

TESIS DOCTORAL

**Técnicas de clustering aplicadas a la
resolución de problemas de
optimización combinatoria con
restricciones espaciales y temporales**

**Autor: Emilio Martín Gallardo
Directores: Pedro Isasi Viñuela
Yago Sáez Achaerandio**

**Doctorado en Ciencia y tecnología informática
Universidad Carlos III de Madrid**

Leganés, septiembre 2018

Este documento está preparado para ser imprimido a doble cara.

Técnicas de clustering aplicadas a la resolución de problemas de optimización combinatoria con restricciones espaciales y temporales

Autor: Emilio Martín Gallardo
Directores: Pedro Isasi Viñuela
Yago Sáez Achaerandio

Doctorado en Ciencia y tecnología informática
Universidad Carlos III de Madrid

Leganés, septiembre 2018

*A mi mujer Laura
y
a nuestro hijo Marcos E.*

A mis padres, Emilio y Dami.

Agradecimientos

*Ingrato es quien niega el beneficio recibido;
ingrato es quien lo disimula, más ingrato es quien
no lo devuelve, y mucho más ingrato quien se
olvida de él.*

Séneca

Esta es quizás la sección más personal de mi Tesis, en ella simplemente quiero agradecer a todas aquellas personas que de un modo u otro han contribuido a que esta investigación sea posible.

En primer lugar a mis **padres**, sin su apoyo y sacrificio nada de lo que ahora escribo sería posible, por haber creído siempre en mí y por animarme de forma incondicional. A mi familia más directa, a mi hermana M^a Luz y a mi sobrino.

A Yago Sáez, quien ha estado presente en toda mi etapa como investigador desde mi PFC hasta finalmente mi Tesis, pasando por mi máster, sin duda el me brindó la oportunidad de iniciarme en tareas de investigación, lo cual le agradezco profundamente. A Pedro Isasi, con el que he tenido el placer de llevar a cabo esta investigación y quien sin duda ha resultado clave para la finalización de la misma. Muchas gracias a ambos por vuestro apoyo, consejos, confianza, dedicación e implicación personal sin las cuales, esta investigación no habría podido realizarse. Muchas gracias a los dos por vuestra infinita paciencia, sé que este no ha sido un camino fácil. Y por supuesto a todos los miembros de grupo EVANNAI que también han colaborado y posibilitado de forma activa esta investigación.

A mis amigos de siempre, a los que he descuido un poco en estos últimos años con “mis cosas que hacer” y con los que espero poder compartir más tiempo de ahora en adelante. A mis compañeros de las distintas compañías en las que he trabajado mientras desarrollaba esta investigación ¹. Muy en especial a mis compañeros de Decide Soluciones (Ángel, Valbuena, Peter, Sergi, Pablum, Dani, Jorge, Marcos y J. Lafuente) y a mi equipo de Cabify (Ares, Dani, Guille, Gabi y Carlos), con todos ellos he aprendido y compartido muchas ideas sobre optimización, aunque sin duda lo que más valoro son los buenos momentos que hemos compartido.

Y por último a mi mujer **Laura**, quien sin duda ocupa un papel muy relevante en esta investigación. Ella fue quien me animó a retomar de nuevo mis estudios para obtener mi doctorado, cuando yo buscaba nuevos másteres con los que continuar formándome. “Gracias por animarme a empezar con todo esto, pero sobre todo por ayudarme y permitirme terminarlo, aun cuando sin duda tú has resultado la mayor perjudicada. Sin tu cariño, ánimo, apoyo y comprensión esta investigación no se habría podido llevar a cabo, así que esta Tesis también es tuya”. Y por último a nuestro hijo Marcos Emilio, quien sin saberlo nos ha transportado en estos últimos meses hasta un nuevo, mejor y todavía algo inexplorado

¹Lo cual desaconsejo encarecidamente a cualquier estudiante que considere hacer un doctorado.

espacio de soluciones.

Gracias a todos.

