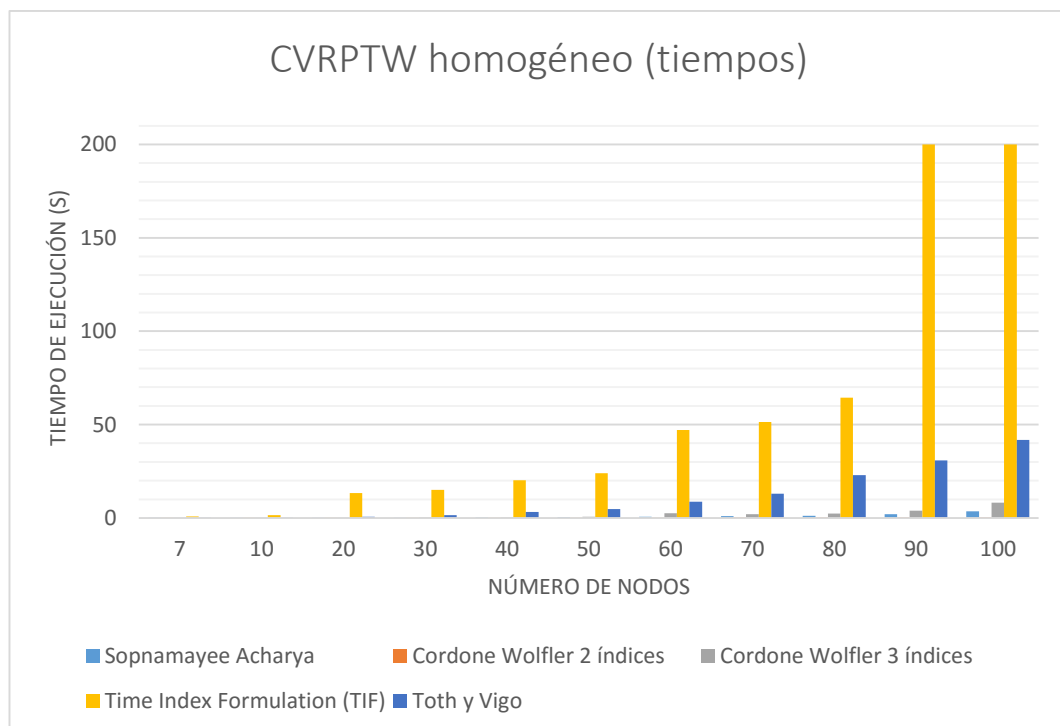


Figura 6.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPTW homogéneo.

6.5.2. Resumen de resultados obtenidos. Caso homogéneo.

Como podemos observar en la Figura 6.2, todos los modelos obtienen las mismas distancias para los problemas que resuelven. Por tanto, este criterio de comparación no será válido para jerarquizar dichos modelos.

Si nos fijamos en los tiempos de ejecución de la Figura 6.3, ya comenzamos a ver diferencias entre los modelos. Así, por ejemplo, el modelo TIF (d), aparte de ser el único que no consigue resolver todos los problemas propuestos (resuelve 9 de 11), es el que con diferencia mayores tiempos de ejecución presenta.

Por otro lado, el modelo de Toth y Vigo (e), pese a resolver los 11 problemas propuestos, presenta unos tiempos de ejecución mayores que los modelos de Acharya (a), Cordone et al. de dos índices (b) y Cordone et al. de tres índices (c). Como dato, para el problema TW30 (de 30 nodos), el tiempo de ejecución de este modelo es alrededor de tres veces mayor con respecto del modelo (c), que es el que obtiene el segundo peor tiempo de ejecución.

Por tanto, y en base a los resultados anteriormente citados, los modelos (d) y (e) han sido descartados, no siendo los más adecuados para resolver problemas CVRPTW. Por su parte, los casos (a), (b) y (c) serán modelados en CPLEX a fin de identificar más claramente las diferencias existentes entre ellos.

En la Tabla 6.1 se puede observar el resumen de resultados obtenidos para el CVRPTW homogéneo empleando Xpress-IVE según los criterios de comparación. El criterio número de problemas resueltos es el número de los problemas planteados para los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas. Por su parte, los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor solución, se cuenta igualmente ya que han llegado a la mejor solución)³⁶.

³⁶ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Sopnamayee Acharya	11 de 11	0 de 11	11 de 11	CPLEX
Cordone et al. 2 ind.	11 de 11	10 de 11	11 de 11	CPLEX
Cordone et al. 3 ind.	11 de 11	1 de 11	11 de 11	CPLEX
TIF	9 de 11	0 de 11	9 de 11	Descartado
Toth y Vigo	11 de 11	0 de 11	11 de 11	Descartado

Tabla 6.1: Resumen de resultados para el problema CVRP homogéneo empleando Xpress-IVE.

En la Tabla 6.2 podemos observar los resultados de distancias y tiempos de ejecución obtenidos para cada uno de los modelos. Además, y como se indicó anteriormente, podemos observar las mejores soluciones conocidas hasta la fecha para este tipo de problemas (caso de 50 y de 100 nodos). Si las comparamos con los resultados que hemos obtenido, vemos que estos son un poco mejores (una distancia ligeramente inferior).

No obstante, hay que tener en cuenta que la finalidad de una heurística no es obtener los mejores resultados posibles, sino ofrecer resultados aceptables en unos tiempos de ejecución muy bajos. En este caso, si bien es cierto que el tiempo de ejecución no es un factor significativo ya que nuestros modelos resuelven los problemas CVRPTW en unos tiempos muy pequeños, que la heurística se aproxime tanto a la solución exacta ofrecida por el solver es un indicador de su gran calidad.

Datos empleados (Solomon, problema C201)		Modelo de Sopnamayee Acharya		Modelo de Cordone Wolflier 2 índices		Modelo de Cordone Wolflier 3 índices		Modelo Time Index Formulation (TIF)		Modelo de Toth y Vigo		
Nombre	Nº de nodos	Mejor solución conocida	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
TW7.TXT	7	No calculada	121,042	0,015	121,042	0,015	121,042	0,012	121,042	0,875	121,042	0,015
TW10.TXT	10	No calculada	149,803	0,013	149,803	0,007	149,803	0,023	149,803	1,548	149,803	0,04
TW20.TXT	20	No calculada	198,184	0,058	198,184	0,014	198,184	0,211	198,184	13,342	198,184	0,703
TW30.TXT	30	No calculada	227,761	0,229	227,761	0,023	227,761	0,284	227,761	14,977	227,761	1,561
TW40.TXT	40	No calculada	266,017	0,317	266,017	0,04	266,017	0,362	266,017	20,187	266,017	3,265
TW50.TXT	50	360,2	357,324	0,432	357,324	0,063	357,324	0,647	357,324	23,936	357,324	4,701
TW60.TXT	60	No calculada	393,358	0,611	393,358	0,219	393,358	2,548	393,358	47,041	393,358	8,679
TW70.TXT	70	No calculada	418,066	0,974	418,066	0,244	418,066	2,018	418,066	51,461	418,066	12,942
TW80.TXT	80	No calculada	515,414	1,23	515,414	0,304	515,414	2,413	515,414	64,486	515,414	22,945
TW90.TXT	90	No calculada	541,424	2,043	541,424	0,359	541,424	3,84	#N/A	200	541,424	30,783
TW100.TXT	100	589,1	588,628	3,591	588,628	0,34	588,628	8,195	#N/A	200	588,628	41,747

Tabla 6.2: Resultados de Xpress-IVE para el problema CVRPTW homogéneo.

6.5.3. Caso heterogéneo.

Para el problema CVRPTW heterogéneo, los resultados obtenidos han sido:

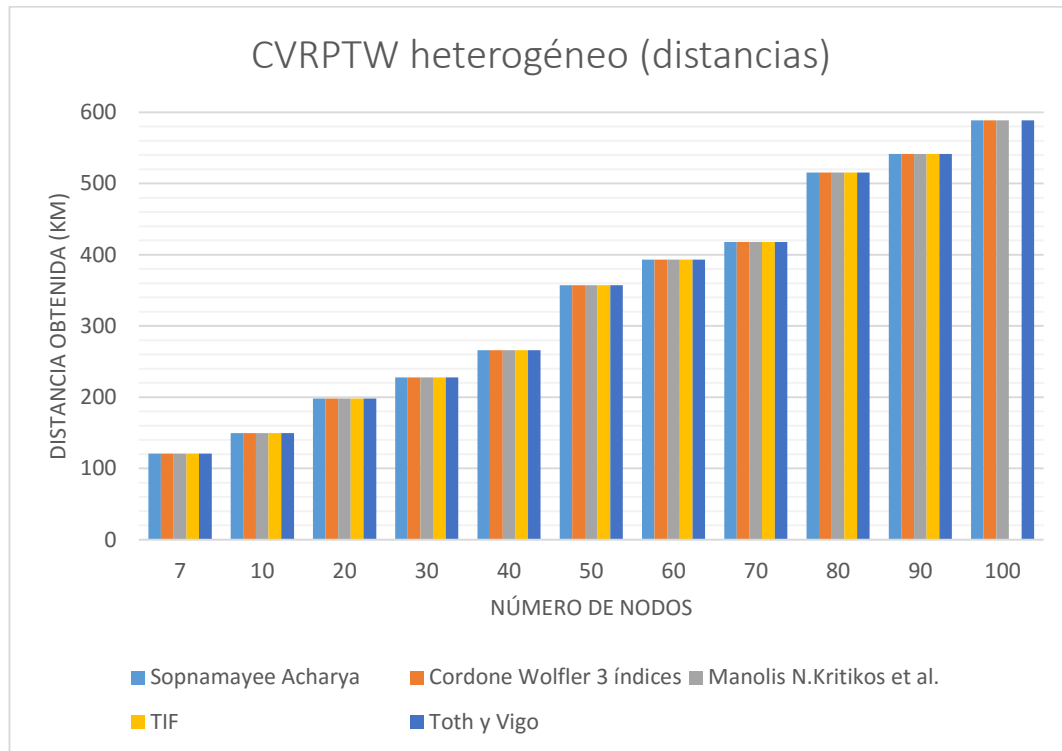


Figura 6.4: Distancias de Xpress-IVE. CVRPTW heterogéneo.

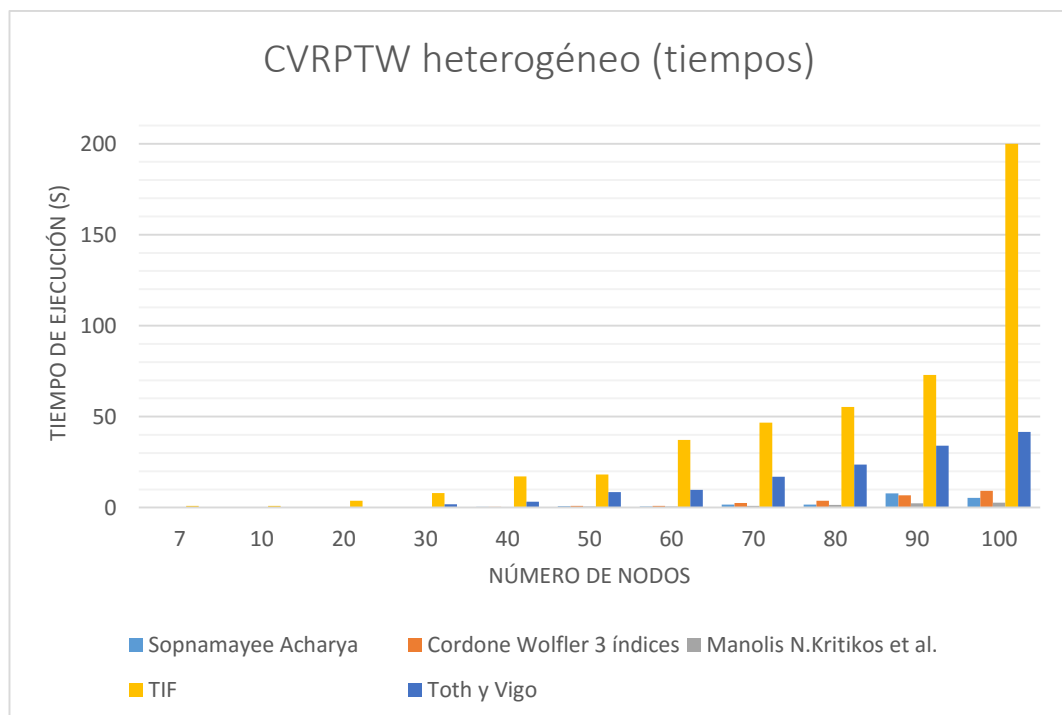


Figura 6.5: Tiempos de ejecución. CVRPTW heterogéneo.

Datos empleados		Modelo de Sopnamayee Acharya		Modelo de Cordone Wolfier		Modelo de Manolis N. Kritikos - George Ioannou		Modelo Time Index Formulation (TIF)		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
TW7_heterog	7	121,042	0,032	121,042	0,01	121,04	0,01	121,042	0,919	121,042	0,028
TW10_heterog	10	149,803	0,025	149,803	0,043	149,8	0,01	149,803	0,91	149,803	0,015
TW20_heterog	20	198,184	0,039	198,184	0,071	198,18	0,04	198,184	3,805	198,184	0,065
TW30_heterog	30	227,761	0,203	227,761	0,264	227,76	0,11	227,761	7,937	227,761	1,906
TW40_heterog	40	266,017	0,328	266,017	0,354	266,02	0,25	266,017	17,174	266,017	3,24
TW50_heterog	50	357,324	0,74	357,324	0,952	357,32	0,55	357,324	18,139	357,324	8,442
TW60_heterog	60	393,358	0,677	393,358	0,887	393,36	0,61	393,358	37,16	393,358	9,806
TW70_heterog	70	418,066	1,573	418,066	2,49	418,07	0,94	418,066	46,686	418,066	16,985
TW80_heterog	80	515,414	1,659	515,414	3,809	515,41	1,51	515,414	55,288	515,414	23,716
TW90_heterog	90	541,424	7,761	541,424	6,834	541,42	2,27	541,424	72,984	541,424	33,96
TW100_heterog	100	588,628	5,36	588,628	9,195	588,63	2,7	#N/A	200	588,628	41,596

Tabla 6.3: Resultados obtenidos con Xpress-IVE para el problema CVRPTW heterogéneo.

6.5.4. Resumen de resultados obtenidos. Caso heterogéneo.

En la Figura 6.4 podemos observar las distancias obtenidas por cada uno de los modelos. Al igual que ocurría en el CVRPTW homogéneo, vemos cómo todos los modelos obtienen los mismos resultados, siempre y cuando consigan encontrar una solución factible. Debido a esto, y a que todos los modelos salvo el (c) Time Index Formulation (TIF) consiguen resolver todos los problemas planteados, no consideramos estos resultados válidos para decidir si un modelo es más adecuado que otro.

Sin embargo, cuando nos fijamos en la Figura 6.5 (en donde vemos los tiempos de ejecución obtenidos), la cosa cambia. Así, a simple vista podemos ver que al igual que ocurría en el caso homogéneo, los modelos (c) TIF y (d) Toth y Vigo presentan unos resultados claramente peores que el resto. Si bien en el modelo (d) esta diferencia es notable a partir del problema TW30, en el caso del modelo (c) desde el problema TW7 podemos ver que presenta unos tiempos de ejecución muy superiores al resto.

Si comparamos los modelos (a) Sopnamayee Acharya, (b) Cordone - Wolfler de 3 índices y (e) M.N.Kritikos et al. observamos que cada uno de ellos consigue resolver los 11 problemas propuestos, obteniendo unos tiempos de ejecución similares hasta TW80, si bien es cierto que el modelo (e) parece que presenta mejores tiempos de ejecución para los problemas TW90 y TW100. No obstante, no consideramos que esta diferencia sea a priori significativa para catalogar el modelo (e) como el más adecuado.

En base a estos resultados (que pueden verse resumidos en la Tabla 6.3) y a las conclusiones anteriormente desarrolladas, concluimos que los modelos (c) y (d) no son válidos para resolver problemas CVRPTW heterogéneos, ya que presentan unos resultados claramente peores al resto. En cuanto a los modelos (a), (b) y (e), se implementarán en CPLEX para intentar llegar a una comparación más profunda.

En la Tabla 6.4 se puede observar el resumen de resultados según los criterios de comparación. El criterio número de problemas resueltos representa los casos en los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas. Los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)³⁷.

³⁷ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Sopnamayee Acharya	11 de 11	1 de 11	11 de 11	CPLEX
Cordone et al. 3 ind.	11 de 11	1 de 11	11 de 11	CPLEX
TIF	10 de 11	0 de 11	10 de 11	Descartado
Toth y Vigo	11 de 11	0 de 11	11 de 11	Descartado
M.Kritikos et al.	11 de 11	10 de 11	11 de 11	CPLEX

Tabla 6.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPTW heterogéneo.

6.6. Resultados obtenidos empleando CPLEX.

Al igual que ocurría en el caso de Xpress-IVE, hemos establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización, quedándose con la mejor solución obtenida hasta ese momento.

6.6.1. Caso homogéneo.

Recordemos que en el apartado 6.5.2 se decidió implementar en CPLEX los modelos propuestos para el CVRPTW homogéneo (a) Acharya, (b) Cordone – Wolfler de 2 índices y (c) Cordone – Wolfler de 3 índices.

En las Figuras 6.6 y 6.7 y en la Tabla 6.5 podemos observar los resultados obtenidos. Al igual que ocurría en la implementación en Xpress-IVE, hemos incluido una columna adicional en donde aparecen los mejores resultados conocidos hasta la fecha empleando heurísticas.

Como podemos observar, los resultados obtenidos empleando solvers son ligeramente mejores que los proporcionados por las heurísticas, lo cual, teniendo en cuenta que la finalidad de una heurística no es obtener la mejor solución sino obtener soluciones aceptables en un tiempo razonable, es un indicador de la gran calidad de las mismas.

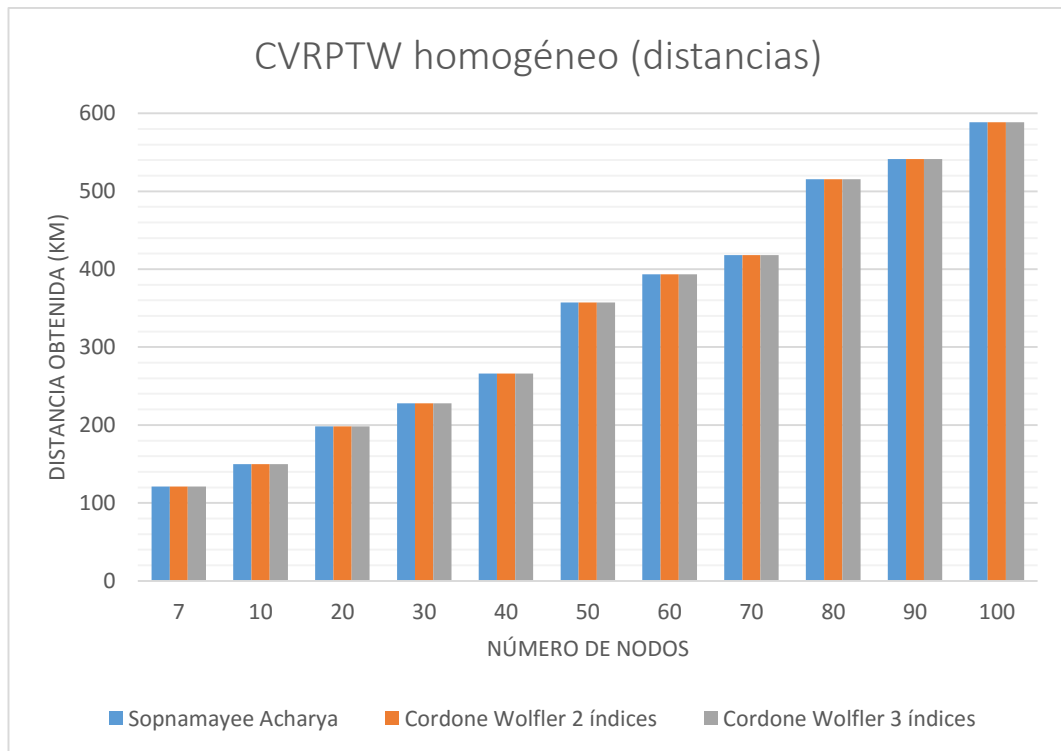


Figura 6.6: Distancias de CPLEX para el problema CVRPTW homogéneo.

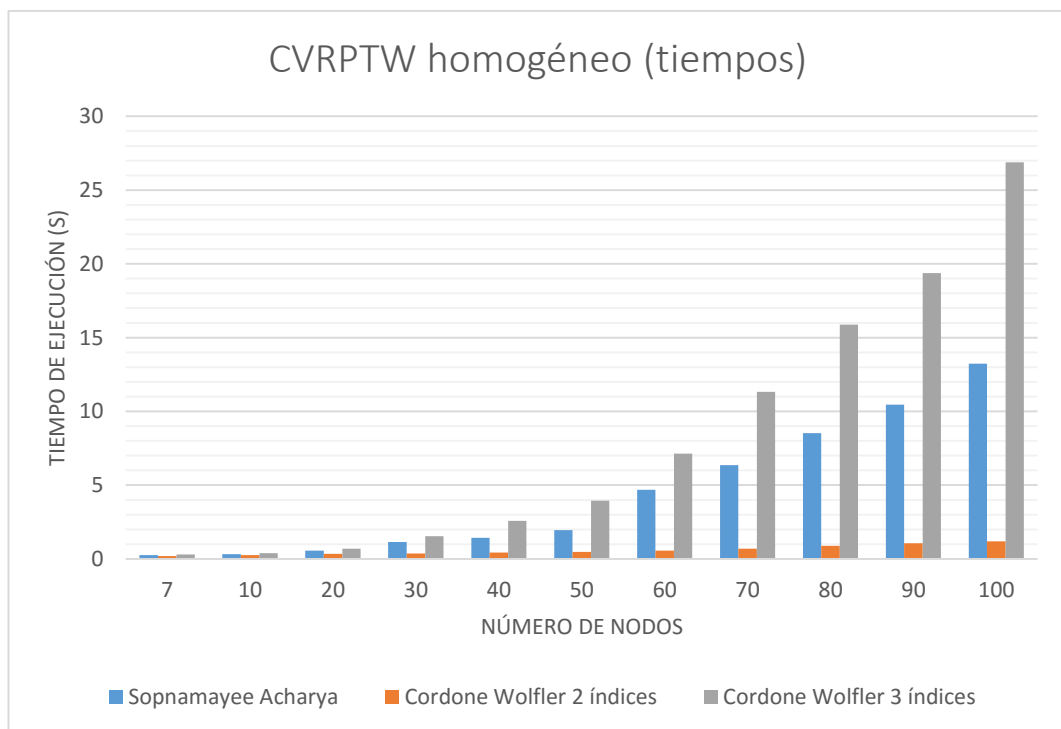


Figura 6.7: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPTW homogéneo.

Datos empleados (Solomon, problema C201)		Modelo de Sopnamayee Acharya		Modelo de Cordone Wolflier 2 índices		Modelo de Cordone Wolflier 3 índices		
Nombre	Nº de nodos	Mejor solución conocida	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
TW7.TXT	7	No calculada	121,0424	0,249	121,0424	0,203	121,0424	0,297
TW10.TXT	10	No calculada	149,8025	0,327	149,8025	0,25	149,8025	0,39
TW20.TXT	20	No calculada	198,1841	0,562	198,1841	0,343	198,1841	0,702
TW30.TXT	30	No calculada	227,7612	1,154	227,7612	0,359	227,7612	1,544
TW40.TXT	40	No calculada	266,0175	1,42	266,0175	0,437	266,0175	2,59
TW50.TXT	50	360,2	357,3244	1,95	357,3244	0,484	357,3244	3,947
TW60.TXT	60	No calculada	393,3582	4,695	393,3582	0,561	393,3582	7,145
TW70.TXT	70	No calculada	418,0656	6,365	418,0656	0,687	418,0656	11,329
TW80.TXT	80	No calculada	515,4136	8,517	515,4136	0,889	515,4136	15,88
TW90.TXT	90	No calculada	541,4242	10,452	541,4242	1,061	541,4242	19,375
TW100.TXT	100	589,1	588,6276	13,244	588,6276	1,186	588,6276	26,879

Tabla 6.5: Resultados obtenidos con CPLEX para el problema CVRPTW homogéneo.

6.6.2. Resumen de resultados. Caso homogéneo.

Al igual que ocurría para estos modelos en Xpress-IVE, el criterio de distancias calculadas no es válido para realizar una comparación entre ellos, debido a que los tres consiguen resolver los 11 problemas propuestos, obteniendo unas distancias exactamente iguales.

Una vez más, cuando nos fijamos en los tiempos de ejecución (Figura 6.7 con resultados desarrollados en Tabla 6.5), nos encontramos en una situación distinta. Los tres modelos comienzan obteniendo unos tiempos de ejecución semejantes, pero a partir del problema TW30 esto cambia, resultando el modelo de Cordone – Wolfler de dos índices (b) mucho más efectivo.

A modo de ejemplo, para el problema TW30, el modelo (b) obtiene un tiempo de ejecución cerca de tres veces inferior al obtenido por el modelo de Acharya (a), modelo con el segundo mejor resultado, incrementándose esta diferencia en los problemas posteriores.

Cabe denotar que los modelos (a) y (c) no obtienen unos tiempos de ejecución excesivamente elevados, es sólo que comparándolos con el modelo (b) lo parecen (pero rondan a lo sumo los 27 segundos para el problema TW100, algo totalmente aceptable).

Por todo ello, se selecciona el modelo de (b) como el más adecuado para resolver problemas CVRPTW homogéneos en base a los criterios establecidos y para aquellos casos en los que se quiera un tiempo de ejecución óptimo.

En la Tabla 6.6 se puede observar el resumen de resultados obtenidos CPLEX según los criterios de comparación establecidos. Los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)³⁸. El criterio número de problemas resueltos es el número de los problemas planteados para los que se ha obtenido una solución factible, en tanto que el estado actual representa las decisiones tomadas.

³⁸ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Sopnamayee Acharya	11 de 11	0 de 11	11 de 11	Descartado
Cordone et al. 2 ind.	11 de 11	11 de 11	11 de 11	SELECCIONADO
Cordone et al. 3 ind.	11 de 11	0 de 11	11 de 11	Descartado

Tabla 6.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPTW homogéneo.

6.6.3. Caso heterogéneo.

Para el problema CVRPTW de flota heterogénea empleando CPLEX se han obtenido los siguientes resultados:

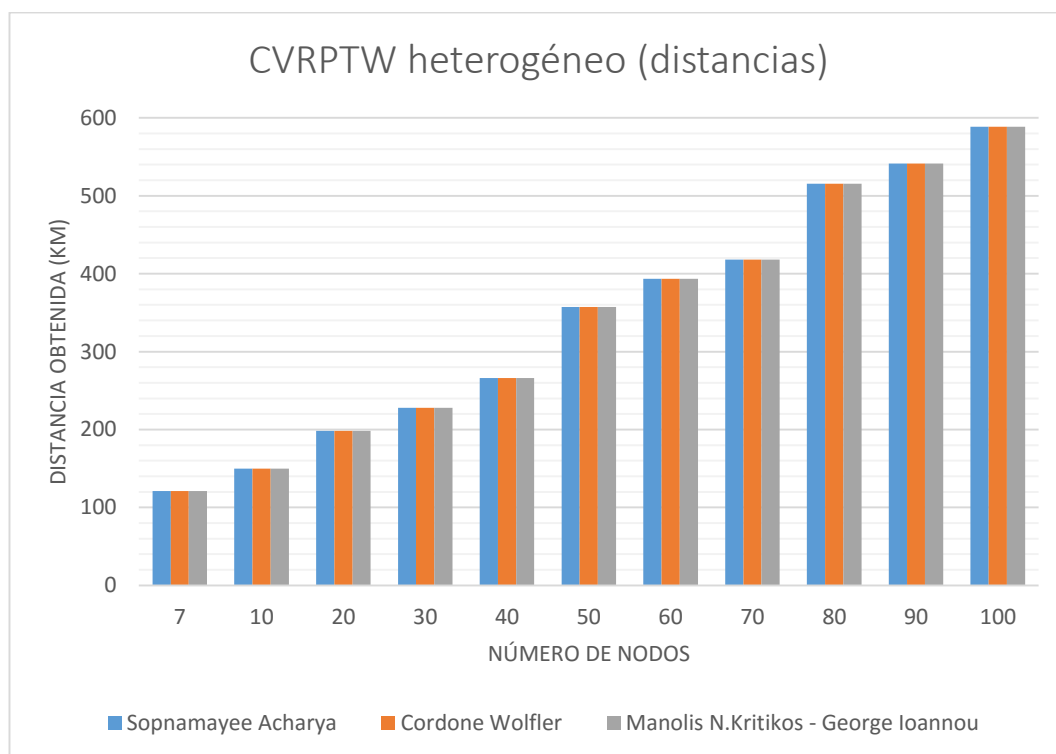


Figura 6.8: Distancias de CPLEX para el CVRPTW heterogéneo.

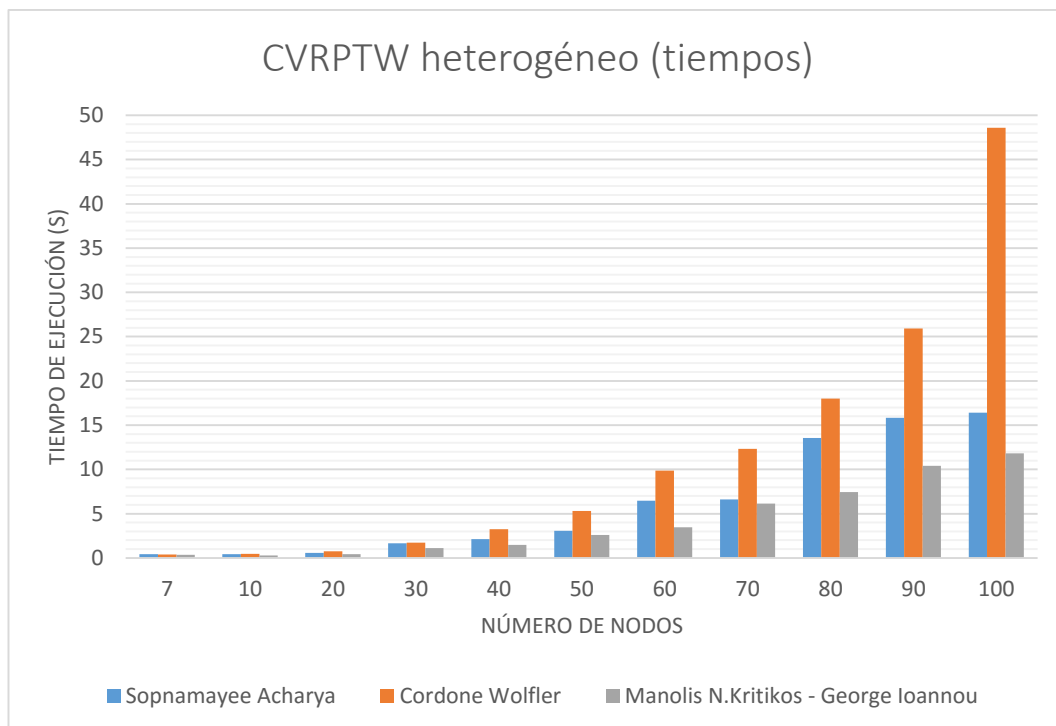


Figura 6.9: Tiempos de CPLEX para el CVRPTW heterogéneo.

6.6.3. Resumen de resultados obtenidos. Caso heterogéneo.

Al igual que nos ha ocurrido al evaluar los modelos seleccionados en el caso homogéneo, el criterio de distancias obtenidas o el de número de problemas resueltos no se demuestran válidos para identificar cuál de todos los modelos es mejor ya que cada uno de los tres modelos resuelve los 11 problemas planteados obteniendo exactamente las mismas distancias. Sin embargo, desde el punto de vista de los tiempos de ejecución obtenidos, sí que vemos que hay un modelo que destaca frente al resto: el modelo de M.N.Kritikos et al. (e).

Dicho modelo presenta, sobre todo a partir del problema TW30_heterog, unos menores tiempos de ejecución que el resto de modelos. Esta diferencia entre modelos se va incrementando a medida que el número de nodos aumenta.

A modo de curiosidad, también podemos observar que el modelo de Acharya (a) presenta mejores tiempos de ejecución que el modelo de Cordone - Wolfler de 3 índices (c).

Por todo ello, se identifica el modelo (e) como el más adecuado para resolver problemas CVRPTW heterogéneos, debido a que es el que mejores tiempos de ejecución presenta. Cabe mencionar que los modelos (a) y (c), pese a obtener peores tiempos de ejecución, no presentan unos tiempos excesivamente

Datos empleados		Modelo de Sopnamayee Acharya		Modelo de Cordone Wolfler		Modelo de Manolis N.Kritikos - George Ioannou	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
TW7_heterog	7	121,0424177	0,437	121,0424177	0,375	121,0424177	0,358
TW10_heterog	10	149,8025479	0,422	149,8025479	0,468	149,8025479	0,296
TW20_heterog	20	198,1840802	0,577	198,1840802	0,749	198,1840802	0,421
TW30_heterog	30	227,7611982	1,669	227,7611982	1,732	227,7611982	1,107
TW40_heterog	40	266,0174832	2,121	266,0174832	3,229	266,0174832	1,482
TW50_heterog	50	357,32437	3,074	357,32437	5,32	357,32437	2,605
TW60_heterog	60	393,3581563	6,458	393,3581563	9,859	393,3581563	3,464
TW70_heterog	70	418,0655749	6,599	418,0655749	12,308	418,0655749	6,131
TW80_heterog	80	515,4135785	13,541	515,4135785	17,986	515,4135785	7,441
TW90_heterog	90	541,4241926	15,834	541,4241926	25,927	541,4241926	10,405
TW100_heterog	100	588,6276245	16,426	588,6276245	48,61	588,6276245	11,825

Tabla 6.7: Resultados obtenidos con CPLEX para el CVRPTW heterogéneo.

elevados, solo que si los comparamos con el modelo (e), sí que se aprecia que son claramente superiores.

En la Tabla 6.8 se puede observar el resumen de resultados obtenidos para el CVRPTW heterogéneo empleando CPLEX³⁹.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Sopnamayee Acharya	11 de 11	0 de 11	11 de 11	Descartado
Cordone et al. 3 ind.	11 de 11	0 de 11	11 de 11	Descartado
M.Kritikos et al.	11 de 11	11 de 11	11 de 11	SELECCIONADO

Tabla 6.8: Resumen de resultados obtenidos con CPLEX para el problema CVRPTW heterogéneo.

6.7. Comparativa Xpress-IVE vs CPLEX para el CVRPTW.

6.7.1. Caso homogéneo.

Para los tres modelos planteados tanto en Xpress-IVE como en CPLEX (modelo de Acharya (a), (b) Cordone – Wolflier de 2 índices y (c) Cordone – Wolflier de 3 índices), comparamos los resultados obtenidos:

³⁹ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

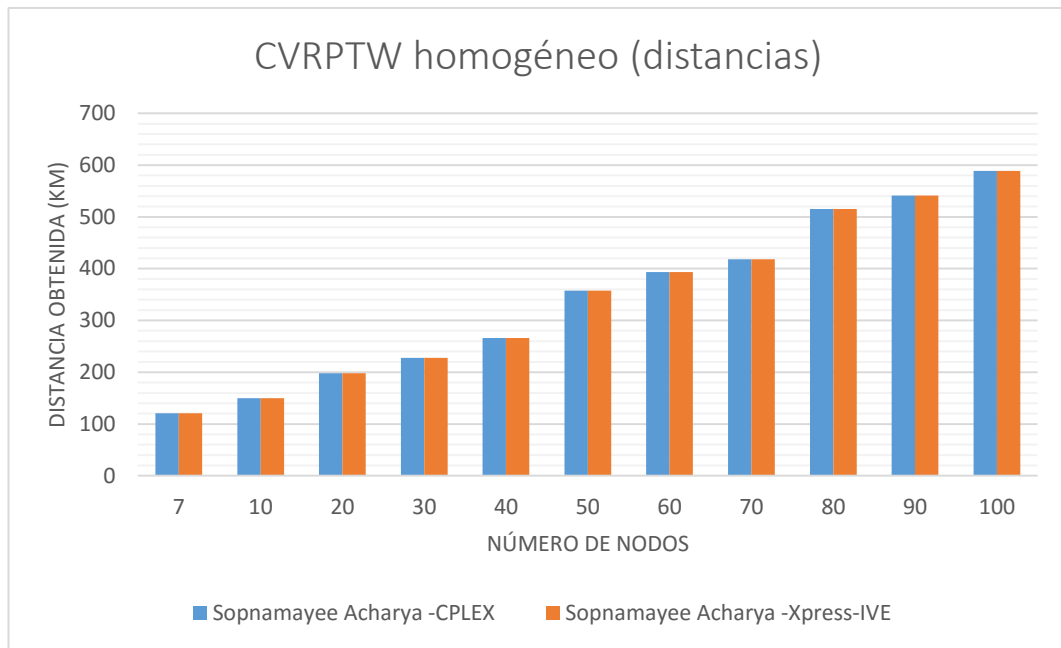


Figura 6.10: Comparativa de distancias modelo de Acharya.

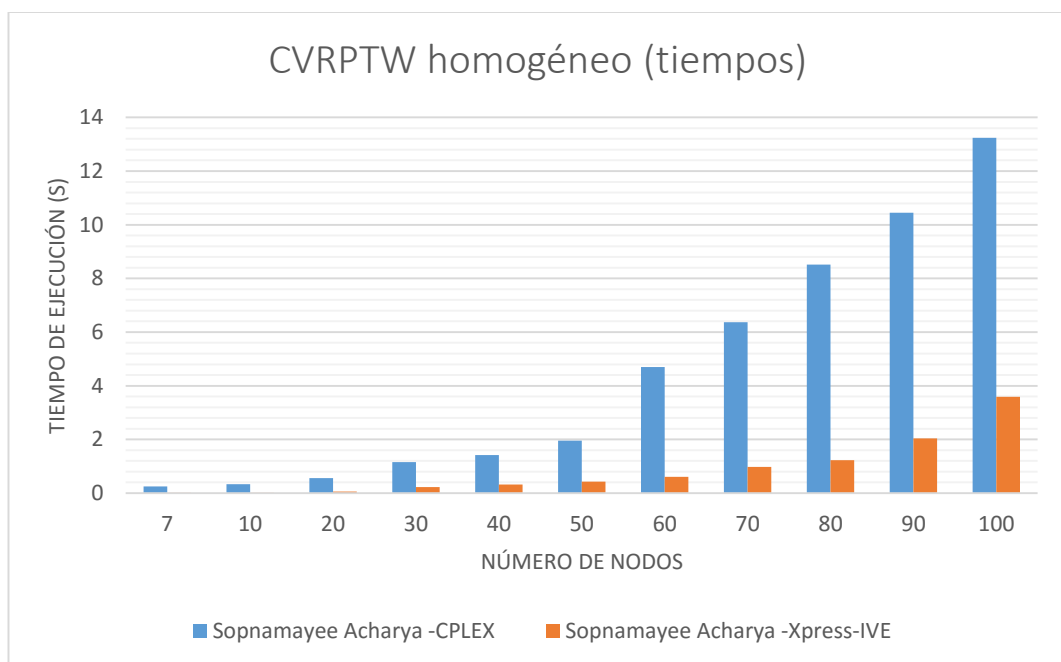


Figura 6.11: Comparativa de tiempos modelo de Acharya.

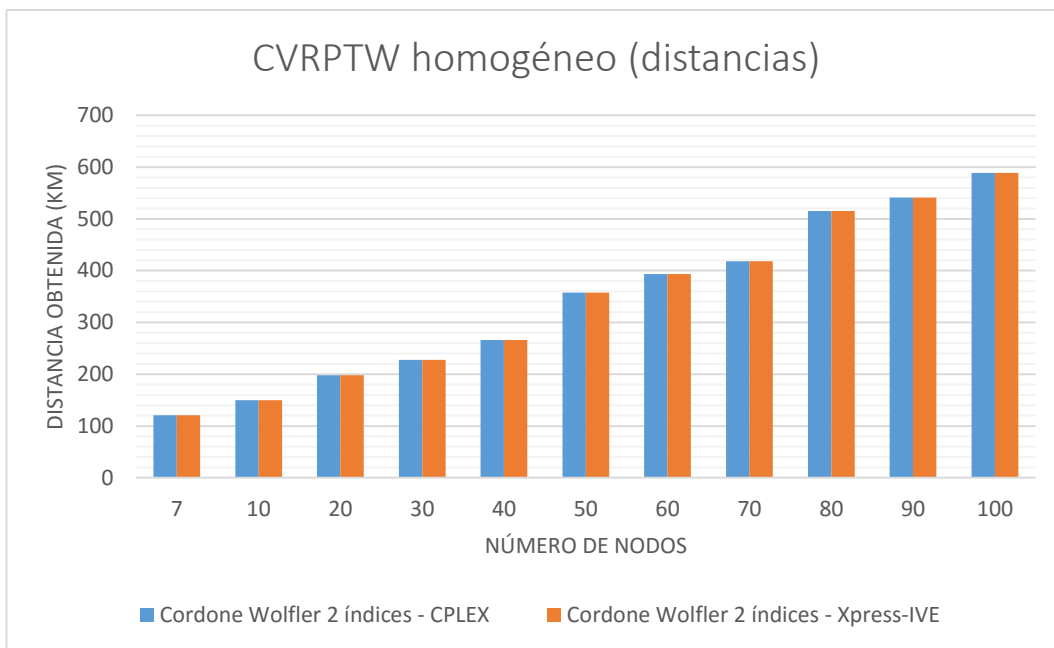


Figura 6.12: Comparativa de distancias modelo de Cordone – Wolfler de 2 índices.

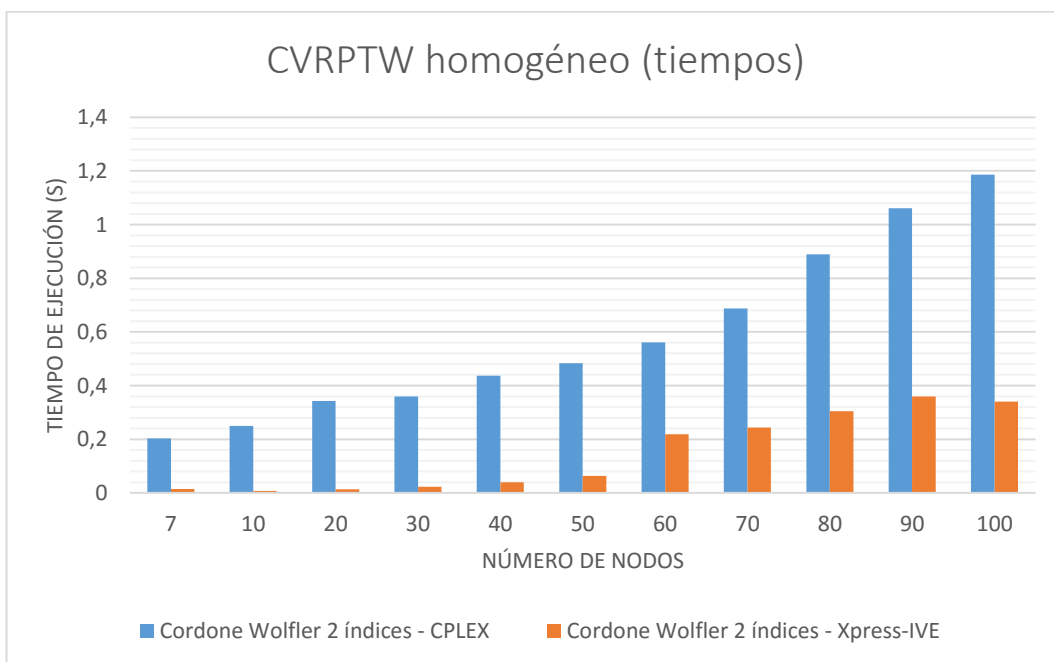


Figura 6.13: Comparativa de tiempos modelo de Cordone – Wolfler de 2 índices.

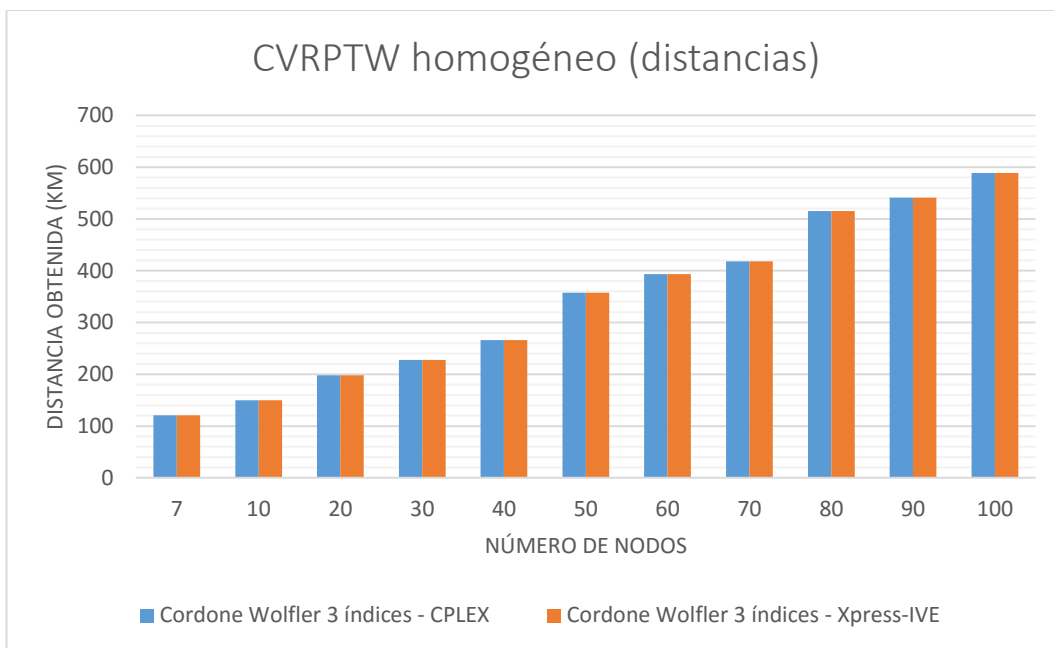


Figura 6.14: Comparativa de distancias modelo de Cordone – Wolfler de 3 índices.

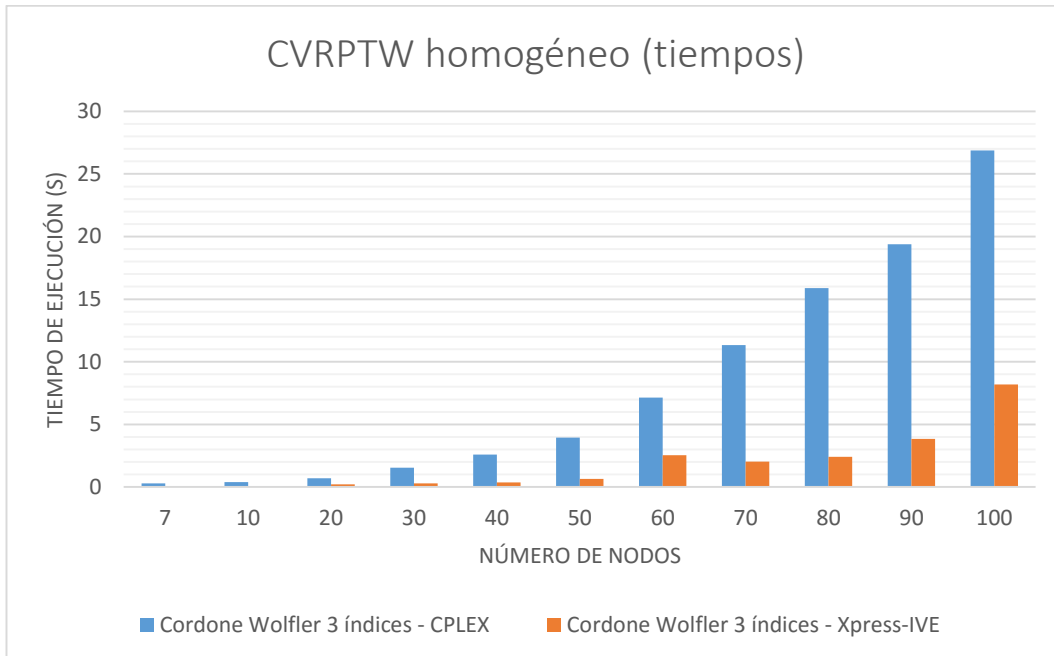


Figura 6.15: Comparativa de tiempos modelo de Cordone – Wolfler de 3 índices.

6.7.2. Resumen de resultados obtenidos. Caso homogéneo.

En la línea de todos los resultados que vamos obteniendo a lo largo de este capítulo, podemos observar en las figuras anteriores cómo tanto Xpress-IVE como CPLEX logran resolver los 33 problemas planteados, obteniendo exactamente los mismos resultados desde el punto de vista de distancias en todos ellos. Por tanto, estos dos criterios no son válidos para realizar la comparación entre ambos solvers.

En el caso de los tiempos de ejecución, la cosa cambia, y vemos cómo Xpress-IVE arroja unos resultados mejores en prácticamente todos los problemas (si bien los tiempos de ejecución de CPLEX no son para nada malos (25 segundos para TW100), sí lo son comparados con los de Xpress-IVE). No hay pie a una mayor discusión, los resultados son a nuestro parecer suficientemente claros.

Por tanto, podemos identificar a Xpress-IVE como el solver más adecuado para resolver problemas CVRPTW homogéneos.

En la Tabla 6.9 se puede observar el resumen de resultados obtenidos para el CVRP homogéneo comparando Xpress-IVE con CPLEX⁴⁰.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	33 de 33	33 de 33	33 de 33	SELECCIONADO
CPLEX	33 de 33	0 de 33	33 de 33	Descartado

Tabla 6.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPTW homogéneo.

6.7.3. Caso heterogéneo.

Comparando los resultados obtenidos en los tres modelos planteados tanto en Xpress-IVE como en CPLEX (modelo de Acharya (a), modelo de Cordone – Wolfier de 3 índices (b) y modelo de M.N.Kritikos et al. (e)), obtenemos:

⁴⁰ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

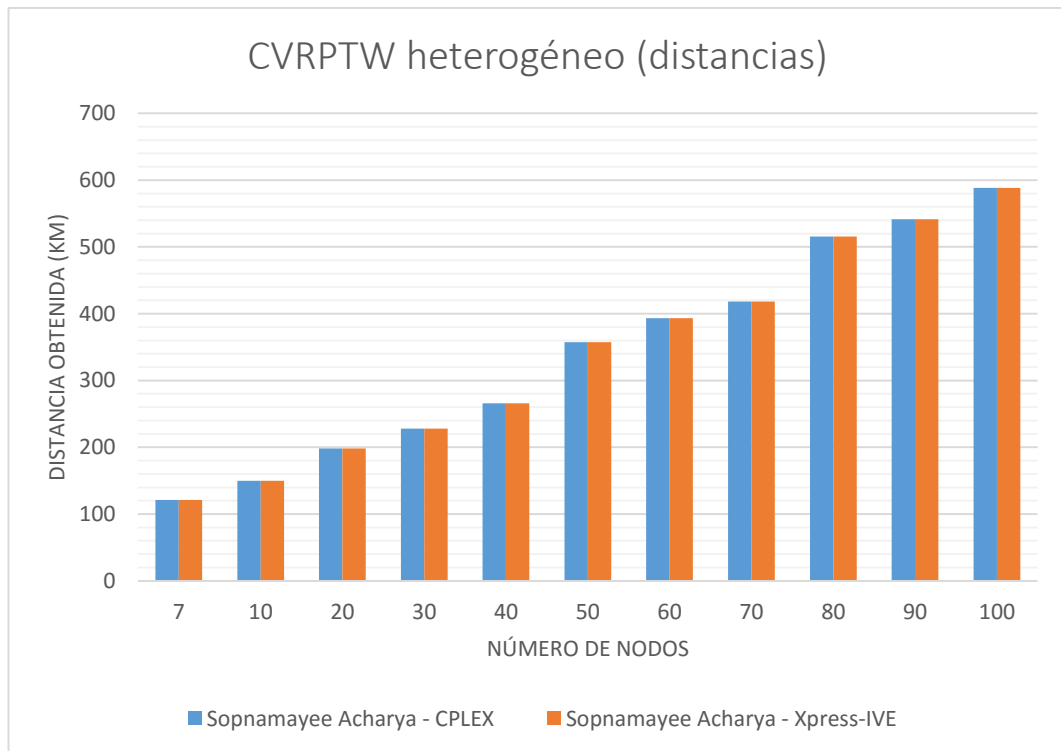


Figura 6.16: Comparativa de distancias modelo de Acharya.

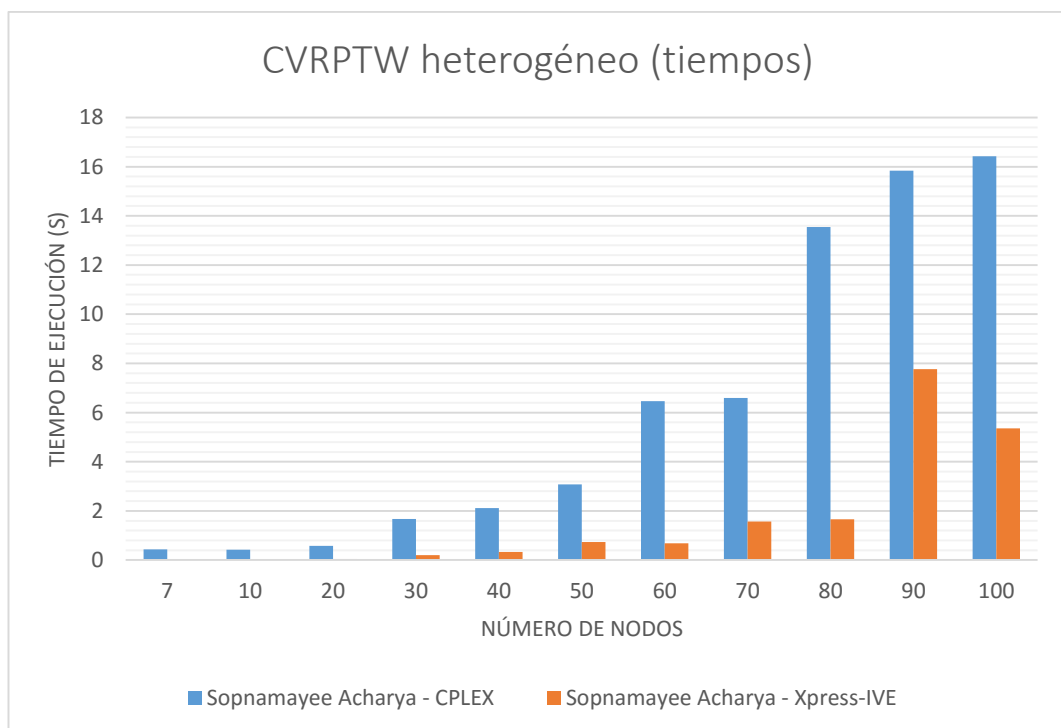


Figura 6.17: Comparativa de tiempos modelo de Acharya.

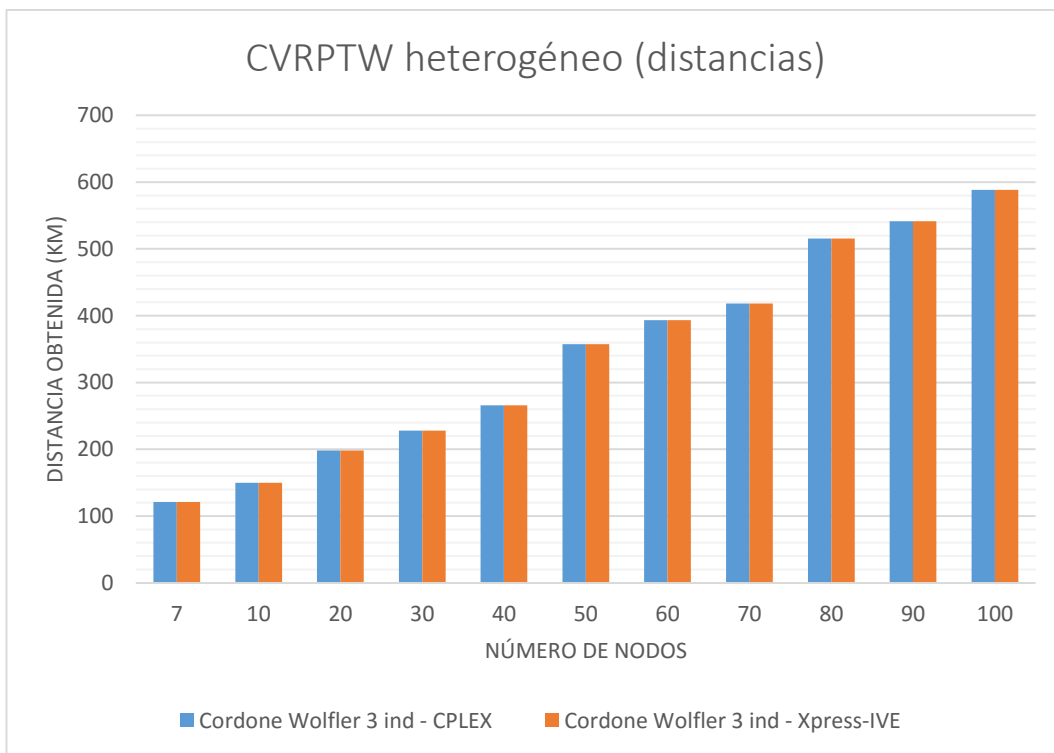


Figura 6.18: Comparativa de distancias modelo de Cordone-Wolfler de 3 índices.

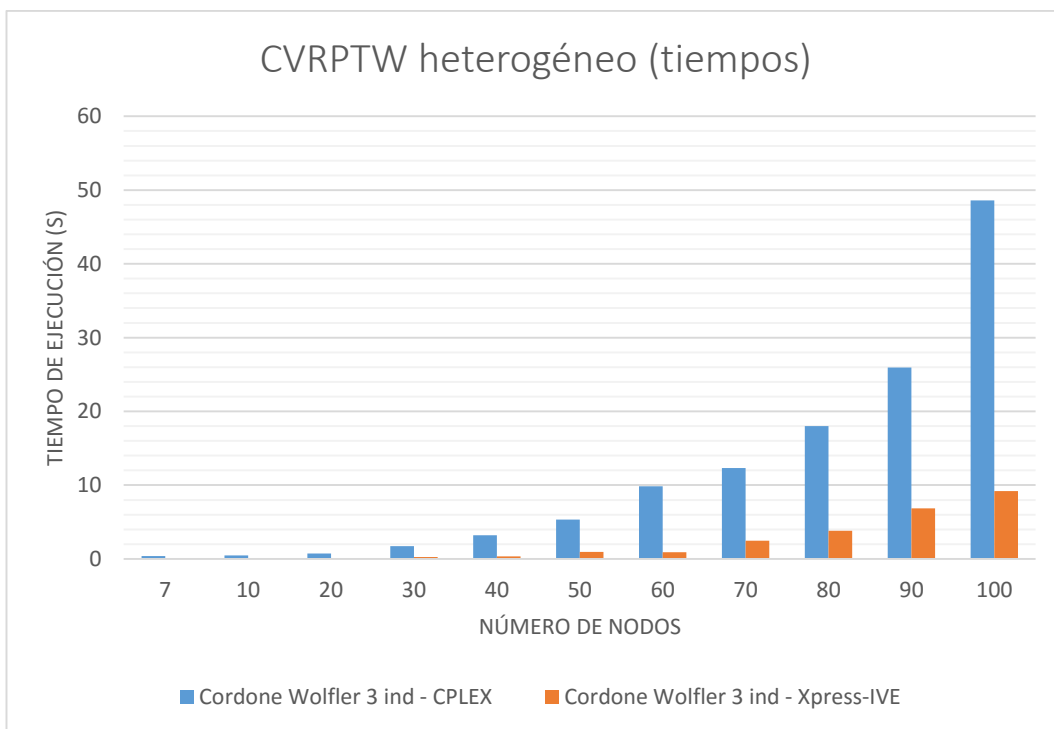


Figura 6.19: Comparativa de tiempos modelo Cordone-Wolfler de 3 índices.

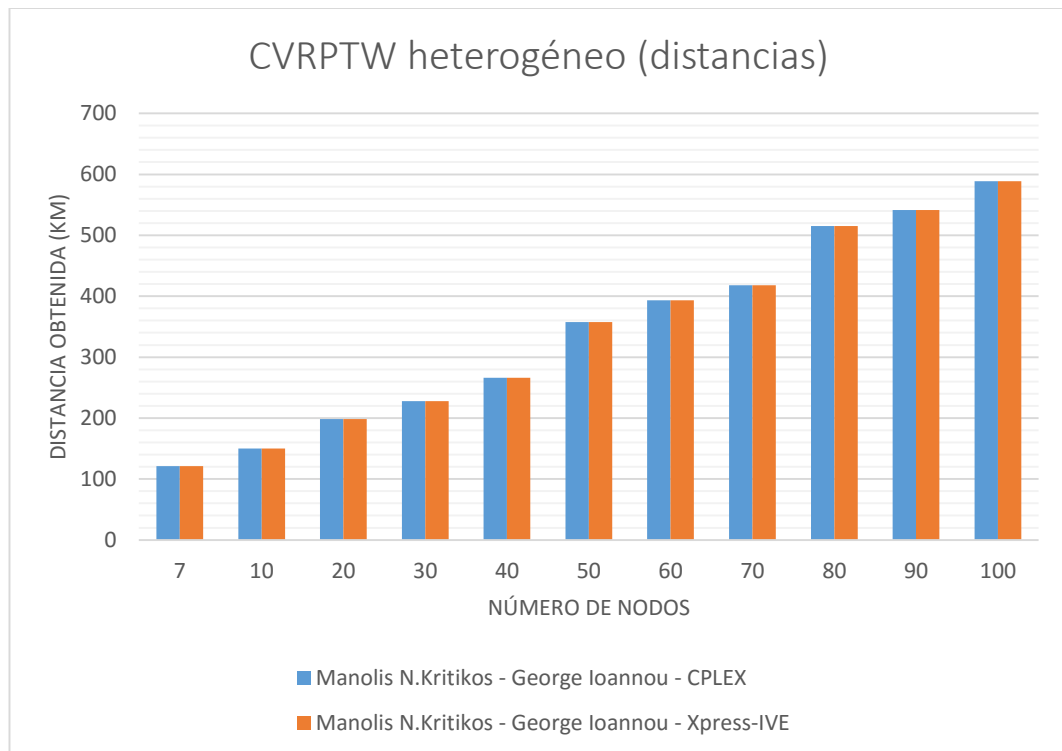


Figura 6.20: Comparativa de distancias modelo de Manolis N.Kritikos et al.

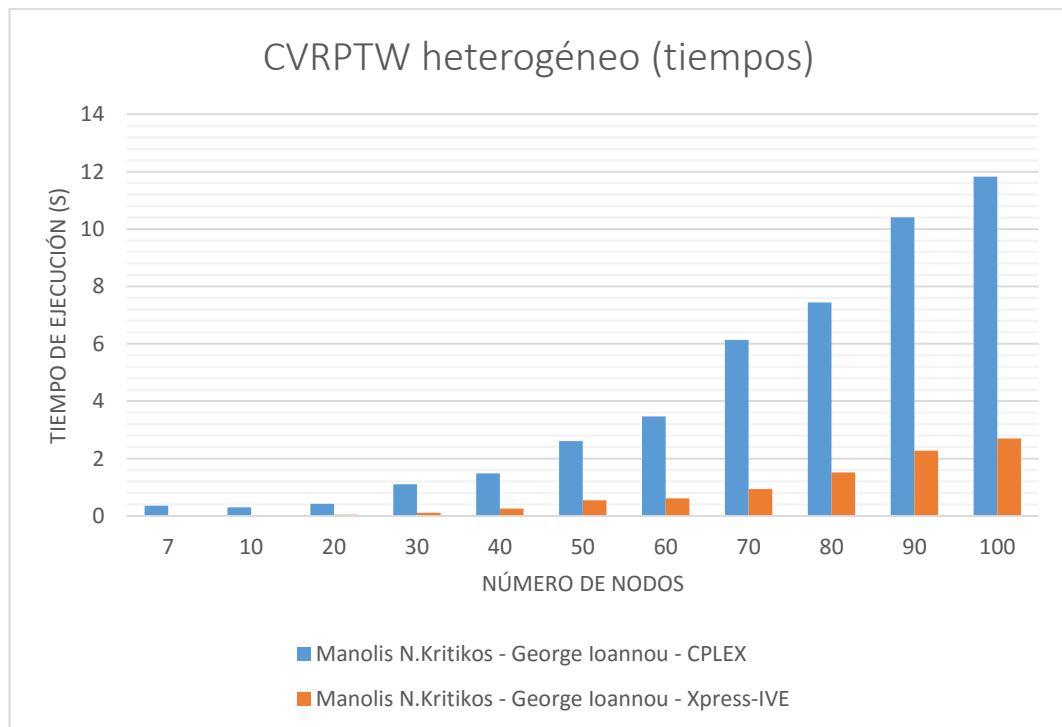


Figura 6.21: Comparativa de tiempos modelo de Manolis N.Kritikos et al.

6.7.4. Resumen de resultados obtenidos. Caso heterogéneo.

Una vez más, se observa en las figuras anteriores que los criterios de distancias obtenidas y número de problemas resueltos no son válidos para realizar la comparación entre solvers, dado que ambos han obtenido exactamente los mismos resultados para los 33 problemas propuestos, resolviendo todos ellos.

De nuevo, son los tiempos de ejecución el factor diferenciador que nos va a permitir clasificar ambos solvers. Así, y como ocurría en el apartado 6.6.2, se puede observar a simple vista que Xpress-IVE presenta unos tiempos de ejecución muy inferiores a CPLEX (aunque los tiempos de este no sean muy elevados, sí lo son comparados con los de Xpress-IVE).

Por lo tanto, y dado que los resultados obtenidos no dejan pie a dudas, se selecciona Xpress-IVE como el solver más adecuado para resolver problemas CVRPTW heterogéneos.

En la Tabla 6.10 se puede observar el resumen de resultados obtenidos para el CVRP heterogéneo comparando Xpress-IVE con CPLEX⁴¹.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	33 de 33	33 de 33	33 de 33	SELECCIONADO
CPLEX	33 de 33	0 de 33	33 de 33	Descartado

Tabla 6.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPTW heterogéneo.

6.8. Conclusiones.

Como resumen final de los resultados obtenidos en este capítulo en base a los criterios establecidos en los apartados 2.4 y 3.2.4., concluimos que:

- Los criterios de distancias obtenidas y número de problemas resueltos no han demostrado ser válidos a la hora de establecer diferencias (ni

⁴¹ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

entre modelos ni entre solvers), siendo el criterio tiempos de ejecución el que ha declinado la balanza por un modelo/solver u otro.

- Para problemas CVRPTW homogéneos, el modelo de Cordone – Wolfier de 2 índices ha demostrado ser el más adecuado, ya que de todos los modelos propuestos es el que menores tiempos de ejecución presenta.
- Para resolver problemas CVRPTW heterogéneos, el modelo de M.N.Kritikos et al. es que menores tiempos de ejecución presenta, por lo que se selecciona como el más efectivo.
- En la comparativa de solvers, se ha seleccionado Xpress-IVE como el que mejores resultados presenta desde el punto de vista de tiempos de ejecución, tanto para el caso homogéneo como para el heterogéneo.

Si bien cuestionarse el por qué un modelo obtiene mejores resultados que otro no tendría mucho sentido debido a que la respuesta reside en la formulación matemática empleada, sí lo tiene en el caso de los solvers. Para responder a esta pregunta, rogamos leer el apartado 3.2.5, en donde se ha tratado este tema con anterioridad.

CAPÍTULO 7

PROBLEMAS DE RUTAS CON RECOGIDAS (PICK UP) Y ENTREGAS (DELIVERY) (CVRPPD).

7.1. Descripción del problema.

Como ya vimos en el apartado 2.3, el CVRPPD (Capacitated Vehicle Routing Problem with Pick up and Delivery) se define como una variante del problema CVRP, en donde una flota de vehículos debe satisfacer un conjunto de pedidos, cada uno de los cuales queda definido por un nodo de pick up (recogida), un nodo de delivery (entrega) y una demanda que debe ser transportada entre estas dos localizaciones (del nodo pick up al nodo delivery) (Toth & Vigo, "The Vehicle Routing Problem", 2002).

Además, según lo visto en ese apartado, y como se trata de un problema de pick up y delivery, estaremos transportando materiales o productos, ya que si transportásemos personas el problema sería catalogado como Dial a Ride. Por último, dentro de la clasificación vista en el citado apartado, se catalogan los modelos manejados como CVRPPD del tipo one-to-one.⁴²

Un ejemplo gráfico de este tipo de problemas puede apreciarse en la Figura 7.1, donde observamos la salida generada por medio del interfaz gráfico de Xpress-IVE para el problema PD9_homog (1 nodo origen, 1 nodo destino, 8 nodos de pick up y 8 nodos de delivery) empleando el modelo de Toth y Vigo (apartado 7.2 (e)). Junto al indicador de cada nodo podemos observar en verde

⁴² (Cordeau, Laporte, & Ropke, 2007) No puede entregarse mercancía recogida en un nodo pick up en ningún nodo delivery distinto del correspondiente a ese nodo pick up.

la demanda de los nodos de pick up (positiva) y en azul la demanda de los nodos de delivery (negativa).

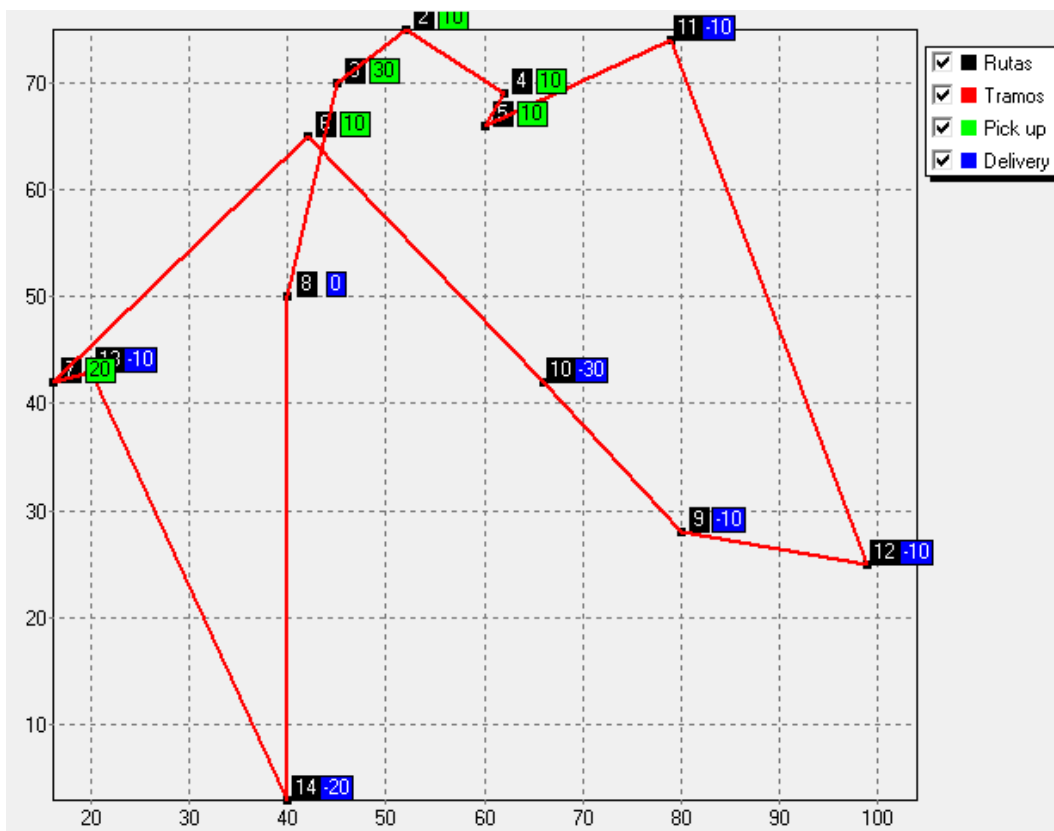


Figura 7.1: ejemplo gráfico del problema CVRPPD.

Como se trata de un caso one-to-one, del nodo origen partirán los vehículos descargados, cargándose/descargándose a medida que vayan recorriendo los distintos nodos de pick up y delivery, llegando descargados al nodo destino.⁴³ Cabe denotar, por tanto, que al contrario de lo que venía ocurriendo hasta ahora, se ha definido un nodo destino, ya que esto facilita el planteamiento y de no hacerlo los resultados pueden no ser fiables. Dicho nodo destino coincide en localización con el nodo origen, simplemente se trata de un nodo del que no parten arcos, equivalente al nodo origen, al que ahora no llegará ningún arco.

Para concluir con este apartado, sólo nos queda indicar los objetivos que hemos perseguido en el presente capítulo, que coinciden con los expresados en capítulos anteriores (ver apartado 1.2):

⁴³ Dependiendo del modelo, nos encontraremos con uno o varios nodos origen y uno o varios nodos destino, debido a que algunos modelos necesitan un planteamiento especial.

- Objetivo 1: establecer, en base a criterios específicos (ver apartado 2.4) cuál de todos los modelos propuestos es el más adecuado para resolver problemas CVRPPD de flota homogénea y flota heterogénea.
- Objetivo 2: intentar reducir el número de vehículos empleado, siempre que eso suponga minimizar la distancia total recorrida (minimizando por tanto los costes asociados al transporte). Para ello ha sido necesario modificar algunas restricciones, que han sido indicadas en el pie de página correspondiente

La elección de este objetivo queda justificada según los resultados obtenidos en el Anexo para el problema CVRPPD.

- Objetivo 3: identificar el solver más adecuado (Xpress-IVE o CPLEX) para resolver problemas CVRPPD de flota homogénea y flota heterogénea, en base a criterios ya establecidos (ver apartado 3.2.4).

7.2. Modelos planteados. Caso homogéneo.

a) Modelo de Jan Dethloff.⁴⁴

El primero de los modelos planteados lo encontramos en el artículo (Dethloff, 2001), formulado como sigue:

Variables:

- x_{ijk} : variable binaria que nos va a indicar si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga del vehículo k tras haber atendido al nodo i . En el artículo original aparece únicamente como L_i , pero hemos considerado que añadiendo el subíndice k resultaba más intuitiva.
- Π_i : variable entera para eliminación de subtours. Puede interpretarse como la posición del nodo i en la ruta.

Parámetros:

⁴⁴ La restricción (7) que plantea no ha sido formulada ya que emplea variables que pueden ser omitidas del modelo.

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- d_i : cantidad de entrega en el nodo i (nodo de delivery).
- p_i : cantidad de recogida en el nodo i (nodo de pick up).
- M : constante de un valor elevado para servir como cota superior para algunas restricciones.

Formulación:

$$Min \sum_{i=1}^{2n} \sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (2_1)$$

(2_1) De todo nodo distinto de los nodos origen/destino debe salir un solo arco.

$$\sum_{\substack{i=1 \\ i \neq \text{destino}}}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq \text{origen}; j \neq \text{destino} \quad (2_2)$$

(2_2) A todo nodo distinto de los nodos origen/destino debe llegar un solo arco.

$$\sum_{\substack{j=1 \\ j \neq \text{destino} \\ j \neq \text{origen}}}^{2n} x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (2_3)$$

(2_3) Para el nodo origen, para cada vehículo puede existir un arco saliente de dicho nodo.⁴⁵

⁴⁵ Esta restricción ha sido modificada respecto de la formulación original.

$$\sum_{i=1}^{2n} x_{ijk} - \sum_{i=1}^{2n} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k = 1..K \quad (3_1)$$

(3_1) El número de arcos que entran en un nodo (distinto de los nodos origen/destino) debe ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j=1 \\ j \neq \text{origen} \\ j \neq \text{destino}}}^{2n} x_{\text{origen } j k} - \sum_{\substack{j=1 \\ j \neq \text{origen} \\ j \neq \text{destino}}}^{2n} x_{j \text{ destino } k} = 0 \quad \forall k = 1..K \quad (3_2)$$

(3_2) El número de arcos que salen del nodo origen debe ser igual al número de arcos que llegan al nodo destino.

$$L_{\text{origen } k} = 0 \quad \forall k = 1..K \quad (4_1)$$

$$L_{\text{destino } k} = 0 \quad \forall k = 1..K \quad (4_2)$$

(4_1), (4_2) Para cualquier vehículo, la carga transportada tras haber atendido al nodo origen/destino debe ser 0.

$$L_{jk} \geq L_{ik} + d_j + p_j - (1 - x_{ijk})M \quad \forall i \neq \text{destino}; i \neq j; k = 1..K \quad (6)$$

(6) Restricción para asegurar una correcta asignación de carga a cada vehículo. Representa que, de existir el arco ij empleando el vehículo k, la carga transportada por el vehículo k al salir del nodo j debe de ser como mínimo la transportada al salir del nodo i, más la demanda del nodo j (positiva si es un nodo de pick up; negativa si es un nodo de delivery). En caso de no existir el nodo ij empleando el vehículo k, se tiene una cota superior para las cargas transportadas.

$$L_{jk} \leq \text{cap} \quad \forall j \neq \text{origen}; j \neq \text{destino}; k = 1..K \quad (8)$$

(8) Cota superior para la carga transportada por cada vehículo.

$$\pi_j \geq \pi_i + 1 - n \left(1 - \sum_{k=1}^K x_{ijk} \right)$$

$\forall i, j \neq \text{destino}; i, j \neq \text{origen}; i \neq j; k = 1..K$ (9_1)

(9_1) Restricciones impuestas para la eliminación de subtours.

$$\pi_i \leq \pi_{i+n} \quad \forall i \neq \text{origen}; i \leq n \quad (9_2)$$

(9_2) Restricción que nos indica que los nodos de pick up deben ocupar una posición anterior a su correspondiente nodo de delivery en la ruta.

$$\sum_{j=1}^{2n} x_{ijk} - \sum_{j=1}^{2n} x_{i+njk} = 0 \quad \forall i \neq \text{origen}; i \leq n; k = 1..K \quad (10)$$

(10) Restricción que nos indica que cada nodo de pick up y su correspondiente nodo de delivery deben ser atendidos por el mismo vehículo.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..2n; k = 1..K$$

$$L_{ik} \geq 0 \quad \forall i = 1..2n; k = 1..K$$

$$\pi_i \geq 0 \quad \forall i = 1..n$$

La restricción (10) no aparece formulada en el artículo, pero de no incluirla, la solución podría ser correcta desde el punto de vista de los resultados, pero violaría las hipótesis consideradas (caso one-to-one, en donde un nodo de pick up sólo puede distribuir carga a su correspondiente nodo de delivery).

Ejemplo: supongamos los nodos de pick up A y B, ambos con demanda de 20. Supongamos, además, los nodos de delivery C y D, ambos con demanda -20. El nodo A se corresponde con el nodo C y el nodo B con el D. De no incluir la restricción (10), no habría ninguna restricción que impidiera satisfacer la demanda del nodo C con las unidades del nodo B, o la del D con las del A. Sin embargo, al considerar problemas del tipo one-to-one, esta opción es completamente inviable.

b) Modelo de Quan Lu y Maged Dessouky.

En (Quan Lu & Dessouky, 2002) encontramos una de los planteamientos más interesantes con las que nos hemos topado al realizar este TFG. En este artículo, los autores definen variables para identificar si un nodo precede a otro en la ruta, de manera que se puede saber qué vehículo recorre cada nodo sin necesidad de un índice específico.

Aspectos de modelado aparte, la forma de definir las restricciones contrasta con la línea seguida hasta ahora, por lo que resulta muy llamativa. Sin más dilación, veamos dicha formulación:

Variables:

- x_{ij} : variable binaria que de valer 1 nos va a indicar que un vehículo va del nodo i al nodo j .
- b_{ij} : variable binaria que de valer 1 nos va a indicar que el nodo i va antes que el nodo j en el tour.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- g_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).

Conjuntos:

- $N_r^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (g_i positivo).
- $N_r^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (g_i negativo).
- $N_q = \{2n+1 \dots 2n+m+1\}$: siendo m el número de vehículos empleados, el nodo $2n+1$ representa el nodo de partida para el primer vehículo, mientras que los nodos $2n+v$, $v=2, 3 \dots m$ representan el nodo de retorno para el vehículo $v-1$, además del nodo de partida para el vehículo v . El nodo $2n+m+1$ representa el nodo de retorno para el vehículo m .
- $N_r = N_r^+ \cup N_r^-$.
- $N = N_r \cup N_q$.

En nuestro caso concreto, para cualquier vehículo se han diseñado los nodos de origen/destino de tal forma que coincidan, es decir, que a efectos prácticos es como si cada vehículo partiera del y retornara al mismo punto, respectivamente. Esto se ha hecho para respetar la notación del artículo, ya que en caso contrario el modelo carecería de sentido.

Formulación:

$$\text{Min} \quad \sum_{i \in N} \sum_{j \in N} x_{ij} d_{ij}$$

$$\sum_{\substack{j \in N \\ j \leq 2n}} x_{ij} \leq 1 \quad \forall i \neq 2n + m + 1; i \in N_q \quad (1)$$

$$\sum_{j \in N} x_{ij} = 1 \quad \forall i \neq 2n + m + 1; i \in N_r \quad (1')$$

$$\sum_{\substack{i \in N \\ i \leq 2n}} x_{ij} \leq 1 \quad \forall j \neq 2n + 1; j \in N_q \quad (2)$$

$$\sum_{i \in N} x_{ij} = 1 \quad \forall j \neq 2n + 1; j \in N_r \quad (2')$$

(1), (1'), (2), (2') Restricciones básicas, nos indican que de cada nodo distinto de los nodos origen/destino debe salir y entrar un solo arco, mientras que de cada nodo origen y destino puede entrar o salir un arco, respectivamente.⁴⁶

$$b_{ki} \leq b_{kj} + (1 - x_{ij}) \quad \forall k \in N/k \neq i; k \neq j; i \neq j \quad (3)$$

$$b_{kj} \leq b_{ki} + (1 - x_{ij}) \quad \forall k \in N/k \neq i; k \neq j; i \neq j \quad (4)$$

⁴⁶ Las restricciones (1) y (2) han sido modificadas respecto de la formulación original.

(3), (4) Estas restricciones nos indican que, de existir el arco ij (lo cual implica que el nodo i va justo antes que el nodo j en la ruta), y siendo el nodo $k \neq i$ se tendrá que $b_{ki} = b_{kj}$. Esto quiere decir que, si el nodo k precede al nodo i en la ruta, también precederá al nodo j , mientras que si es el nodo i el que precede al nodo k , el nodo j también precederá al nodo k , por ir el nodo j justo después del nodo i en la ruta.

$$x_{ij} \leq b_{ij} \quad \forall i, j \in N \quad (5)$$

(5) Nos indica que de existir el arco ij , el nodo i irá antes del nodo j en la ruta obligatoriamente, pero que la inexistencia del arco ij no implica que el nodo i no pueda ir antes que el nodo j (imaginemos, por ejemplo, que existe el arco ik y también el arco kj , de manera que el nodo i va antes que el nodo j , siendo el valor de $b_{ij}=1$ pese a ser $x_{ij}=0$).

$$b_{ii} = 0 \quad \forall i \in N \quad (6)$$

(6) Se impone que un nodo no puede precederse a sí mismo.

$$b_{n+i i} = 0 \quad \forall i \in N^+_r \quad (7)$$

(7) Un nodo de delivery no puede preceder a su nodo de pick up en la ruta.

$$b_{i n+i} = 1 \quad \forall i \in N^+_r \quad (8)$$

(8) Un nodo de pick up debe preceder siempre a su correspondiente nodo de delivery en la ruta.

$$b_{i 2n+j} = b_{n+i 2n+j} \quad \forall i \in N^+_r; 2n+j \in N_q; 2n+j \leq 2n+m+1 \quad (9)$$

(9) Garantiza que cada par de nodos pick up – delivery sea atendido por el mismo vehículo.

$$g_j + \sum_{i \in N} (b_{ij} \cdot g_i) \leq cap \quad \forall j \in N \quad (10)$$

(10) Restricciones de capacidad, que fuerzan a que la carga que transporta cualquier vehículo no supere la capacidad máxima de carga de dicho vehículo.

$$b_{i \ 2n+1} = 0 \quad \forall i \neq 2n + 1 \quad (11)$$

(11) El nodo $2^*n + 1$ se identifica como el nodo de partida del primer vehículo, y por definición se trata del primer nodo de la ruta, no pudiendo ser precedido por ningún otro nodo.

$$b_{2n+k \ 2n+j} = 1 \quad \forall k < j; 2n + k \in N_q; 2n + j \in N_q \quad (12)$$

$$b_{2n+j \ 2n+k} = 0 \quad \forall k < j; 2n + k \in N_q; 2n + j \in N_q \quad (13)$$

(12), (13) Restricciones que nos aseguran que los nodos depot deben seguir la secuencia $2^*n+1 \rightarrow 2^*n+2 \rightarrow \dots \rightarrow 2^*n+m+1$.

$$b_{i \ 2n+m+1} = 1 \quad \forall i \neq 2n + m + 1 \quad (14)$$

(14) Identificamos el nodo $2^*n + m + 1$ como el último nodo de la ruta, por lo que cualquier nodo debe precederle.

$$b_{ki} + b_{k \ i+n} \geq x_{ik} + x_{ki} \quad \forall i \in N^+_r; k \neq i; k \neq i + n \quad (22)$$

$$b_{ik} + b_{i+n \ k} \geq x_{i+n \ k} + x_{k \ i+n} \quad \forall i \in N^+_r; k \neq i; k \neq i + n \quad (23)$$

(22), (23) Probamos la inecuación (22):

- Si el nodo k y el nodo i no son adyacentes (no existen ni el arco ik ni el arco ki), entonces la parte derecha de la inecuación valdrá 0. VÁLIDO.
- Si el nodo i va inmediatamente antes del nodo k en la ruta, entonces el nodo $i + n$ deberá ir después del nodo k , por lo que $b_{ki}=0$ y $b_{k \ i+n}=1$. VÁLIDO.
- Si el nodo i va inmediatamente después del nodo k en la ruta, entonces el nodo $i + n$ deberá ir después del nodo k también, con lo que $b_{ki}=1$ y $b_{k \ i+n}=1$. VÁLIDO.

Se puede realizar el mismo razonamiento para probar la inecuación (23).

$$\sum_{k \in N} b_{ki} + 1 \leq \sum_{k \in N} b_{k i+n} \quad \forall i \in N \quad (24_1)$$

$$\sum_{k \in N} b_{ik} + 1 \geq \sum_{k \in N} b_{i+n k} + 1 \quad \forall i \in N \quad (24_2)$$

(24_1), (24_2) Prueba: para todo $i \in N$, definimos $S_i = \sum_{k=1}^{2n+m+1} b_{ki}$. Por definición de las variables b , S_i representa la secuencia del nodo i en la ruta (ejemplo: $S_{2n+1}=0$ y $S_{2n+m+1}=2n+m \rightarrow$ En consecuencia $S_{i+1} \leq S_{i+n}$).

$$\sum_{k \in N} (b_{ki} \cdot g_k) = 0 \quad \forall i \in N_q \quad (25)$$

(25) La carga transportada por cualquier vehículo, al partir de o llegar a cualquier nodo depot ($\in N_q$) debe ser nula. Esto se debe a que, por definición del problema, los vehículos partirán descargados de los nodos depot, y a lo largo de la ruta recogerán carga en los nodos pick up y la entregarán en su correspondiente nodo delivery, no existiendo ninguna cantidad de carga sobrante.

$$b_{ij} + b_{ji} = 1 \quad \forall ij/i \neq j \quad (26)$$

(26) Restricción que impone que cualquier nodo debe ser asignado a una ruta (si el nodo i no va antes que el nodo j en la ruta implica que el nodo j irá antes que el nodo i , no pudiendo existir ningún caso en que algún nodo no preceda o sea precedido por ningún otro nodo).

$$x_{ij} \in \{0,1\} \quad \forall i,j/ i \neq n+j; i \neq 2n+m+1; j \neq 2n+1; i \neq j$$

Es decir, hemos excluid de los posibles x_{ij} aquellos arcos tales que sean $(n+i, i)$, $(\dots, 2n+1)$ o $(2n+m+1, \dots)$ por ser inviables.

$$b_{ij} \geq 0 \quad \forall ij \in N$$

c) Modelo basado en redes multiproducto.

Partiendo de la formulación planteada en (Sáez Aguado, "El problema de flujo multiproducto de coste mínimo en redes: descripción y propiedades", 2016) hemos extendido el problema para adaptarlo al caso de disponer de múltiples vehículos para realizar las rutas.

Además, hemos implementado una programación especial, de manera que cada par de nodos pick up – delivery maneja un tipo de producto único, en consonancia con el planteamiento one-to-one.

Por último, se han añadido unas restricciones adicionales para eliminación de subtours, consistentes en variables de tiempo para cada nodo.

La formulación final es la siguiente:

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ijkp} : variable entera para indicar la cantidad del producto p transportada del nodo i al nodo j con el vehículo k .
- D_{ik} : variables de tiempo extras creadas para eliminación de subtours.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- d_{ip} : cantidad demandada en el nodo i del producto p . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
Como ya hemos indicado anteriormente, la idea es que cada nodo pick up tenga oferta para un único producto, que coincida con el producto demandado por su correspondiente nodo delivery (es por ello que el número de productos es n).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).

- M: constante de valor elevado empleada para penalizar aquellas d_{ip} que no nos sean útiles debido a que no hayamos asignado al nodo i el producto p .

Formulación:

$$\text{Min} \sum_{i=1}^{2n} \sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{j=1 \\ j \neq \text{destino}}}^{2n} x_{\text{origen } j k} - \sum_{\substack{j=1 \\ j \neq \text{origen}}}^{2n} x_{j \text{ destino } k} = 0 \quad \forall k = 1..K \quad (0_1)$$

(0_1) El número de arcos que salen del nodo origen debe ser igual al número de nodos que llegan al nodo destino.

$$\sum_{\substack{j=1 \\ j \neq \text{origen} \\ j > n}}^{2n} \sum_{k=1}^K x_{j \text{ destino } k} \leq K \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (0_2)$$

(0_2) El número de nodos que llegan al nodo destino debe ser a lo sumo igual al número de vehículos empleado.⁴⁷

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{j \text{ origen } k} = 0 \quad (0_3)$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{\text{destino } j k} = 0 \quad (0_4)$$

(0_3), (0_4) No pueden existir arcos salientes del nodo destino, ni entrantes al nodo origen.

⁴⁷ Restricción para intentar reducir el número de vehículos empleado.

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (2)$$

(2) Para cualquier nodo distinto de los nodos origen/destino debe existir un solo arco saliente de dicho nodo.

$$\sum_{\substack{j=1 \\ j \neq i}}^{2n} x_{ijk} - \sum_{\substack{j=1 \\ j \neq i}}^{2n} x_{jn+ik} = 0 \\ \forall i \leq n; i \neq \text{origen}; k = 1..K \quad (3)$$

(3) Restricción adicional que hemos formulado para garantizar que cada par pick up – delivery sea atendido por el mismo vehículo.

$$\sum_{\substack{j=1 \\ j \leq n \\ j \neq \text{origen}}}^{2n} x_{\text{origen} j k} \leq 1 \quad \forall k = 1..K \quad (4)$$

(4) Para cualquier vehículo, del nodo origen debe salir a lo sumo un arco.⁴⁸

$$\sum_{\substack{i=1 \\ i \neq j}}^{2n} x_{ijk} - \sum_{\substack{i=1 \\ i \neq j}}^{2n} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k = 1..K \quad (5)$$

(5) Para cualquier nodo distinto de los nodos origen/destino, el número de arcos que entran en ese nodo debe ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j=1 \\ j > n \\ j \neq \text{destino}}}^{2n} x_{j \text{ destino} k} \leq 1 \quad \forall k = 1..K \quad (6)$$

(6) Para todo vehículo, al nodo destino debe llegar a lo sumo un arco.⁴⁹

⁴⁸ Restricción para intentar reducir el número de vehículos empleado.

⁴⁹ Restricción para intentar reducir el número de vehículos empleado.

$$\sum_{j=1}^{2n} \sum_{k=1}^K y_{ijkp} - \sum_{j=1}^{2n} \sum_{k=1}^K y_{jikp} = d_{ip} \quad \forall i = 1..2n; p = 1..n/d_{ip} \neq M \quad (7)$$

(7) Restricciones de flujo para eliminación de subtours, similares a las restricciones de Kirchhoff vistas para el problema TSP (apartado 4.4.2 (c)). Indican que la cantidad de un producto que llega a un nodo menos la cantidad de dicho producto que sale de ese nodo debe ser igual a la demanda que dicho nodo tenga de ese producto. Sólo se tienen en cuenta aquellos nodos con demandas existentes para cada producto.

$$\sum_{p=1}^n y_{ijkp} \leq x_{ijk} \cdot cap \quad \forall i, j = 1..2n; k = 1..K \quad (8)$$

(8) De existir el arco ij empleando el vehículo k, la suma de las cantidades de cada producto que transporte ese vehículo deberá ser menor que su capacidad máxima. Esta restricción garantiza la relación entre las variables binarias y las de flujo.

$$\sum_{j=1}^{2n} \sum_{k=1}^K y_{jikp} = \sum_{j=1}^{2n} \sum_{k=1}^K y_{ijkp} \quad \forall i = 1..2n; p = 1..n/d_{ip} = M \quad (9)$$

(9) Una extensión de la restricción 7, que asegura que en los casos en que no exista demanda de un producto en un nodo, la cantidad de ese producto que entre en dicho nodo deberá ser igual a la cantidad que salga.

$$D_{origen k} = 0 \quad \forall k = 1..K \quad (10)$$

(10) El tiempo de comienzo de servicio en el nodo origen será 0 para cualquier vehículo j.

$$D_{ik} + t_{ij} - D_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq destino; i \neq j; k = 1..K \quad (11)$$

(11) Restricción de tiempo para eliminación de subtours. Aseguran que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en

el nodo j deberá ser como mínimo el tiempo de comienzo de servicio en el nodo i , más el tiempo empleado en ir del nodo i al nodo j . En caso de no existir el arco ij empleando el vehículo k , se tiene una cota superior para los tiempos de comienzo de servicio.

$$D_{ik} + t_{i \rightarrow j} \leq D_{jk} \quad \forall i \neq \text{origen}; i \leq n; k = 1..K \quad (12)$$

(12) Restricción que nos garantiza que cada nodo de pick up vaya antes que su correspondiente nodo de delivery en la ruta.

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..2n; k = 1..K$$

$$D_{ik} \geq 0 \quad \forall i = 1..2n; k = 1..K$$

$$y_{ijkp} \geq 0 \quad \forall i, j = 1..2n; k \in K; p = 1..n$$

d) Modelo de M.W.P.Savelsbergh y M.Sol.

Modelo que aparece formulado en (Savelsbergh & M.Sol, 1995) como sigue:

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- z_{ik} : variable binaria que de valer 1 nos va a indicar que el nodo i va a ser atendido por el vehículo k .
- D_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en cada uno de los nodos.
- y_i : variable entera que especifica la carga del vehículo correspondiente cuando sale del nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.

- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- T : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

Este es uno de los dos tipos de problemas para los que se ha optado por una distribución especial, quedando esta como sigue:

- $N^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $N^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $N = N^+ \cup N^-$.
- $M = \{1 \dots K\}$, con K el número máximo de vehículos empleado.
- $M^+ = \{2n+1 \dots 2n+K\}$: el nodo $2n + i$, con $i \leq K$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+K+1 \dots 2n+2K\}$: el nodo $2n + i$, con $K < i \leq 2K$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.

Formulación:

$$\text{Min} \quad \sum_{i \in N \cup W} \sum_{j \in N \cup W} \sum_{k \in M} x_{ijk} d_{ij}$$

$$\sum_{k=1}^K z_{ik} = 1 \quad \forall i \in N \quad (1)$$

(1) Cada nodo debe ser atendido por un solo vehículo.

$$z_{ik} = z_{i+n k} \quad \forall i \in N^+; k \in M \quad (1_1)$$

(1_1) Hemos añadido esta restricción para garantizar que cada par de nodos pick up – delivery sea atendido por el mismo vehículo.

$$\sum_{\substack{j \in N \cup M_k^- \\ i \neq n+j}} x_{ijk} = z_{ik} \quad \forall i \in N; k \in M \quad (2_1)$$

$$\sum_{\substack{j \in N \cup M_k^+ \\ j \neq n+i}} x_{jik} = z_{ik} \quad \forall i \in N; k \in M \quad (2_2)$$

(2_1), (2_2) Si un nodo no es atendido por un vehículo, no podrán existir arcos entrantes o salientes a dicho nodo empleando ese vehículo. Recordemos que M_k^- representando el nodo de destino y M_k^+ el nodo de origen del vehículo k.

$$\sum_{j \in N} x_{2n+k j k} \leq 1 \quad \forall k \in M \quad (3)$$

(3) Del nodo origen, para cada vehículo, puede salir a lo sumo un arco.⁵⁰

$$\sum_{\substack{j \in N^+ \\ k \in M}} x_{2n+k j k} \geq 1 \quad (3_2)$$

(3_2) De los nodos origen debe partir como mínimo un arco.

$$\sum_{i \in N^-} x_{i 2n+k k} = \sum_{j \in N^+} x_{2n+k j k} \quad \forall k \in M \quad (4)$$

(4) El número de arcos que salen del nodo origen de un vehículo debe ser igual al número de arcos que lleguen al nodo destino de dicho vehículo.

$$D_{2n+k} = 0 \quad \forall k \in M \quad (5)$$

(5) El tiempo de comienzo de servicio para los nodos origen de cada vehículo deberá ser 0.

⁵⁰ Esta restricción ha sido modificada respecto de la formulación original.

$$D_i + t_{i i+n} \leq D_{i+n} \quad \forall i \in N^+ \quad (6)$$

(6) Restricción empleada para garantizar que cada nodo de pick up sea atendido antes que su correspondiente nodo de delivery.

$$D_i + t_{ij} - D_j \leq T(1 - x_{ijk}) \quad \forall i \in N \cup M^+; i \neq j; k \in M \quad (7)$$

(7) Restricciones de tiempo. Indican que de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j deberá ser como mínimo el tiempo de comienzo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de que no exista el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$y_{2n+k} = 0 \quad \forall k \in M \quad (8_1)$$

$$y_{2n+K+k} = 0 \quad \forall k \in M \quad (8_2)$$

(8_1), (8_2) La carga transportada por cada vehículo al salir de sus nodos origen/destino deberá ser nula.

$$y_i \leq \sum_{k \in M} (cap \cdot z_{ik}) \quad \forall i \in N \quad (9)$$

(9) De ser el nodo i atendido por el vehículo k, la carga transportada por el vehículo k al salir del nodo i deberá ser menor que la capacidad máxima de carga de dicho vehículo.

$$y_i + d_j - y_j \leq cap(1 - x_{ijk}) \quad \forall i \in N \cup M^+; i \neq j; k \in M \quad (10)$$

(10) Restricciones de flujo para eliminación de subtours. Indican que, de existir el arco ij empleando el vehículo k, la carga transportada por el vehículo al salir del nodo j deberá ser como mínimo la carga transportada al salir del nodo i, más la demanda del nodo j. En caso de no existir el arco ij empleando el vehículo k, tenemos una cota superior para la carga transportada.

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in N \cup W; k \in M$$

$$z_{ik} \in \{0,1\} \quad \forall i \in N \cup W; k \in M$$

$$D_i \geq 0 \quad \forall i \in N$$

$$y_i \geq 0 \quad \forall i \in N$$

e) Modelo de Paolo Toth y Daniele Vigo.

Modelo planteado en (Toth & Vigo, "The Vehicle Routing Problem", 2002), formulado como sigue:

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga del vehículo k cuando abandona el nodo i .
- D_{ik} : variable de tiempo que nos va a indicar el momento de comienzo de servicio en el nodo i , atendido por el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo). El nodo i , con $i \in P / d_i=0$ será el nodo origen.
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo). El nodo $n+i$, con i el nodo origen, será el nodo destino.

- $N = P \cup D$.
- $K = \{1 \dots k\}$: conjunto del número máximo de vehículos a emplear.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{j \in N} x_{origen j k} - \sum_{j \in N} x_{j destino k} = 0 \quad \forall k = 1..K \quad (0_1)$$

(0_1) El número de arcos que salen del nodo origen debe ser igual al número de arcos que llegan al nodo destino.

$$\sum_{j \in N} \sum_{k \in K} x_{j origen k} = 0 \quad (0_4)$$

$$\sum_{j \in N} \sum_{k \in K} x_{destino j k} = 0 \quad (0_5)$$

(0_4), (0_5) Ni al nodo origen pueden llegar arcos, ni salir del nodo destino.

$$\sum_{k \in K} \sum_{\substack{j \in N \\ i \neq j \\ j \neq origen \\ j \neq destino}} x_{ijk} = 1 \quad \forall i \in P; i \neq origen \quad (2)$$

(2) Para todo nodo distinto del nodo origen debe existir un arco saliente de dicho nodo.

$$\sum_{\substack{j \in N \\ i \neq j \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{ijk} - \sum_{\substack{j \in N \\ i \neq j \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{jn+ik} = 0 \quad \forall i \in P; i \neq \text{origen}, \text{destino}; k \in K \quad (3)$$

(3) Cada par de nodos pick up – delivery debe ser atendido por el mismo vehículo.

$$\sum_{\substack{j \in P \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{\text{origen}jk} \leq 1 \quad \forall k \in K \quad (4)$$

(4) Del nodo origen, para cada vehículo, deberá salir a lo sumo un arco.⁵¹

$$\sum_{\substack{i \in N \\ i \neq j}} x_{ijk} - \sum_{\substack{i \in N \\ i \neq j}} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k \in K \quad (5)$$

(5) El número de arcos que entran en un nodo distinto de los nodos origen y destino deberá ser igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{j \in D \\ j \neq \text{destino}}} x_{j\text{destino}k} \leq 1 \quad \forall k \in K \quad (6)$$

(6) Para el nodo destino, para cada vehículo, deberá llegar a lo sumo un arco.⁵²

$$L_{ik} + d_j - L_{jk} \leq \text{cap}(1 - x_{ijk}) \\ \forall i, j \in N; i \neq j; i \neq \text{destino}; j \neq \text{origen}; k \in K \quad (10_1)$$

$$L_{jk} - d_j - L_{ik} \leq \text{cap}(1 - x_{ijk}) \\ \forall i, j \in N; i \neq j; i \neq \text{destino}; j \neq \text{origen}; k \in K \quad (10_2)$$

⁵¹ Restricción modificada respecto de la formulación original.

⁵² Restricción modificada respecto de la formulación original.

(10_1), (10_2) Restricciones de flujo para eliminación de subtours.

$$d_i \leq L_{ik} \leq cap \quad \forall i \in N; k \in K \quad (11_1 \text{ y } 11_2)$$

$$0 \leq L_{ik} \quad \forall i \in N; k \in K \quad (12_1)$$

$$L_{ik} \leq cap - d_{i-n} \quad \forall i \in D; i \neq destino; k \in K \quad (12_2)$$

(11_1), (11_2), (12_1), (12_2) Cotas superiores e inferiores para la carga transportada por un vehículo al salir de un nodo.

$$L_{origen\ k} = 0; L_{destino\ k} = 0 \quad \forall k \in K \quad (13_1 \text{ y } 13_2)$$

(13_1), (13_2) La carga transportada por cada vehículo al salir de los nodos origen/destino debe ser nula.

$$D_{origen\ k} = 0 \quad \forall k \in K \quad (14)$$

(14) El nodo origen debe ser atendido en el instante 0.

$$D_{ik} + t_{ij} - D_{jk} \leq M(1 - x_{ijk}) \\ \forall i, j \in N; i \neq destino; i \neq j; k \in K \quad (15)$$

(15) Restricciones de tiempo para eliminación de subtours.

$$D_{ik} + t_{i\ i+n} \leq D_{i+n\ k} \quad \forall i \in P; i \neq origen; k \in K \quad (16)$$

(16) Para asegurarnos de que cada nodo pick up sea atendido antes que su correspondiente nodo delivery.

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in N; k \in K$$

$$L_{ik} \geq 0 \quad \forall i \in N; k \in K$$

$$D_{ik} \geq 0 \quad \forall i \in N; k \in K$$

7.3. Modelos planteados. Caso heterogéneo.

En este caso, todas las formulaciones del caso heterogéneo son extensiones de las del caso homogéneo, adaptando algunas restricciones para las distintas capacidades de cada vehículo.

a) Modelo de Jan Dethloff.

VARIABLES:

- x_{ijk} : variable binaria que nos va a indicar si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga del vehículo k tras haber atendido al nodo j . En el artículo original aparece como L_i , pero hemos considerado que así resultaba más intuitiva.
- Π_i : variable entera para eliminación de subtours. Puede interpretarse como la posición del nodo i en la ruta.

PARÁMETROS:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : cantidad de entrega en el nodo i (nodo de delivery).
- p_i : cantidad de recogida en el nodo i (nodo de pick-up).
- M : constante de un valor elevado para servir como cota superior para algunas restricciones.

FORMULACIÓN:

$$\text{Min} \quad \sum_{i=1}^{2n} \sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{j=1 \\ j \neq \text{origen}}}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (2_1)$$

$$\sum_{\substack{i=1 \\ i \neq destino}}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall j \neq origen; j \neq destino \quad (2_2)$$

$$\sum_{\substack{j=1 \\ j \neq destino \\ j \neq origen}}^{2n} x_{origen j k} \leq 1 \quad \forall k = 1..K \quad (2_3)^{53}$$

$$\sum_{i=1}^{2n} x_{ijk} - \sum_{i=1}^{2n} x_{jik} = 0 \quad \forall j \neq origen; j \neq destino; k = 1..K \quad (3_1)$$

$$\sum_{\substack{j=1 \\ j \neq origen \\ j \neq destino}}^{2n} x_{origen j k} - \sum_{\substack{j=1 \\ j \neq origen \\ j \neq destino}}^{2n} x_{j destino k} = 0 \quad \forall k = 1..K \quad (3_2)$$

$$L_{origen k} = 0 \quad \forall k = 1..K \quad (4_1)$$

$$L_{destino k} = 0 \quad \forall k = 1..K \quad (4_2)$$

$$L_{jk} \geq L_{ik} + d_j + p_j - (1 - x_{ijk})M \quad \forall i \neq destino; i \neq j; k = 1..K \quad (6)$$

$$L_{jk} \leq cap_k \quad \forall j \neq origen; j \neq destino; k = 1..K \quad (8)$$

$$\pi_j \geq \pi_i + 1 - n \left(1 - \sum_{k=1}^K x_{ijk} \right)$$

$$\forall i \neq destino; i \neq origen; j \neq destino; j \neq origen; i \neq j; k = 1..K \quad (9_1)$$

$$\pi_i \leq \pi_{i+n} \quad \forall i \neq origen; i \leq n \quad (9_2)$$

⁵³ Restricción modificada respecto a la formulación original.

$$\sum_{j=1}^{2n} x_{ijk} - \sum_{j=1}^{2n} x_{i+njk} = 0 \quad \forall i \neq \text{origen}; i \leq n; k = 1..K \quad (10)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..2n; k = 1..K$$

$$L_{ik} \geq 0 \quad \forall i = 1..2n; k = 1..K$$

$$\pi_i \geq 0 \quad \forall i = 1..n$$

b) Modelo de Quan Lu y Maged Dessouky.

Variables:

- x_{ij} : variable binaria que de valer 1 nos va a indicar que un vehículo va del nodo i al nodo j .
- b_{ij} : variable binaria que de valer 1 nos va a indicar que el nodo i va antes que el nodo j en el tour.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- g_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- f_i : sean cap_k , con $k=1..m$ las capacidades de cada vehículo, para representar las restricciones de capacidad en el caso heterogéneo empleamos este parámetro, definido como:
 - cap_i $i=2n+1$
 - $cap_{i-2n} - cap_{i-2n-1}$ $i=2n+2...2n+m$
 - 0 para cualquier otro caso

Conjuntos:

Este es uno de los dos tipos de problemas para los que se ha formulado una distribución especial, quedando esta como sigue:

- $N_r^+ = \{1..n\}$: representa a todos los nodos de pick up (g_i positivo).
- $N_r^- = \{n+1..2n\}$: representa a todos los nodos de delivery (g_i negativo).
- $N_q = \{2n+1..2n+m+1\}$: siendo m el número de vehículos empleados. El nodo $2n+1$ representa el nodo de partida para el primer vehículo. Los nodos $2n+v$, $v=2, 3, \dots, m$ representa el nodo de retorno para el vehículo

v-1, además del nodo de partida para el vehículo v. El nodo $2n+m+1$ representa el nodo de retorno para el vehículo m.

- $N_r = N_r^+ \cup N_r^-$.
- $N = N_r \cup N_q$.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} x_{ij} d_{ij}$$

$$\sum_{\substack{j \in N \\ j \leq 2n}} x_{ij} \leq 1 \quad \forall i \neq 2n + m + 1; i \in N_q \quad (1)^{54}$$

$$\sum_{j \in N} x_{ij} = 1 \quad \forall i \neq 2n + m + 1; i \in N_r \quad (1')$$

$$\sum_{\substack{i \in N \\ i \leq 2n}} x_{ij} \leq 1 \quad \forall j \neq 2n + 1; j \in N_q \quad (2)^{55}$$

$$\sum_{i \in N} x_{ij} = 1 \quad \forall j \neq 2n + 1; j \in N_r \quad (2)$$

$$b_{ki} \leq b_{kj} + (1 - x_{ij}) \quad \forall k \in N/k \neq i; k \neq j; i \neq j \quad (3)$$

$$b_{kj} \leq b_{ki} + (1 - x_{ij}) \quad \forall k \in N/k \neq i; k \neq j; i \neq j \quad (4)$$

$$x_{ij} \leq b_{ij} \quad \forall i, j \in N \quad (5)$$

$$b_{ii} = 0 \quad \forall i \in N \quad (6)$$

$$b_{n+i} = 0 \quad \forall i \in N_r^+ \quad (7)$$

⁵⁴ Restricción modificada respecto de la formulación original.

⁵⁵ Restricción modificada respecto de la formulación original.

$$b_{i\ n+i} = 1 \quad \forall i \in N^+_r \quad (8)$$

$$b_{i\ 2n+j} = b_{n+i\ 2n+j} \quad \forall i \in N^+_r; 2n+j \in N_q; 2n+j \leq 2n+m+1 \quad (9)$$

$$g_j + \sum_{i \in N} (b_{ij} \cdot g_i) \leq \sum_{k=1}^K (f_{2n+k} \cdot b_{2n+k\ j}) \quad \forall j \in N \quad (10)$$

(10) Restricciones de capacidad para el caso heterogéneo.

$$b_{i\ 2n+1} = 0 \quad \forall i \neq 2n+1 \quad (11)$$

$$b_{2n+k\ 2n+j} = 1 \quad \forall k < j; 2n+k \in N_q; 2n+j \in N_q \quad (12)$$

$$b_{2n+j\ 2n+k} = 0 \quad \forall k < j; 2n+k \in N_q; 2n+j \in N_q \quad (13)$$

$$b_{i\ 2n+m+1} = 1 \quad \forall i \neq 2n+m+1 \quad (14)$$

$$b_{ki} + b_{k\ i+n} \geq x_{ik} + x_{ki} \quad \forall i \in N^+_r; k \neq i; k \neq i+n \quad (22)$$

$$b_{ik} + b_{i+n\ k} \geq x_{i+n\ k} + x_{k\ i+n} \quad \forall i \in N^+_r; k \neq i; k \neq i+n \quad (23)$$

$$\sum_{k \in N} b_{ki} + 1 \leq \sum_{k \in N} b_{k\ i+n} \quad \forall i \in N \quad (24_1)$$

$$\sum_{k \in N} b_{ik} + 1 \geq \sum_{k \in N} b_{i+n\ k} + 1 \quad \forall i \in N \quad (24_2)$$

$$\sum_{k \in N} (b_{ki} \cdot g_k) = 0 \quad \forall i \in N_q \quad (25)$$

$$b_{ij} + b_{ji} = 1 \quad \forall ij/i \neq j \quad (26)$$

$$x_{ij} \in \{0,1\} \quad \forall i, j/i \neq n+j; i \neq 2n+m+1; j \neq 2n+1; i \neq j$$

$$b_{ij} \geq 0 \quad \forall ij \in N$$

c) Modelo basado en redes multiproducto

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ijkp} : variable entera para indicar la cantidad del producto p transportada del nodo i al nodo j con el vehículo k .
- D_{ik} : variables de tiempo creadas para eliminar los subtours generados por las restricciones para intentar reducir el número de vehículos utilizados.

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_{ip} : cantidad demandada en el nodo i del producto p . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).

La idea es que cada nodo de pick up tenga oferta para un único producto, que coincida con el producto demandado por su nodo correspondiente de delivery (es por ello que el número de productos es n).

- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- M : constante de valor elevado empleada para quitarnos de en medio aquellas d_{ip} que no nos sean útiles debido a que no hayamos asignado al nodo i el producto p .

Formulación:

$$Min \sum_{i=1}^{2n} \sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} d_{ij}$$

$$\sum_{\substack{j=1 \\ j \neq destino}}^{2n} x_{origen j k} - \sum_{\substack{j=1 \\ j \neq origen}}^{2n} x_{j destino k} = 0 \quad \forall k = 1..K \quad (0_1)$$

$$\sum_{\substack{j=1 \\ j \neq \text{origen} \\ j > n}}^{2n} \sum_{k=1}^K x_{j \text{ destino } k} \leq K \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (0_2)^{56}$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{j \text{ origen } k} = 0 \quad (0_3)$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{\text{destino } j k} = 0 \quad (0_4)$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K x_{ijk} = 1 \quad \forall i \neq \text{origen}; i \neq \text{destino} \quad (2)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{2n} x_{ijk} - \sum_{\substack{j=1 \\ j \neq i}}^{2n} x_{jn+ik} = 0 \quad \forall i \leq n; i \neq \text{origen}; k = 1..K \quad (3)$$

$$\sum_{\substack{j=1 \\ j \leq n \\ j \neq \text{origen}}}^{2n} x_{\text{origen } j k} \leq 1 \quad \forall k = 1..K \quad (4)^{57}$$

$$\sum_{\substack{i=1 \\ i \neq j}}^{2n} x_{ijk} - \sum_{\substack{i=1 \\ i \neq j}}^{2n} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k = 1..K \quad (5)$$

$$\sum_{\substack{j=1 \\ j > n \\ j \neq \text{destino}}}^{2n} x_{j \text{ destino } k} \leq 1 \quad \forall k = 1..K \quad (6)^{58}$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K y_{ijkp} - \sum_{j=1}^{2n} \sum_{k=1}^K y_{jikp} = d_{ip} \quad \forall i = 1..2n; p = 1..n/d_{ip} \neq M \quad (7)$$

⁵⁶ Restricción que nos permite intentar reducir el número de vehículos empleado.

⁵⁷ Restricción que nos permite intentar minimizar el número de vehículos empleado.

⁵⁸ Restricción que nos permite intentar minimizar el número de vehículos empleado.

$$\sum_{p=1}^n y_{ijkp} \leq x_{ijk} \cdot cap_k \quad \forall i, j = 1..2n; k = 1..K \quad (8)$$

$$\sum_{j=1}^{2n} \sum_{k=1}^K y_{jikp} = \sum_{j=1}^{2n} \sum_{k=1}^K y_{ijkp} \quad \forall i = 1..2n; p = 1..n/d_{ip} = M \quad (9)$$

$$D_{origen k} = 0 \quad \forall k = 1..K \quad (10)$$

$$D_{ik} + t_{ij} - D_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq destino; i \neq j; k = 1..K \quad (11)$$

$$D_{ik} + t_{i i+n} \leq D_{i+n k} \quad \forall i \neq origen; i \leq n; k = 1..K \quad (12)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j = 1..2n; k = 1..K$$

$$D_{ik} \geq 0 \quad \forall i = 1..2n; k = 1..K$$

$$y_{ijkp} \geq 0 \quad \forall i, j = 1..2n; k \in K; p = 1..n$$

d) Modelo de M.W.P.Savelsbergh y M.Sol.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- z_{ik} : variable binaria que de valer 1 nos va a indicar que el nodo i va a ser atendido por el vehículo k .
- D_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en cada uno de los nodos.
- y_i : variable entera que especifica la carga del vehículo correspondiente cuando sale del nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).

- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente entre la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90).
- T : constante de valor elevado para servirnos de cota superior en determinadas restricciones. Tiene el mismo nombre que uno de los conjuntos definidos a continuación.

Conjuntos:

- $N^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $N^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $N = N^+ \cup N^-$.
- $M = \{1 \dots K\}$, con K el número máximo de vehículos empleado.
- $M^+ = \{2n+1 \dots 2n+K\}$: el nodo $2n + i$, con $i \leq K$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+K+1 \dots 2n+2K\}$: el nodo $2n + i$, con $K < i \leq 2K$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.

Formulación:

$$\text{Min} \sum_{i \in N \cup W} \sum_{j \in N \cup W} \sum_{k \in M} x_{ijk} d_{ij}$$

$$\sum_{k=1}^K z_{ik} = 1 \quad \forall i \in N \quad (1)$$

$$z_{ik} = z_{i+n_k} \quad \forall i \in N^+; k \in M \quad (1_1)$$

$$\sum_{\substack{j \in N \cup M_k^- \\ i \neq n+j}} x_{ijk} = z_{ik} \quad \forall i \in N; k \in M \quad (2_1)$$

$$\sum_{\substack{j \in N \cup M_k^+ \\ j \neq n+i}} x_{jik} = z_{ik} \quad \forall i \in N; k \in M \quad (2_2)$$

$$\sum_{j \in N} x_{2n+kjk} \leq 1 \quad \forall k \in M \quad (3)^{59}$$

$$\sum_{\substack{j \in N^+ \\ k \in M}} x_{2n+kjk} \geq 1 \quad (3_2)$$

$$\sum_{i \in N^-} x_{i2n+k+k} = \sum_{j \in N^+} x_{2n+kjk} \quad \forall k \in M \quad (4)$$

$$D_{2n+k} = 0 \quad \forall k \in M \quad (5)$$

$$D_i + t_{i+n} \leq D_{i+n} \quad \forall i \in N^+ \quad (6)$$

$$D_i + t_{ij} - D_j \leq T(1 - x_{ijk}) \quad \forall i \in N \cup M^+; i \neq j; k \in M \quad (7)$$

$$y_{2n+k} = 0 \quad \forall k \in M \quad (8_1)$$

$$y_{2n+k+k} = 0 \quad \forall k \in M \quad (8_2)$$

$$y_i \leq \sum_{k \in M} (cap_k \cdot z_{ik}) \quad \forall i \in N \quad (9)$$

$$y_i + d_j - y_j \leq cap_k(1 - x_{ijk}) \quad \forall i \in N \cup M^+; i \neq j; k \in M \quad (10)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in N \cup W; k \in M$$

$$z_{ik} \in \{0,1\} \quad \forall i \in N \cup W; k \in M$$

$$D_i \geq 0 \quad \forall i \in N$$

$$y_i \geq 0 \quad \forall i \in N$$

⁵⁹ Esta restricción ha sido modificada respecto de la formulación original.

e) Modelo de Paolo Toth y Daniele Vigo.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga del vehículo k cuando este sale del nodo i .
- D_{ik} : variable de tiempo que nos va a indicar el momento de comienzo de servicio en el nodo i , que es atendido por el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo). El nodo i , con $i \in P / d_i=0$ será el nodo origen.
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo). El nodo $n+i$, con i el nodo origen, será el nodo destino.
- $N= P \cup D$.
- $K=\{1\dots k\}$: conjunto del número de vehículos a emplear **teóricamente**.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{j \in N} x_{origen j k} - \sum_{j \in N} x_{j destino k} = 0 \quad \forall k = 1..K \quad (0_1)$$

$$\sum_{j \in N} \sum_{k \in K} x_{j origen k} = 0 \quad (0_4)$$

$$\sum_{j \in N} \sum_{k \in K} x_{destino j k} = 0 \quad (0_5)$$

$$\sum_{k \in K} \sum_{\substack{j \in N \\ i \neq j \\ j \neq origen \\ j \neq destino}} x_{ijk} = 1 \quad \forall i \in P; i \neq origen \quad (2)$$

$$\sum_{\substack{j \in N \\ i \neq j \\ j \neq origen \\ j \neq destino}} x_{ijk} - \sum_{\substack{j \in N \\ i \neq j \\ j \neq origen \\ j \neq destino}} x_{j n+i k} = 0 \quad \forall i \in P; i \neq origen; i \neq destino; k \in K \quad (3)$$

$$\sum_{\substack{j \in P \\ j \neq origen \\ j \neq destino}} x_{origen j k} \leq 1 \quad \forall k \in K \quad (4)^{60}$$

$$\sum_{\substack{i \in N \\ i \neq j}} x_{ijk} - \sum_{\substack{i \in N \\ i \neq j}} x_{jik} = 0 \quad \forall j \neq origen; j \neq destino; k \in K \quad (5)$$

$$\sum_{\substack{j \in D \\ j \neq destino}} x_{j destino k} \leq 1 \quad \forall k \in K \quad (6)^{61}$$

$$L_{ik} + d_j - L_{jk} \leq cap_k(1 - x_{ijk}) \quad \forall i, j \in N; i \neq j; i \neq destino; j \neq origen; k \in K \quad (10_1)$$

⁶⁰ Restricción modificada respecto de la formulación original.

⁶¹ Restricción modificada respecto de la formulación original.

$$L_{jk} - d_j - L_{ik} \leq cap_k(1 - x_{ijk})$$

$$\forall i, j \in N; i \neq j; i \neq destino; j \neq origen; k \in K \quad (10_2)$$

$$d_i \leq L_{ik} \leq cap_k \quad \forall i \in N; k \in K \quad (11_1 \text{ y } 11_2)$$

$$0 \leq L_{ik} \quad \forall i \in N; k \in K \quad (12_1)$$

$$L_{ik} \leq cap_k - d_{i-n} \quad \forall i \in D; i \neq destino; k \in K \quad (12_2)$$

$$L_{origen\ k} = 0; L_{destino\ k} = 0 \quad \forall k \in K \quad (13_1 \text{ y } 13_2)$$

$$D_{origen\ k} = 0 \quad \forall k \in K \quad (14)$$

$$D_{ik} + t_{ij} - D_{jk} \leq M(1 - x_{ijk})$$

$$\forall i, j \in N; i \neq destino; i \neq j; k \in K \quad (15)$$

$$D_{ik} + t_{i\ i+n} \leq D_{i+n\ k} \quad \forall i \in P; i \neq origen; k \in K \quad (16)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in N; k \in K$$

$$L_{ik} \geq 0 \quad \forall i \in N; k \in K$$

$$D_{ik} \geq 0 \quad \forall i \in N; k \in K$$

7.4. Datos empleados.

Al contrario que ocurría en los problemas anteriores, para este caso no hemos sido capaces de encontrar una librería de la que extraer ficheros de datos correctos. En su lugar, se ha optado por adaptar los datos del problema CVRPTW (ejemplo C201 de (Librería CVRPTW homogéneo, s.f.)), eliminando las columnas correspondientes a las ventanas de tiempo y quedándonos sólo con las de coordenadas y demandas. Dichas demandas serán modificadas en un

pretratamiento dentro de la programación para hacer que las correspondientes a los nodos delivery sean iguales a las de su nodo pick up cambiadas de signo.

Para el caso heterogéneo, se han empleado los mismos datos, solo que se han asignado las capacidades de cada uno de los vehículos a mano.

Asimismo, se tiene que de los 5 modelos planteados tanto para el caso homogéneo como heterogéneo 3 de ellos presentan una estructura basada en la existencia de un nodo origen y un nodo destino, en tanto que los otros dos presentan una formulación con múltiples nodos de origen y de destino. Por ello, se han generado dos tipos de ficheros de datos:

- Para los modelos (a), (c) y (e) de los casos homogéneo y heterogéneo, con la siguiente estructura:
 - Nodos pick up: del nodo 1 al nodo n , incluidos. El nodo origen viene representado el nodo i tal que $i \leq n$ con d_i nula.
 - Nodos delivery: del nodo $n+1$ al nodo $2n$, incluidos. El nodo $n+i$, tal que el nodo i sea el nodo origen, representa el nodo destino y cada nodo $n+i \forall i \leq n / i \neq$ origen representa el nodo delivery del nodo pick up i .

La nomenclatura empleada para este tipo de problemas ha sido: PD7_homog, PD9_homog, PD10_homog, PD15_homog, PD20_homog, PD30_homog, PD40_homog, PD50_homog, PD60_homog, PD70_homog, PD80_homog, PD90_homog, PD100_homog para el caso homogéneo y PD7_heterog, PD9_heterog, PD10_heterog, PD15_heterog, PD20_heterog, PD30_heterog, PD40_heterog, PD50_heterog, PD60_heterog, PD70_heterog, PD80_heterog, PD90_heterog, PD100_heterog para el caso heterogéneo.

- Para los modelos (b) y (d):
 - Nodos pick up: del nodo 1 al nodo n , incluidos.
 - Nodos delivery: del nodo $n+1$ al nodo $2n$, incluidos. Cada nodo $n+i \forall i \leq n$ representa el nodo delivery del nodo pick up i .
 - Nodo $2n+1$: representa las coordenadas y demanda de los nodos origen/destino, que como se ha descrito han sido establecidos con la misma localización. A partir de este dato, cada uno de estos modelos realiza un pretratamiento de datos en donde se genera el conjunto de nodos origen/destino correspondiente.

Este tipo de problemas han sido denotados como sigue: PD7_QuanLu_homog, PD9_QuanLu_homog, PD10_QuanLu_homog, PD15_QuanLu_homog, PD20_QuanLu_homog, PD30_QuanLu_homog,

PD40_QuanLu_homog, PD50_QuanLu_homog, PD60_QuanLu_homog, PD70_QuanLu_homog, PD80_QuanLu_homog, PD90_QuanLu_homog, PD100_QuanLu_homog para el caso homogéneo y PD7_QuanLu_heterog, PD9_QuanLu_heterog, PD10_QuanLu_heterog, PD15_QuanLu_heterog, PD20_QuanLu_heterog, PD30_QuanLu_heterog, PD40_QuanLu_heterog, PD50_QuanLu_heterog, PD60_QuanLu_heterog, PD70_QuanLu_heterog, PD80_QuanLu_heterog, PD90_QuanLu_heterog, PD100_QuanLu_heterog para el caso heterogéneo.

7.5. Resultados obtenidos empleando Xpress-IVE.

Como ya sabemos, se ha establecido un tiempo de ejecución máximo de 200 segundos, deteniéndose la optimización cuando llegamos a ese punto y quedándonos con la mejor solución factible obtenida hasta ese momento.

7.5.1. Caso homogéneo.

Los resultados que hemos obtenidos han sido:

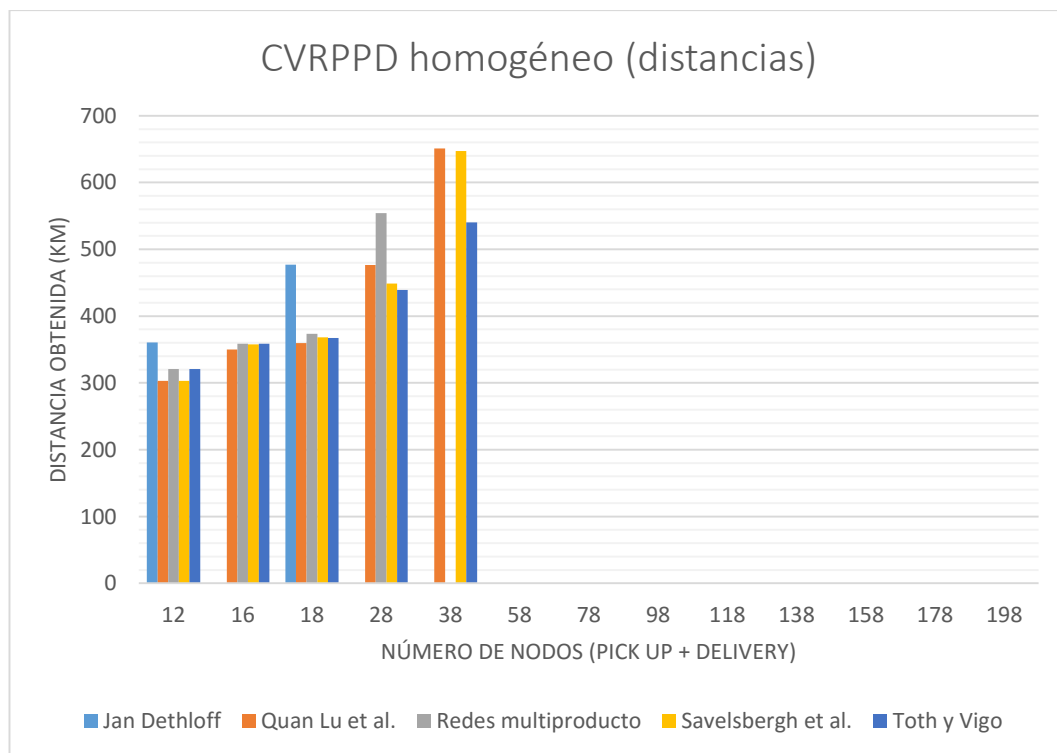


Figura 7.2: Distancias de Xpress-IVE para el problema CVRPPD homogéneo.

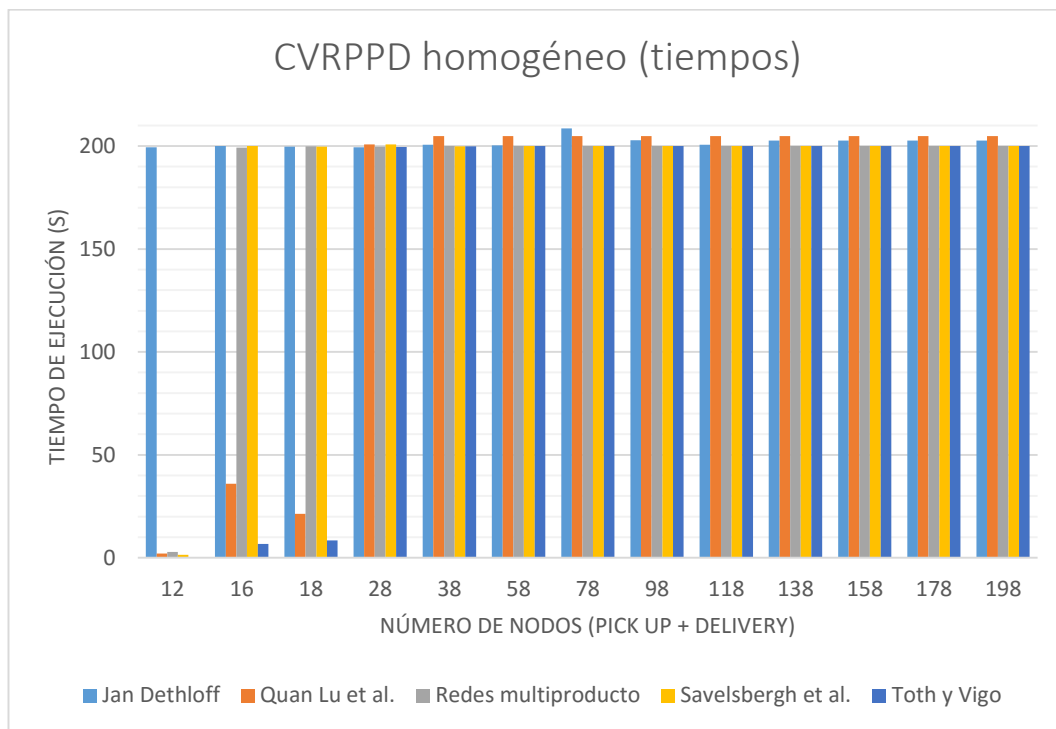


Figura 7.3: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPPD homogéneo.

7.5.2. Resumen de resultados obtenidos. Caso homogéneo.

Como se puede observar en la Figura 7.2 y en la Tabla 7.2, el número de casos resueltos para este tipo de problema de rutas desciende notoriamente respecto a los casos CVRP y CVRPTW, debido sobre todo a que lo que nosotros hemos denominado por ejemplo PD30_homog se corresponde con un problema de más de 58 nodos (29 de pick up y 29 de delivery + número de nodos origen/destino), número muy elevado que junto a la mayor complejidad de los problemas CVRPPD limita la capacidad de obtención de una solución factible por parte del solver.

Para los 5 ficheros de datos para los que algún modelo ha encontrado alguna solución factible, se puede observar en las Figuras 7.2 y 7.3 que el modelo de Jan Dethloff (a) únicamente encuentra solución factible (que no óptima) para dos de ellos, llegándose al tiempo de ejecución máximo de 200 segundos establecido. Esto, añadido a que la solución obtenida en los dos casos que sí logra resolver es con diferencia la peor cuando la comparamos con el resto de modelos, hace que descartemos este modelo para futuras evaluaciones, de

manera que lo consideremos no adecuado para resolver problemas CVRPPD homogéneos.

En el caso del modelo Redes multiproducto (c), si bien sus resultados no son tan malos como en el modelo (a), vemos que resuelve 4 de los 5 problemas propuestos, por 5 que resuelven los modelos de Quan Lu et al. (b), Savelsbergh et al. (d) y Toth y Vigo (e). Además, en 1 de esos 4 problemas obtiene la peor solución con diferencia, llegándose al tiempo máximo de ejecución en 3 de ellos. En base a estos resultados, y a que se ha observado que se trata de un modelo en el que el pretratamiento de datos por parte del solver dura mucho tiempo (tiempo que no se ve reflejado en el de ejecución), se ha desestimado este modelo como válido para la resolución de problemas CVRPPD homogéneos.

En lo referente a los modelos (b), (d) y (e), la comparativa resulta mucho más difusa, ya que se encuentran bastante parejos entre ellos en todos los aspectos: todos resuelven 5 problemas (llegando a soluciones muy próximas en 4 de ellos) y obtienen elevados tiempos de ejecución. Por lo tanto, se ha decidido implementar estos tres modelos en CPLEX, a fin de poder evaluar con mayor claridad las diferencias existentes entre ellos.

En la Tabla 7.1 se puede observar el resumen de resultados obtenidos según los criterios de comparación. Cabe mencionar que, como viene siendo habitual, los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor solución, se cuenta igualmente ya que han llegado a la mejor solución)⁶².

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Jan Dethloff	0 de 13	0 de 13	2 de 13	Descartado
QuanLu et al.	3 de 13	0 de 13	5 de 13	CPLEX
Redes multiproc	0 de 13	0 de 13	4 de 13	Descartado
Savelsbergh et a	1 de 13	0 de 13	5 de 13	CPLEX
Toth y Vigo	2 de 13	3 de 13	5 de 13	CPLEX

Tabla 7.1: Resumen de resultados para el problema CVRPPD homogéneo empleando Xpress-IVE.

⁶² Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Datos empleados		Modelo de Jan Dethloff		Modelo de Quan Lu et al.		Modelo de Redes multiproducto		Modelo de Savelsbergh et al.		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PD7_homog.txt o PD7_QuanLu_homog	12	360,459	199,423	303,213	2,061	320,859	2,805	303,213	1,487	320,859	0,39
PD9_homog.txt o PD9_QuanLu_homog	16	#N/A	200	350,253	35,901	358,582	199,275	357,574	199,964	358,582	6,731
PD10_homog.txt o PD10_QuanLu_homog	18	477,159	199,626	359,805	21,355	373,707	199,83	368,019	199,667	367,314	8,365
PD15_homog.txt o PD15_QuanLu_homog	28	#N/A	199,437	476,64	200,725	554,351	199,683	448,891	200,837	439,184	199,611
PD20_homog.txt o PD20_QuanLu_homog	38	#N/A	200,618	651,159	204,886	#N/A	200	647,074	199,916	540,076	199,794
PD30_homog.txt o PD30_QuanLu_homog	58	#N/A	200,263	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD40_homog.txt o PD40_QuanLu_homog	78	#N/A	208,518	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD50_homog.txt o PD50_QuanLu_homog	98	#N/A	202,843	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD60_homog.txt o PD60_QuanLu_homog	118	#N/A	200,579	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD70_homog.txt o PD70_QuanLu_homog	138	#N/A	202,626	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD80_homog.txt o PD80_QuanLu_homog	158	#N/A	202,627	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD90_homog.txt o PD90_QuanLu_homog	178	#N/A	202,628	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200
PD100_homog.txt o PD100_QuanLu_homog	198	#N/A	202,629	#N/A	204,886	#N/A	200	#N/A	200	#N/A	200

Tabla 7.2: Resultados de Xpress-IVE para el problema CVRPPD homogéneo.

7.5.3. Caso heterogéneo.

Para el caos heterogéneo, los resultados obtenidos han sido:

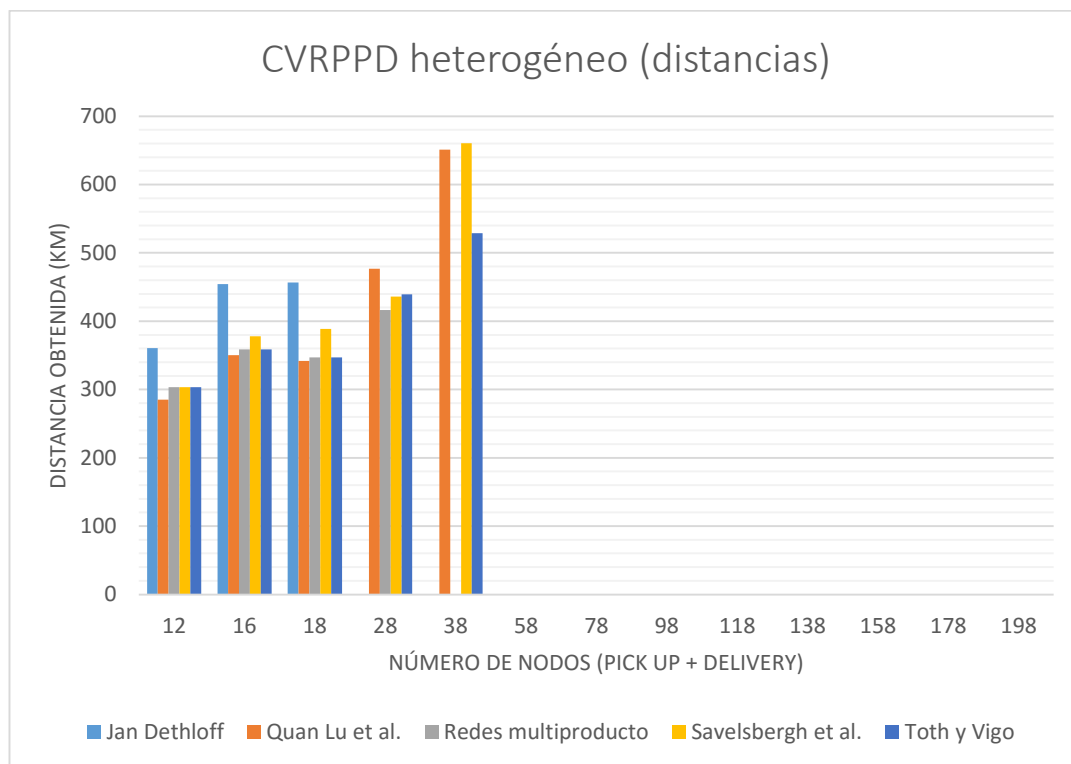


Figura 7.4: Distancias Xpress-IVE. Problema CVRPPD heterogéneo.

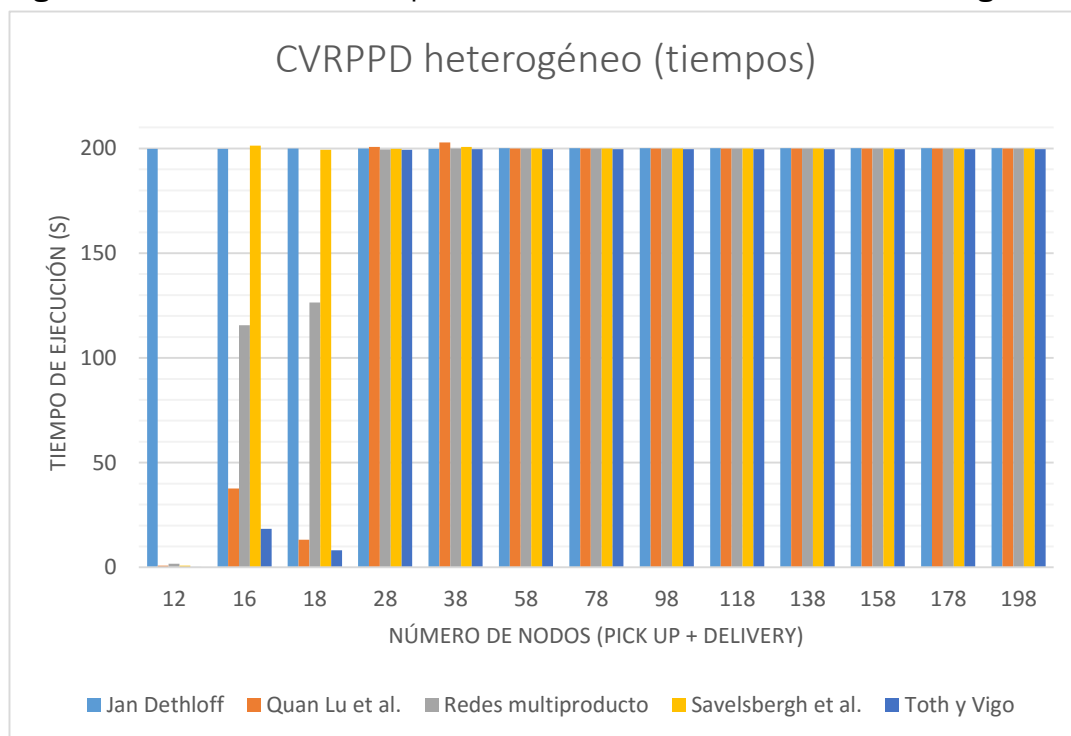


Figura 7.5: Tiempos de ejecución Xpress-IVE. CVRPPD heterogéneo.

Datos empleados		Modelo de Jan Dethloff		Modelo de Quan Lu		Modelo de Redes multiproducto		Modelo de Savelsbergh		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PD7_heterog o PD7_QuanLu_heterog	12	360,459	199,752	285,218	0,822	303,252	1,828	303,213	0,761	303,252	0,279
PD9_heterog o PD9_QuanLu_heterog	16	454,478	199,869	350,253	37,717	358,582	115,617	377,717	201,421	358,582	18,472
PD10_heterog o PD10_QuanLu_heterog	18	456,564	199,939	341,609	13,182	347,22	126,524	388,576	199,382	347,22	8,117
PD15_heterog o PD15_QuanLu_heterog	28	#N/A	199,965	476,64	200,721	416,271	199,524	435,933	199,856	439,184	199,309
PD20_heterog o PD20_QuanLu_heterog	38	#N/A	199,884	651,159	202,907	#N/A	200	660,693	200,696	528,909	199,713
PD30_heterog o PD30_QuanLu_heterog	58	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD40_heterog o PD40_QuanLu_heterog	78	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD50_heterog o PD50_QuanLu_heterog	98	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD60_heterog o PD60_QuanLu_heterog	118	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD70_heterog o PD70_QuanLu_heterog	138	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD80_heterog o PD80_QuanLu_heterog	158	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD90_heterog o PD90_QuanLu_heterog	178	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713
PD100_heterog o PD100_QuanLu_heterog	198	#N/A	200,104	#N/A	200	#N/A	200	#N/A	200	#N/A	199,713

Tabla 7.3: Resultados obtenidos con Xpress-IVE para el problema CVRPPD heterogéneo.

7.5.4. Resumen de resultados obtenidos. Caso heterogéneo.

Al igual que ocurría en el caso homogéneo, puede apreciarse que los resultados obtenidos empeoran bastante con respecto a capítulos anteriores en el sentido de número de problemas resueltos.

De nuevo nos encontramos con que el modelo de Jan Dethloff (a) presenta unos resultados bastante malos: si bien es cierto que resuelve un problema más que en el caso homogéneo, sólo logra resolver 3 de los 5 problemas para los que algún modelo ha conseguido encontrar solución. Por otra parte, en esos 3 problemas sus resultados resultan los peores con diferencia, llegándose en todos ellos al límite establecido para el tiempo de ejecución.

Sin embargo, al comparar según las Figuras 7.4 y 7.5 y la Tabla 7.3 el resto de modelos, la situación es diferente, ya que vemos que tres de ellos (Quan Lu et al. (b), Savelsbergh(d) y Toth y Vigo (e)) logran resolver los 5 problemas para los que algún modelo ha encontrado solución factible, mientras que el modelo de redes multiproducto (c) resuelve sólo 4 de ellos, pero obteniendo unos resultados bastante buenos tanto desde el punto de vista de distancias como de tiempos de ejecución.

Por todo ello, hemos llegado a dos conclusiones: primera, el modelo (a) no resulta efectivo para resolver problemas CVRPPD heterogéneos debido a que sus resultados son claramente peores a los obtenidos con el resto de modelos en base a los criterios establecidos; segunda, para el resto de modelos planteados no se ha logrado discernir si alguno de ellos no resulta adecuado para resolver este tipo de problemas, por lo que serán los cuatro formulados en CPLEX.

En la Tabla 7.4 se puede observar el resumen de resultados según los criterios de comparación. Como viene siendo habitual, los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)⁶³.

⁶³ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Jan Dethloff	0 de 13	0 de 13	3 de 13	Descartado
Quan Lu et al.	3 de 13	0 de 13	5 de 13	CPLEX
Redes multiproducto	1 de 13	0 de 13	4 de 13	CPLEX
Savelsbergh et al.	0 de 13	0 de 13	5 de 13	CPLEX
Toth y Vigo	1 de 13	3 de 13	5 de 13	CPLEX

Tabla 7.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPPD heterogéneo.

7.6. Resultados obtenidos empleando CPLEX.

Al igual que ocurría en el caso de Xpress-IVE, hemos establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización.

7.6.1. Caso homogéneo.

Como vimos en el apartado 7.5.2, se ha decidido implementar en CPLEX los modelos propuestos para el CVRPPD homogéneo (b) Quan Lu et al., (d) Savelsbergh et al. y (e) Toth y Vigo. Las Figuras 7.6 y 7.7 y en la Tabla 7.5 muestran los resultados obtenidos.

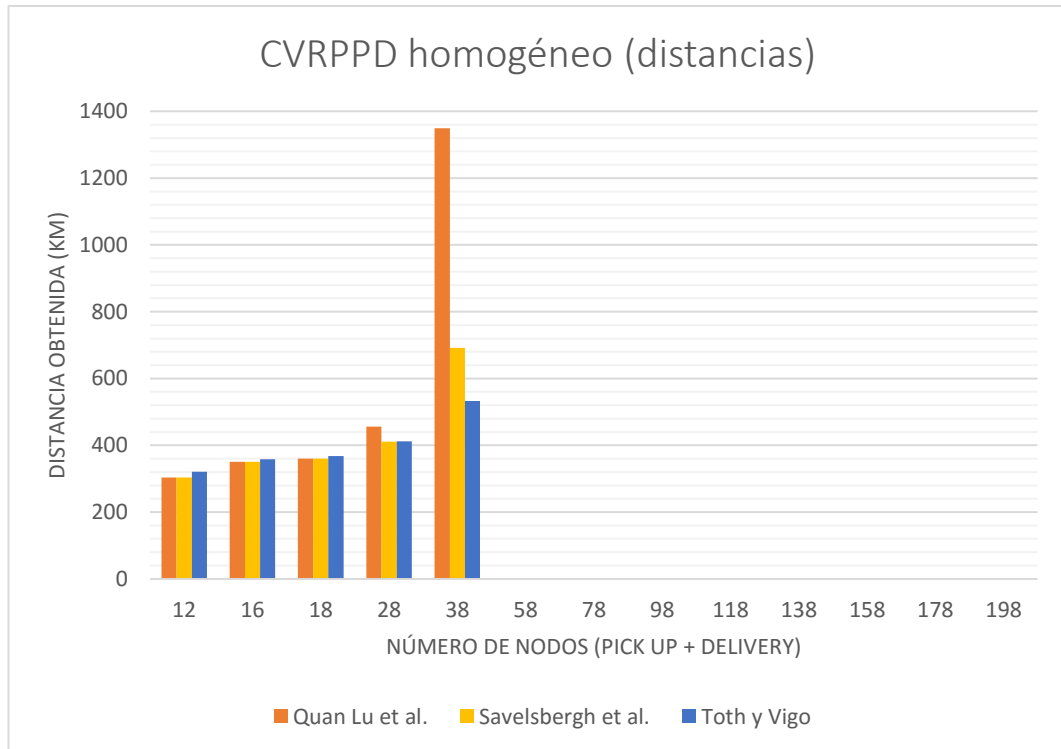


Figura 7.6: Distancias de CPLEX para el problema CVRPPD homogéneo.

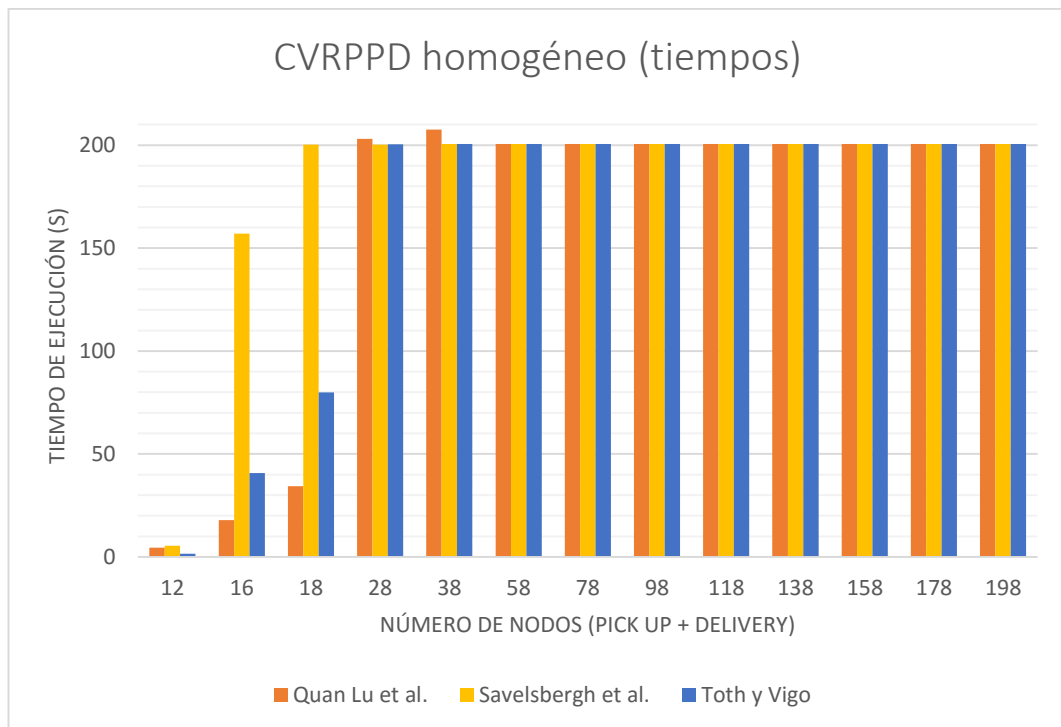


Figura 7.7: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPPD homogéneo.

Datos empleados		Modelo de Quan Lu et al.		Modelo de Savelsbergh et al.		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PD7_homog.txt o PD7_QuanLu_homog	12	303,213	4,493	303,213	5,507	320,859	1,56
PD9_homog.txt o PD9_QuanLu_homog	16	350,253	17,815	350,253	157,061	358,582	40,716
PD10_homog.txt o PD10_QuanLu_homog	18	359,805	34,305	359,805	200,332	367,314	79,959
PD15_homog.txt o PD15_QuanLu_homog	28	456,038	203,081	411,365	200,258	412,334	200,461
PD20_homog.txt o PD20_QuanLu_homog	38	1349,243	207,527	692,151	200,534	532,559	200,538
PD30_homog.txt o PD30_QuanLu_homog	58	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD40_homog.txt o PD40_QuanLu_homog	78	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD50_homog.txt o PD50_QuanLu_homog	98	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD60_homog.txt o PD60_QuanLu_homog	118	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD70_homog.txt o PD70_QuanLu_homog	138	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD80_homog.txt o PD80_QuanLu_homog	158	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD90_homog.txt o PD90_QuanLu_homog	178	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD100_homog.txt o PD100_QuanLu_homog	198	#N/A	200,538	#N/A	200,538	#N/A	200,538

Tabla 7.5: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD homogéneo.

7.6.2. Resumen de resultados. Caso homogéneo.

A la hora de comparar estos resultados, nos encontramos con una situación mucho menos intuitiva que en los casos anteriores. A simple vista, todos los modelos logran resolver el mismo número de problemas, obteniendo resultados similares en la mayoría de ellos.

Si nos fijamos un poco más en detalle, vemos cómo los modelos de Quan Lu et al. (b) y Savelsbergh et al. (d) logran calcular unas distancias idénticas hasta el problema PD20_homog_QuanLu, en donde el modelo (d) se demuestra claramente superior. Analizando los tiempos de ejecución, se observa que el modelo (b) obtiene mejores resultados hasta el problema PD15_QuanLu_homog, momento a partir del cual pasan a ser iguales para ambos.

El modelo de Toth y Vigo (e), por su parte, obtiene la peor distancia para el problema PD7_homog (con un hueco del 5,82% respecto del mejor resultado) y a partir de ahí comienza una convergencia que le lleva progresivamente a reducir ese hueco, logrando calcular el mejor resultado cuando llegamos al problema PD20_homog. Si analizamos sus tiempos de ejecución, vemos que son completamente aceptables, y que se encuentran en un punto intermedio entre el modelo (b) y el (d), si bien para el problema PD7_homog son los mejores.

En base a este razonamiento y a la estabilidad de sus resultados, que no presentan picos como ocurre con el resto de modelos a medida que incrementamos el número de nodos estudiado, se ha seleccionado el modelo (e) como el más adecuado para resolver problemas CVRPPD homogéneos. Sin embargo, si lo que se va a manejar es un fichero de datos con un número de nodos (pick up + delivery) inferior a 18, se puede emplear el modelo (b) como modelo alternativo que presenta resultados un poco mejores para estos problemas pequeños.

La Tabla 7.6 resume los resultados obtenidos en CPLEX según los criterios de comparación establecidos. Al igual que en casos anteriores, los criterios distancias obtenidas y tiempos de ejecución se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)⁶⁴.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Quan Lu et al.	3 de 13	2 de 13	5 de 13	Opción si nº nodos ≤ 18
Savelsbergh et al.	4 de 13	0 de 13	5 de 13	Descartado
Toth y Vigo	1 de 13	1 de 13	5 de 13	SELECCIONADO

Tabla 7.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD homogéneo.

⁶⁴ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

7.6.3. Caso heterogéneo.

Para el caso heterogéneo, hemos obtenido los siguientes resultados:

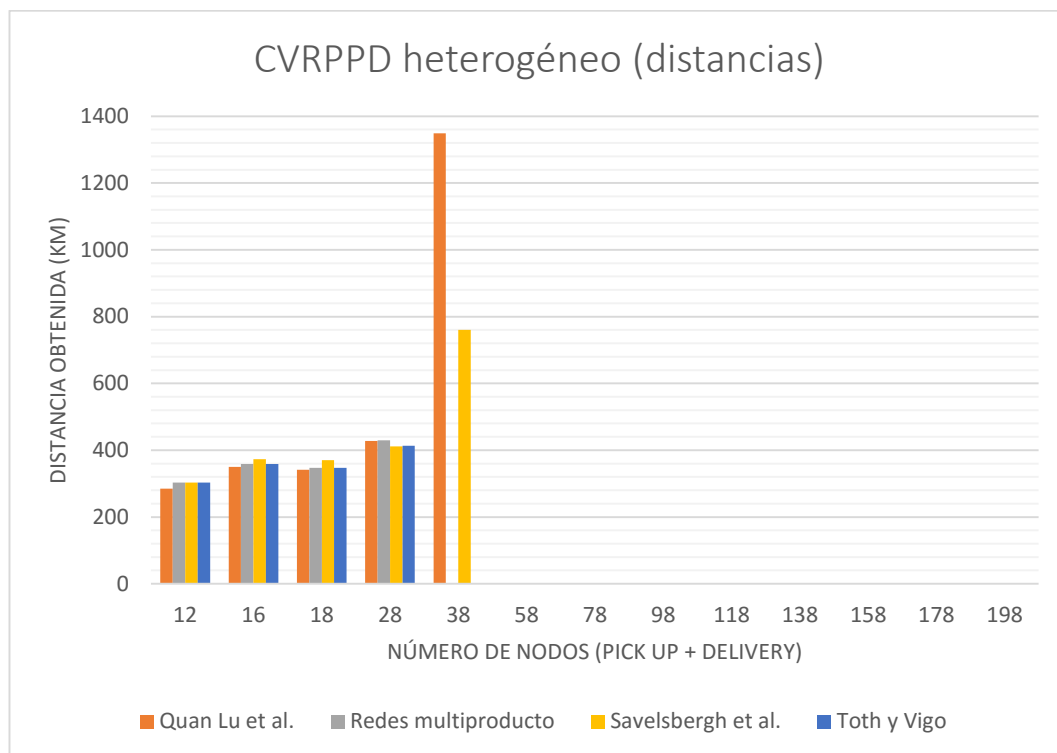


Figura 7.8: Distancias de CPLEX para el CVRPPD heterogéneo.

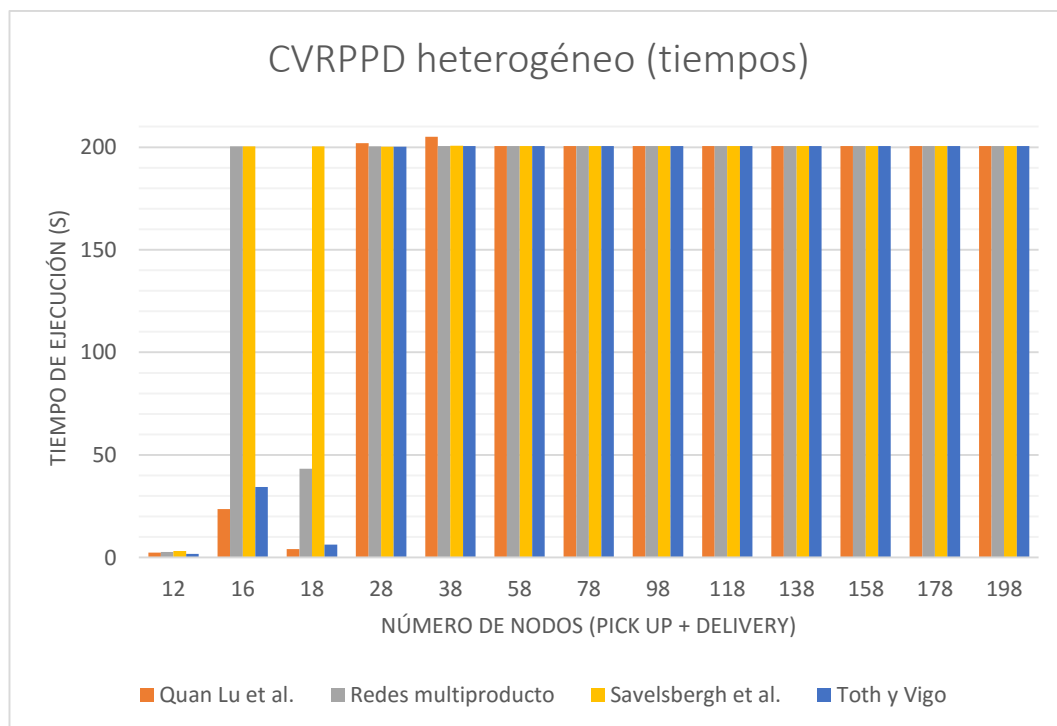


Figura 7.9: Tiempos de CPLEX para el CVRPPD heterogéneo.

7.6.3. Resumen de resultados obtenidos. Caso heterogéneo.

En la Figura 7.8, observamos que los modelos de redes multiproducto (c) y de Toth y Vigo (e) logran resolver 4 problemas, por 5 de los modelos de Quan Lu et al. (b) y Savelsbergh (d). Además de esto, los modelos (c) y (e) obtienen siempre iguales o peores distancias que los modelos (b) y (d). Esto se repite en el caso de la Figura 7.9, en donde se aprecia que los modelos (c) y (e) obtienen peores tiempos de ejecución siempre que el modelo (b).

Por tanto, se considera a los modelos (c) y (e) no válidos para resolver problemas CVRPPD heterogéneo.

Comparando los modelos (b) y (d) vemos que ambos resuelven 5 problemas, que el modelo (b) obtiene en 3 de esos casos mejores distancias (por 2 del modelo (d)) y en otros 3 casos mejores tiempos de ejecución (llegando, como el modelo (d) al límite de tiempo de ejecución en los 2 casos restantes).

En definitiva, el modelo (b) se antoja más adecuado para resolver problemas pequeños (inferior a 18 nodos), en tanto que el modelo (d) resulta más eficiente para un número de nodos igual o superior. Es la primera vez en el presente TFG que llegamos a una doble solución, pero nos ha parecido adecuado en presencia de los resultados analizados.

La Tabla 7.8 muestra el resumen de resultados obtenidos para el CVRPPD heterogéneo empleando CPLEX⁶⁵.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
QuanLu et al.	3 de 13	2 de 13	5 de 13	Opción si n ^o nodos < 18
Redes multiproducto	0 de 13	0 de 13	4 de 13	Descartado
Savelsbergh et al.	2 de 13	0 de 13	5 de 13	Opción si n ^o nodos ≥ 18
Toth y Vigo	0 de 13	1 de 13	4 de 13	Descartado

Tabla 7.7: Resumen de resultados obtenidos con CPLEX para el problema CVRPPD heterogéneo.

⁶⁵ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

Datos empleados		Modelo de Quan Lu et al.		Modelo de Redes multiproducto		Modelo de Savelsbergh et al.		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PD7_heterog o PD7_QuanLu_heterog	12	285,218	2,324	303,252	2,73	303,213	3,23	303,252	1,778
PD9_heterog o PD9_QuanLu_heterog	16	350,253	23,649	358,582	200,35	372,995	200,32	358,582	34,32
PD10_heterog o PD10_QuanLu_heterog	18	341,609	4,196	347,220	43,306	370,636	200,351	347,220	6,24
PD15_heterog o PD15_QuanLu_heterog	28	428,154	202,005	429,711	200,414	411,365	200,242	413,349	200,312
PD20_heterog o PD20_QuanLu_heterog	38	1349,243	205,073	#N/A	200,538	760,674	200,718	#N/A	200,538
PD30_heterog o PD30_QuanLu_heterog	58	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD40_heterog o PD40_QuanLu_heterog	78	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD50_heterog o PD50_QuanLu_heterog	98	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD60_heterog o PD60_QuanLu_heterog	118	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD70_heterog o PD70_QuanLu_heterog	138	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD80_heterog o PD80_QuanLu_heterog	158	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD90_heterog o PD90_QuanLu_heterog	178	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538
PD100_heterog o PD100_QuanLu_heterog	198	#N/A	200,538	#N/A	200,538	#N/A	200,538	#N/A	200,538

Tabla 7.8: Resultados obtenidos con CPLEX para el CVRPPD heterogéneo.

7.7. Comparativa Xpress-IVE vs CPLEX para el CVRPPD.

7.7.1. Caso homogéneo.

A continuación, se presentan los resultados obtenidos para los tres modelos planteados tanto en Xpress-IVE como en CPLEX para el problema CVRPPD homogéneo:

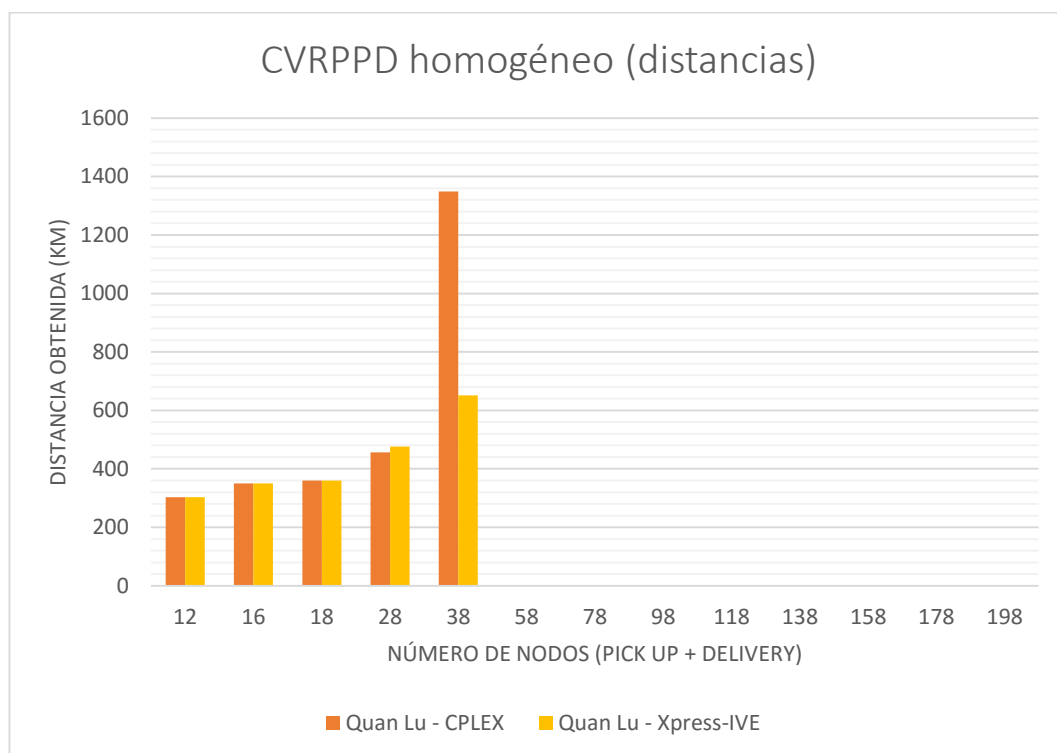


Figura 7.10: Comparativa de distancias modelo de Quan Lu et al.

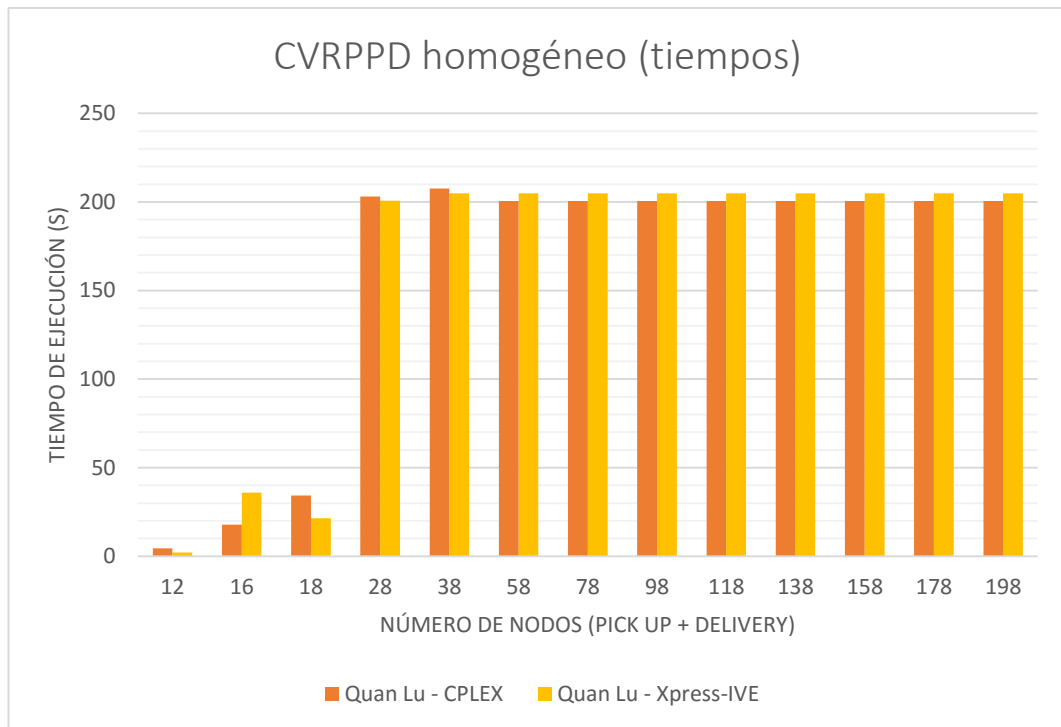


Figura 7.11: Comparativa de tiempos modelo de Quan Lu et al.

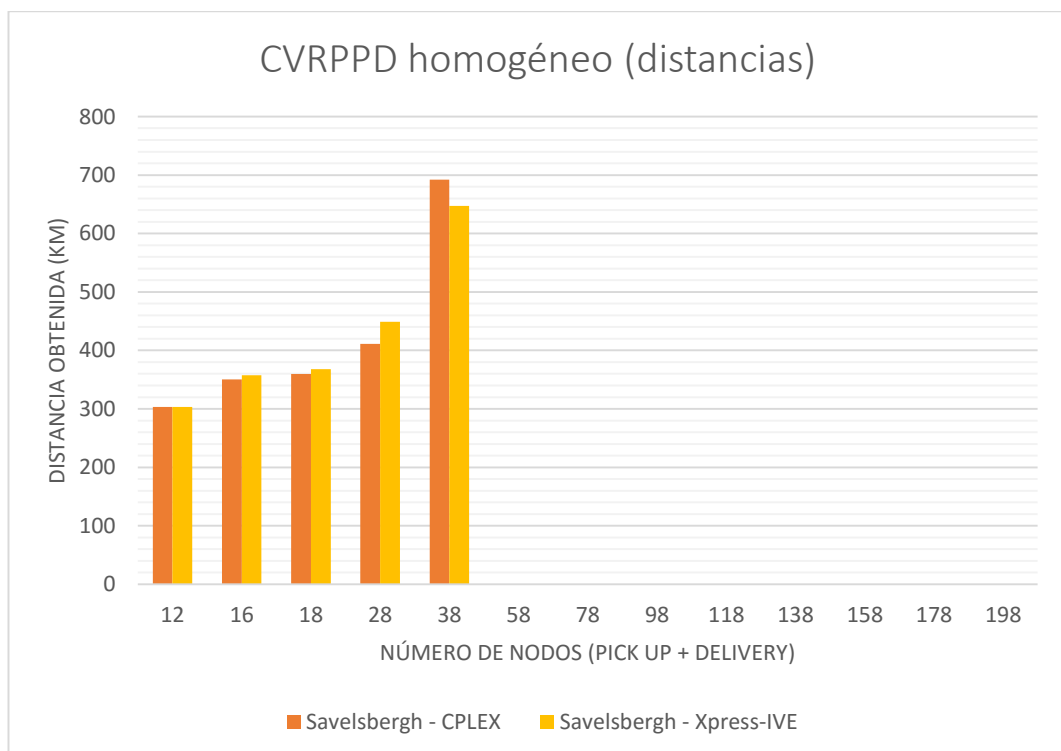


Figura 7.12: Comparativa distancias modelo de Savelsbergh et al.

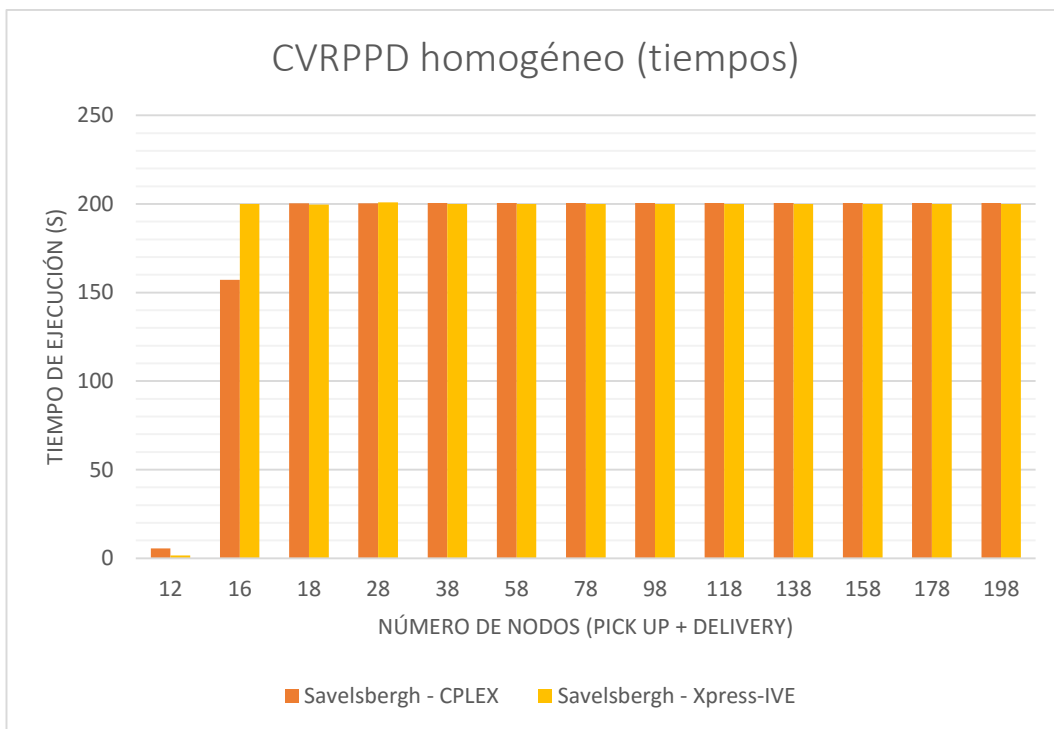


Figura 7.13: Comparativa tiempos modelo de Savelsbergh et al.

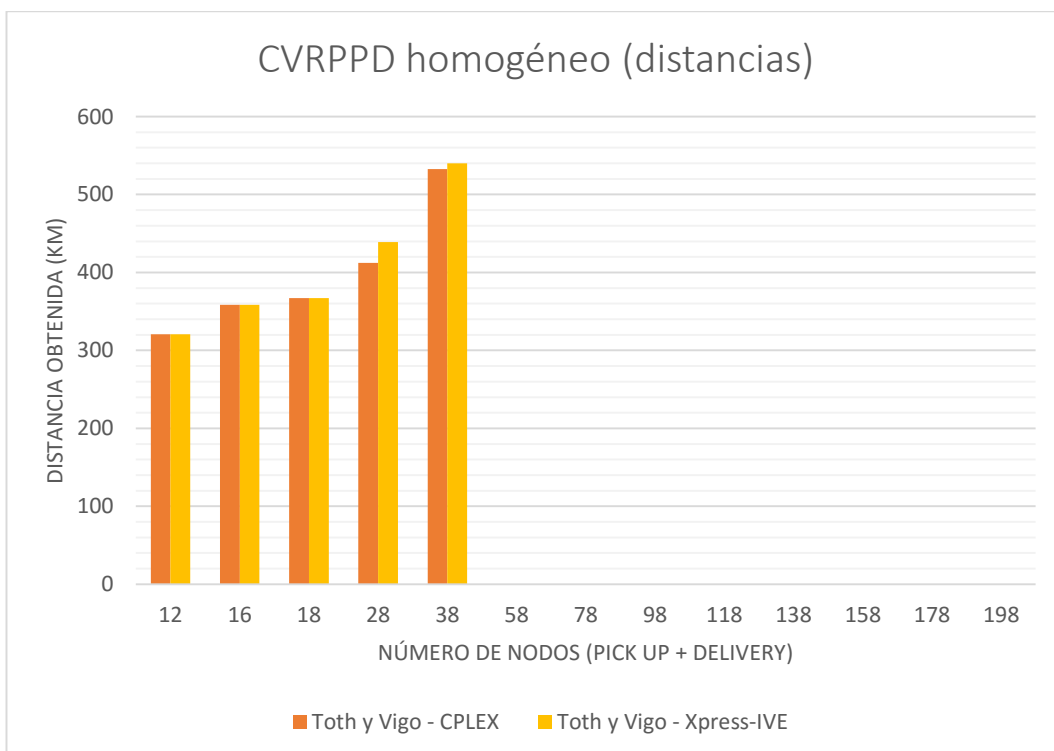


Figura 7.14: Comparativa de distancias modelo de Toth y Vigo.

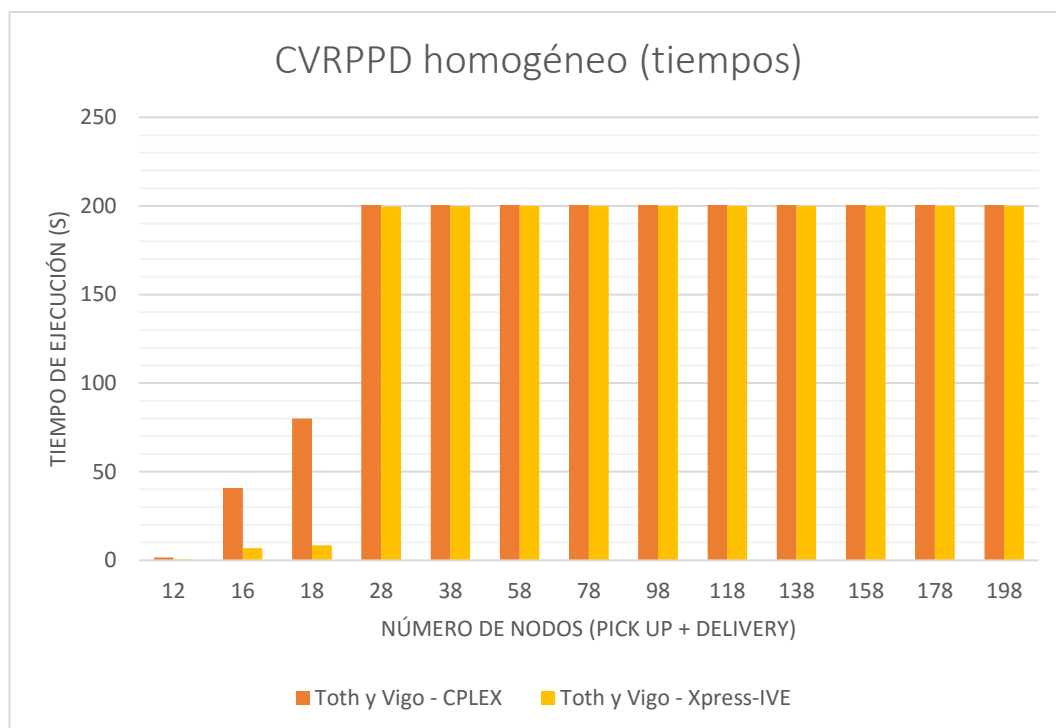


Figura 7.15: Comparativa de tiempos modelo de Toth y Vigo.

7.7.2. Resumen de resultados obtenidos. Caso homogéneo.

Analizando los resultados obtenidos, observamos lo siguiente:

- Ambos solvers resuelven 15 problemas sobre los 39 planteados.
- De esos 15 problemas, CPLEX obtiene mejores distancias en 6, por 2 de Xpress-IVE y 7 empates.
- En relación a los tiempos de ejecución, CPLEX obtiene mejores resultados en 2 de los 15 casos, por 6 de Xpress-IVE y 7 casos en los que se ha llegado al tiempo máximo de ejecución permitido.

Como se puede comprobar, no existen diferencias tan notorias entre ambos solvers como en los capítulos anteriores que permitan seleccionar uno como el más adecuado para resolver problemas CVRPPD homogéneos. Una decisión aceptable podría ser emplear CPLEX para resolver problemas en los que la distancia sea un factor crítico, en tanto que si lo que se busca es minimizar los tiempos de ejecución, podría elegirse Xpress-IVE como solución más acertada.

En la Tabla 7.9 muestra el resumen de resultados obtenidos para el CVRP homogéneo comparando Xpress-IVE con CPLEX⁶⁶.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	9 de 39	6 de 39	15 de 39	Opción si tiempo es crítico
CPLEX	13 de 39	2 de 39	15 de 39	Opción si distancia es crítica

Tabla 7.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPD homogéneo.

7.7.3. Caso heterogéneo.

A continuación, se muestran los resultados obtenidos para los 4 modelos del problema CVRPPD heterogéneo planteados tanto en Xpress-IVE como en CPLEX:

⁶⁶ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

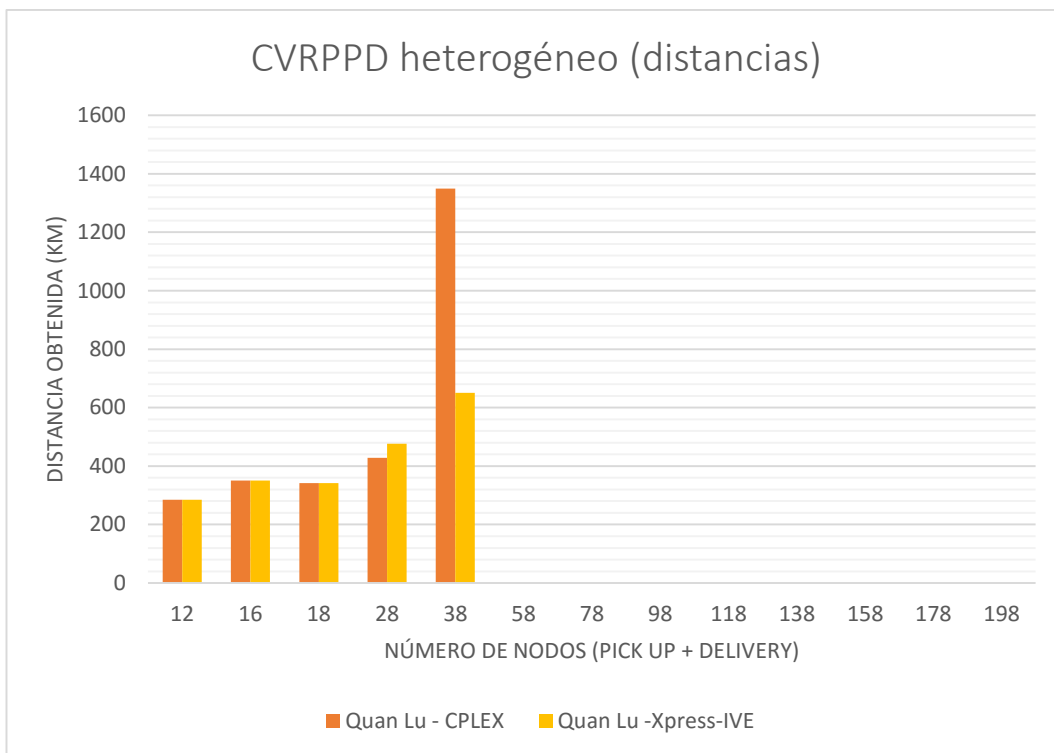


Figura 7.16: Comparativa de distancias modelo de Quan Lu et al.

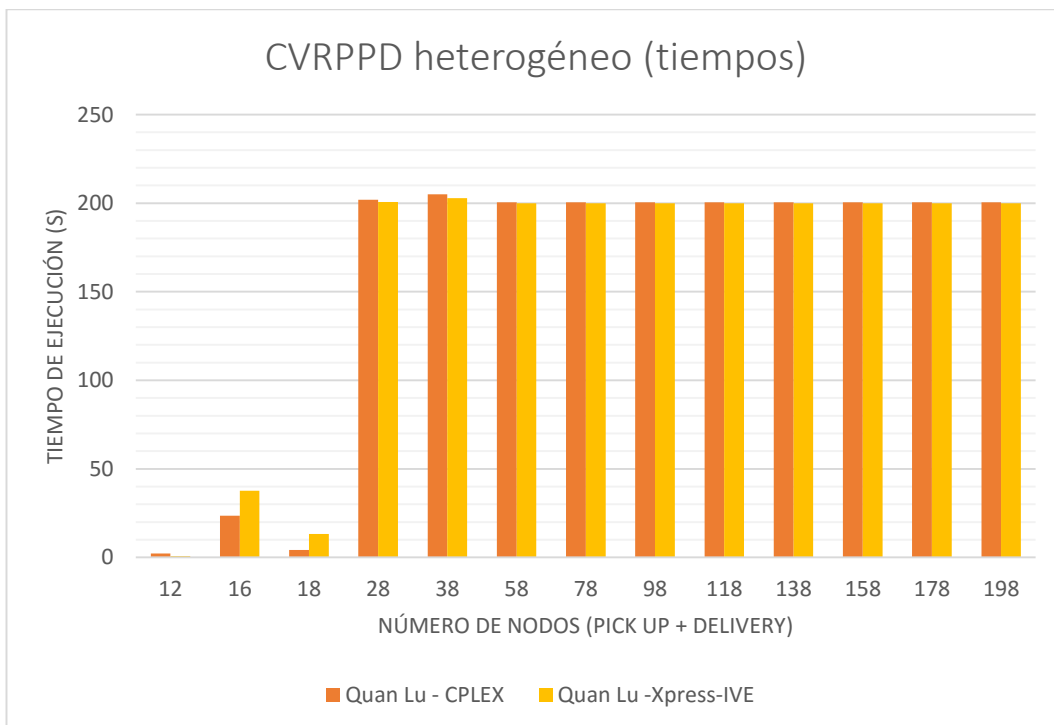


Figura 7.17: Comparativa de tiempos modelo de Quan Lu et al.

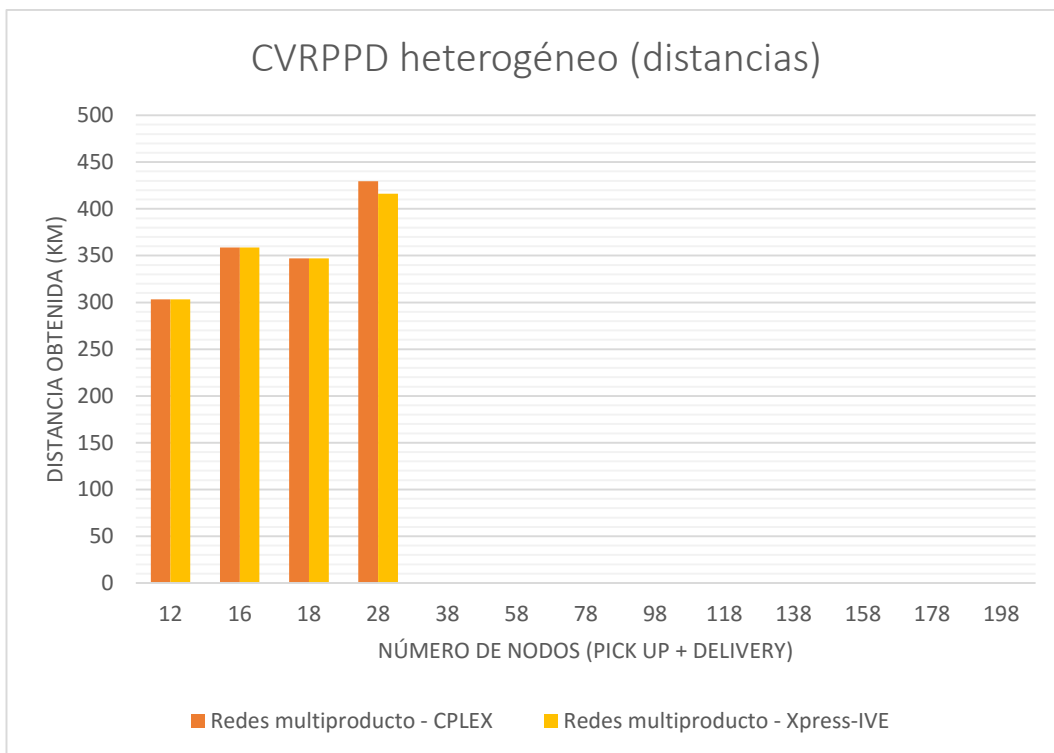


Figura 7.18: Comparativa de distancias modelo de redes multiproducto.

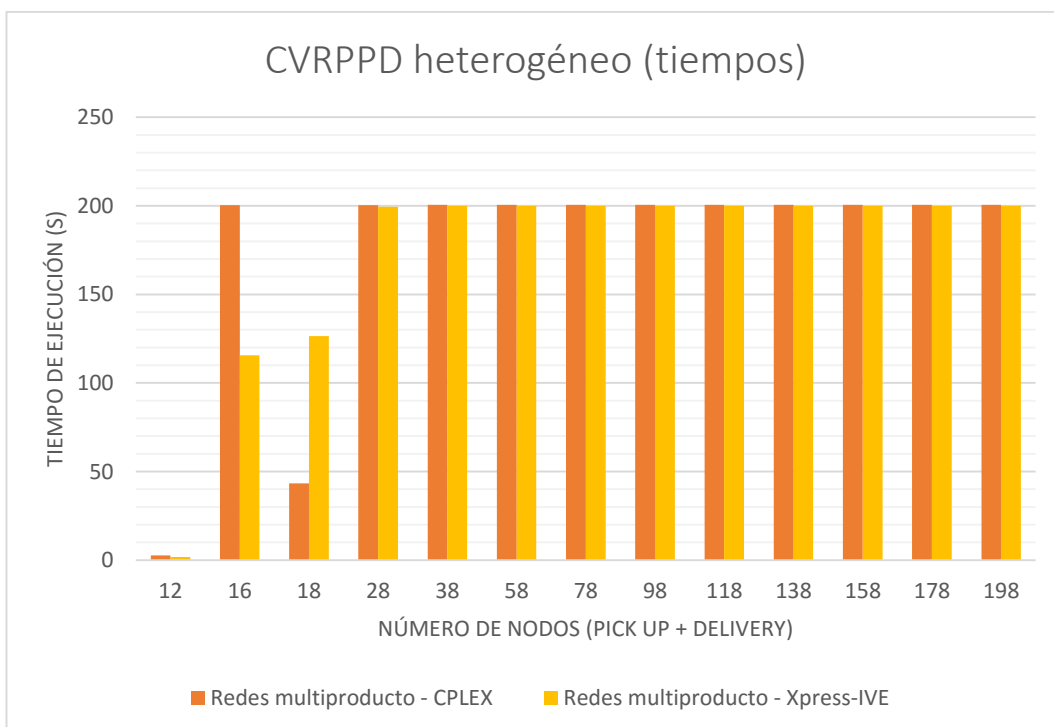


Figura 7.19: Comparativa de tiempos modelo de redes multiproducto.

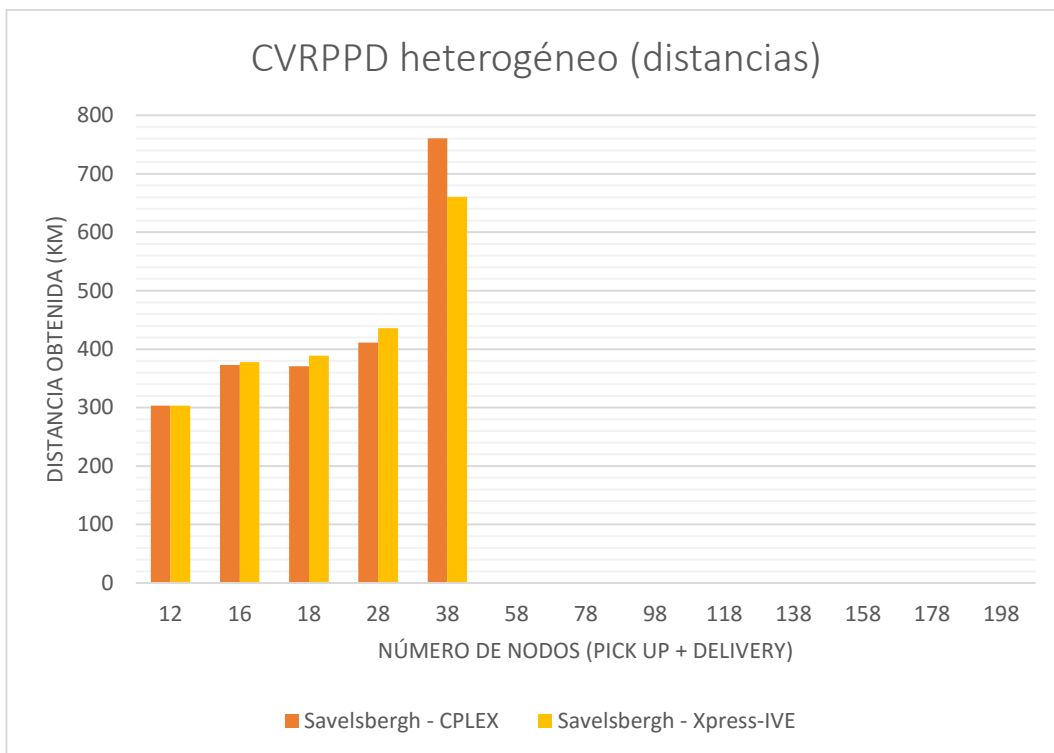


Figura 7.20: Comparativa distancias modelo de Savelsbergh et al.

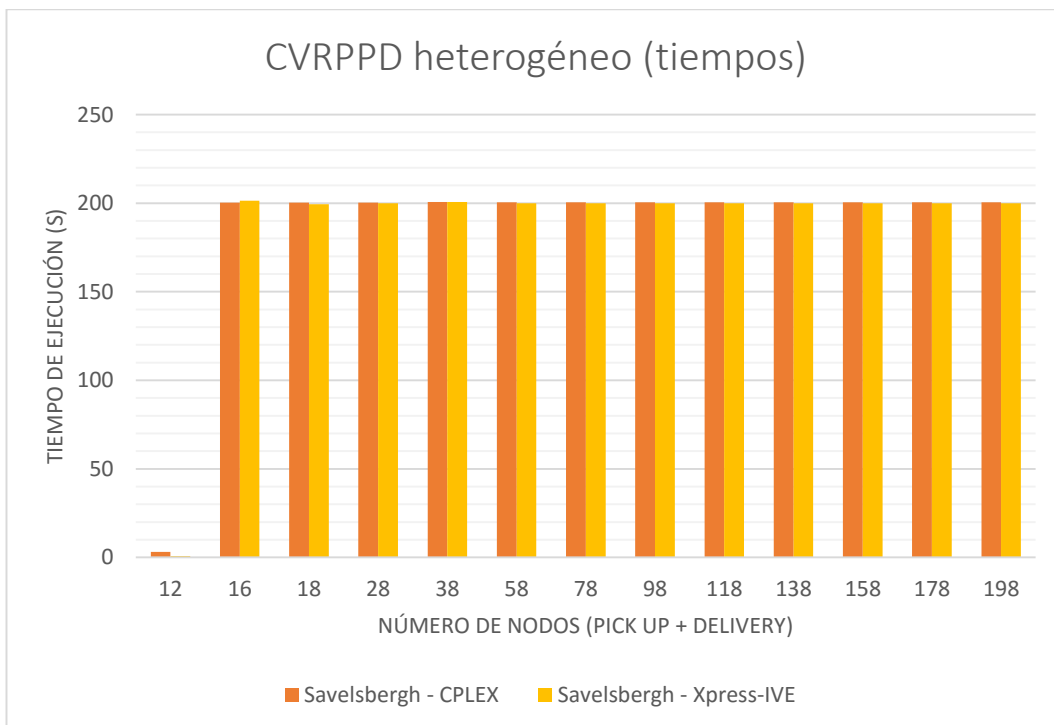


Figura 7.21: Comparativa de tiempos modelo de Savelsbergh et al.

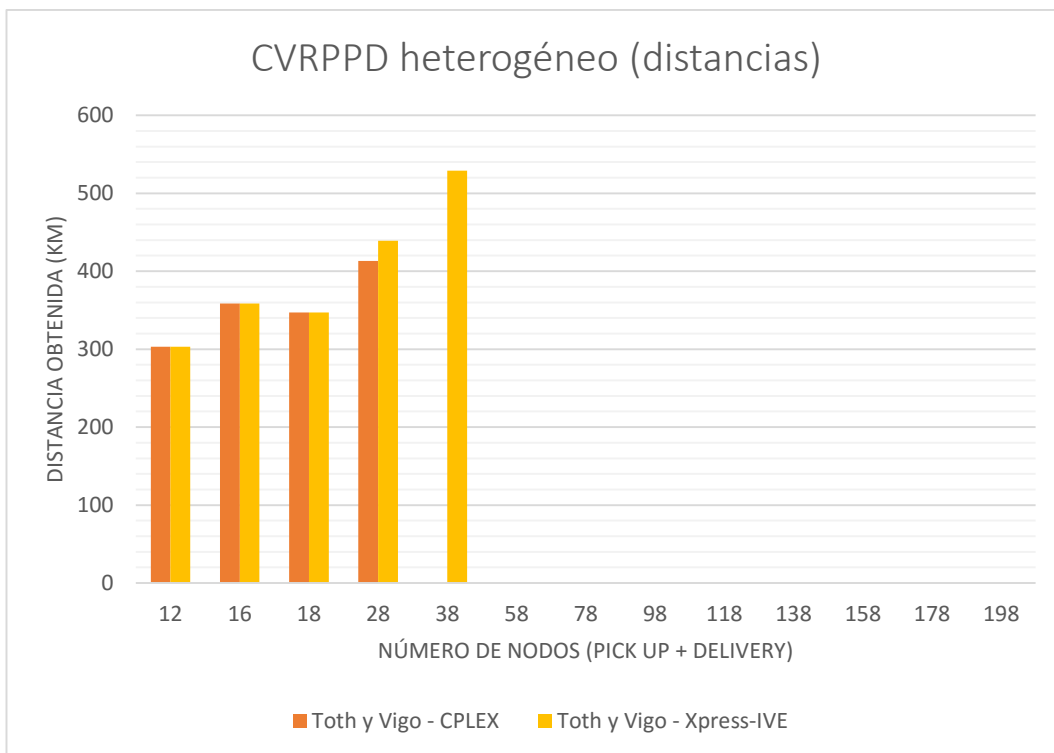


Figura 7.22: Comparativa distancias modelo de Toth y Vigo.

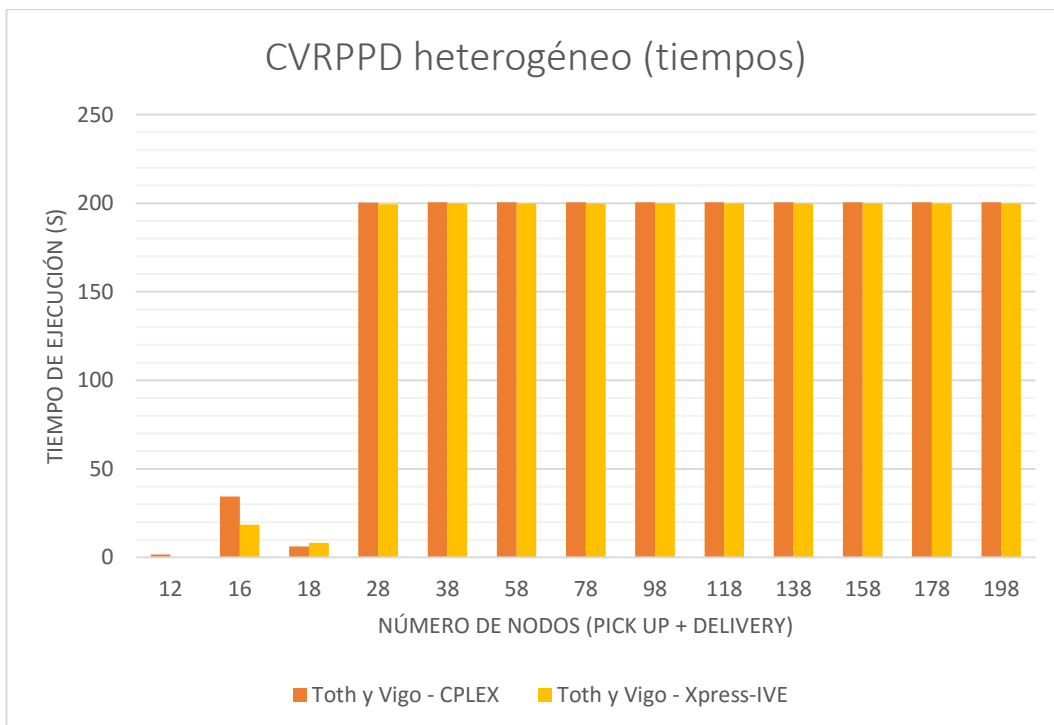


Figura 7.23: Comparativa de tiempos modelo de Toth y Vigo.

7.7.4. Resumen de resultados obtenidos. Caso heterogéneo.

Analizando los resultados presentados anteriormente, extraemos las siguientes estadísticas:

- CPLEX resuelve 18 de los 52 problemas planteados, por 19 de Xpress-IVE.
- De los 18 problemas resueltos por ambos solvers, CPLEX obtiene mejores distancias en 5 de ellos, por 3 de Xpress-IVE y 10 empates.
- En relación a los tiempos de ejecución, CPLEX obtiene mejores resultados en 4 de los 18 casos, por 6 de Xpress-IVE y 8 casos en los que se ha llegado al tiempo máximo de ejecución permitido.

Como podemos observar, los dos solvers se encuentran en un punto intermedio en cuanto a los criterios de distancias y tiempos de ejecución obtenidos, no presentando una inclinación tan clara como en el caso homogéneo.

Si nos fijamos en el tamaño de los problemas resueltos, sí que logra extraerse un patrón en el cuál Xpress-IVE se demuestra más eficiente resolviendo problemas de 38 nodos, incluso dándose un caso en el que este logra encontrar solución factible, pero CPLEX no.

Por tanto, y en base al análisis anterior se concluye que, de evaluarse problemas de un número elevado de nodos de pick up y delivery (a partir de unos 38), Xpress-IVE es el solver más adecuado para resolver problemas CVRPPD heterogéneos, no siendo significativa la elección de uno u otro solver en caso contrario.

En la Tabla 7.10 se puede observar el resumen de resultados obtenidos para el CVRPPD heterogéneo comparando Xpress-IVE con CPLEX⁶⁷.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	13 de 52	6 de 52	19 de 52	Opción si nº de nodos ≥ 38, indiferente en caso contrario
CPLEX	10 de 39	3 de 39	11 de 39	Indiferente si nº de nodos < 38

⁶⁷ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

Tabla 7.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPD heterogéneo.

7.8. Conclusiones.

Como resumen final de los resultados anteriormente citados en este capítulo, y teniendo en cuenta los criterios establecidos en los apartados 2.4 y 3.2.4., concluimos que:

- Para problemas CVRPPD homogéneos, el modelo Toth y Vigo se ha seleccionado como el más adecuado, si bien el modelo de Quan Lu et al. puede llegar a ser una opción siempre y cuando el número de nodos (pick up + delivery) sea inferior a 18.
- Para resolver problemas CVRPPD heterogéneos, si el número de nodos (pick up + delivery) es inferior a 18, el modelo de Quan Lu et al. es el más efectivo. En caso contrario, se selecciona el modelo de Savelsbergh et al.
- Comparando solvers para el caso homogéneo, se recomienda emplear de Xpress-IVE en casos en que el tiempo de ejecución sea un factor crítico. El uso de CPLEX, por su parte, se aconseja si el factor crítico es la distancia calculada.
- Comparando solvers para el caso heterogéneo, hemos llegado a la conclusión de que resulta indiferente qué solver emplear para problemas de un tamaño inferior a 38 nodos (pick up + delivery). De tenerse o superarse este valor, recomendamos el empleo de Xpress-IVE ya que es el que mejores resultados presenta para problemas de tamaños elevados.

Por último, y como hemos indicado en todos los capítulos, cuestionarse el por qué un modelo obtiene mejores resultados que otro consideramos que no tiene mucho sentido debido a que la respuesta reside en la formulación matemática empleada. Sin embargo, sí lo tiene en el caso de los solvers. Por ello, se recomienda al lector consultar el apartado 3.2.5, en donde se han tratado estos motivos con anterioridad.

CAPÍTULO 8

PROBLEMAS DE RUTAS CON RECOGIDAS (PICK UP), ENTREGAS (DELIVERY) Y VENTANAS DE TIEMPO (CVRPPDTW).

8.1. Descripción del problema.

La Figura 8.1 muestra una imagen que quizás le resulte familiar al lector, ya que fue mostrada anteriormente en el apartado 2.3 (concretamente se trata de la Figura 2.8 del Capítulo 2).

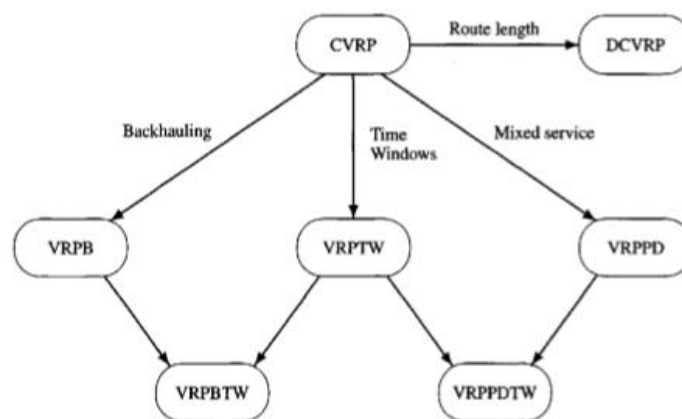


Figura 8.1: tipos de problemas de rutas de vehículos.

Como se puede apreciar en dicha imagen (extraída de (Toth & Vigo, "The Vehicle Routing Problem", 2002)), varios problemas de rutas pueden juntarse para generar un problema nuevo, que reúne propiedades de los anteriores.

Este es el caso del CVRPPDTW (Capacitated Vehicle Routing Problem with Pick up, Delivery and Time Windows), problema generado de la unión del CVRPPD y el CVRPTW. Su descripción es muy intuitiva, y podría resumirse como un CVRPPD en el cual cada depot, nodo de pick up y nodo de delivery tiene asociada una ventana de tiempo $[a_i, b_i]$ que nos acota los momentos mínimo y máximo en los que cada uno de estos nodos puede ser visitado.

La Figura 8.2 nos muestra un ejemplo de este tipo de problema. Se trata del ejemplo PDTW7_homog (14 nodos contando los de pick up y delivery) ejecutado según el modelo de Savelsbergh et al. (ver apartado 8.2 (b)). Junto a cada nodo se muestra por una parte su demanda en verde si es un nodo de pick up (demanda positiva) o en amarillo si es de delivery (demanda negativa) y por otra en azul la ventana de tiempo asociada a dicho nodo.

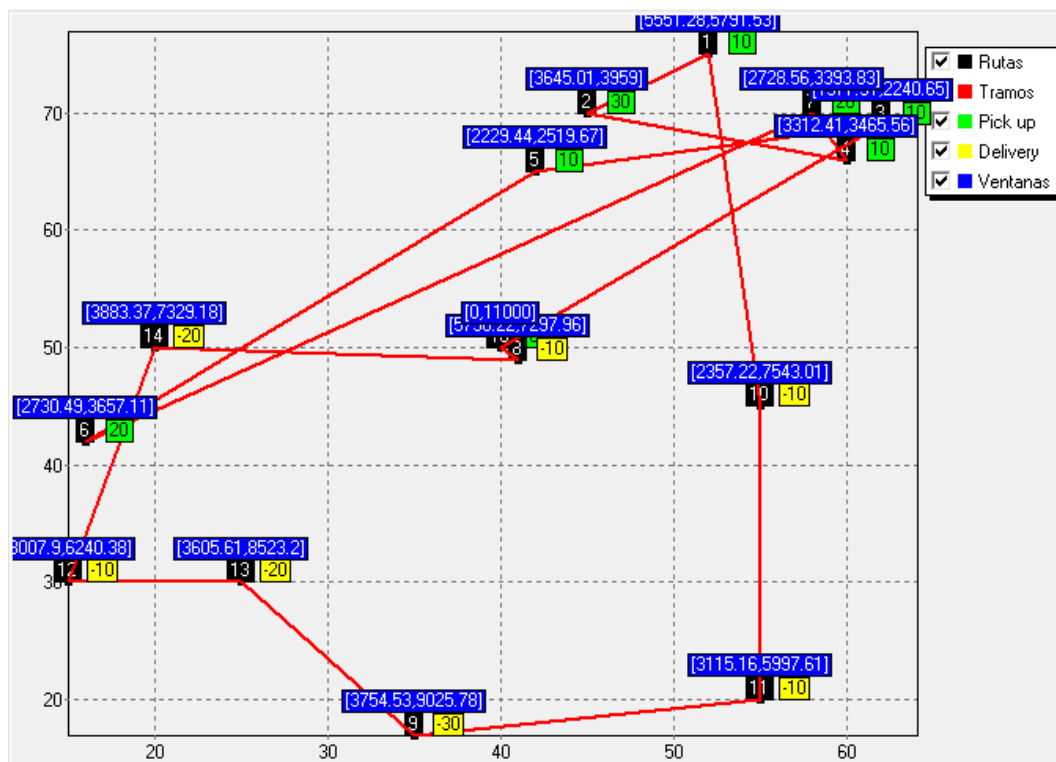


Figura 8.2: ejemplo gráfico del problema CVRPPDTW.

Al igual que ocurría en el problema CVRPPD, consideramos el problema tipo one-to-one, es decir, del nodo origen partirán los vehículos descargados, cargándose/descargándose a medida que vayan recorriendo los distintos nodos de pick up y delivery y llegando descargados al nodo destino.⁶⁸ Dicho nodo destino coincide en localización con el nodo origen, simplemente se trata

⁶⁸ Dependiendo del modelo, nos encontraremos con uno o varios nodos origen y uno o varios nodos destino, debido a que algunos modelos necesitan un planteamiento especial.

de un nodo del que no parten arcos, equivalente al nodo origen, al que ahora no llegará ningún arco.

Por último, a continuación aparecen enumerados los objetivos que hemos perseguido en el presente capítulo, que difieren un poco respecto de capítulos anteriores⁶⁹:

- **Objetivo 1:** establecer, en base a diversos criterios (ver apartado 2.4) qué modelo de entre todos los propuestos resulta el más efectivo resolviendo problemas CVRPPDTW homogéneos y/o heterogéneos.
- **Objetivo 2:** identificar el solver más efectivo (Xpress-IVE o CPLEX) para resolver los problemas CVRPPDTW propuestos, en base a los criterios establecidos en el apartado 3.2.4.

8.2. Modelos planteados. Caso homogéneo.

a) Modelo de Paolo Toth y Daniele Vigo.

En el libro (Toth & Vigo, "The Vehicle Routing Problem", 2002) encontramos el primero de los modelos que vamos a presentar, que queda formulado como sigue:

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga transportada por el vehículo k al salir del nodo i .
- T_{ik} : variable de tiempo que nos va a indicar el momento de comienzo de servicio en el nodo i , atendido por el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo, todas las capacidades son idénticas.

⁶⁹ En base a los resultados que se pueden ver en el Anexo para el problema CVRPPDTW, no se han reformulado algunas restricciones de los modelos para intentar reducir la distancia total recorrida, por lo que uno de los tres objetivos perseguidos hasta ahora desaparece.

- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $W=\{2n+1, 2n+2\}$: el nodo $2n+1$ representa el nodo de origen. El nodo $2n+2$ representa el nodo destino.
- $N= P \cup D \cup W$.
- $K=\{1\dots k\}$: conjunto del número de vehículos a emplear.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{k \in K} \sum_{\substack{j \in N \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{ijk} = 1 \quad \forall i \in P \quad (2)$$

(2) Para cualquiera de los nodos de pick up debe existir un solo arco saliente de dicho nodo, que no puede ir al nodo origen ni al nodo destino.

$$\sum_{j \in N} x_{ijk} - \sum_{j \in N} x_{j n+i k} = 0 \quad \forall i \in P; k \in K \quad (3)$$

(3) Un nodo de pick up y su correspondiente nodo de delivery deben de ser atendidos por el mismo vehículo.

$$\sum_{\substack{j \in N \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{\text{origen } j k} = 1 \quad \forall k \in K \quad (4)$$

(4) Para cada vehículo, debe existir un arco saliente del nodo origen.

$$\sum_{i \in N} x_{ijk} - \sum_{i \in N} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k \in K \quad (5)$$

(5) Para cada nodo distinto de los nodos origen/destino, se debe cumplir que el número de arcos que entran sea igual al número de arcos que salen de dicho nodo.

$$\sum_{\substack{i \in N \\ i \neq \text{destino} \\ i \neq \text{origen}}} x_{i \text{ destino } k} = 1 \quad \forall k \in K \quad (6)$$

(6) Al nodo destino, para cada vehículo, debe llegar un arco.

$$T_{ik} + s_i + t_{ij} - T_{jk} \leq M(1 - x_{ijk}) \\ \forall k \in K; i, j \in N / i \neq \text{destino}; a_i + s_i + t_{ij} \leq b_j \quad (7)$$

(7) Restricciones de tiempo. Expresan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo el tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \leq T_{ik} \leq b_i \quad \forall i \in N; k \in K \quad (8_1 \text{ y } 8_2)$$

(8_1 y 8_2) Acotación de los tiempos de comienzo de servicio gracias al empleo de las ventanas de tiempo.

$$T_{ik} + t_{i+n} + s_i \leq T_{i+n k} \quad \forall i \in P; k \in K \quad (9)$$

(9) Restricción empleada para que cada nodo de pick up vaya antes que su correspondiente nodo de delivery en la ruta.

$$L_{ik} + d_j - L_{jk} \leq cap(1 - x_{ijk}) \\ \forall k \in K; i, j \in N / i \neq destino ; a_i + s_i + t_{ij} \leq b_j \quad (10)$$

(10) Restricciones para asignación de carga a lo largo de la ruta.

$$d_i \leq L_{ik} \leq cap \quad \forall i \in P; k \in K \quad (11_1 \text{ y } 11_2)$$

$$0 \leq L_{ik} \leq cap \quad \forall i \in D; k \in K \quad (12_1 \text{ y } 12_2)$$

(11_1 y 11_2), (12_1 y 12_2) Cotas superior e inferior para la carga transportada por cada vehículo al salir de cada nodo. En el caso de abandonar un nodo de pick up ($\in P$), la carga nunca podrá ser nula, sí pudiendo serlo en el caso de abandonar un nodo de delivery ($\in D$).

$$L_{origen\ k} = 0 ; L_{destino\ k} = 0 \quad \forall k \in K \quad (13_1 \text{ y } 13_2)$$

(13_1 y 13_2) La carga inicial y final de cada vehículo deberá ser nula.

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in N / a_i + s_i + t_{ij} \leq b_j^{70} \\ L_{ik} \geq 0 \quad \forall k \in K; i \in N \\ D_{ik} \geq 0 \quad \forall k \in K; i \in N$$

b) Modelo de M.W.P.Savelsbergh y M.Sol

Realizando unas pequeñas modificaciones al modelo propuesto en (Savelsbergh & M.Sol, 1995), conseguimos la formulación siguiente:

Variables:

⁷⁰ Es decir, sólo generamos variables x_{ij} para aquellos arcos que puedan existir, en base a las ventanas de tiempo existentes.

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- z_{ik} : variable binaria que de valer 1 nos va a indicar que el nodo i es atendido por el vehículo k .
- D_i : variable de tiempo que nos va a indicar el momento de salida del nodo i del vehículo que atiende el nodo.
- y_i : variable entera que especifica la carga del vehículo correspondiente cuando sale del nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $N^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $N^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $N = N^+ \cup N^-$.
- $M = \{1 \dots K\}$, con K el número de vehículos empleados.
- $M^+ = \{2n+1 \dots 2n+K\}$: el nodo $2n + i$, con $i \leq K$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+K+1 \dots 2n+2K\}$: el nodo $2n + i$, con $K < i \leq 2K$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.

Formulación:

$$\text{Min} \sum_{i \in N \cup W} \sum_{j \in N \cup W} \sum_{k \in M} x_{ijk} d_{ij}$$

$$\sum_{k=1}^K z_{ik} = 1 \quad \forall i \in N \quad (1)$$

(1) Cada nodo debe ser atendido por un solo vehículo.

$$z_{ik} = z_{i+n_k} \quad \forall i \in N^+; k \in M \quad (1_1)$$

(1_1) El nodo de pick up y su correspondiente nodo de delivery deben ser atendidos por el mismo vehículo. Hemos añadido esta restricción para asegurar que esta propiedad se cumpla.

$$\sum_{j \in N \cup M_k^-} x_{ijk} = z_{ik} \quad \forall i \in N; k \in M \quad (2_1)$$

$$\sum_{j \in N \cup M_k^+} x_{jik} = z_{ik} \quad \forall i \in N; k \in M \quad (2_2)$$

(2_1), (2_2) En el caso de ser atendido el nodo i por el vehículo k , deberá existir un arco saliente y un arco entrante en el nodo i , a no ser que se trate del nodo origen o el nodo destino del vehículo k , en cuyo caso sólo existirá un nodo saliente o un nodo entrante, respectivamente.

Como ya hemos visto, M_k^- representa el nodo de destino para el vehículo k , en tanto que M_k^+ representa el nodo de origen de dicho vehículo.

$$\sum_{j \in N} x_{2n+k j k} = 1 \quad \forall k \in M \quad (3)$$

(3) Del nodo origen de cada vehículo debe salir un arco.

$$\sum_{i \in N} x_{i 2n+K+k k} = 1 \quad \forall k \in M \quad (4)$$

(4) A cada nodo destino de cada vehículo debe llegar un arco.

$$D_{2n+k} = 0 \quad \forall k \in M \quad (5)$$

(5) El tiempo de comienzo del servicio en cada nodo origen será 0 (todos están disponibles en el instante 0).

$$D_i + t_{i \rightarrow n} \leq D_{i+n} \quad \forall i \in N^+ \quad (6)$$

(6) Restricciones para garantizar que cada nodo pick up vaya antes que su correspondiente nodo delivery en la ruta.

$$D_i + t_{ij} + s_i - D_j \leq M(1 - x_{ijk}) \\ \forall k \in M; i \in N \cup M^+; j \in N \cup W / a_i + s_i + t_{ij} \leq b_j \quad (7)$$

(7) Restricciones de tiempo. Expresan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo el tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \leq D_i \leq b_i \quad \forall i \in N^+ \quad (7_1 \text{ y } 7_2)$$

(7_1 y 7_2) Cotas superior e inferior para los tiempos de comienzo del servicio en cada nodo, en base a las ventanas de tiempo de dicho nodo.

$$y_{2n+k} = 0 \quad \forall k \in M \quad (8_1)$$

$$y_{2n+K+k} = 0 \quad \forall k \in M \quad (8_2)$$

(8_1), (8_2) La carga transportada por cada vehículo al atender los nodos origen y destino debe ser nula.

$$y_i \leq \sum_{k \in M} (cap \cdot z_{ik}) \quad \forall i \in N \quad (9)$$

(9) En el caso de que el nodo i sea atendido por el vehículo k , la carga transportada por el vehículo al salir de dicho nodo no podrá ser mayor que la capacidad máxima de carga del mencionado vehículo.

$$y_i + d_j - y_j \leq cap(1 - x_{ijk})$$

$$\forall k \in M; i \in N \cup M^+; j \in N \cup W / a_i + s_i + t_{ij} \leq b_j \quad (10)$$

(10) Restricciones para asignación de carga a lo largo de la ruta.

$$x_{ijk} \in \{0,1\} \quad \forall k \in M; i, j \in N / a_i + s_i + t_{ij} \leq b_j$$

$$z_{ik} \in \{0,1\} \quad \forall i \in N \cup W; k \in M$$

$$D_i \geq 0 \quad \forall i \in N$$

$$y_i \geq 0 \quad \forall i \in N$$

c) Modelo basado en redes multiproducto.

Al igual que en el problema CVRPPD, partimos de la formulación planteada en (Sáez Aguado, "El problema de flujo multiproducto de coste mínimo en redes: descripción y propiedades", 2016), extendiendo el problema para adaptarlo al caso de disponer de múltiples vehículos para realizar las rutas.

Además, hemos implementado una programación especial, de manera que cada par de nodos pick up - delivery maneja un tipo de producto único, en consonancia con el planteamiento one-to-one.

Por último, se han añadido restricciones para representar la acotación dada por las ventanas de tiempo.

El modelo queda definido como sigue:

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $W=\{2n+1, 2n+2\}$: el nodo $2n+1$ representa el nodo de origen. El nodo $2n+2$ representa el nodo destino.
- $N= P \cup D \cup W$.

- $K=\{1\dots k\}$: conjunto del número de vehículos a emplear.

Variables:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ijkp} : variable entera para indicar la cantidad del producto p transportada del nodo i al nodo j con el vehículo k .
- D_{ik} : variable que indica el tiempo de comienzo del servicio en el nodo i , que es atendido por el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo es la misma para todos los vehículos.
- d_{ip} : cantidad demandada en el nodo i del producto p . Para los nodos de pick up será una cantidad positiva, mientras que para los nodos de delivery será negativa (por convenio).

La idea es que cada nodo de pick up tenga oferta para un único producto, que coincida con el producto demandado por su nodo correspondiente de delivery (es por ello que el número de productos es n).

- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo de servicio en el nodo i .
- b_i : fecha más tardía de comienzo de servicio en el nodo i .
- M : constante de valor elevado empleada para quitarnos de en medio aquellas d_{ip} que no nos sean útiles debido a que no hayamos asignado al nodo i el producto p .

Formulación:

$$\text{Min} \quad \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{j \in N} \sum_{k \in K} x_{j \text{ origen } k} = 0 \quad (0_4)$$

(0_4) Al nodo origen no pueden llegar arcos.

$$\sum_{j \in N} \sum_{k \in K} x_{\text{destino } j k} = 0 \quad (0_5)$$

(0_5) Del nodo destino no pueden salir arcos.

$$\sum_{j \in N} \sum_{k \in K} x_{ijk} = 1 \quad \forall i \in P \cup D \quad (2)$$

(2) Para cualquier nodo distinto de los nodos origen o destino, debe existir un arco saliente de dicho nodo.

$$\sum_{\substack{j \in N \\ j \neq i}} x_{ijk} - \sum_{\substack{j \in N \\ j \neq i}} x_{jn+ik} = 0 \quad \forall i \in P; k \in K \quad (3)$$

(3) Restricción que asegura que cada par de nodos pick up – delivery sea atendido por el mismo vehículo.

$$\sum_{j \in P} x_{\text{origen } j k} = 1 \quad \forall k = 1..K \quad (4)$$

(4) Para el nodo origen, para cada vehículo, debe existir un arco saliente de dicho nodo.

$$\sum_{\substack{i \in N \\ i \neq j}} x_{ijk} - \sum_{\substack{i \in N \\ i \neq j}} x_{jik} = 0 \quad \forall j \in P \cup D; k \in K \quad (5)$$

(5) El número de arcos entrantes en cualquier nodo distinto de los nodos origen o destino debe ser igual al número de arcos salientes de dicho nodo.

$$\sum_{j \in D} x_j \text{ destino } k = 1 \quad \forall k \in K \quad (6)$$

(6) En el nodo destino, para cada vehículo, debe existir un arco entrante a dicho nodo.

$$\sum_{j \in N} \sum_{k \in K} y_{ijkp} - \sum_{j \in N} \sum_{k \in K} y_{jikp} = d_{ip} \quad \forall i \in N; p = 1..n/d_{ip} \neq M \quad (7)$$

(7) Restricciones similares a las de Kirchhoff empleadas en el problema TSP para eliminación de subtours. Indican que la cantidad de un producto que llega a un nodo menos la cantidad de ese producto que sale de ese nodo debe ser igual a la demanda que ese nodo tiene de ese producto. Consideramos sólo aquellos casos en los que el nodo tenga demanda de un producto.

$$\sum_{p=1}^n y_{ijkp} \leq x_{ijk} \cdot cap \quad \forall i, j \in N; k \in K \quad (8)$$

(8) En caso de existir el arco ij empleando el vehículo k, la suma de las cantidades de cada producto transportadas no puede ser superior a la demanda de dicho vehículo. Esta restricción garantiza la relación entre las variables binarias y de flujo.

$$\sum_{j \in N} \sum_{k \in K} y_{jikp} = \sum_{j \in N} \sum_{k \in K} y_{ijkp} \quad \forall i \in N; p = 1..n/d_{ip} = M \quad (9)$$

(9) Extensión de la restricción (7), en este caso para aquellos nodos que no tengan demanda de un producto.

$$D_{origen } k = 0 \quad \forall k \in K \quad (10)$$

(10) El tiempo de comienzo de servicio para el nodo origen debe ser 0.

$$D_{ik} + t_{ij} + s_i - D_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq \text{destino}; i \neq j; k \in K \quad (11)$$

(11) Restricciones de tiempo para eliminación de subtours. Expresan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en

el nodo j debe ser como mínimo el tiempo de comienzo de servicio en el nodo i , más el tiempo de servicio en el nodo i , más el tiempo empleado en ir de i a j . En caso de no existir el arco ij empleando el vehículo k , se tiene una cota superior para los tiempos de comienzo de servicio.

$$D_{ik} + t_{i+n} + s_i \leq D_{i+n k} \quad \forall i \in P; k \in K \quad (12)$$

(12) El nodo de pick up debe ir siempre antes que su correspondiente nodo delivery en la ruta.

$$a_i \leq D_{ik} \leq b_i \quad \forall i \in N; k \in K \quad (13_1 \text{ y } 13_2)$$

(13_1 y 13_2) Acotación de los tiempos de comienzo del servicio dada por las ventanas de tiempo de cada nodo.

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$D_{ik} \geq 0 \quad \forall i \in N; k \in K$$

$$y_{ijkp} \geq 0 \quad \forall i, j \in N; k \in K; p = 1..n$$

d) Modelo de Yvan Dumas, Jacques Desrosiers y François Soumis.

Este nuevo modelo aparece formulado en (Dumas, Desrosiers, & Soumis, 1991) como sigue:

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- y_i : variable entera que nos indica la carga transportada por cada vehículo al salir del nodo i .
- T_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en el nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .

- cap: capacidad de cada vehículo. Como nos encontramos en el caso homogéneo, todas las capacidades son idénticas.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva, mientras que para los nodos de delivery será negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $P^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $P = P^+ \cup P^-$.
- $M = \{1 \dots V\}$, con V el número de vehículos empleados.
- $M^+ = \{2n+1 \dots 2n+V\}$: el nodo $2n + i$, con $i \leq V$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+V+1 \dots 2n+2V\}$: el nodo $2n + i$, con $V < i \leq 2V$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.⁷¹
- $N = P \cup W$.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{v \in M} x_{ijv} d_{ij}$$

⁷¹ En el artículo denota al nodo origen como nodo 0 y al nodo destino como nodo $2*n+1$. Nosotros hemos creado $2*n+v$ orígenes y $2*n+V+v$ destinos. Esto es debido a que si no las restricciones de tiempo generaban problemas.

$$\sum_{v \in M} \sum_{j \in P \cup M_v^-} x_{ijv} = 1 \quad \forall i \in P^+ \quad (6)$$

(6) De cualquier nodo de pick up debe existir un arco saliente de dicho nodo. Recordemos que M_v^- representa al nodo destino del vehículo v , en tanto que M_v^+ representa al nodo origen de dicho vehículo.

$$\sum_{j \in P \cup M_v^-} x_{ijv} - \sum_{j \in P \cup M_v^+} x_{jiv} = 0 \quad \forall i \in P; v \in M \quad (7)$$

(7) Para cualquier nodo distinto de los nodos origen o destino se debe cumplir que el número de nodos que entran debe ser igual al número de nodos que salen de dicho nodo.

$$\sum_{j \in P^+} x_{M_v^+ j v} = 1 \quad \forall v \in M \quad (8)$$

$$\sum_{i \in P^-} x_{i M_v^- v} = 1 \quad \forall v \in M \quad (9)$$

(8), (9) Para cada vehículo, para cada nodo origen y para cada nodo destino, debe existir un arco saliente y un arco entrante a dicho nodo, respectivamente.

$$\sum_{j \in N} x_{ijv} - \sum_{j \in N} x_{j_{n+i}v} = 0 \quad \forall i \in P^+; v \in M \quad (10)$$

(10) Restricción empleada para que cada par de nodos pick up – delivery sea atendido por el mismo vehículo.

$$T_i + t_{i+n} + s_i \leq T_{i+n} \quad \forall i \in P^+ \quad (11)$$

(11) Nos indica que cada nodo de pick up debe ir antes que su correspondiente nodo de delivery en la ruta.

$$T_i + s_i + t_{ij} - T_j \leq M(1 - x_{ijv})$$

$$\forall v \in M; i, j \in P / a_i + s_i + t_{ij} \leq b_j \quad (12)$$

$$T_{M_v^+} + t_{M_v^+ j} - T_j \leq M(1 - x_{M_v^+ jv}) \quad \forall v \in M; j \in P^+ \quad (13)$$

$$T_i + s_i + t_{i M_v^-} - T_{M_v^-} \leq M(1 - x_{i M_v^- v}) \quad \forall v \in M; i \in P^- \quad (14)$$

(12), (13), (14) Restricciones de tiempo para eliminación de subtours. Expresan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo el tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \leq T_i \leq b_i \quad \forall i \in P \quad (15_1 \text{ y } 15_2)$$

$$a_{M_v^+} \leq T_{M_v^+} \leq b_{M_v^+} \quad \forall v \in M \quad (16_1 \text{ y } 16_2)$$

$$a_{M_v^-} \leq T_{M_v^-} \leq b_{M_v^-} \quad \forall v \in M \quad (17_1 \text{ y } 17_2)$$

(15_1 y 15_2), (16_1 y 16_2), (17_1 y 17_2) Cotas superiores e inferiores para los tiempos de comienzo de servicio, dadas por las correspondientes ventanas de tiempo de cada nodo.

$$y_i + d_j - y_j \leq cap(1 - x_{ijv}) \\ \forall v \in M; i \leq 2n + V; j \in N / a_i + s_i + t_{ij} \leq b_j \quad (18)$$

(18) Restricción para garantizar la correcta asignación de carga durante la ruta.⁷²

$$d_i \leq y_i \leq cap \quad \forall i \in P^+ \quad (21_1 \text{ y } 21_2)$$

(21_1 y 21_2) Cotas superior e inferior para la carga transportada al salir de cada nodo distinto de los nodos origen o destino.

⁷² En el artículo aparecen junto a la restricción (18), otro par de restricciones, denotadas como (19) y (20), que intentan descomponer las restricciones típicas de asignación de carga en tres restricciones. Sin embargo, estas tres dan problemas cuando las planteamos juntas, por lo que se ha decidido agruparlas en una sola, como venía ocurriendo hasta ahora.

$$y_{M_v^+} = 0; y_{M_v^-} = 0 \quad \forall v \in M \quad (22_1 \text{ y } 22_2)$$

(22_1 y 22_2) La carga transportada al salir de los nodos origen o destino debe ser 0.

$$x_{ijv} \in \{0,1\} \quad \forall v \in M; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$y_i \geq 0 \quad \forall i \in N$$

$$T_i \geq 0 \quad \forall i \in N$$

e) Modelo de Enrico Angelelli y Renata Mansini.

En el artículo (Angelelli & Mansini, 2002) aparece una curiosa formulación que difiere ligeramente respecto de las planteadas hasta ahora, veámosla:

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- D_{ik} : variable entera que nos indica la cantidad restante de deliveries transportadas por el vehículo k cuando parte del nodo i .
- P_{ik} : variable entera que nos indica la cantidad de pick ups recogidas y que son transportadas por el vehículo k cuando parte del nodo i .
- T_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en el nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap : capacidad de cada vehículo. Como nos encontramos en el caso homogéneo, todas las capacidades son idénticas.
- d_i : requerimiento de entrega (delivery) del nodo i .
- p_i : requerimiento de recogida (pick up) del nodo i .
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .

- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $N=\{1...2n\}$: representa a todos los nodos. Del nodo 1 al nodo n se trata de nodos de pick up y del nodo n al nodo $2n$ de los nodos de delivery. No se incluyen ni el nodo de origen ni el nodo de destino
- $V=N \cup \{2n+1, 2n+2\}$: representa al conjunto de todos los nodos, incluyendo además el nodo de origen y el nodo de destino.
- $K=\{1...k\}$, con k el número de vehículos empleados.

Formulación:

$$\text{Min} \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1 \quad \forall i \in N \quad (4)$$

(4) Para cada nodo distinto de los nodos origen o destino debe existir un arco saliente de dicho nodo.

$$\sum_{i \in V} x_{ijk} = \sum_{i \in V} x_{jik} \quad \forall j \in N; k \in K \quad (5)$$

(5) Para cualquier nodo distinto de los nodos origen o destino el número de arcos que entran debe ser igual al número de arcos que salen de dicho nodo.

$$\sum_{j \in N} x_{2n+1jk} = 1 \quad \forall k \in K \quad (6)$$

(6) Para el nodo origen, para cada vehículo, debe existir un arco saliente de dicho nodo.

$$\sum_{i \in N} x_{i 2n+2 k} = \sum_{j \in N} x_{2n+1 j k} \quad \forall k \in K \quad (7)$$

(7) El número de arcos que salen del nodo origen debe ser igual al número de arcos que llegan al nodo destino.

$$D_{ik} + P_{ik} \leq cap \quad \forall i \in N; k \in K \quad (8)$$

(8) La carga transportada por cada vehículo debe ser menor que la capacidad del mismo.

$$D_{2n+2 k} = 0 \quad \forall k \in K \quad (9)$$

$$D_{2n+1 k} = \sum_{i \in V} \sum_{j \in V} (x_{ijk} \cdot d_i) \quad \forall k \in K \quad (10)$$

$$P_{2n+2 k} = \sum_{i \in V} \sum_{j \in V} (x_{ijk} \cdot p_i) \quad \forall k \in K \quad (11)$$

$$P_{2n+1 k} = 0 \quad \forall k \in K \quad (12)$$

(10), (12) Establecen que cada vehículo abandone el nodo origen cargado con los productos a ser distribuidos (delivery), mientras que la carga a recoger (pick up) es nula.

(9), (11) Establecen que cada vehículo llegue al nodo destino cargado con las mercancías recogidas (pick up), mientras que la carga a entregar (delivery) será nula.

$$P_{ik} + p_j - P_{jk} \leq cap(1 - x_{ijk}) \quad \forall k \in K, i, j \in V/a_i + s_i + t_{ij} \leq b_j \quad (13)$$

$$D_{ik} - d_j - D_{jk} \leq cap(1 - x_{ijk}) \quad \forall k \in K, i, j \in V/a_i + s_i + t_{ij} \leq b_j \quad (14)$$

(13), (14) De existir el arco ij y ser este recorrido por el vehículo k, entonces la cantidad a ser entregada por ese vehículo debe decrementarse por los nodos

de delivery, mientras que la cantidad recogida debe incrementarse con los nodos de pick up.

$$T_{ik} + s_i + t_{ij} - T_{jk} \leq M(1 - x_{ijk})$$

$$\forall k \in K; i, j \in P/a_i + s_i + t_{ij} \leq b_j \quad (15)$$

(15) Restricciones de tiempo para eliminación de subtours. Expresan que, de existir el arco ij empleando el vehículo k, el tiempo de comienzo de servicio en el nodo j debe ser como mínimo el tiempo de comienzo de servicio en el nodo i, más el tiempo de servicio en el nodo i, más el tiempo empleado en ir de i a j. En caso de no existir el arco ij empleando el vehículo k, se tiene una cota superior para los tiempos de comienzo de servicio.

$$a_i \leq T_{ik} \leq b_i \quad \forall i \in V, k \in K \quad (16_1 \text{ y } 16_2)$$

(16_2 y 16_2) Cotas superior e inferior para los tiempos de comienzo de servicio, que vienen dadas por las ventanas de tiempo de cada nodo.

$$\sum_{j \in V} x_{ijk} - \sum_{j \in V} x_{j_{n+i}k} = 0 \quad \forall I \in N; k \in K \quad (17)$$

(17) Hemos añadido esta restricción para garantizar que cada par de nodos pick up – delivery sea atendido por el mismo vehículo, ya que si no, no quedaba definido implícitamente en el artículo.

$$T_{ik} + s_i + t_{i_{n+i}} \leq T_{n+i k} \quad \forall k \in K, i \in N/i \leq n \quad (18)$$

(18) El nodo de pick up debe ir siempre antes que su correspondiente nodo delivery en la ruta. Tampoco viene formulada en el artículo, la hemos añadido para garantizar que esto se cumpla, al igual que en el caso de la restricción anterior.

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in V/a_i + s_i + t_{ij} \leq b_j$$

$$y_i \geq 0 \quad \forall i \in V$$

$$T_i \geq 0 \quad \forall i \in V$$

8.3. Modelos planteados. Caso heterogéneo.

Para el problema CVRPPDTW heterogéneo, todas las formulaciones que se han seleccionado son extensiones de las del caso homogéneo, y sólo ha sido necesario adaptar algunas restricciones para las distintas capacidades de cada vehículo.

a) Modelo de Paolo Toth y Daniele Vigo.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- L_{ik} : variable entera que nos indica la carga transportada por el vehículo k al salir del nodo i .
- T_{ik} : variable de tiempo que nos va a indicar el momento de comienzo de servicio en el nodo i , atendido por el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).

- $W=\{2n+1, 2n+2\}$: el nodo $2n+1$ representa el nodo de origen. El nodo $2n+2$ representa el nodo destino.
- $N= P \cup D \cup W$.
- $K=\{1\dots k\}$: conjunto del número de vehículos a emplear.

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{k \in K} \sum_{\substack{j \in N \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{ijk} = 1 \quad \forall i \in P \quad (2)$$

$$\sum_{j \in N} x_{ijk} - \sum_{j \in N} x_{j n+i k} = 0 \quad \forall i \in P; k \in K \quad (3)$$

$$\sum_{\substack{j \in N \\ j \neq \text{origen} \\ j \neq \text{destino}}} x_{\text{origen } j k} = 1 \quad \forall k \in K \quad (4)$$

$$\sum_{i \in N} x_{ijk} - \sum_{i \in N} x_{jik} = 0 \quad \forall j \neq \text{origen}; j \neq \text{destino}; k \in K \quad (5)$$

$$\sum_{\substack{i \in N \\ i \neq \text{destino} \\ i \neq \text{origen}}} x_{i \text{ destino } k} = 1 \quad \forall k \in K \quad (6)$$

$$T_{ik} + s_i + t_{ij} - T_{jk} \leq M(1 - x_{ijk}) \\ \forall k \in K; i, j \in N / i \neq \text{destino}; a_i + s_i + t_{ij} \leq b_j \quad (7)$$

$$a_i \leq T_{ik} \leq b_i \quad \forall i \in N; k \in K \quad (8_1 \text{ y } 8_2)$$

$$T_{ik} + t_{i+n} + s_i \leq T_{i+n k} \quad \forall i \in P; k \in K \quad (9)$$

$$\begin{aligned} L_{ik} + d_j - L_{jk} &\leq cap_k(1 - x_{ijk}) \\ \forall k \in K; i, j \in N/i \neq destino; a_i + s_i + t_{ij} &\leq b_j \end{aligned} \quad (10)$$

$$d_i \leq L_{ik} \leq cap_k \quad \forall i \in P; k \in K \quad (11_1 \text{ y } 11_2)$$

$$0 \leq L_{ik} \leq cap_k \quad \forall i \in D; k \in K \quad (12_1 \text{ y } 12_2)$$

$$L_{origen k} = 0; L_{destino k} = 0 \quad \forall k \in K \quad (13_1 \text{ y } 13_2)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$L_{ik} \geq 0 \quad \forall k \in K; i \in N$$

$$D_{ik} \geq 0 \quad \forall k \in K; i \in N$$

b) Modelo de M.W.P.Savelsbergh y M.Sol.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- z_{ik} : variable binaria que de valer 1 nos va a indicar que el nodo i va a ser atendido por el vehículo k .
- D_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en cada uno de los nodos.
- y_i : variable entera que especifica la carga del vehículo correspondiente cuando sale del nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $N^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $N^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $N = N^+ \cup N^-$.
- $M = \{1 \dots K\}$, con K el número de vehículos empleados.
- $M^+ = \{2n+1 \dots 2n+K\}$: el nodo $2n + i$, con $i \leq K$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+K+1 \dots 2n+2K\}$: el nodo $2n + i$, con $K < i \leq 2K$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.

Formulación:

$$\text{Min} \quad \sum_{i \in N \cup W} \sum_{j \in N \cup W} \sum_{k \in M} x_{ijk} d_{ij}$$

$$\sum_{k=1}^K z_{ik} = 1 \quad \forall i \in N \quad (1)$$

$$z_{ik} = z_{i+n k} \quad \forall i \in N^+; k \in M \quad (1_1)$$

$$\sum_{j \in N \cup M_k^-} x_{ijk} = z_{ik} \quad \forall i \in N; k \in M \quad (2_1)$$

$$\sum_{j \in N \cup M_k^+} x_{jik} = z_{ik} \quad \forall i \in N; k \in M \quad (2_2)$$

$$\sum_{j \in N} x_{2n+kjk} = 1 \quad \forall k \in M \quad (3)$$

$$\sum_{i \in N} x_{i2n+k+k} = 1 \quad \forall k \in M \quad (4)$$

$$D_{2n+k} = 0 \quad \forall k \in M \quad (5)$$

$$D_i + t_{i+n} \leq D_{i+n} \quad \forall i \in N^+ \quad (6)$$

$$D_i + t_{ij} + s_i - D_j \leq M(1 - x_{ijk})$$

$$\forall k \in M; i \in N \cup M^+; j \in N \cup W / a_i + s_i + t_{ij} \leq b_j \quad (7)$$

$$a_i \leq D_i \leq b_i \quad \forall i \in N^+ \quad (7_1 \text{ y } 7_2)$$

$$y_{2n+k} = 0 \quad \forall k \in M \quad (8_1)$$

$$y_{2n+k+k} = 0 \quad \forall k \in M \quad (8_2)$$

$$y_i \leq \sum_{k \in M} (cap_k \cdot z_{ik}) \quad \forall i \in N \quad (9)$$

$$y_i + d_j - y_j \leq cap_k(1 - x_{ijk})$$

$$\forall k \in M; i \in N \cup M^+; j \in N \cup W / a_i + s_i + t_{ij} \leq b_j \quad (10)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in M; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$z_{ik} \in \{0,1\} \quad \forall i \in N \cup W; k \in M$$

$$D_i \geq 0 \quad \forall i \in N$$

$$y_i \geq 0 \quad \forall i \in N$$

c) Modelo basado en redes multiproducto.

Conjuntos:

- $P=\{1\dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $D=\{n+1\dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $W=\{2n+1, 2n+2\}$: el nodo $2n+1$ representa el nodo de origen. El nodo $2n+2$ representa el nodo destino.
- $N= P \cup D \cup W$.
- $K=\{1\dots k\}$: conjunto del número de vehículos a emplear.

VARIABLES:

- x_{ijk} : variable binaria para indicar si el arco ij se recorre con el vehículo k .
- y_{ijkp} : variable entera para indicar la cantidad del producto p transportada del nodo i al nodo j con el vehículo k .
- D_{ik} : variable que indica el tiempo de comienzo de servicio en el nodo i con el vehículo k .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_{ip} : cantidad demandada en el nodo i del producto p . Para los nodos de pick up será una cantidad y para los nodos de delivery negativa (por convenio).

La idea es que cada nodo de pick up tenga oferta para un único producto, que coincida con el producto demandado por su nodo correspondiente de delivery (es por ello que el número de productos es n).

- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).

- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado empleada para quitarnos de en medio aquellas d_{ip} que no nos sean útiles debido a que no hayamos asignado al nodo i el producto p .

Formulación:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{j \in N} \sum_{k \in K} x_{j \text{ origen } k} = 0 \quad (0_4)$$

$$\sum_{j \in N} \sum_{k \in K} x_{\text{destino } j k} = 0 \quad (0_5)$$

$$\sum_{j \in N} \sum_{k \in K} x_{ijk} = 1 \quad \forall i \in P \cup D \quad (2)$$

$$\sum_{\substack{j \in N \\ j \neq i}} x_{ijk} - \sum_{\substack{j \in N \\ j \neq i}} x_{jn+ik} = 0 \quad \forall i \in P; k \in K \quad (3)$$

$$\sum_{j \in P} x_{\text{origen } j k} = 1 \quad \forall k = 1..K \quad (4)$$

$$\sum_{\substack{i \in N \\ i \neq j}} x_{ijk} - \sum_{\substack{i \in N \\ i \neq j}} x_{jik} = 0 \quad \forall j \in P \cup D; k \in K \quad (5)$$

$$\sum_{j \in D} x_j \text{ destino } k = 1 \quad \forall k \in K \quad (6)$$

$$\sum_{j \in N} \sum_{k \in K} y_{ijkp} - \sum_{j \in N} \sum_{k \in K} y_{jikp} = d_{ip} \quad \forall i \in N; p = 1..n/d_{ip} \neq M \quad (7)$$

$$\sum_{p=1}^n y_{ijkp} \leq x_{ijk} \cdot \text{cap}_k \quad \forall i, j \in N; k \in K \quad (8)$$

$$\sum_{j \in N} \sum_{k \in K} y_{jikp} = \sum_{j \in N} \sum_{k \in K} y_{ijkp} \quad \forall i \in N; p = 1..n/d_{ip} = M \quad (9)$$

$$D_{\text{origen } k} = 0 \quad \forall k \in K \quad (10)$$

$$D_{ik} + t_{ij} + s_i - D_{jk} \leq M(1 - x_{ijk}) \quad \forall i \neq \text{destino}; i \neq j; k \in K \quad (11)$$

$$D_{ik} + t_{i+n} + s_i \leq D_{i+nk} \quad \forall i \in P; k \in K \quad (12)$$

$$a_i \leq D_{ik} \leq b_i \quad \forall i \in N; k \in K \quad (13_1 \text{ y } 13_2)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$D_{ik} \geq 0 \quad \forall i \in N; k \in K$$

$$y_{ijkp} \geq 0 \quad \forall i, j \in N; k \in K; p = 1..n$$

d) Modelo de Yvan Dumas, Jacques Desrosiers y François Soumis.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .

- y_i : variable entera que nos indica la carga del vehículo correspondiente al salir del nodo i .
- T_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en el nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_v : capacidad de cada vehículo.
- d_i : cantidad demandada en el nodo i . Para los nodos de pick up será una cantidad positiva y para los nodos de delivery negativa (por convenio).
- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $P^+ = \{1 \dots n\}$: representa a todos los nodos de pick up (d_i positivo).
- $P^- = \{n+1 \dots 2n\}$: representa a todos los nodos de delivery (d_i negativo).
- $P = P^+ \cup P^-$.
- $M = \{1 \dots V\}$, con V el número de vehículos empleados.
- $M^+ = \{2n+1 \dots 2n+V\}$: el nodo $2n+i$, con $i \leq V$, representa el nodo de origen para el vehículo i .
- $M^- = \{2n+V+1 \dots 2n+2V\}$: el nodo $2n+i$, con $V < i \leq 2V$, representa el nodo de destino para el vehículo i .
- $W = M^+ \cup M^-$.⁷³
- $N = P \cup W$.

Formulación:

⁷³ En el artículo denota al nodo origen como nodo 0 y al nodo destino como nodo 2^*n+1 . Nosotros hemos definido 2^*n+v orígenes y 2^*n+V+v destinos. Esto es debido a que si no las restricciones de tiempo dan problemas.

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{v \in M} x_{ijv} d_{ij}$$

$$\sum_{v \in M} \sum_{j \in P \cup M_v^-} x_{ijv} = 1 \quad \forall i \in P^+ \quad (6)$$

$$\sum_{j \in P \cup M_v^-} x_{ijv} - \sum_{j \in P \cup M_v^+} x_{jiv} = 0 \quad \forall i \in P; v \in M \quad (7)$$

$$\sum_{j \in P^+} x_{M_v^+ j v} = 1 \quad \forall v \in M \quad (8)$$

$$\sum_{i \in P^-} x_{i M_v^- v} = 1 \quad \forall v \in M \quad (9)$$

$$\sum_{j \in N} x_{ijv} - \sum_{j \in N} x_{j n+i v} = 0 \quad \forall i \in P^+; v \in M \quad (10)$$

$$T_i + t_{i+n} + s_i \leq T_{i+n} \quad \forall i \in P^+ \quad (11)$$

$$\begin{aligned} T_i + s_i + t_{ij} - T_j &\leq M(1 - x_{ijv}) \\ \forall v \in M; i, j \in P / a_i + s_i + t_{ij} &\leq b_j \end{aligned} \quad (12)$$

$$T_{M_v^+} + t_{M_v^+ j} - T_j \leq M(1 - x_{M_v^+ jv}) \quad \forall v \in M; j \in P^+ \quad (13)$$

$$T_i + s_i + t_{i M_v^-} - T_{M_v^-} \leq M(1 - x_{i M_v^- v}) \quad \forall v \in M; i \in P^- \quad (14)$$

$$a_i \leq T_i \leq b_i \quad \forall i \in P \quad (15_1 \text{ y } 15_2)$$

$$a_{M_v^+} \leq T_{M_v^+} \leq b_{M_v^+} \quad \forall v \in M \quad (16_1 \text{ y } 16_2)$$

$$a_{M_v^-} \leq T_{M_v^-} \leq b_{M_v^-} \quad \forall v \in M \quad (17_1 \text{ y } 17_2)$$

$$\begin{aligned} & y_i + d_j - y_j \leq cap_v(1 - x_{ijv}) \\ \forall v \in M; i \leq 2n + V; j \in N/a_i + s_i + t_{ij} \leq b_j \end{aligned} \quad (18)$$

$$d_i \leq y_i \leq cap_v \quad \forall i \in P^+ \quad (21_1 \text{ y } 21_2)$$

$$y_{M_v^+} = 0; y_{M_v^-} = 0 \quad \forall v \in M \quad (22_1 \text{ y } 22_2)$$

$$x_{ijv} \in \{0,1\} \quad \forall v \in M; i, j \in N/a_i + s_i + t_{ij} \leq b_j$$

$$y_i \geq 0 \quad \forall i \in N$$

$$T_i \geq 0 \quad \forall i \in N$$

e) Modelo de Enrico Angelelli y Renata Mansini.

Variables:

- x_{ijk} : variable binaria que nos indica si el arco ij se recorre empleando el vehículo k .
- D_{ik} : variable entera que nos indica la cantidad restante de entregas (deliveries) transportadas por el vehículo k cuando parte del nodo i .
- P_{ik} : variable entera que nos indica la cantidad de recogidas (pick ups) que son transportadas por el vehículo k cuando parte del nodo i .
- T_i : variable de tiempo que nos va a indicar el momento de comienzo del servicio en el nodo i .

Parámetros:

- d_{ij} : distancia entre el nodo i y el nodo j .
- cap_k : capacidad de cada vehículo.
- d_i : requerimiento de entrega (delivery) del nodo i .
- p_i : requerimiento de recogida (pick up) del nodo i .

- t_{ij} : tiempo que tardamos en ir del nodo i al nodo j . Se define como el cociente de la distancia entre el par de nodos ij y la velocidad con la que se realiza el trayecto (tomamos, por ejemplo, una velocidad de referencia de 90 km/h).
- s_i : tiempo de servicio en el nodo i .
- a_i : fecha más temprana de comienzo del servicio en el nodo i .
- b_i : fecha más tardía de comienzo del servicio en el nodo i .
- M : constante de valor elevado para servirnos de cota superior en determinadas restricciones.

Conjuntos:

- $N=\{1...2n\}$: representa a todos los nodos. De 1 a n son los nodos de pick up y de n a $2n$ los nodos de delivery. No se incluyen ni el nodo de origen ni el nodo de destino
- $V=N \cup \{2n+1, 2n+2\}$: representa al conjunto de todos los nodos, incluyendo además el nodo de origen y el nodo de destino.
- $K=\{1...K\}$, con K el número de vehículos empleados.

Formulación:

$$\text{Min} \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} d_{ij}$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1 \quad \forall i \in N \quad (4)$$

$$\sum_{i \in V} x_{ijk} = \sum_{i \in V} x_{jik} \quad \forall j \in N; k \in K \quad (5)$$

$$\sum_{j \in N} x_{2n+1jk} = 1 \quad \forall k \in K \quad (6)$$

$$\sum_{i \in N} x_{i2n+2k} = \sum_{j \in N} x_{2n+1jk} \quad \forall k \in K \quad (7)$$

$$D_{ik} + P_{ik} \leq cap_k \quad \forall i \in N; k \in K \quad (8)$$

$$D_{2n+2 \ k} = 0 \quad \forall k \in K \quad (9)$$

$$D_{2n+1 \ k} = \sum_{i \in V} \sum_{j \in V} (x_{ijk} \cdot d_i) \quad \forall k \in K \quad (10)$$

$$P_{2n+2 \ k} = \sum_{i \in V} \sum_{j \in V} (x_{ijk} \cdot p_i) \quad \forall k \in K \quad (11)$$

$$P_{2n+1 \ k} = 0 \quad \forall k \in K \quad (12)$$

$$P_{ik} + p_j - P_{jk} \leq cap_k(1 - x_{ijk}) \quad \forall k \in K, i, j \in V/a_i + s_i + t_{ij} \leq b_j \quad (13)$$

$$D_{ik} - d_j - D_{jk} \leq cap_k(1 - x_{ijk}) \quad \forall k \in K, i, j \in V/a_i + s_i + t_{ij} \leq b_j \quad (14)$$

$$T_{ik} + s_i + t_{ij} - T_{jk} \leq M(1 - x_{ijk}) \\ \forall k \in K; i, j \in P/a_i + s_i + t_{ij} \leq b_j \quad (15)$$

$$a_i \leq T_{ik} \leq b_i \quad \forall i \in V, k \in K \quad (16_1 \text{ y } 16_2)$$

$$\sum_{j \in V} x_{ijk} - \sum_{j \in V} x_{j \ n+i \ k} = 0 \quad \forall i \in N; k \in K \quad (17)$$

$$T_{ik} + s_i + t_{i \ n+i} \leq T_{n+i \ k} \quad \forall k \in K, i \in N/i \leq n \quad (18)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in K; i, j \in V/a_i + s_i + t_{ij} \leq b_j$$

$$y_i \geq 0 \quad \forall i \in V$$

$$T_i \geq 0 \quad \forall i \in V$$

8.4. Datos empleados.

Como ya nos ocurría en el Capítulo 7, no hemos sido capaces de encontrar una librería de la que extraer ficheros de datos en el formato deseado para el problema CVRPPDTW. En su lugar, se han empleado los datos del problema CVRPPD (ver apartado 7.4), añadiendo dos columnas correspondientes a las ventanas de tiempo de cada uno de los nodos.⁷⁴ Las demandas para los nodos delivery serán modificadas en un pretratamiento dentro de la programación para hacer que sean iguales a las de su nodo pick up cambiadas de signo, tal y como ocurría en el capítulo anterior.

Para el caso heterogéneo, se han empleado los mismos ficheros de datos, asignando las distintas capacidades de cada uno de los vehículos a mano.

Por último, cabe destacar que al contrario de lo que ocurría en el capítulo anterior, sólo se ha generado una estructura (tipo) de fichero de datos:

- Nodos pick up: del nodo 1 al nodo n , incluidos.
- Nodos delivery: del nodo $n+1$ al nodo $2n$, incluidos.
- Nodos origen /destino: dentro de cada uno de los distintos ejemplos propuestos, se define un nodo depot (con sus coordenadas y ventanas de tiempo específicas), denominado nodo 0, el cuál mediante un tratamiento en la programación se transforma en el número de nodos depot deseados, adecuándose a las requisiciones del modelo en el que nos encontremos⁷⁵:

La nomenclatura de cada uno de los ficheros de datos empleados es la siguiente: PDTW7_homog, PDTW9_homog, PDTW10_homog, PDTW20_homog, PDTW30_homog, PDTW40_homog, PDTW50_homog, PDTW60_homog, PDTW70_homog, PDTW80_homog, PDTW90_homog, PDTW100_homog para

⁷⁴ Para la generación de las ventanas de tiempo, se ha empleado un programa que hemos desarrollado en Xpress-IVE. Dicho programa se encarga de generar números aleatorios entre determinados valores para obtener dichas ventanas, garantizando en todo momento que cada par de nodos pick up – delivery pueda ser atendido en la misma ruta sin que estas interfieran en ello (por ejemplo, si un nodo pick up tiene una ventana de tiempo [10, 20] y su nodo delivery de [1, 5] tendríamos un problema infactible, ya que no podríamos atender al nodo pick up antes del nodo delivery con el mismo vehículo lo cual, como ya hemos visto, es imprescindible).

⁷⁵ Hemos visto que algunos modelos, como el de Dumas et al., necesitan para su formulación la existencia de más de un nodo origen/destino. Este pretratamiento en la programación nos permite prescindir de dos estructuras de ficheros de datos (como ocurría en el problema CVRPPD), garantizando una flexibilidad adecuada de los modelos.

el caso homogéneo y PDTW7_heterog, PDTW9_heterog, PDTW10_heterog, PDTW20_heterog, PDTW30_heterog, PDTW40_heterog, PDTW50_heterog, PDTW60_heterog, PDTW70_heterog, PDTW80_heterog, PDTW90_heterog, PDTW100_heterog para el caso heterogéneo.

8.5. Resultados obtenidos empleando Xpress-IVE.

Al igual que en todos los casos anteriores, se ha establecido un tiempo de ejecución máximo de 200 segundos, deteniéndose la optimización cuando llegamos a ese punto y quedándonos con la mejor solución factible obtenida hasta ese momento.

8.5.1. Caso homogéneo.

Los resultados que hemos obtenidos han sido:

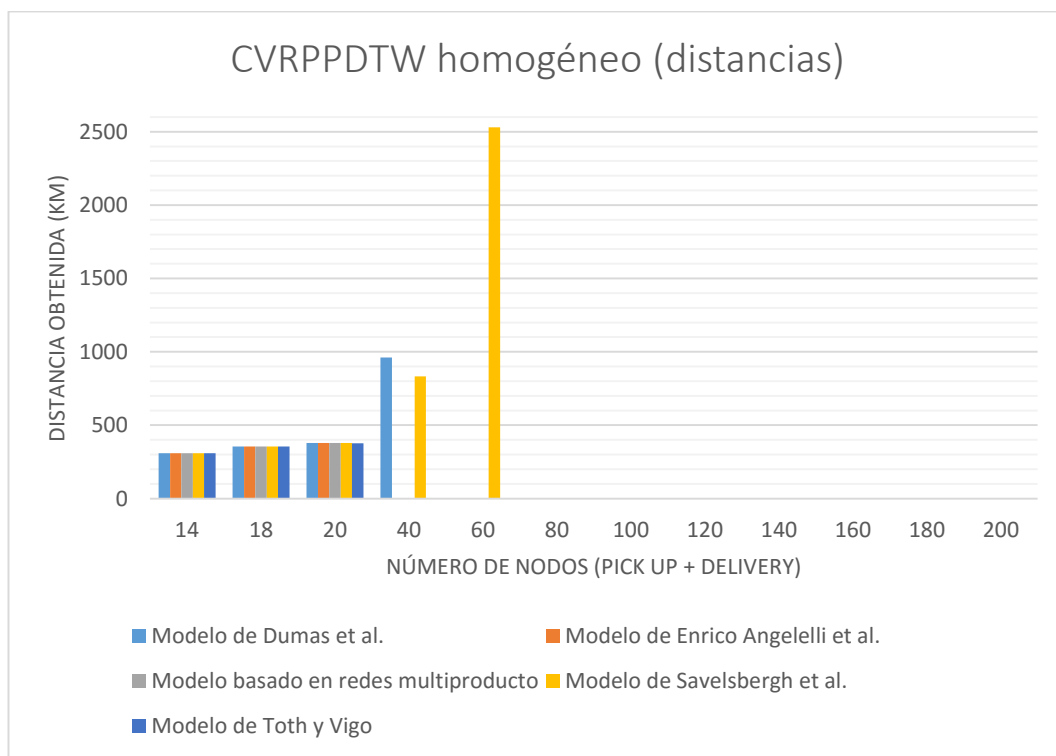


Figura 8.3: Distancias de Xpress- IVE para el problema CVRPPDTW homogéneo.

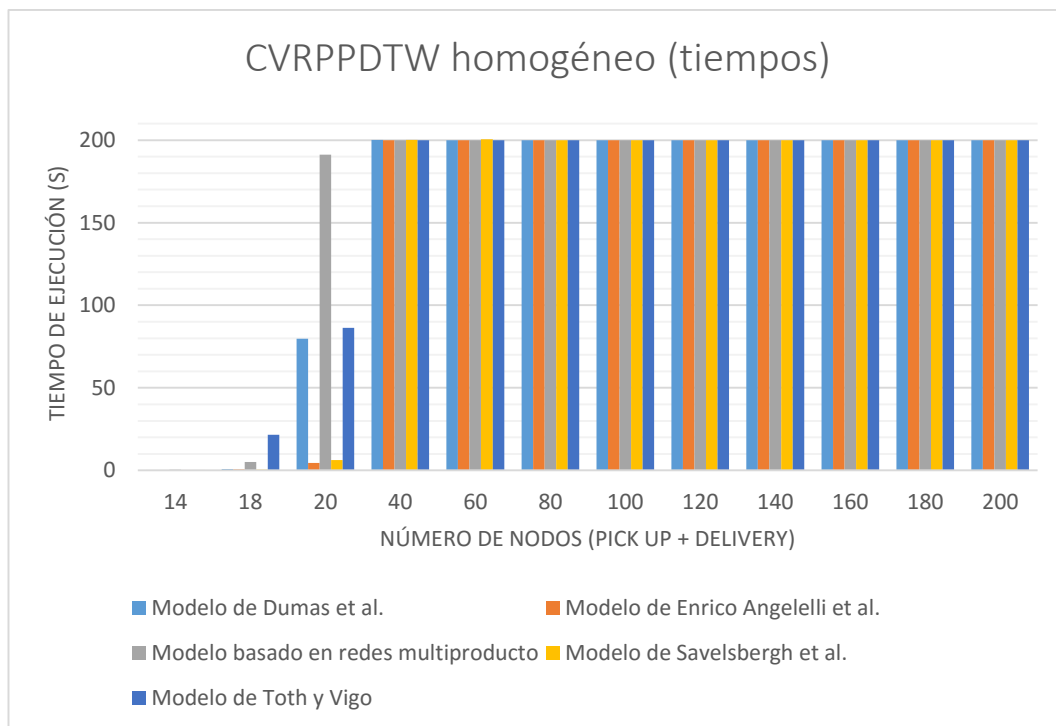


Figura 8.4: Tiempos de ejecución obtenidos con Xpress-IVE para el problema CVRPPDTW homogéneo.

8.5.2. Resumen de resultados obtenidos. Caso homogéneo.

Según podemos observar en la Figura 8.2 y en la Tabla 8.2, y tal y como ocurría para el problema CVRPPD, el número de problemas resueltos ha decaído considerablemente con respecto a los problemas CVRP o CVRPPDTW, debido en gran medida a que estos problemas (CVRPPD y CVRPPDTW) tienen una carga computacional mucho más elevada que los anteriores, dificultando que se pueda encontrar una solución factible dentro del tiempo de ejecución máximo estipulado cuando el número de nodos aumenta.

Centrándonos en los resultados, vemos cómo todos los modelos han logrado encontrar una solución factible para los problemas PDTW7_homog, PDTW9_homog y PDTW10_homog (de 14, 18 y 20 nodos de pick up + delivery, respectivamente). Sin embargo, a partir de este punto, sólo los modelos de Dumas et al. (d) y Savelsbergh et al. (b) logran resolver el problema PDTW20_homog, en tanto que sólo el modelo (b) consigue una solución factible para el problema PDTW30_homog.

Si nos fijamos en las distancias obtenidas, vemos que todos los modelos obtienen prácticamente los mismos valores para los tres primeros problemas (si bien es cierto que el modelo basado en redes multiproducto (c) obtiene una

distancia 2 km mayor que el resto de modelos para el problema PDTW10_homog, se considera esta diferencia ínfima), en tanto que para el problema PDTW20_homog, el modelo (b) obtiene mejores resultados que el modelo (d).

Analizando los tiempos de ejecución para los problemas en que todos los modelos logran encontrar solución factible, se aprecia que los modelos (b), (d) y el modelo de Enrico Angelelli et al. (e) obtienen los mejores resultados.

Por lo tanto, y en base a los resultados anteriormente mostrados, se concluye que los modelos (b) y (d) por lograr resolver el mayor número de problemas (obteniendo resultados muy buenos) y el modelo (e) por poseer unos tiempos de ejecución muy buenos se implementarán en CPLEX para una comparación más detallada de los mismos.

En la Tabla 8.1 se puede observar el resumen de resultados obtenidos según los criterios de comparación. Como siempre, los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados, de manera que si todos los modelos obtienen la mejor solución, se cuenta igualmente ya que han llegado a la mejor solución⁷⁶.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Toth y Vigo	3 de 12	0 de 12	3 de 12	Descartado
Savelsbergh et al.	5 de 12	2 de 12	5 de 12	CPLEX
Redes multiproducto	2 de 12	0 de 12	3 de 12	Descartado
Dumas et al.	3 de 12	0 de 12	4 de 12	CPLEX
Enrico Angelelli et al.	3 de 12	1 de 12	3 de 12	CPLEX

Tabla 8.1: Resumen de resultados para el problema CVRPPDTW homogéneo empleando Xpress-IVE.

⁷⁶ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Datos empleados		Modelo de Dumas et al.		Modelo de Enrico Angelelli et al.		Modelo basado en redes multiproducto		Modelo de Savelbergh et al.		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PDTW7_homog	7	308,083	0,134	308,083	0,173	308,083	0,388	308,083	0,115	308,083	0,272
PDTW9_homog	9	354,532	0,917	354,532	0,649	354,807	5,031	354,532	0,546	354,532	21,622
PDTW10_homog	10	377,813	79,697	377,813	4,593	379,769	191,132	377,813	6,306	377,213	86,321
PDTW20_homog	20	961,197	200,153	#N/A	200	#N/A	200	831,543	200,064	#N/A	200
PDTW30_homog	30	#N/A	200	#N/A	200	#N/A	200	2529,3	200,545	#N/A	200
PDTW40_homog	40	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW50_homog	50	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW60_homog	60	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW70_homog	70	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW80_homog	80	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW90_homog	90	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW100_homog	100	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200

Tabla 8.2: Resultados de Xpress-IVE para el problema CVRPPDTW homogéneo.

8.5.3. Caso heterogéneo.

Para el caos heterogéneo, los resultados obtenidos han sido:

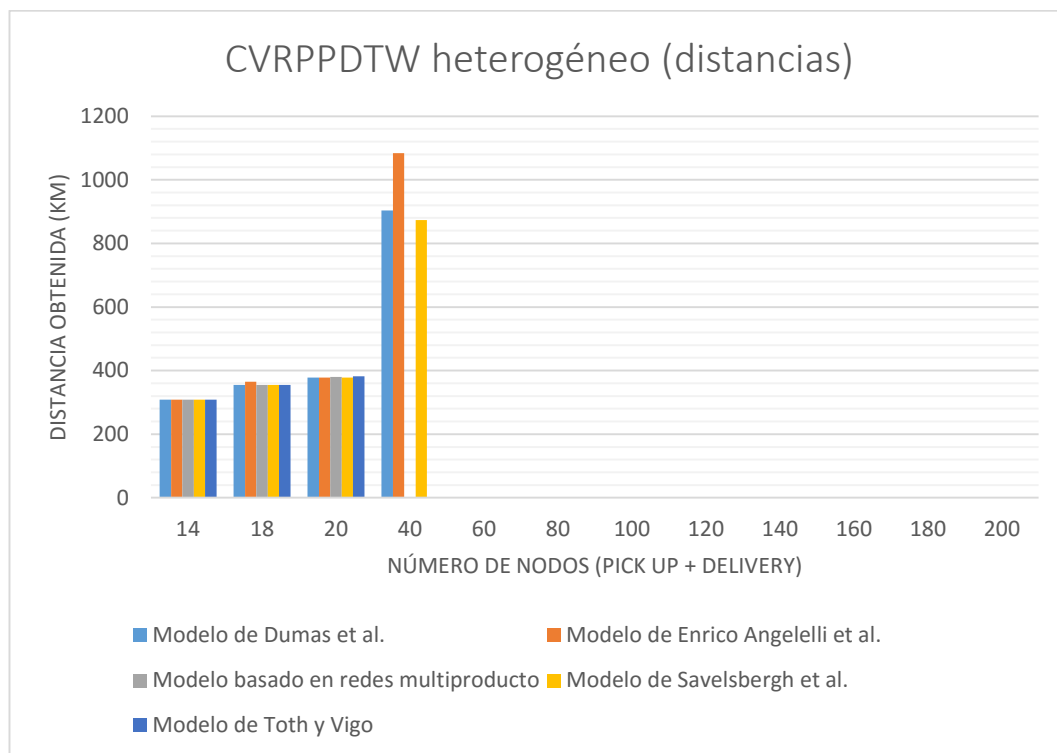


Figura 8.5: Distancias Xpress-IVE. Problema CVRPPDTW heterog.

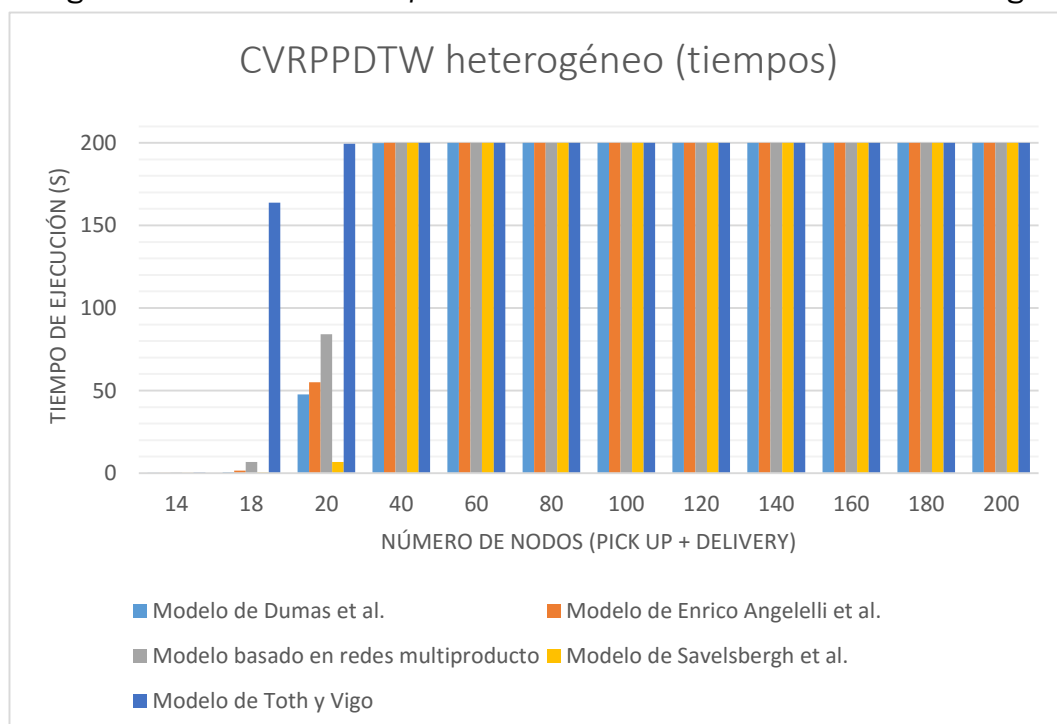


Figura 8.6: Tiempos de ejecución Xpress-IVE. CVRPPDTW heterog.

Datos empleados		Modelo de Dumas et al.		Modelo de Enrico Angelelli et al.		Modelo basado en redes multiproducto		Modelo de Savelsbergh et al.		Modelo de Toth y Vigo	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PDTW7_heterog	7	308,083	0,201	308,083	0,204	308,083	0,44	308,083	0,096	308,083	0,437
PDTW9_heterog	9	354,532	0,576	364,531	1,564	354,807	6,845	354,532	0,422	354,532	163,837
PDTW10_heterog	10	377,813	47,7	377,813	54,945	379,769	84,025	377,813	6,872	382,233	199,427
PDTW20_heterog	20	903,556	199,741	1083,65	200,082	#N/A	200	873,851	199,948	#N/A	200
PDTW30_heterog	30	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW40_heterog	40	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW50_heterog	50	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW60_heterog	60	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW70_heterog	70	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW80_heterog	80	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW90_heterog	90	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200
PDTW100_heterog	100	#N/A	200	#N/A	200	#N/A	200	#N/A	200	#N/A	200

Tabla 8.3: Resultados obtenidos con Xpress-IVE para el problema CVRPPDTW heterogéneo.

8.5.4. Resumen de resultados obtenidos. Caso heterogéneo.

Al igual que ocurría en el caso homogéneo, puede apreciarse que los resultados obtenidos empeoran bastante con respecto a los problemas CVRP y CVRPTW en el sentido de número de problemas resueltos. Así, todos los modelos logran resolver los tres primeros problemas (14, 18 y 20 nodos de pick up + delivery, respectivamente), en tanto que sólo los modelos de Savelsbergh et al. (b), Dumas et al. (d) y Enrico Angelelli et al. (e) resuelven el problema siguiente (40 nodos de pick up + delivery), no existiendo ningún modelo que resuelva problemas de un mayor tamaño.

En relación a las distancias obtenidas, para los tres primeros problemas, todos los modelos logran obtener prácticamente la misma distancia total recorrida (existiendo alguna pequeña fluctuación en el modelo (e) para el problema PDTW9_homog y en los modelos de Toth y Vigo (a) y redes multiproducto (c) para el problema PDTW10_homog, si bien estas fluctuaciones no suponen una variación mayor del 2% respecto del resto de soluciones). Si nos fijamos en el problema que sólo logran resolver los modelos (b), (d) y (e), el modelo (b) se obtiene los mejores resultados.

Analizando los tiempos de ejecución, y como ocurría en el caso homogéneo, los modelos (b), (d) y (e) obtienen los mejores resultados. De entre estos tres, el modelo (b) es el que logra encontrar una solución óptima en el menor tiempo posible.

Por todo ello, y al igual que ocurría en el caso homogéneo, los modelos (a) y (c) se consideran no válidos para resolver problemas CVRPPDTW heterogéneos. Por su parte los modelos (b), (d) y (e) se implementarán en CPLEX para compararlos más en profundidad.

La Tabla 8.4 muestra el resumen de resultados según los criterios de comparación. Como siempre, los criterios distancias y tiempos se representan por medio del número de veces que un modelo obtiene los mejores resultados, de manera que si todos los modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución⁷⁷.

⁷⁷ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Toth y Vigo	2 de 12	0 de 12	3 de 12	Descartado
Savelsbergh et al.	4 de 12	3 de 12	4 de 12	CPLEX
Redes multiproducto	2 de 12	0 de 12	3 de 12	Descartado
Dumas et al.	3 de 12	0 de 12	4 de 12	CPLEX
Enrico Angelelli et al.	2 de 12	0 de 12	4 de 12	CPLEX

Tabla 8.4: Resumen de resultados obtenidos con Xpress-IVE para el problema CVRPPDTW heterogéneo.

8.6. Resultados obtenidos empleando CPLEX.

Al igual que ocurría en el caso de Xpress-IVE, hemos establecido un tiempo de ejecución máximo de 200 segundos, tiempo a partir del cual el solver detiene la optimización.

8.6.1. Caso homogéneo.

Como vimos en el apartado 8.5.2, se ha decidido implementar en CPLEX los modelos propuestos para el CVRPPDTW homogéneo (b) Savelsbergh et al., (d) Dumas et al. y (e) Enrico Angelelli et al. En las Figuras 8.6 y 8.7 y la Tabla 8.5 podemos observar los resultados obtenidos.

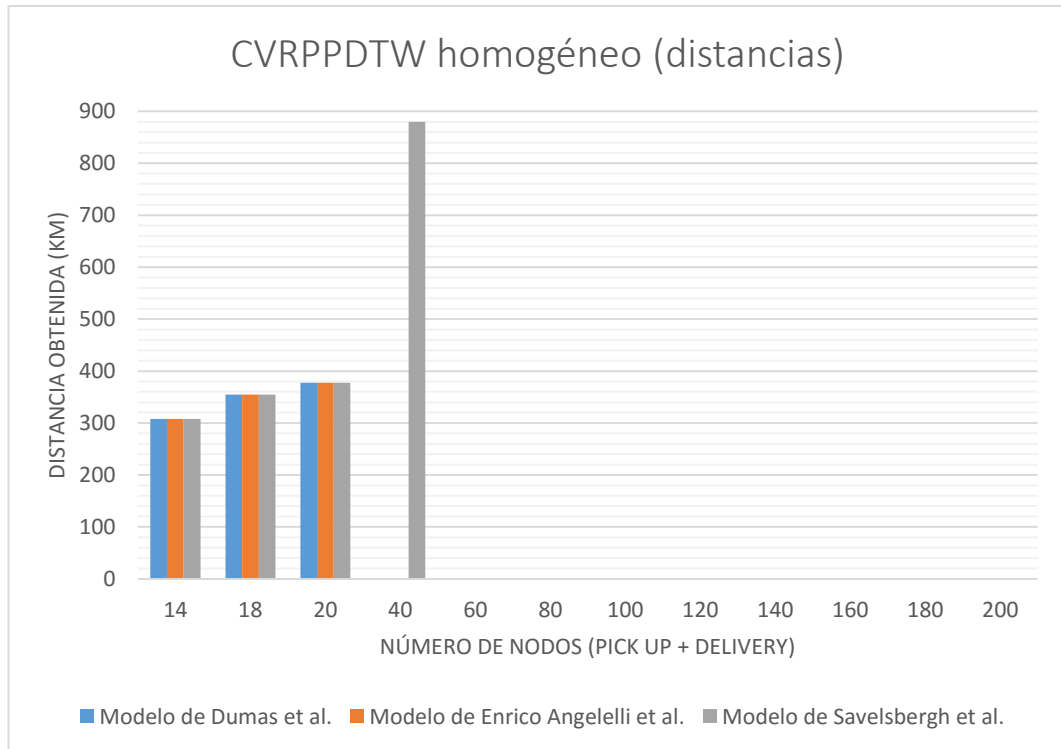


Figura 8.7: Distancias de CPLEX para el problema CVRPPDTW homogéneo.

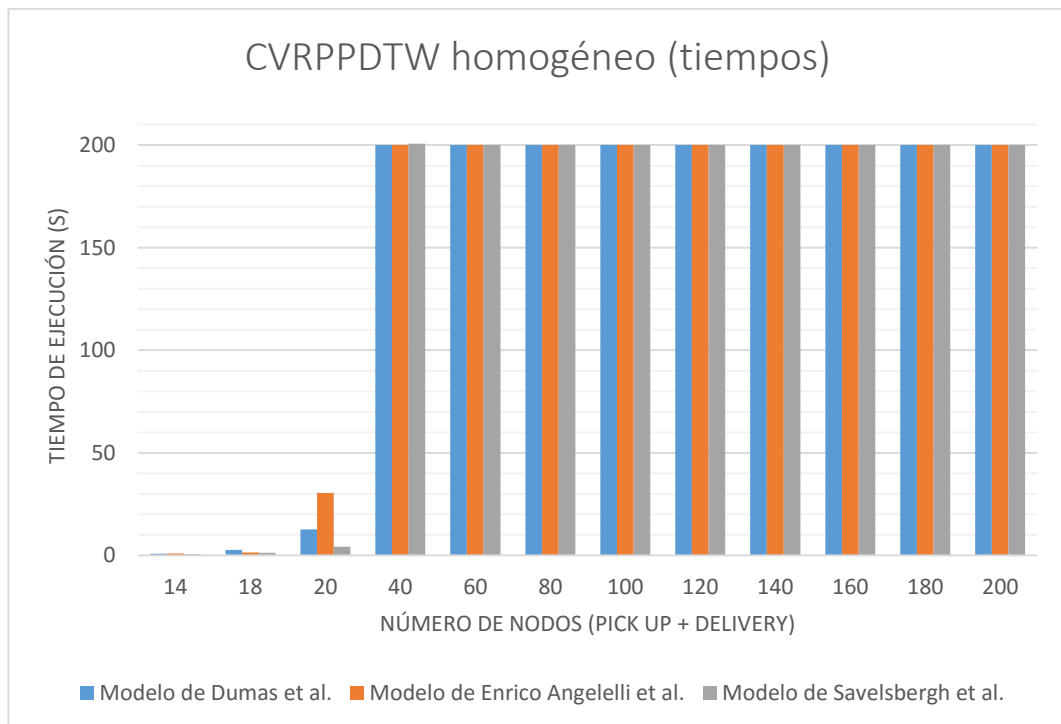


Figura 8.8: Tiempos de ejecución obtenidos con CPLEX para el problema CVRPPDTW homogéneo.

Datos empleados		Modelo de Dumas et al.		Modelo de Enrico Angelelli et al.		Modelo de Savelsbergh et al.	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PDTW7_homog	14	308,0825579	0,889	308,0825579	0,999	308,0825579	0,733
PDTW9_homog	18	354,5321836	2,761	354,5321836	1,482	354,5321836	1,326
PDTW10_homog	20	377,8134273	12,667	377,8134273	30,529	377,8134273	4,274
PDTW20_homog	40	#N/A	200	#N/A	200	879,7900954	200,664
PDTW30_homog	60	#N/A	200	#N/A	200	#N/A	200
PDTW40_homog	80	#N/A	200	#N/A	200	#N/A	200
PDTW50_homog	100	#N/A	200	#N/A	200	#N/A	200
PDTW60_homog	120	#N/A	200	#N/A	200	#N/A	200
PDTW70_homog	140	#N/A	200	#N/A	200	#N/A	200
PDTW80_homog	160	#N/A	200	#N/A	200	#N/A	200
PDTW90_homog	180	#N/A	200	#N/A	200	#N/A	200
PDTW100_homog	200	#N/A	200	#N/A	200	#N/A	200

Tabla 8.5: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW homogéneo.

8.6.2. Resumen de resultados. Caso homogéneo.

Como podemos observar, nos encontramos en una situación bastante parecida a la que acabamos de ver para los resultados empleando Xpress-IVE: para los tres primeros problemas, todos los modelos logran obtener la misma solución óptima y, además, existe un modelo (Savelsbergh et al. (b)) que consigue encontrar una solución factible para un problema adicional.

Así pues, el criterio de distancias totales obtenidas no se demuestra significativo para realizar una clasificación entre modelos, por lo que nos centraremos en los tiempos de ejecución. Según este criterio, el modelo (b) es el que mejores resultados obtiene para todos los casos tratados.

En nuestra opinión los resultados no ofrecen lugar a dudas, el modelo (b) es el más adecuado para resolver problemas CVRPPDTW homogéneos ya que, aparte de ser el que mayor número de ejemplos resuelve, presenta los mejores tiempos de ejecución y distancias en todos estos casos.

La Tabla 8.6 resume los resultados obtenidos en CPLEX según los criterios de comparación establecidos. Al igual que en casos anteriores, los criterios distancias obtenidas y tiempos de ejecución se representan por medio del número de veces que un modelo obtiene los mejores resultados (si todos los

modelos obtienen la mejor distancia, se cuenta igualmente ya que han llegado a la mejor solución)⁷⁸.

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Savelsbergh et al.	4 de 12	4 de 12	4 de 12	SELECCIONADO
Dumas et al.	3 de 12	0 de 12	3 de 12	Descartado
Enrico Angelelli et al.	3 de 12	0 de 12	3 de 12	Descartado

Tabla 8.6: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW homogéneo.

8.6.3. Caso heterogéneo.

Para el caso heterogéneo, hemos obtenido los siguientes resultados:

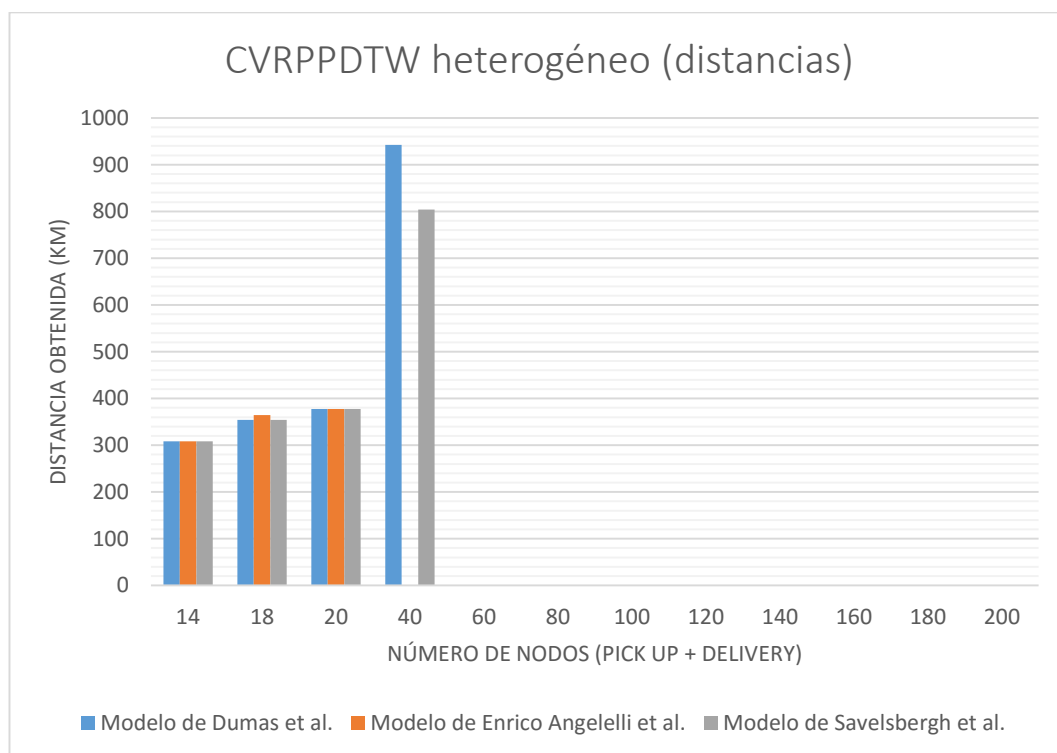


Figura 8.9: Distancias de CPLEX para el CVRPPDTW heterogéneo.

⁷⁸ Al contrario que ocurre cuando comparamos dos modelos entre sí, ya que contamos únicamente los casos en que un modelo obtiene mejores distancias que otro. Sin embargo, no se cuenta el caso en que todos los modelos lleguen al tiempo máximo de ejecución establecido.

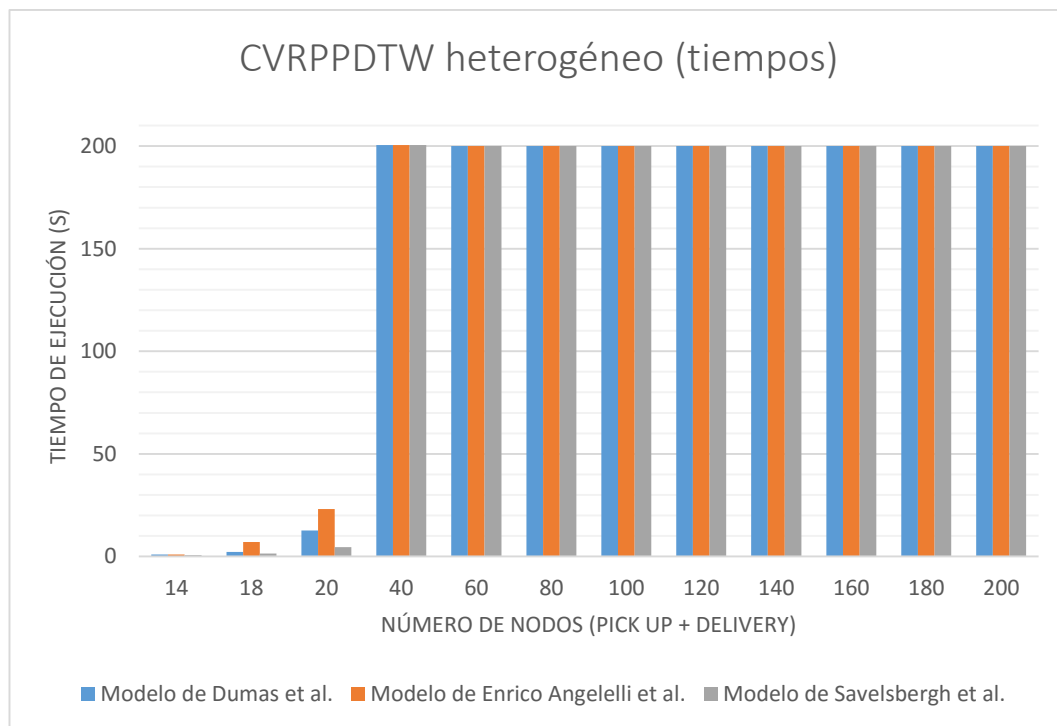


Figura 8.10: Tiempos de CPLEX para el CVRPPDTW heterogéneo.

Datos empleados		Modelo de Dumas et al.		Modelo de Enrico Angelelli et al.		Modelo de Savelsbergh et al.	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PDTW7_heterog	14	308,083	1,014	308,083	0,967	308,083	0,639
PDTW9_heterog	18	354,532	2,247	364,531	7,098	354,532	1,467
PDTW10_heterog	20	377,813	12,729	377,813	23,104	377,813	4,571
PDTW20_heterog	40	942,323	200,476	#N/A	200,476	804,025	200,46
PDTW30_heterog	60	#N/A	200	#N/A	200	#N/A	200
PDTW40_heterog	80	#N/A	200	#N/A	200	#N/A	200
PDTW50_heterog	100	#N/A	200	#N/A	200	#N/A	200
PDTW60_heterog	120	#N/A	200	#N/A	200	#N/A	200
PDTW70_heterog	140	#N/A	200	#N/A	200	#N/A	200
PDTW80_heterog	160	#N/A	200	#N/A	200	#N/A	200
PDTW90_heterog	180	#N/A	200	#N/A	200	#N/A	200
PDTW100_heterog	200	#N/A	200	#N/A	200	#N/A	200

Tabla 8.7: Resultados obtenidos con CPLEX para el CVRPPDTW heterogéneo.

8.6.3. Resumen de resultados obtenidos. Caso heterogéneo.

Nos encontramos en un caso bastante parecido al anterior (CPLEX para caso homogéneo), es decir, tenemos que todos los modelos logran resolver los tres

problemas más pequeños (14, 18 y 20 nodos de pick up + delivery, respectivamente) y que el modelo de Savelsbergh (b) logra resolver un problema extra. A mayores, aparece el modelo de Dumas et al. (d), el cual también logra resolver ese problema extra.

En lo relativo a las distancias obtenidas vemos que, para los tres primeros problemas, todos los modelos obtienen prácticamente la misma solución (salvo el modelo de Enrico Angelelli et al. (e) que difiere de la mejor solución en un 2%, aproximadamente). Para el caso PDTW20_heterog, que es resuelto por los modelos (b) y (d), tenemos que el primero obtiene un resultado menor.

Analizando los tiempos de ejecución obtenidos, vemos que el modelo (b) es el que mejores resultados presenta sin lugar a dudas, obteniendo en 3 casos los mejores tiempos, por 1 en el que llega al límite máximo de ejecución establecido.

En definitiva, dado que logra obtener soluciones factibles para un mayor número de problemas, con las mejores distancias y tiempos de ejecución, seleccionamos el modelo (b) como el más adecuado para resolver problemas CVRPPDTW heterogéneos.

La Tabla 8.8 muestra el resumen de resultados obtenidos empleando CPLEX⁷⁹:

Modelo	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Savelsbergh et al.	4 de 12	3 de 12	4 de 12	SELECCIONADO
Dumas et al.	3 de 12	0 de 12	4 de 12	Descartado
Enrico Angelelli et al	2 de 12	0 de 12	3 de 12	Descartado

Tabla 8.8: Resumen de resultados obtenidos con CPLEX para el problema CVRPPDTW heterogéneo.

8.7. Comparativa Xpress-IVE vs CPLEX para el CVRPPDTW.

8.7.1. Caso homogéneo.

⁷⁹ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

A continuación, se presentan los resultados obtenidos para los tres modelos planteados tanto en Xpress-IVE como en CPLEX para el problema CVRPPDTW homogéneo:

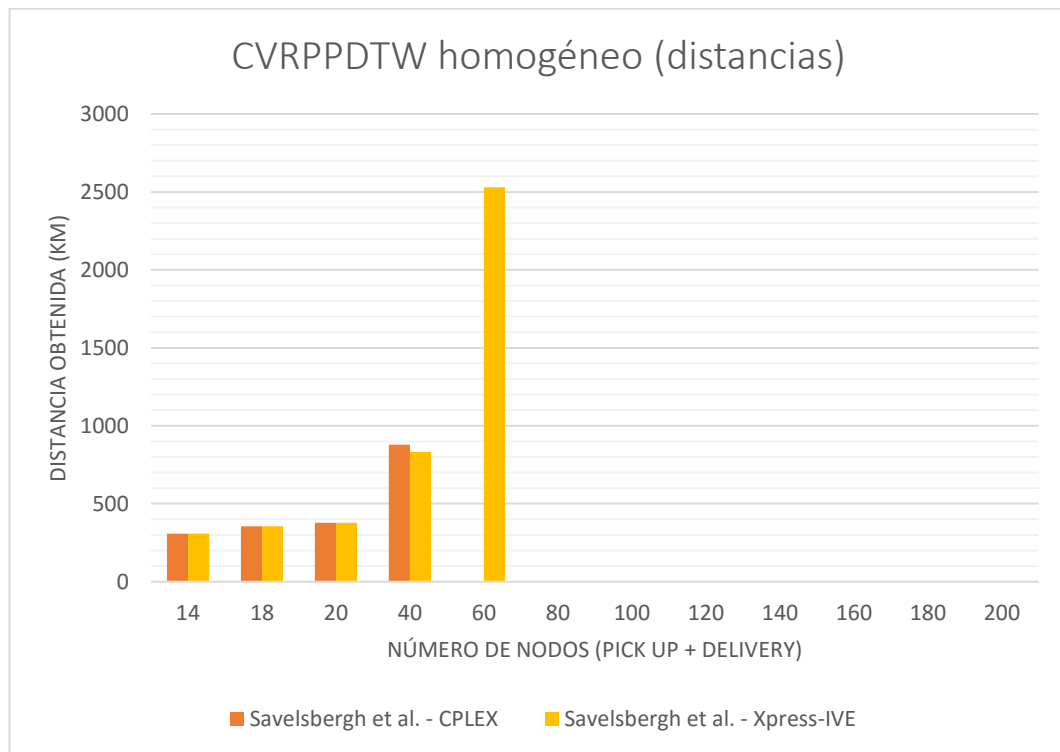


Figura 8.11: Comparativa distancias modelo de Savelsbergh et al.

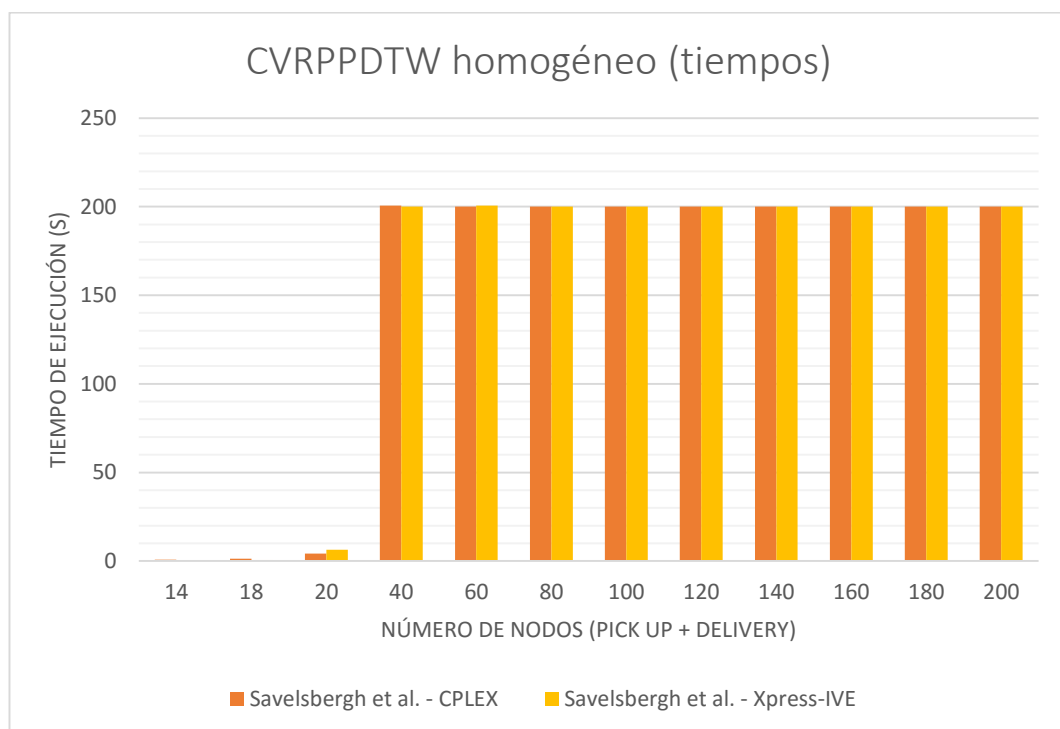


Figura 8.12: Comparativa tiempos modelo de Savelsbergh et al.

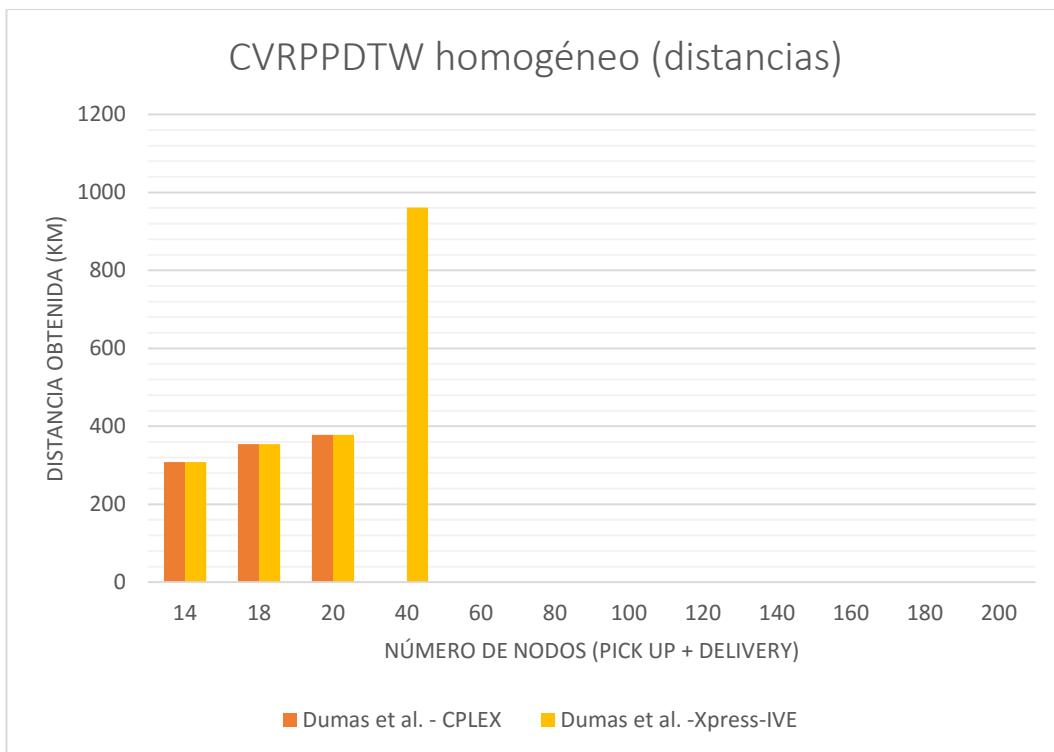


Figura 8.13: Comparativa distancias modelo de Dumas et al.

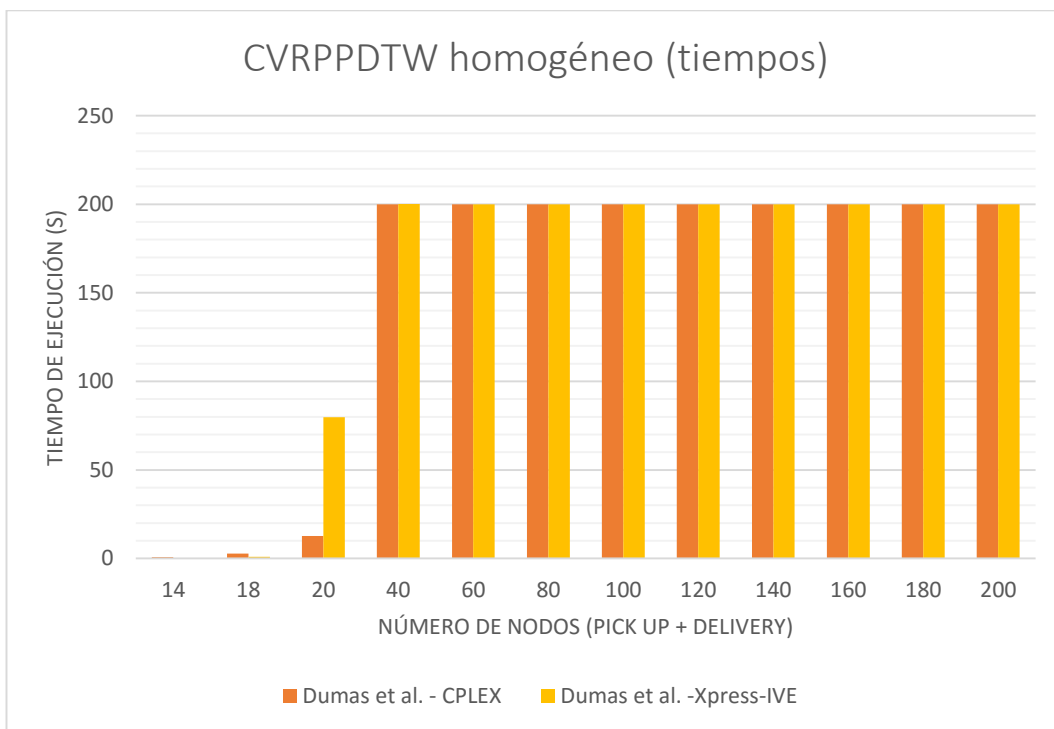


Figura 8.14: Comparativa tiempos modelo de Dumas et al.

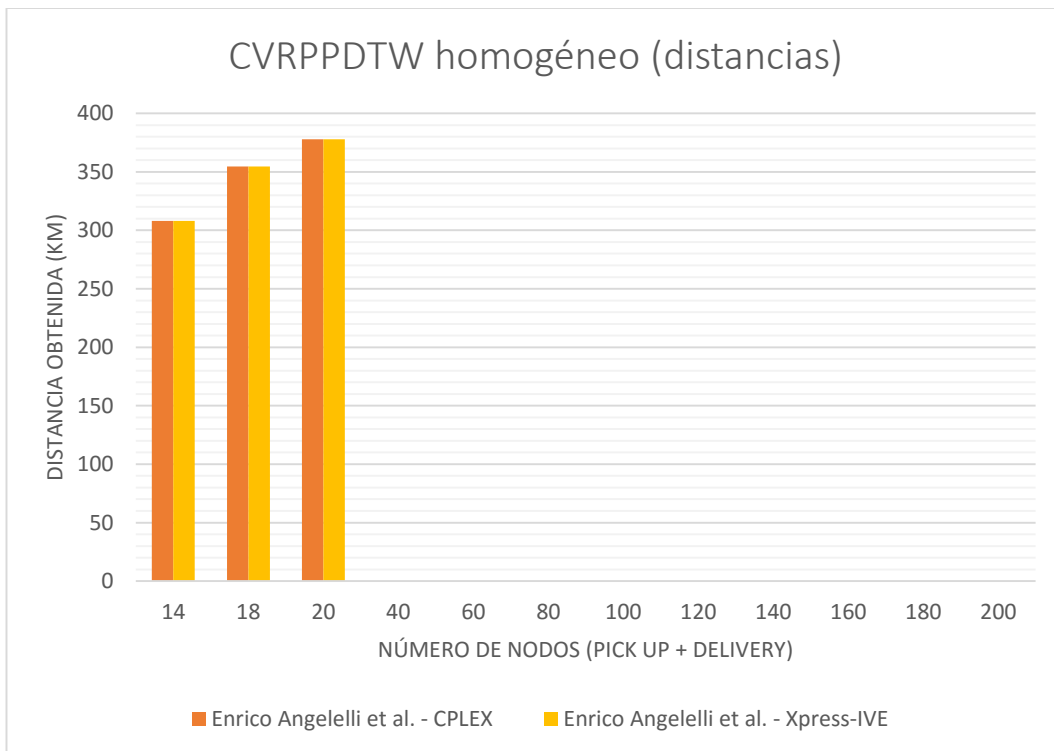


Figura 8.15: Comparativa de distancias modelo de Enrico Angelelli et al.

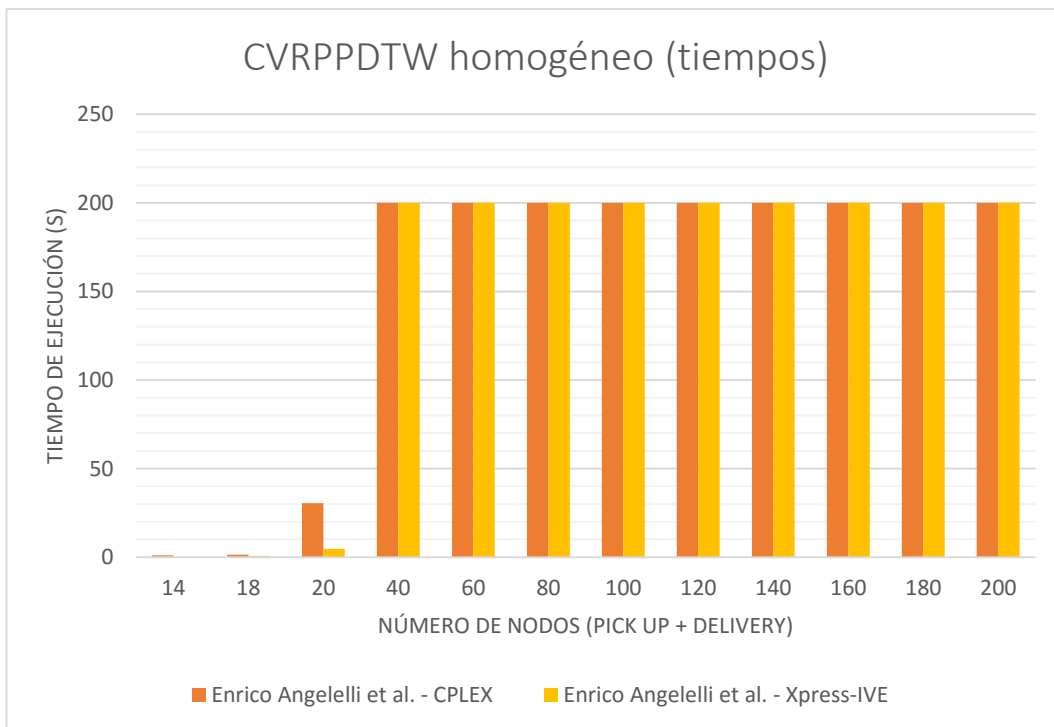


Figura 8.16: Comparativa de tiempos modelo de Enrico Angelelli et al.

8.7.2. Resumen de resultados obtenidos. Caso homogéneo.

Hemos obtenido los siguientes resultados:

- Xpress-IVE logra resolver 12 de los 39 problemas planteados, por los 10 que logra resolver CPLEX.
- De esos 10 problemas resueltos en común por ambos solvers, CPLEX no obtiene mejores distancias en ninguno de ellos, por 1 de Xpress-IVE y 9 empates. Cabe mencionar que en ese caso en que Xpress-IVE obtiene un resultado mejor, CPLEX empeora la solución en aproximadamente un 5% por lo que, si bien no es un porcentaje elevado, sí que resulta significativo (48 km, aproximadamente).
- En relación a los tiempos de ejecución, CPLEX obtiene mejores resultados en 2 de los 10 casos, por 7 de Xpress-IVE y 1 caso en que se ha llegado al tiempo máximo de ejecución permitido.

Pese a que el criterio de distancias obtenidas no se demuestra muy significativo para determinar qué solver es el más adecuado a la hora de resolver problemas CVRPPDTW homogéneos, el criterio de número de problemas resueltos y especialmente el de tiempos de ejecución obtenidos sí. Según estos criterios significativos, Xpress-IVE se trata del solver más adecuado, ya que logra resolver un mayor número de problemas en unos tiempos de ejecución menores. Además, en un caso logra obtener una distancia menor que CPLEX, lo que refuerza la decisión tomada.

La Tabla 8.9 muestra el resumen de resultados obtenidos para el problema CVRPPDTW homogéneo comparando Xpress-IVE con CPLEX⁸⁰.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	10 de 39	7 de 39	12 de 39	SELECCIONADO
CPLEX	9 de 39	2 de 39	10 de 39	Descartado

Tabla 8.9: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPDTW homogéneo.

⁸⁰ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

8.7.3. Caso heterogéneo.

A continuación, se muestran los resultados obtenidos para los 4 modelos del problema CVRPPDTW heterogéneo planteados tanto en Xpress-IVE como en CPLEX:

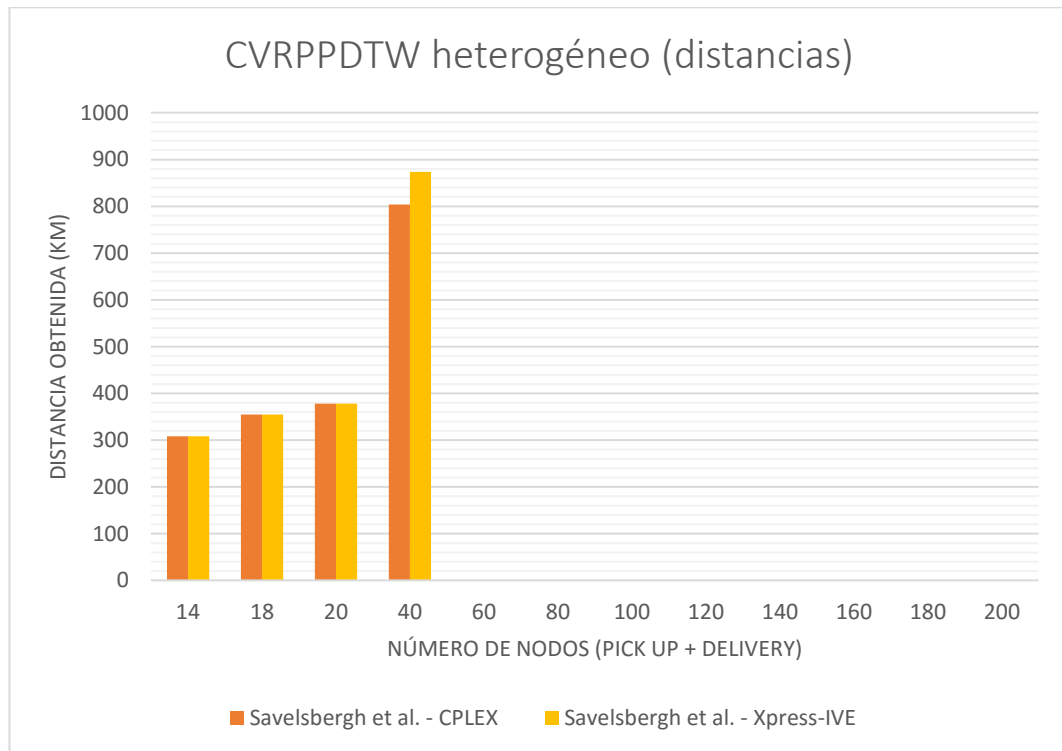


Figura 8.17: Comparativa distancias modelo de Savelsbergh et al.

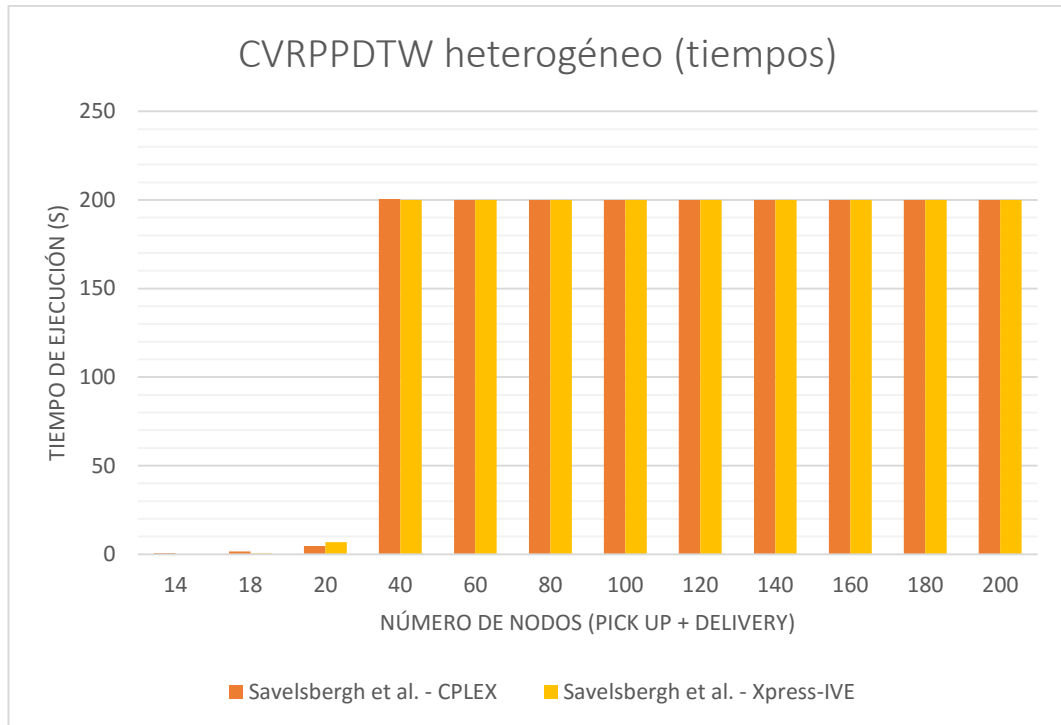


Figura 8.18: Comparativa de tiempos modelo de Savelsbergh et al.

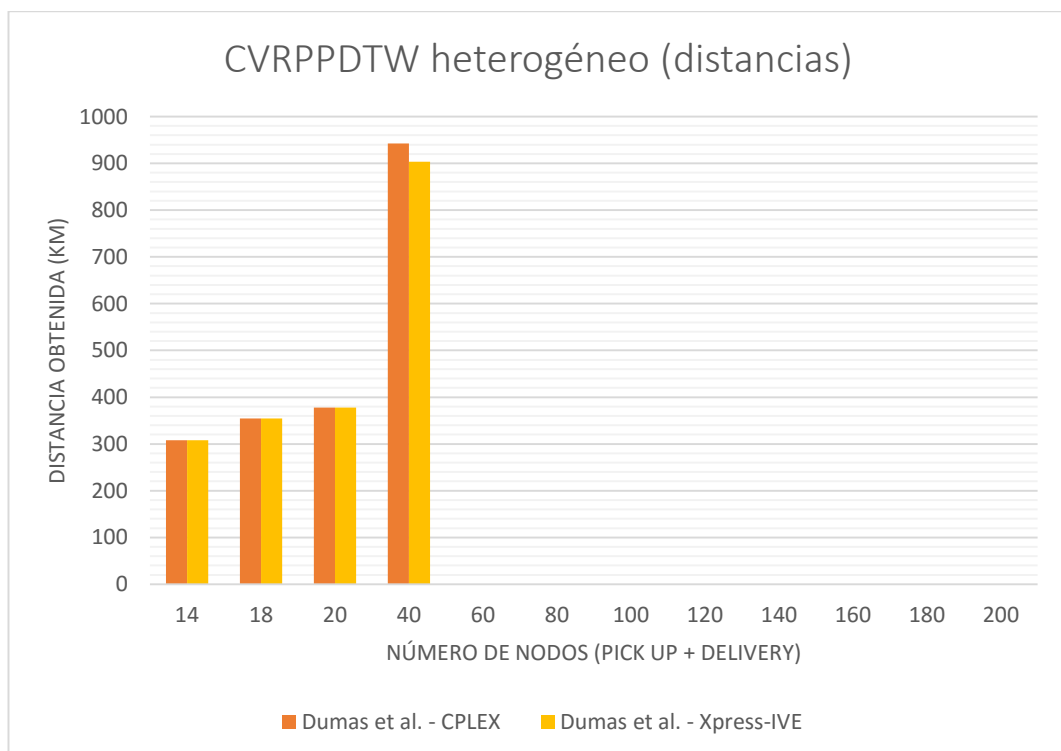


Figura 8.19: Comparativa de distancias modelo Dumas et al.

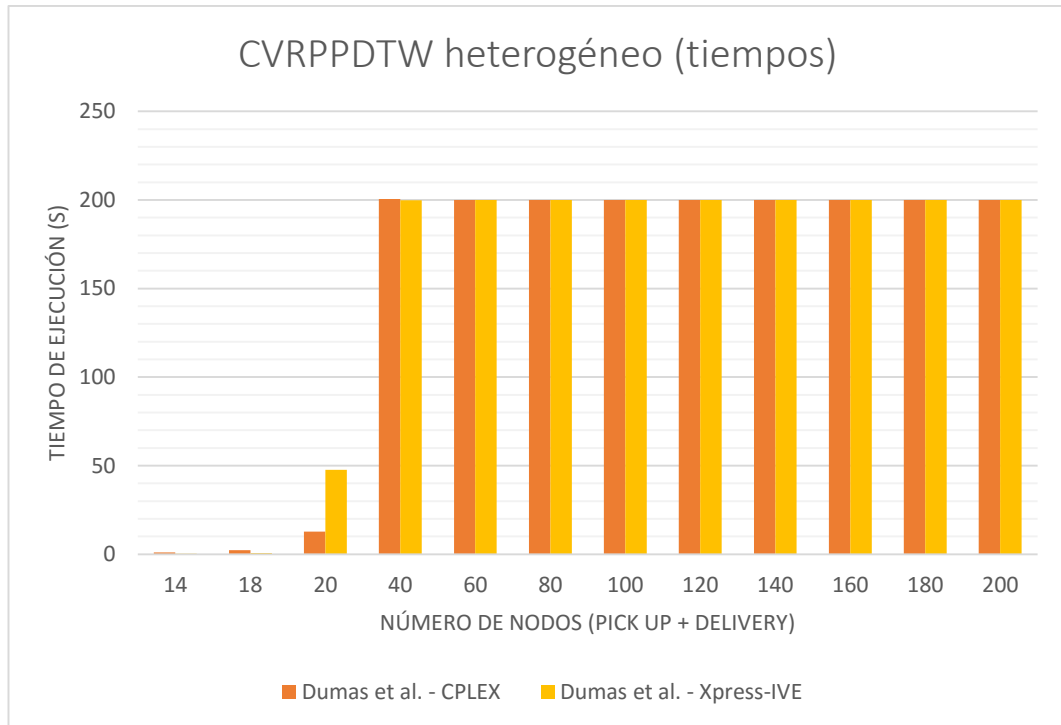


Figura 8.20: Comparativa de tiempos modelo de Dumas et al.

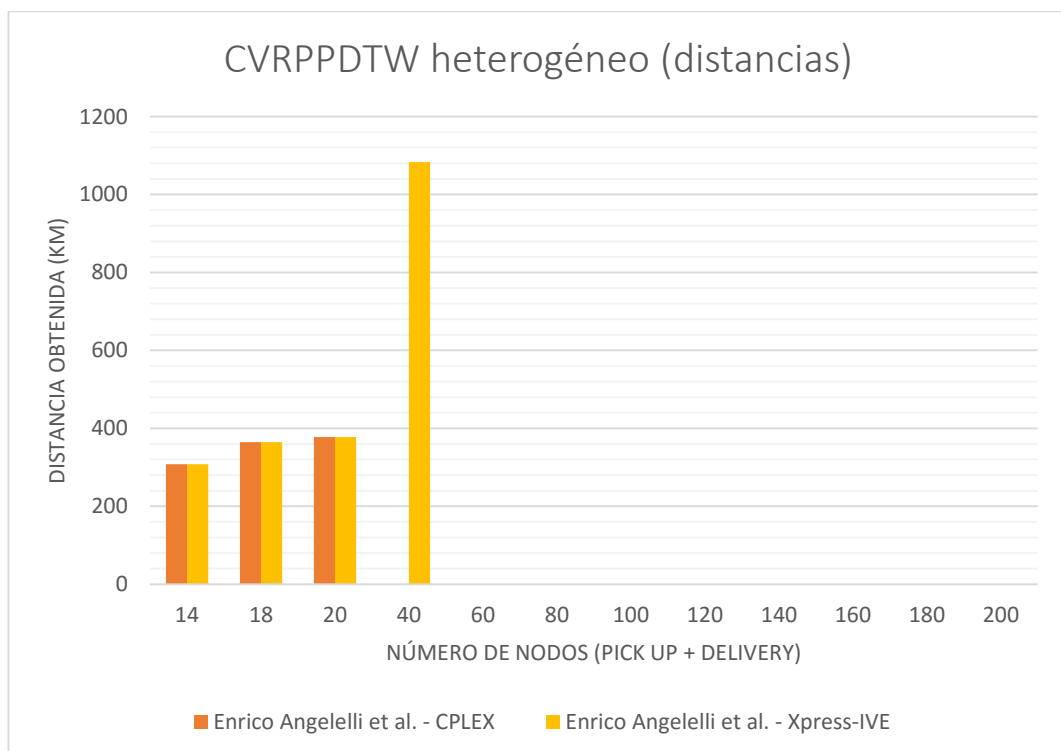


Figura 8.21: Comparativa distancias modelo de Enrico Angelelli et al.

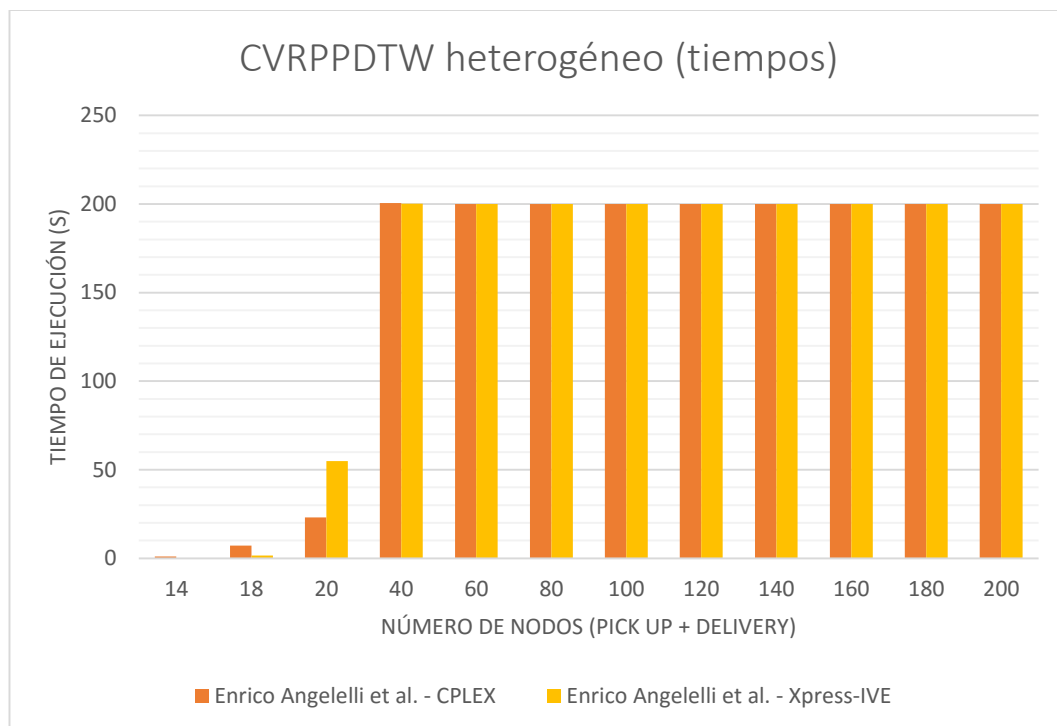


Figura 8.22: Comparativa de tiempos modelo de Enrico Angelelli et al.

8.7.4. Resumen de resultados obtenidos. Caso heterogéneo.

Analizando los resultados presentados en las tablas y figuras anteriores, extraemos las siguientes estadísticas:

- CPLEX resuelve 11 de los 39 problemas planteados, por 12 de Xpress-IVE.
- De los 11 problemas resueltos por ambos solvers, CPLEX obtiene mejores distancias en 1 de ellos, por 1 de Xpress-IVE y 9 empates.
- En relación a los tiempos de ejecución, CPLEX obtiene mejores resultados en 3 de los 11 casos, por 6 de Xpress-IVE y 2 casos en los que se ha llegado al tiempo máximo de ejecución permitido.

Como podemos observar, los dos solvers se encuentran igualados en cuanto al criterio de distancias obtenidas, no siendo un criterio significativo para determinar cuál de los dos es mejor.

En cuanto a los criterios de número de problemas resueltos y tiempos de ejecución obtenidos, ocurre lo mismo que en el caso homogéneo: Xpress-IVE logra resolver más problemas que CPLEX, obteniendo en líneas generales mejores tiempos de ejecución.

Por tanto, y teniendo en cuenta que se trata de una solución más ajustada que en el caso homogéneo, se ha seleccionado Xpress-IVE como el solver más adecuado para resolver problemas CVRPPDTW heterogéneos.

En la Tabla 8.10 se puede observar el resumen de resultados obtenidos comparando Xpress-IVE con CPLEX⁸¹.

Solver	Criterios de comparación			Estado actual
	Mejores distancias obtenidas (km)	Mejores tiempos de ejecución (s)	Número de problemas resueltos	
Xpress-IVE	10 de 39	6 de 39	12 de 39	SELECCIONADO
CPLEX	10 de 39	3 de 39	11 de 39	Descartado

Tabla 8.10: Resumen de resultados comparando Xpress-IVE con CPLEX para el problema CVRPPDTW heterogéneo.

8.8. Conclusiones.

A modo de recopilación de todos los resultados anteriormente citados en este capítulo, y teniendo en cuenta los criterios establecidos en los apartados 2.4 y 3.2.4., concluimos que:

- Para problemas CVRPPDTW homogéneos, el modelo de Savelsbergh et al. se ha seleccionado como el más eficiente, tanto desde el punto de vista de número de problemas resueltos, distancias obtenidas y tiempos de ejecución.
- Igualmente, para el problema CVRPPDTW heterogéneo, el modelo de Savelsbergh et al. ha resultado ser el más adecuado desde el punto de vista de todos los criterios establecidos.
- Comparando solvers para el caso homogéneo, se recomienda emplear de Xpress-IVE dado que logra resolver algún problema más que CPLEX, obteniendo en líneas generales mejores resultados que este último en

⁸¹ Ocurre como en los casos anteriores, si todos los modelos coinciden al encontrar la mejor solución (en distancias o en tiempos), se cuenta igualmente a no ser que se haya alcanzado el máximo tiempo de ejecución permitido.

relación a tiempos de ejecución. El criterio de distancias obtenidas no se ha demostrado significativo.

- Para el caso heterogéneo, ocurre exactamente lo mismo: se recomienda el empleo de Xpress-IVE ya que logra resolver algún problema más que CPLEX y a que resuelve los problemas en un tiempo de ejecución en general menor. El criterio de distancias obtenidas no se demuestra significativo para ayudar a la toma de decisiones.

Por último, y como hemos indicado en todos los capítulos, cuestionarse el por qué un modelo obtiene mejores resultados que otro consideramos que no tiene mucho sentido debido a que la respuesta reside en la formulación matemática empleada. Sin embargo, sí lo tiene en el caso de los solvers. Por ello, se recomienda al lector consultar el apartado 3.2.5, en donde se han tratado estos motivos con anterioridad.

CAPÍTULO 9

CONCLUSIONES FINALES Y FUTURAS LÍNEAS DE TRABAJO.

9.1. Conclusiones finales.

Hemos podido comprobar de primera mano la gran cantidad de modelos existentes para cada problema de rutas de vehículos, debido en parte al creciente interés que la investigación operativa está suscitando en la sociedad actual. Aunque es técnicamente imposible recopilarlos todos (y menos para varios tipos de problemas), creemos que se han mostrado los principales, los más citados y los más populares, por lo que bien este estudio podría valer como una revisión al estado del arte actual

Una vez recopilada la información, se han formulado dichos modelos de la forma más precisa posible, intentando respetar al máximo las formulaciones originales (salvo para las restricciones modificadas para intentar reducir el número de vehículos empleado), y teniendo que lidiar con los errores de redacción o de formulación que alguno presentaba. A partir de aquí, la carga de trabajo era para los solvers.

En líneas generales los resultados creemos que han sido muy satisfactorios, ya que hemos cumplido los tres objetivos principales indicados al comienzo del TFG, identificando el modelo/solver más adecuado para resolver cada tipo de problema de rutas propuesto, con el añadido adicional que la mejora de intentar reducir el número de vehículos ha supuesto en los resultados obtenidos.

A modo de resumen, la Tabla 9.1 nos muestra todos los resultados obtenidos:

Problema	Tipo	Modelo más adecuado	Solver más adecuado
CVRP	Homogéneo	- Modelo basado en 2 índices de TMZ. - Modelo de redes para problemas pequeños si el tiempo es un factor crítico	CPLEX
	Heterogéneo	Modelo de Christofides et al.	CPLEX
CVRPTW	Homogéneo	Modelo de Cordone - Wolfler de 2 índices	Xpress-IVE
	Heterogéneo	Modelo de M.Kritikos et al.	Xpress-IVE
CVRPPD	Homogéneo	- Modelo de Toth y Vigo. - Modelo de Quan Lu si el nº de nodos es reducido	Xpress-IVE si el tiempo es un factor crítico CPLEX si la distancia es un factor crítico
	Heterogéneo	- Modelo de Quan Lu para un nº de nodos inferior a 18 - Modelo de Savelsbergh et al. en caso contrario	Xpress-IVE para problemas de ≥ 38 nodos Indiferente en caso contrario
CVRPPDTW	Homogéneo	Modelo de Savelsbergh et al.	Xpress-IVE
	Heterogéneo	Modelo de Savelsbergh et al.	Xpress-IVE

Tabla 9.1: Resumen de resultados obtenidos en para cada problema de rutas.

Como podemos observar, Xpress-IVE presenta en líneas generales unos resultados mejores que CPLEX, siendo escogido como mejor solver en 2 de los 4 casos planteados, con 1 empate en el que resulta elegido si el tiempo de ejecución se considera un factor crítico (homogéneo) o para un número de nodos elevado (heterogéneo). Si bien es cierto que ambos solvers resultan muy eficientes a la hora de resolver los problemas planteados, parece ser que Xpress-IVE toma cierta ventaja en la carrera.

Por último, destacar nuestra convicción en la capacidad de los modelos seleccionados para suponer importantes ahorros para las empresas, que cada vez con más frecuencia están confiando en estas herramientas para gestionar su actividad.

9.2. Futuras líneas de trabajo.

9.2.1. Heurísticas y meta heurísticas.

A continuación, aparece, para cada problema de rutas propuesto, para cada número de nodos considerado, el porcentaje de ejecuciones que han logrado encontrar al menos una solución factible. Como se ha podido comprobar que tanto Xpress-IVE como CPLEX resuelven prácticamente el mismo número de problemas, los datos están referidos sólo a uno de estos solvers, en este caso Xpress-IVE.

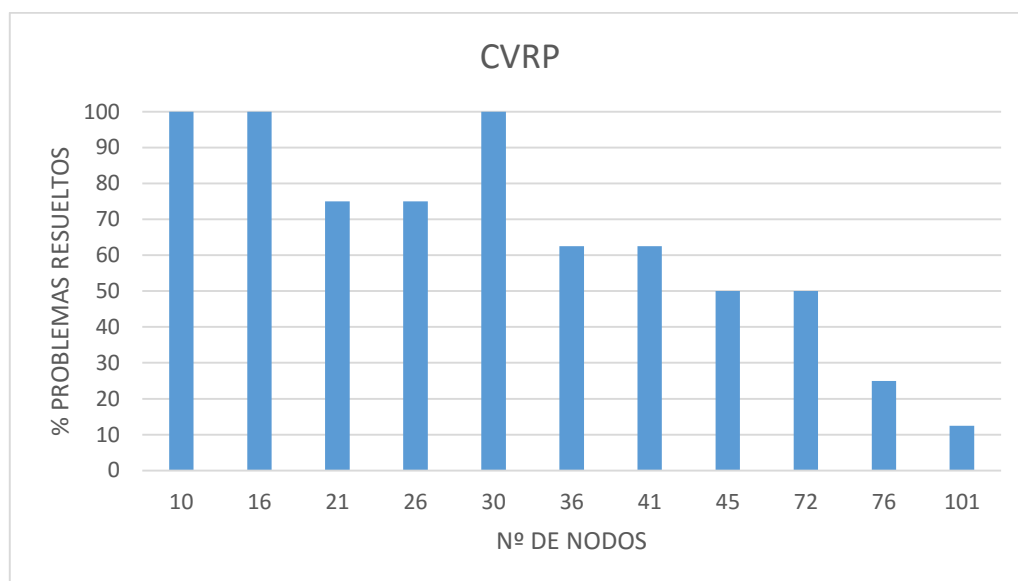


Figura 9.1: % de problemas resueltos. Problema CVRP.

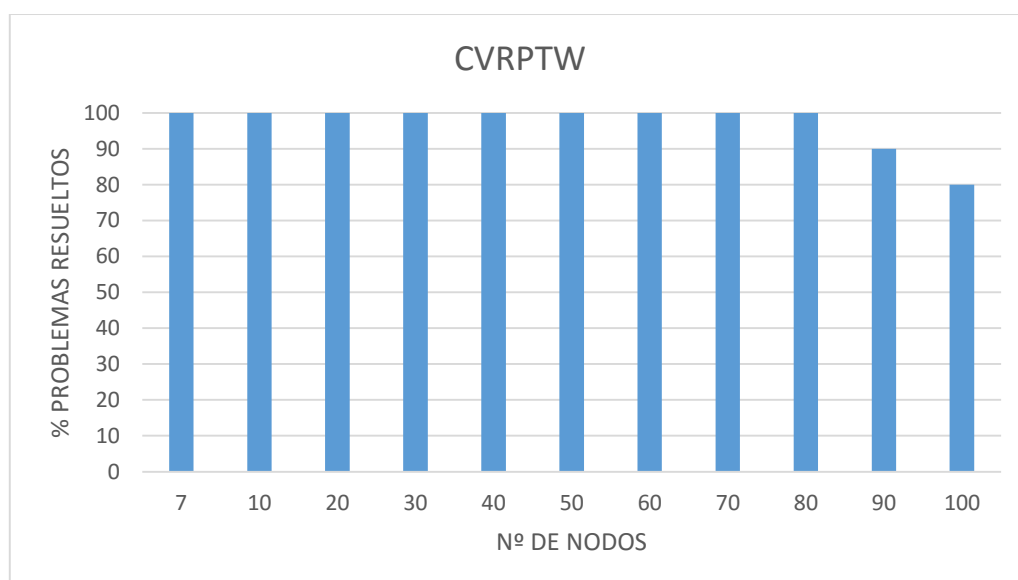


Figura 9.2: % de problemas resueltos. Problema CVRPTW.

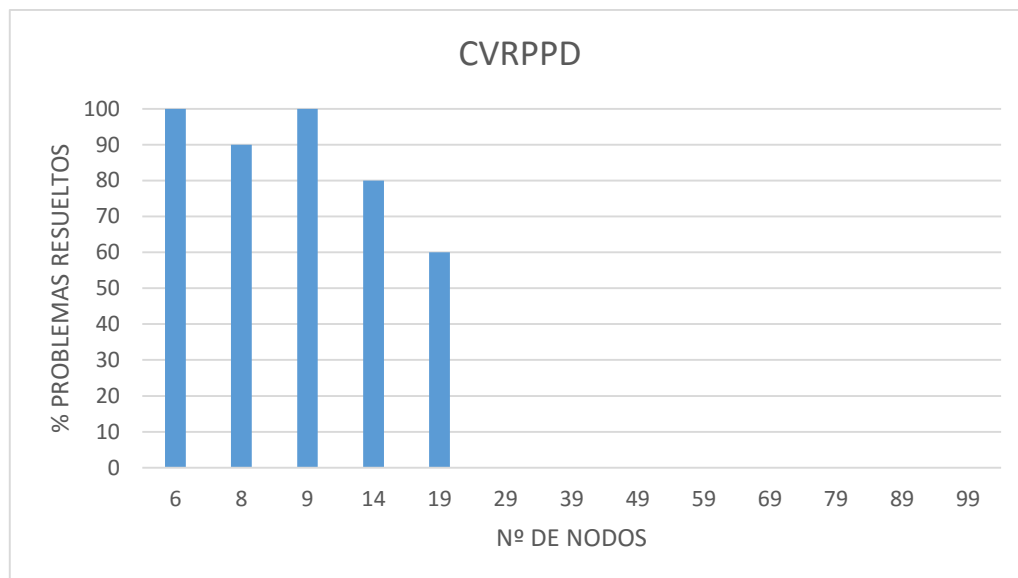


Figura 9.3: % de problemas resueltos. Problema CVRPPD.

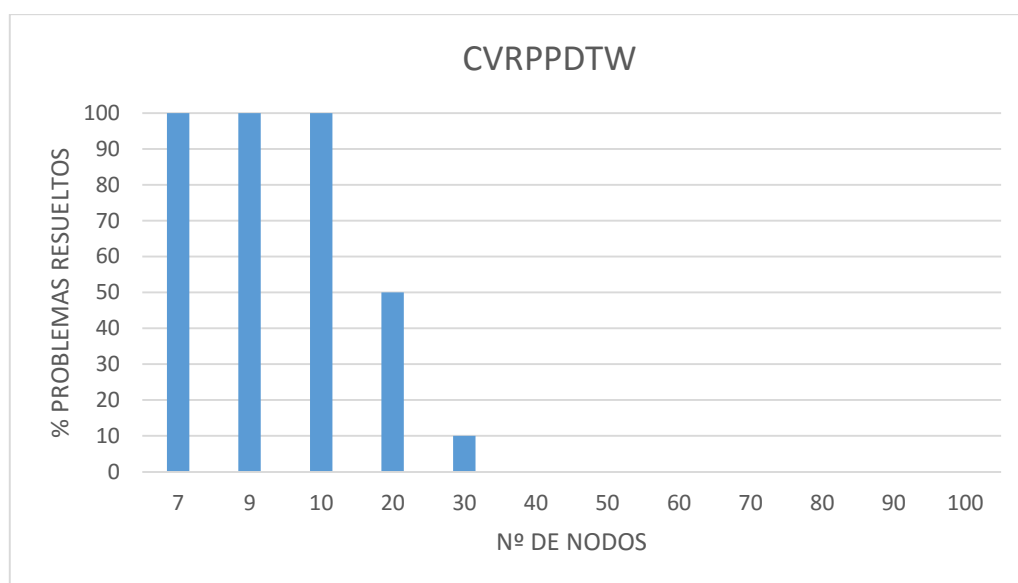


Figura 9.4: % de problemas resueltos. Problema CVRPPDTW.

Como podemos observar en las figuras anteriores, a medida que el número de nodos va aumentando, se resuelve una menor cantidad de problemas. Además, a medida que la complejidad del problema es mayor (problemas CVRPP y CVRPTW son más sencillos que el CVRPPD o el CVRPPDTW), el número de problemas resueltos disminuye considerablemente.

Ya se comentó en el apartado 2.5 que los métodos exactos son aquellos en los que formulamos un problema por medio de un modelo, por lo que necesitan de

un solver específico que posea algoritmos implantados para lograr resolver dicho problema. Esto, conlleva que conforme la complejidad del problema es mayor, el solver tarde más tiempo en lograr encontrar soluciones factibles.

Las heurísticas y meta heurísticas, por su parte, evitan esta dependencia de un software específico, pudiendo implementarse en cualquier lenguaje de programación y ejecutarse sin necesidad de una interfaz particular. Además, si bien las soluciones que obtienen tienden a ser ligeramente peores que en los métodos exactos, los tiempos de computación son muy inferiores.

La Figuras 9.5 y 9.6 muestra una comparación para el modelo de 2 índices basado en TMZ del problema CVRP entre los resultados obtenidos con CPLEX y los obtenidos empleando la heurística de inserción de Mole & Jameson como solución inicial, mejorando esta con el método de String Relocate (Fernández Hernández, 2016):

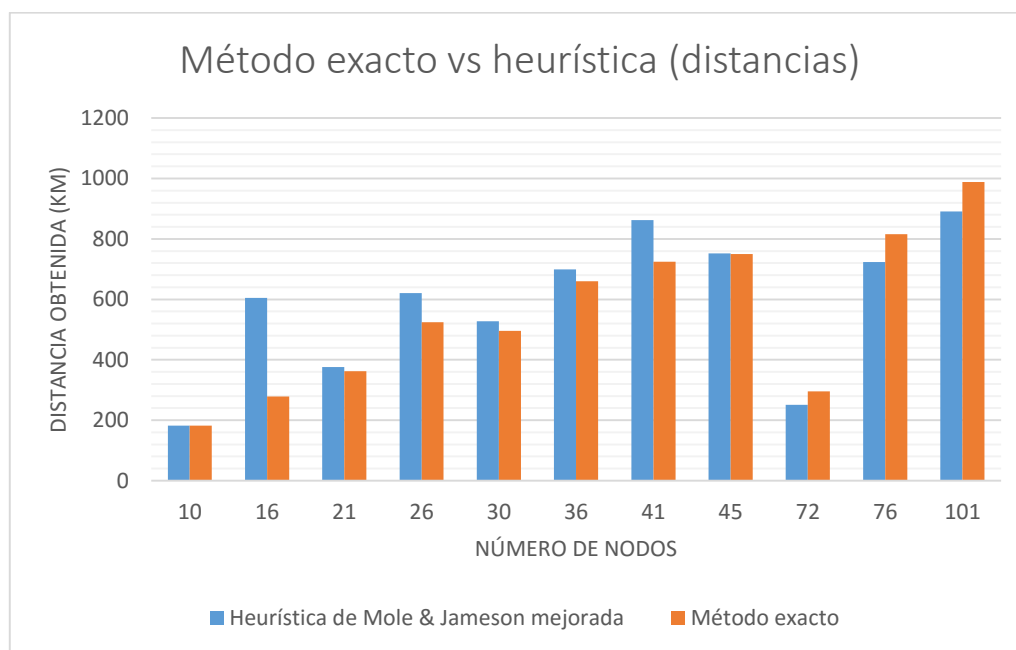


Figura 9.5: comparación método exacto vs heurística. Distancias obtenidas.

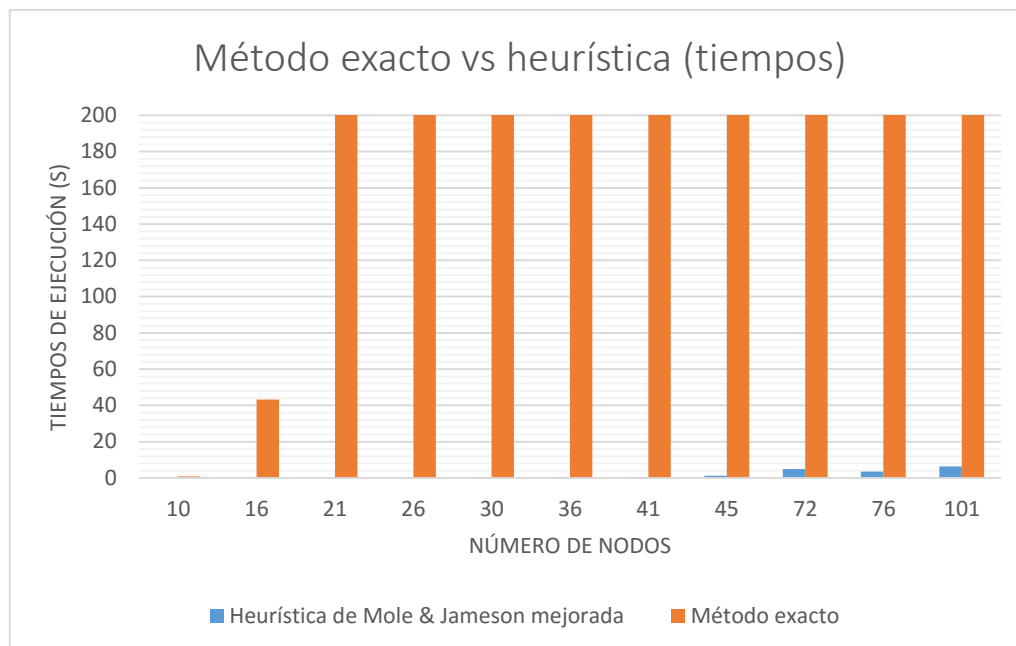


Figura 9.6: comparación método exacto vs heurística. Tiempos de ejecución.

En base a estos resultados, queda demostrado que las heurísticas y meta heurísticas pueden suponer un punto de mejora a los tiempos de ejecución tan elevados obtenidos empleando métodos exactos, aunque esto suponga un ligero empeoramiento de la función objetivo.

Por tanto, como futura línea de trabajo se propone realizar el mismo planteamiento del presente TFG, pero enfocado a las heurísticas y meta heurísticas, y no a métodos exactos. Para ello se realizará una revisión del estado del arte de las heurísticas actuales para los problemas CVRP, CVRPTW, CVRPPD y CVRPPDTW, que serán planteadas y comparadas como hemos hecho hasta ahora.

9.2.2. Herramientas de tuning.

En el apartado 3.2.5 se indicó que las diferencias existentes entre Xpress-IVE y CPLEX iban a ser asociadas a los valores predefinidos que tuvieran los parámetros de cada uno de estos solvers.

La modificación de esos valores puede suponer una mejora en las soluciones obtenidas por el solver, por lo que sería interesante realizar pruebas para ver qué modificaciones pueden ser adecuadas. Sin embargo, cada solver cuenta

con más de 200 parámetros, por lo que una manipulación manual de sus valores resulta cuanto menos tedioso.

Para superar este inconveniente, se cuenta con herramientas implementadas dentro de los solvers, llamadas comúnmente herramientas de tuning, cuyo funcionamiento es el siguiente: para un modelo dado, esta herramienta prueba distintas combinaciones para los valores de los parámetros, resolviendo el modelo para cada una de ellas, y devolviendo aquella combinación que aporte unos mejores resultados.

Por tanto, se propone desarrollar el proceso de tuning para los modelos propuestos, lo que podría radicar en mejoras en las soluciones obtenidas.

9.2.3. Otros solvers.

Si bien en el presente TFG se han resuelto problemas de rutas empleando Xpress-IVE y CPLEX, en el Capítulo 3 se vio que existen muchos otros solvers, por lo que resultaría interesante realizar una comparación entre un mayor número de ellos, incluyendo algunos solvers de código abierto (ya que generalmente son los que obtienen unos peores resultados, pero puede ser que con los últimos avances esta situación haya cambiado).

9.2.4. Otros modelos.

Como ya se ha indicado previamente, recopilar todos los modelos existentes para cada tipo de problema de rutas es una tarea muy complicada. Sin embargo, ir recopilando poco a poco una mayor cantidad de ellos y compararlos con los ya seleccionados como más eficientes sería interesante desde el punto de vista de la mejora continua.

9.2.5. Otros problemas.

En este TFG se han tratado los problemas CVRP, CVRPTW, CVRPPD y CVRPPDTW, pero existen muchos otros, menos importantes, pero utilizados igualmente para determinadas situaciones de transporte. Un estudio como los planteados anteriormente supondría una situación muy favorable a la hora de lidiar con estos problemas menos conocidos.

Especialmente, se recomienda el estudio del problema del autobús escolar (school bus), un tipo especial de problema CVRPPDTW en el que se transportan personas y no mercancías, y en donde se debe determinar previamente la posición de las distintas paradas en las que el autobús deberá detenerse en unas ventanas de tiempo predefinidas.

CAPÍTULO 10

AGRADECIMIENTOS.

10.1. Agradecimientos.

Quisiera agradecer en primer lugar al Departamento de Estadística e Investigación Operativa de la Universidad de Valladolid en general, y a Jesús Sáez Aguado, tutor del presente TFG, en particular, el haberme dado la oportunidad de realizar este trabajo en el régimen de una beca de colaboración, dándome todos los medios necesarios para realizar las partes correspondientes a Xpress-IVE y documentación.

A Jesús Sáez Aguado en especial, gracias por aportarme su experiencia y sus amplios conocimientos en el campo de la investigación operativa y la programación lineal.

En segundo lugar, agradecer al AILN de Valladolid (Renault España S.A.) en general y a Javier Moral Blanco, en particular, el haberme permitido aprovechar durante el periodo de prácticas en empresa la licencia de CPLEX que allí tienen instalada, constituyendo una parte fundamental y un factor distintivo de este TFG.

Por último, agradecer a Alba Fernández Hernández el haberme autorizado a utilizar parte de los resultados obtenidos en su TFG como justificante de la futura línea de trabajo de las heurísticas y meta heurísticas.

ANEXO

TRATANDO DE REDUCIR EL NÚMERO DE VEHÍCULOS EMPLEADO.

A.1. Motivación.

A lo largo del presente TFG se plantean multitud de modelos que tratan de resolver de la forma más óptima posible el problema de rutas de vehículos para el que han sido diseñados (CVRP, CVRPTW, CVRPPD o CVRPPDTW).

Con el fin de generar unos resultados cuya evaluación pueda servir para decidir qué modelo o solver es el más adecuado para resolver cada uno de esos tipos de problemas de rutas, se han establecido unos ficheros de datos que servirán como entradas a los modelos. De esta forma, al garantizar que cada uno de los modelos de un tipo de problema de rutas emplea los mismos datos, la comparación entre estos resulta significativa.

Cada uno de estos ficheros de datos tiene una estructura propia que depende del problema de rutas en el que nos encontremos. Sin embargo, en líneas generales, todos estos ficheros presentan algunos aspectos comunes, tales como número de nodos que va a tener el problema, número de vehículos empleado o capacidad de los mismos, entre otros.

Este hecho puede ser apreciado en la Figura A.1, que muestra parte del fichero de datos E01603m, empleado para resolver el problema CVRP homogéneo. En la citada figura podemos observar aspectos tales como el nombre del fichero de datos, la dimensión (número de nodos) del problema tratado, el número de vehículos a emplear, la capacidad de dichos vehículos, las coordenadas de los nodos empleados...

```

NAME : E016-03m
COMMENT : Christofides, Mingozzi and Toth, 1981
TYPE : CVRP
DIMENSION : 16
EDGE_WEIGHT_TYPE : EUC_2D
CAPACITY : 90
VEHICLES : 3
NODE_COORD_SECTION
1 30 40
2 37 52

```

Figura A.1: Ejemplo fichero de datos para el problema CVRP homogéneo.

Si bien el número de nodos es un elemento identificativo del fichero, otros como capacidades y número de vehículos a emplear podrían no serlo tanto, ya que pueden modificarse a fin de comprobar cómo responde cada modelo ante distintas situaciones.⁸²

No obstante, en las formulaciones de la mayoría de modelos vemos cómo se toma el número de vehículos que aparece en el fichero de datos como un parámetro fijo, de forma que se debe emplear exactamente ese número. Esta situación puede apreciarse en la Figura A.2, en donde se muestra la restricción (3) del modelo basado en redes para el problema CVRP homogéneo tal y como viene formulada en el artículo original (Sáez Aguado, "Modelo de flujo en redes que no precisa conocer de antemano el número de vehículos", 2016).

$$\sum_{j=1}^n x_{origen j} = K \quad (3)$$

Figura A.2: Restricción (3) sin modificar, modelo basado en redes para el problema CVRP homogéneo.

En este contexto, nos planteamos la siguiente pregunta: ¿podría darse el caso de que un modelo obtenga una distancia (función objetivo planteada) menor empleando un número de vehículos inferior al que indica el fichero de datos? Esto supondría una solución mejor del problema y un coste inferior (si consideramos que cada vehículo tiene un coste fijo).

⁸² Muchos autores calculan el número de vehículos a emplear dividiendo la suma de demandas de los nodos entre la capacidad de cada vehículo y redondeando al entero mayor. Sin embargo, esto no siempre es así, ya que podemos encontrar casos en los que el número de vehículos que aparezca en el fichero esté sobredimensionado por cualquier motivo.

En definitiva, lo que proponemos es comprobar para cada uno de los problemas de rutas propuestos si la implantación de restricciones que permitan al solver “jugar” con el número de vehículos empleado puede suponer la obtención de una distancia total recorrida menor.⁸³ La Figura A.3 muestra la restricción de la Figura A.2 modificada según este propósito.⁸⁴

$$\sum_{j=1}^n x_{origen j} \leq K \quad (3)$$

Figura A.3: Restricción (3) modificada, modelo basado en redes para el problema CVRP homogéneo.

A.2. Planteamiento.

Para cada uno de los problemas de rutas planteados (CVRP, CVRPTW, CVRPPD, CVRPPDTW) se ha seleccionado un modelo, que ha sido formulado de dos formas diferentes, según se vio anteriormente:

- **Formulación 1:** la formulación del artículo original, sin modificar ninguna restricción, como en la Figura A.2.
- **Formulación 2:** una formulación adicional, alterando la del artículo original y modificando las restricciones oportunas, tal y como se vio en la Figura A.3.

Dicho modelo ha sido resuelto empleando el solver Xpress-IVE con un tiempo de ejecución máximo establecido de 200 segundos. Según los resultados que se obtengan para cada problema de rutas, se evaluarán las distancias, número de problemas resueltos y tiempos de ejecución obtenidos para justificar o no el empleo de la Formulación 2 en los modelos de dicho problema.

Los modelos que han sido seleccionados para realizar esta comparación han sido:

- Modelo de dos índices basado en TMZ para el problema CVRP (apartado 5.2 (b)).

⁸³ De igual forma, a parte de estas restricciones, podría añadirse en la función objetivo un coste por vehículo empleado, de forma que se tienda a minimizar el número total de vehículos. Sin embargo, dado que estamos minimizando la distancia total recorrida, no queremos distorsionar este valor, por lo que ese coste no será añadido.

⁸⁴ A veces, incluso puede ser necesario añadir alguna restricción, para evitar problemas con la generación de subtours que este tipo de restricciones podría provocar al no indicar específicamente que del nodo origen deba salir algún arco.

- Modelo de Sopnamayee Acharya para el problema CVRPTW (apartado 6.2 (a)).
- Modelo de Toth y Vigo para el problema CVRPPD (apartado 7.2 (e)).
- Modelo de Savelsbergh et al. para el problema CVRPPDTW (apartado 8.2 (b)).

A.3. Resultados.

A continuación, se muestran los resultados obtenidos para cada uno de estos casos junto con las conclusiones extraídas.

A.3.1. Problema CVRP.

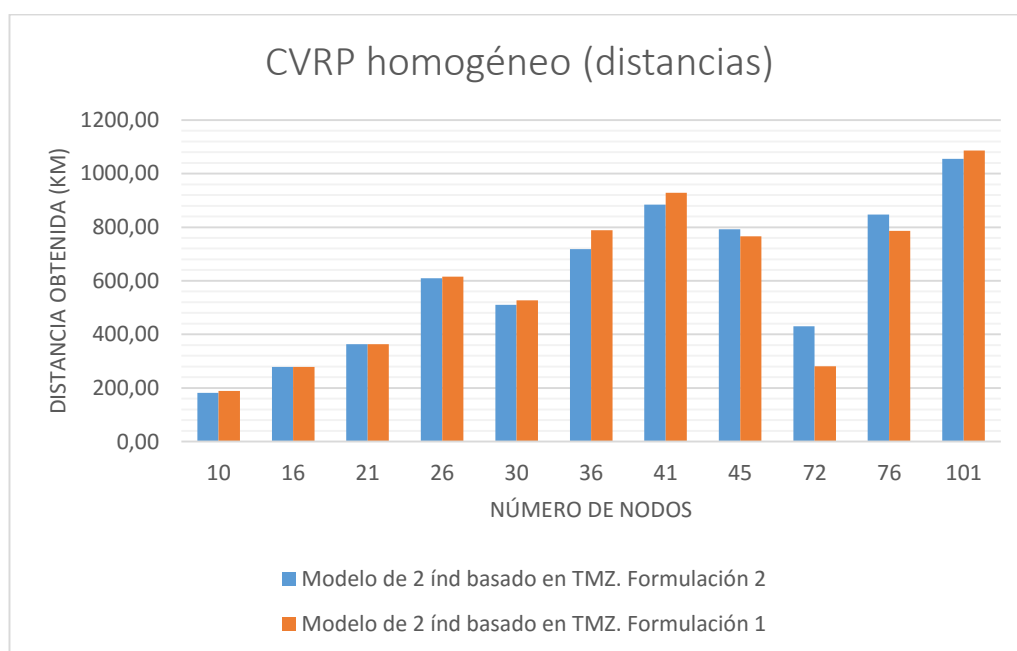


Figura A.4: Distancias obtenidas. Modelo basado en TMZ de 2 índ.

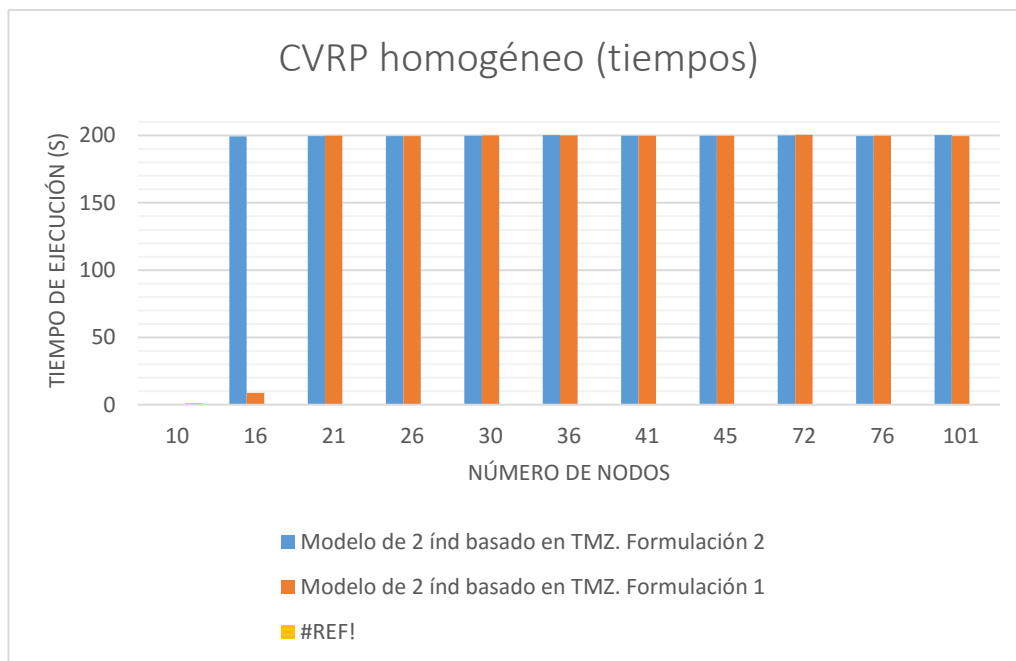


Figura A.5: Tiempos de ejecución. Modelo basado en TMZ de 2 índ.

Datos empleados		Modelo de 2 índ basado en TMZ . Formulación 2		Modelo de 2 índ basado en TMZ. Formulación 1	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
E010-03m	10	181,59	0,20	189,551	0,039
E016-03m	16	278,73	199,41	278,726	8,851
E021-04m	21	363,45	199,46	363,704	199,735
E026-08m	26	609,76	199,57	615,444	199,661
E030-04s	30	510,38	199,72	527,54	199,965
E036-11h	36	718,48	200,20	789,153	200,138
E041-14h	41	884,75	199,87	929,005	199,846
E045-04f	45	791,86	199,76	766,314	199,703
E072-04f	72	430,01	199,94	280,638	200,421
E076-07s	76	847,32	199,65	786,581	199,75
E101-08e	101	1054,91	200,30	1086,47	199,63

Tabla A.1: Resumen de resultados obtenidos. Modelo basado en TMZ de 2 índices.

A.3.2. Conclusiones obtenidas. Problema CVRP.

Como se puede observar en las tablas y figuras anteriores, tanto la Formulación 1 como la Formulación 2 logran resolver todos los problemas planteados, de manera que este no será un factor decisivo.

Si nos fijamos en las distancias obtenidas, vemos que la Formulación 2 obtiene mejores resultados que la Formulación 1 en 6 casos, por 3 casos favorables a la Formulación 1 y 2 empates.

En cuanto a los tiempos de ejecución, en ambos casos son muy similares, si bien para el problema E016-03m se obtiene un tiempo de ejecución mucho más elevado en la Formulación 2. Esto no necesariamente es un punto negativo, ya que, si nos fijamos en las distancias obtenidas para este caso, vemos que ambos modelos han obtenidos los mismos resultados, de manera que el tiempo de ejecución más elevado se traduce en que la Formulación 1 encontró ya la solución óptima, en tanto que la Formulación 2 aún sigue buscando dicha solución. Por tanto, de no ser por los 200 segundos máximos establecidos, quizás la hubiera encontrado.

Por último, y en base a las conclusiones anteriormente expuestas, se considera que la Formulación 2 puede suponer un punto de mejora a los resultados obtenidos en los modelos CVRP, por lo que se implantará en todos los modelos de dicho problema.

A.3.3. Problema CVRPTW.

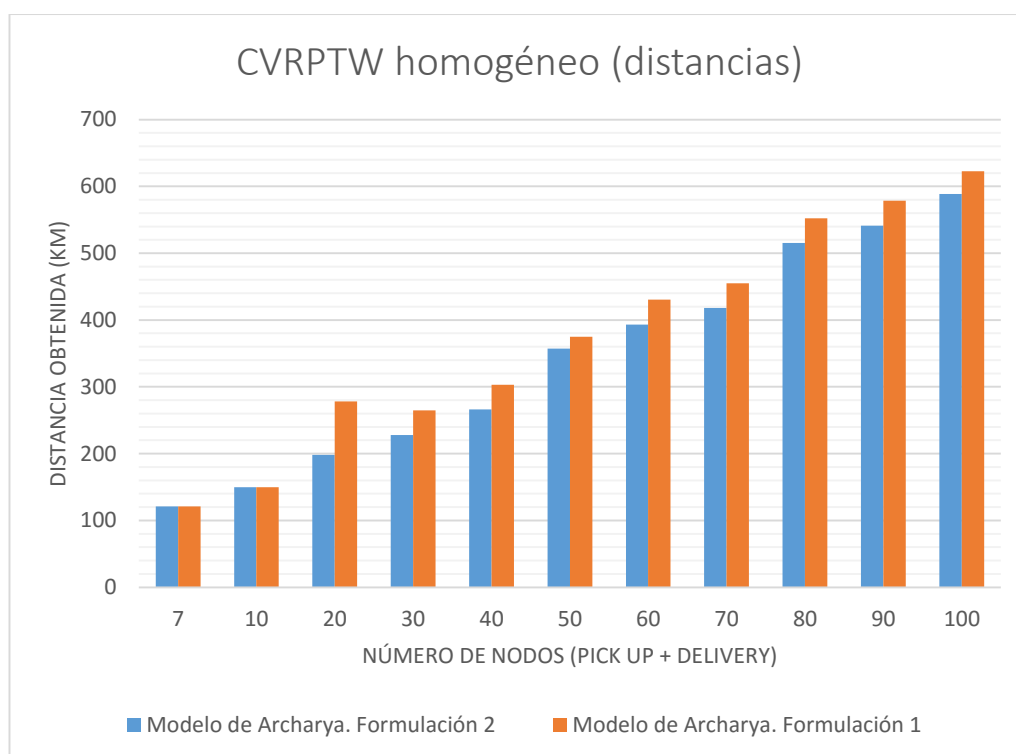


Figura A.6: Distancias obtenidas. Modelo de Acharya.

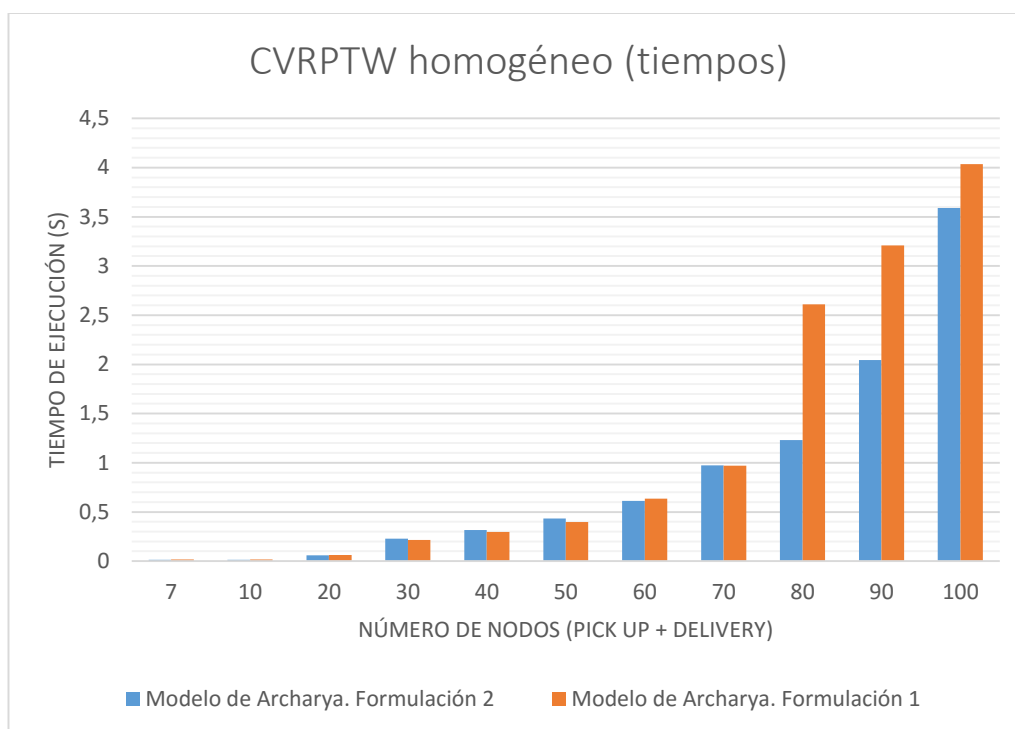


Figura A.7: Tiempos de ejecución. Modelo de Acharya.

Datos empleados		Modelo de 2 índ basado en TMZ . Formulación 2		Modelo de 2 índ basado en TMZ. Formulación 1	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
TW7	7	121,042	0,015	121,042	0,017
TW10	10	149,803	0,013	149,803	0,016
TW20	20	198,184	0,058	278,082	0,061
TW30	30	227,761	0,229	264,765	0,216
TW40	40	266,017	0,317	303,022	0,296
TW50	50	357,324	0,432	374,991	0,399
TW60	60	393,358	0,611	430,362	0,635
TW70	70	418,066	0,974	455,070	0,970
TW80	80	515,414	1,23	552,418	2,609
TW90	90	541,424	2,043	578,428	3,208
TW100	100	588,628	3,591	622,674	4,037

Tabla A.2: Resumen de resultados obtenidos. Modelo de Acharya.

A.3.4. Conclusiones obtenidas. Problema CVRPTW.

A vista de los resultados mostrados, creemos que la conclusión está clara, ya que, además de conseguir mejores resultados en cuanto a distancias calculadas para todos los problemas propuestos, la Formulación 2 obtiene unos tiempos de ejecución similares (e incluso mejores a medida que el número de nodos se va incrementando) que la Formulación 1.

Por lo tanto, creemos firmemente que la Formulación 2 puede suponer un punto de mejora para los resultados obtenidos en el problema CVRPTW, por lo que será implantada en todos sus modelos.

A.3.5. Problema CVRPPD.

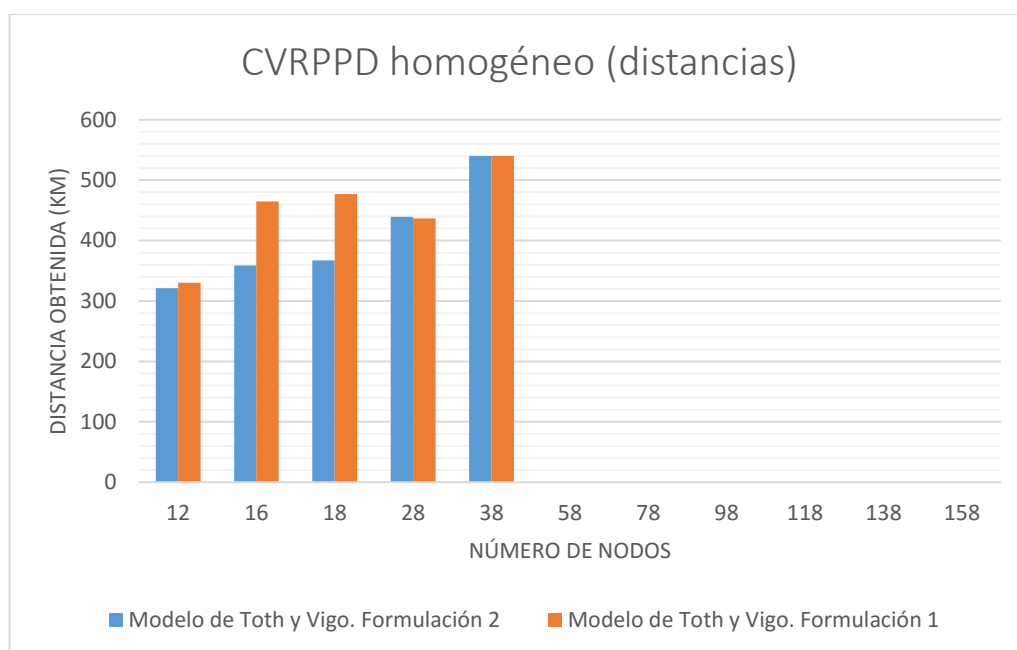


Figura A.8: Distancias obtenidas. Modelo de Toth y Vigo.

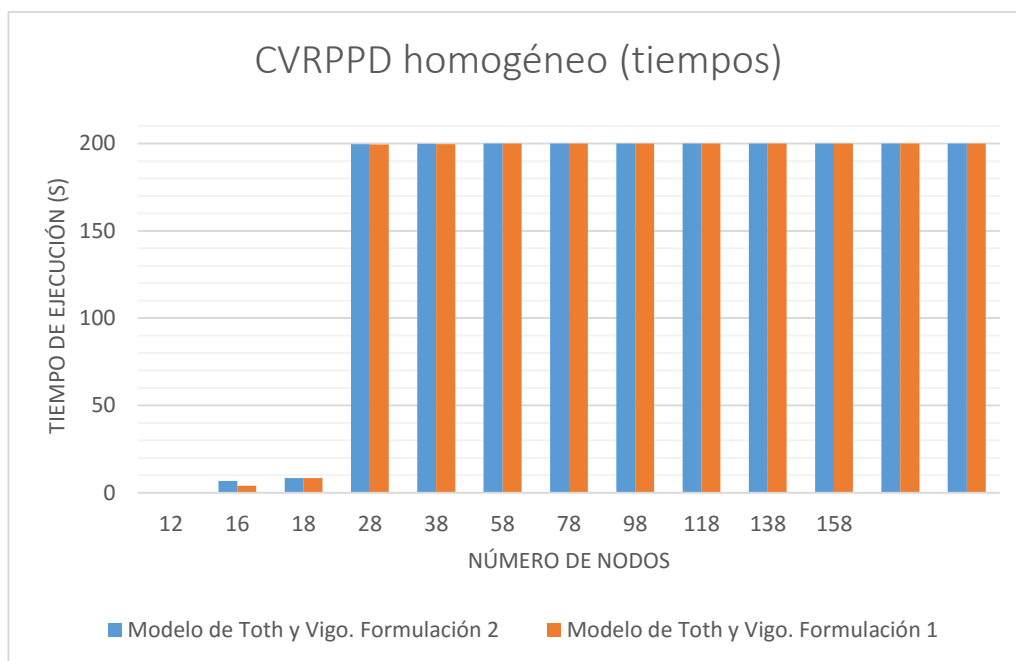


Figura A.9: Tiempos de ejecución. Modelo de Toth y Vigo.

Datos empleados		Modelo de Toth y Vigo. Formulación 2		Modelo de Toth y Vigo. Formulación 1	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PD7_homog	12	320,859	0,39	329,95	0,19
PD9_homog	16	358,582	6,731	464,62	4,07
PD10_homog	18	367,314	8,365	476,59	8,33
PD15_homog	28	439,184	199,611	436,612	199,444
PD20_homog	38	540,076	199,794	540,08	199,57
PD30_homog	58	#N/A	200	#N/A	200
PD40_homog	78	#N/A	200	#N/A	200
PD50_homog	98	#N/A	200	#N/A	200
PD60_homog	118	#N/A	200	#N/A	200
PD70_homog	138	#N/A	200	#N/A	200
PD80_homog	158	#N/A	200	#N/A	200

Tabla A.3: Resumen de resultados obtenidos. Modelo de Toth y Vigo.

A.3.6. Conclusiones obtenidas. Problema CVRPPD.

De las tablas y figuras anteriores extraemos los siguientes resultados:

- Tanto la Formulación 1 como la Formulación 2 consiguen resolver el mismo número de problemas.
- La Formulación 2 obtiene mejores distancias que la Formulación 1 en 3 de 5 casos, por 1 favorable a la Formulación 1 (en el cual la diferencia de distancias entre ambas es muy pequeña, de unos 3 km) y un empate entre ambas.
- En relación a los tiempos de ejecución, existen pequeñas diferencias favorables a la Formulación 1. Sin embargo, debido a que se trata de diferencias de, a lo sumo, 2 segundos, no se consideran significativas.

Por todo ello, se considera que la Formulación 2 logra obtener en general mejores resultados que la Formulación 1, razón por la cual se implantará dicha formulación en todos los modelos del problema CVRPPD.

A.3.7. Problema CVRPPDTW.

Por último, a continuación se muestran los resultados correspondientes a la comparación de la Formulación 1 y la Formulación 2 para el problema CVRPPDTW:

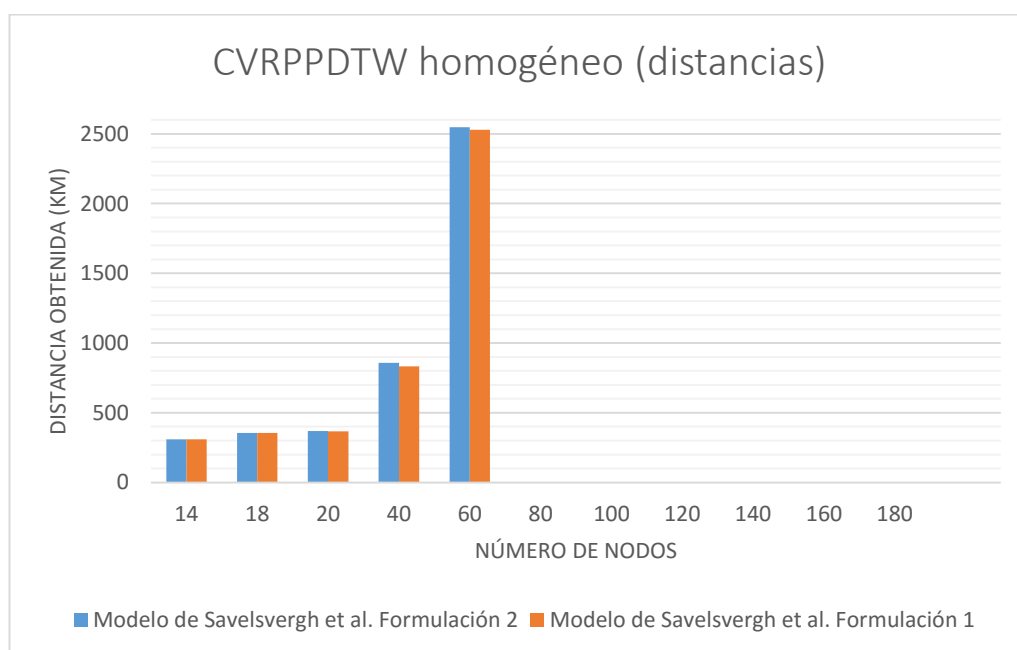


Figura A.10: Distancias obtenidas. Modelo de Savelsbergh et al.

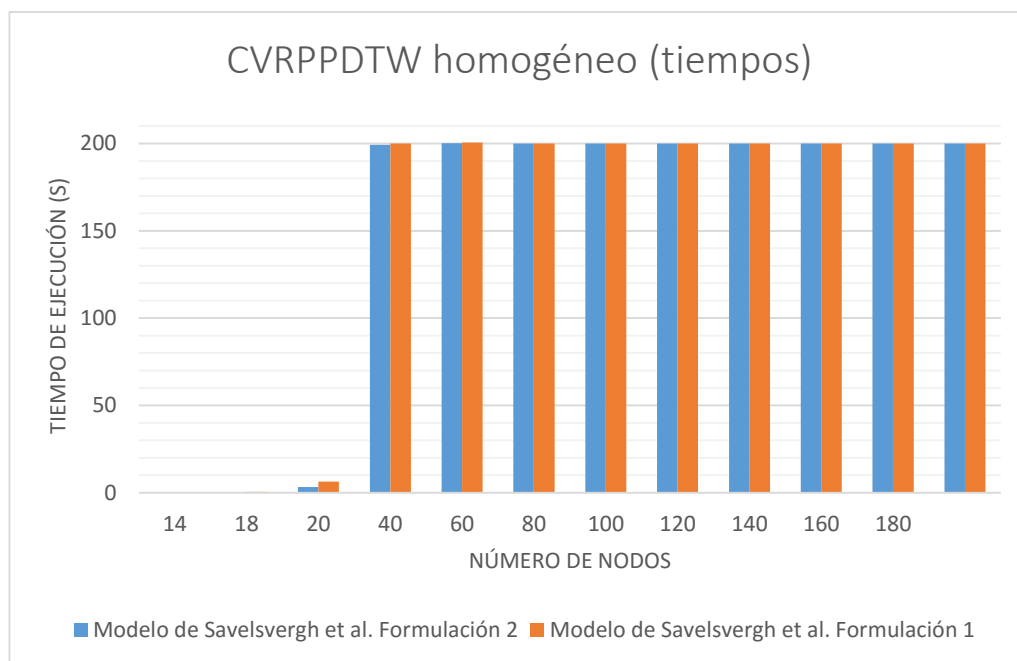


Figura A.11: Tiempos de ejecución. Modelo de Savelsbergh et al.

Datos empleados		Modelo de Savelsvergh et al. Formulación 2		Modelo de Savelsvergh et al. Formulación 1	
Nombre	Nº de nodos	Mejor solución factible (km)	Tiempo de ejecución (s)	Mejor solución factible (km)	Tiempo de ejecución (s)
PDTW7_homog	14	308,083	0,11	308,083	0,115
PDTW9_homog	18	354,532	0,382	354,532	0,546
PDTW10_homog	20	368,903	3,322	367,361	6,306
PDTW20_homog	40	857,148	199,255	831,543	200,064
PDTW30_homog	60	2548,56	200,122	2529,3	200,545
PDTW40_homog	80	#N/A	200	#N/A	200
PDTW50_homog	100	#N/A	200	#N/A	200
PDTW60_homog	120	#N/A	200	#N/A	200
PDTW70_homog	140	#N/A	200	#N/A	200
PDTW80_homog	160	#N/A	200	#N/A	200
PDTW90_homog	180	#N/A	200	#N/A	200

Tabla A.4: Resumen de resultados obtenidos. Modelo de Savelsbergh et al.

A.3.8. Conclusiones obtenidas. Problema CVRPPDTW.

Al contrario de lo que ocurría en los problemas anteriores, en este caso la Formulación 2 no logra calcular unas distancias mejores que la Formulación 1, obteniendo peores resultados en 3 de 5 casos, por 2 empates.

Los tiempos de ejecución, sin embargo, sí que muestran una leve mejoría respecto a la Formulación 1. Pero como lo que estamos buscando es que la Formulación 2 nos valga para obtener unas distancias totales recorridas menores reduciendo el número de vehículos empleado, no consideramos esta leve mejoría como significativa.

En base a las conclusiones anteriores, consideramos que la Formulación 2 no es válida para conseguir reducir el número de vehículos empleado siempre y cuando esto suponga una distancia total recorrida menor en el problema CVRPPDTW. Por ello, los modelos del citado problema serán planteados según su formulación original, sin realizar modificaciones en ninguna de sus restricciones.

Bibliografía

- Acharya, S. (2013). "Vehicle Routing and Scheduling Problems with time windows constraints. Optimization Based Models". *Int Jr. of Mathematical Sciences & Applications*.
- Angelelli, E., & Mansini, R. (2002). "The Vehicle Routing Problem with Time Windows and Simultaneous Pick-up and Delivery". *Quantitative Approaches to Distribution Logistics and Supply Chain Management*.
- Araújo Araúzo, J. (2016). "Dirección de operaciones". Apuntes de clase.
- Benito Quintanilla, A. (2015). "Problemas de rutas de vehículos: modelos, aplicaciones logísticas y métodos de resolución". Valladolid: Trabajo Fin de Grado.
- Berbeglia, G., Cordeau, J.-F., & Laporte, G. (2010). "Dynamic pickup and delivery problems". *European Journal of Operational Research*, 8-15.
- Caric, T., & Gold, H. (2008). "Vehicle routing problem". In-teh.
- Cook, W. (n.d.). *Concorde sitio web*. Retrieved from <http://www.math.uwaterloo.ca/tsp/concorde/index.html>
- Cook, W., & Rich, J. (1999). "A parallel cutting plane algorithm for the vehicle routing problem with time windows". *Computational and Applied Mathematics, Rice University, Houston*.
- Cordeau, J.-F., Desaulniers, G., Desrosiers, J., M.Solomon, M., & Soumis, F. (2000). "The VRP with Time Windows". *Les Cahiers du GERARD*.

- Cordeau, J.-F., Laporte, G., & Ropke, S. (2007). "Recent Models and Algorithms for One-to-One Pickup and Delivery Problems". *Canada Research Chair in Logistics and Transportation, HEC Montréal*, 3-4.
- Cordone, R., & Wolfler Calvo, R. (2001). "A Heuristic for the Vehicle Routing Problem with Time Windows". *Journal of Heuristics*, 107-129.
- Dash, S., Günlük, O., Lodi, A., & Tramontani, A. (2009). "A Time Bucket Formulation for the TSP with Time Windows".
- Desrosiers, J., Soumis, F., Desrochers, M., & Sauvé, M. (1986). "Vehicle routing and scheduling with time windows". *Mathematical Programming Study*, 249-251.
- Dethloff, J. (2001). "Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up". *Operations Research Spektrum*, 79-96.
- Dumas, Y., Desrosiers, J., & Soumis, F. (1991). "The pickup and delivery problem with time windows". *European Journal of Operational Research*, 7-22.
- E. Miller, C., W. Tucker, A., & A. Zemlin, R. (n.d.). *Integer Programming Formulation of Traveling Salesman Problems*.
- Euler, L. (1741). "Comm. academiae scientiarum Petropolitanae". 128-40.
- Euler, L. (1766). "Solution d'une question curieuse qui ne paraît soumise à aucune analyse". *Opera Omnia*, 26-56.
- Fernández Hernández, A. (2016). *"Trabajo de Fin de Grado"*.

- Fico. (2012). *Xpress Optimization Suite. Software*. Retrieved from <http://www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx>
- Flatberg, T. (2009). "A short OPL tutorial".
- GAMS Development Corporation. (2016). "GAMS-Commercial Price List-March 15, 2016".
- García de Jalón, J., Ignacio Rodríguez, J., & Brazález, A. (1999). "Aprenda Visual Basic 6.0 como si estuviera en primero".
- Gheysens, F., Golden, B., & Assad, A. (1984). "A Comparison of Techniques for Solving the Fleet Size and Mix Vehicle Routing Problem". *Operations Research Spektrum*, 207-216.
- Golden, B., Assad, A., Levy, L., & Gheysens, F. (1984). "The fleet size and mix vehicle routing problem". *Operations Research Spektrum*, 49-66.
- Golden, B., Raghavan, S., & Wasil, E. (2008). "The Vehicle Routing Problem. Latest Advances and New Challenges". Springer.
- Gurobi. (n.d.). *Gurobi optimization software*. Retrieved from <http://www.gurobi.com/>
- IBM. (n.d.). "IBM ILOG OPL V6.3 Language Reference Manual". International Business Machines Corporation.
- IBM. (n.d.). "ILOG OPL Studio Introduction".

- IBM. (2012). *IBM ILOG CPLEX Optimization Studio. Software*. Retrieved from <http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/>
- J.Cook, W. (2012). *"In Pursuit of the Traveling Salesman"*. Princeton University Press.
- Laporte, G. (1991). "The Vehicle Routing Problem: an overview of exact and approximate algorithms". *European Journal of Operational Research*, 345-358.
- Laporte, G. (2006). *"A Short History Of Traveling Salesman Problem"*. HEC Montréal, Canada.
- Larsen, J. (1999). *"Parallelization of the vehicle routing problem with time windows"*. Lyngby, Denmark: Ph.D. Thesis IMM-PHD-1999-62, Department of Mathematical Modelling, Technical University of Denmark.
- Librería CVRP homogéneo. (n.d.). Retrieved from <http://or.dei.unibo.it/library/vrplib-vehicle-routing-problem-library>
- Librería CVRPTW homogéneo. (n.d.). *Obtención ejemplo C201 como base de los problemas CVRPTW. Obtención mejores resultados conocidos Solomon para problema CVRPTW*. Retrieved from <http://w.cba.neu.edu/~msolomon/problems.htm>
- Librería TSP. (n.d.). Retrieved from <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/>
- M.Solomon, M. (1987). "Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints". *Operations Research*, 254-265.

- Meindl, B., & Templ, M. (2012). "Analysis of commercial and free and open source solvers for linear optimization problems".
- N.Christofides, A.Mingozzi, & P.Toth. (1979). "The vehicle routing problem". *Combinatorial Optimization, Wiley, Chichester, UK.*
- N.Kritikos, M., & Ioannou, G. (2013). "The heterogeneous fleet vehicle routing problem with overloads and time windows". *Int.J.Production Economics*, 68-75.
- N.Parragh, S., F.Doerner, K., & F.Hartl, R. (2008). "A survey on pickup and delivery problems". Part1. *JfB*, 21-51.
- N.Parragh, S., F.Doerner, K., & F.Hartl, R. (2008). "A survey on pickup and delivery problems".Part2. *JfB*, 81-117.
- Pérez de Vargas Moreno, B. (2015). "Resolución del Problema del Viajante de Comercio (TSP) y su variante con Ventanas de Tiempo (TSPTW) usando métodos heurísticos de búsqueda local". Trabajo Fin de Grado.
- Quan Lu, & Dessouky, M. (2002). "An Exact Algorithm for the Multiple Vehicle Pickup and Delivery Problem".
- Rocha, L., González , C., & Orijuela, J. (2011). "Una revisión al estado del arte del problema del ruteo de vehículos: Evolución histórica y métodos de resolución". *Ingeniería*, 35-55.
- Sáez Aguado, J. (2014). "Introducción a Xpress-Mosel". Apuntes de clase.
- Sáez Aguado, J. (2014). "Métodos matemáticos en ingeniería de organización I". Apuntes de clase.

Sáez Aguado, J. (2016). "Arrays y conjuntos estáticos y dinámicos. Uso de *finalize* y de *create*". Apuntes de clase.

Sáez Aguado, J. (2016). "El problema de flujo multiproducto de coste mínimo en redes: descripción y propiedades".

Sáez Aguado, J. (2016). "El problema TSP".

Sáez Aguado, J. (2016). "Modelo de dos índices para el problema de rutas con capacidades". Apuntes de clase.

Sáez Aguado, J. (2016). "Modelo de flujo en redes que no precisa conocer de antemano el número de vehículos".

Sáez Aguado, J. (2016). "Problemas de rutas de vehículos".

Savelsbergh, M., & M.Sol. (1995). "The General Pickup and Delivery Problem".

Toth, P., & Vigo, D. (2002). "Models, relaxations and exact approaches for the capacitated vehicle routing problem". *Discrete Applied Mathematics* 123, 487-512.

Toth, P., & Vigo, D. (2002). "The Vehicle Routing Problem". *Editorial siam*.

Xpress, F. (2014). "Xpress-Mosel Reference manual".