Resumen

En esta investigación se aborda la resolución de la familia de problemas denominada problemas de planificación de asistentes de atención domiciliaria, conocida por sus siglas en inglés HCSP (Home Care Scheduling Problem). La importancia en la resolución de esta familia de problemas ha ido en aumento durante los últimos años, importancia que también se ha visto reflejada en el mundo académico con un creciente número de publicaciones tal y como reflejan dos recientes revisiones sobre estado del arte [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#). Este interés está motivado principalmente por dos factores, a saber, el envejecimiento de nuestra población y la necesidad de buscar alternativas más eficientes en la prestación de servicios de atención domiciliaria. El envejecimiento progresivo de nuestra población, es quizás uno de los retos más importantes que deberemos afrontar como sociedad, reto que ha sido señalado por diversos organismos tanto europeos como internacionales ([World Health Organization, 2015](#)). En dicho contexto demográfico, y con una creciente demanda de peticiones de asistencia domiciliaria, las empresas prestadoras de servicios de atención domiciliaria deben buscar formas más eficientes y alternativas de satisfacer la creciente demanda, preservando la calidad del servicio y garantizando unos servicios de atención domiciliaria de calidad y sostenibles para nuestros mayores.

Esta investigación se centra en la resolución de un problema real y de fácil transferencia a la industria, reportado por la compañía EULEN. Dicha compañía presta servicios de atención domiciliaria en la Comunidad de Madrid y debe atender anualmente alrededor de 1.5 millones de servicios, lo que se traduce en 2.1 millones de horas de trabajo, no incluyendo estas cifras las ineficiencias inherentes a la prestación de servicios, como son los desplazamientos y los tiempos de espera incurridos por los asistentes. El problema es abordado semanalmente, debiendo atender a 13.344 servicios cada semana. Estos servicios a su vez están formados por tareas, que deben prestarse a una hora concreta y en una localización particular. De estos servicios, el 80% debe ser atendido de lunes a viernes en horario matinal (07:00 a 14:30) con lo que alrededor de 10.700 servicios deben ser planificados y asignados a la vez. El tamaño del problema abordado, el cual está un orden de magnitud por encima de los abordados en el estado del arte, junto con la imposibilidad de dividir el problema en instancias más pequeñas ha requerido el diseño e implementación de técnicas específicas, a fin de poderlo resolver en un tiempo y con un coste que posibilite la operativa diaria de la compañía prestadora de servicio.

El problema se aborda desde la perspectiva del clustering, consistiendo su resolución en la agrupación de servicios dentro de grupos de servicios o clústers, los cuales conformarán el horario de trabajo de cada asistente de atención domiciliaria. La utilización de dicha perspectiva presenta varios desafíos que han sido resueltos a lo largo de la presente investigación, entre ellas destacan: el inusual tamaño de las instancias a resolver, las restricciones a respetar que contemplan aspectos espaciales y temporales, así como la necesidad de definir un concepto de similitud o distancia entre. Dicho concepto de similitud es definido a fin de tener en cuenta las componentes espaciales y temporales del problema. Una vez definido, el problema se aborda con tres técnicas novedosas: la primera de ellas es un método de clustering jerárquico inspirado en el método de [Ward \(1963\)](#), mientras que

las dos aproximaciones restantes se basan en la metaheurística conocida como optimización basada en colonias de hormigas [Dorigo y Gambardella \(1997\)](#) y se denominan ACS-HCSP y IACS-HCSP.

Todas las técnicas son evaluadas experimentalmente de un modo exhaustivo, con un total de 96 configuraciones distintas adaptadas a diferentes entornos. En primer lugar, se comparan las técnicas propuestas entre sí a fin de determinar su rendimiento, tanto como en calidad como en tiempo de ejecución, realizándose los pertinentes análisis de significación estadística. Una vez determinada las técnicas que tienen un mejor rendimiento se pasa a comparar de modo exhaustivo las técnicas ACS-HCSP y IACS-HCSP a fin de determinar si las modificaciones propuestas para la técnica IACS-HCSP producen mejoras significativas, obteniéndose un resultado positivo. Finalmente y con el objetivo de comparar las técnicas propuestas con técnicas existentes en el estado del arte, las técnicas propuestas son comparadas con las heurísticas propuestas por [Quintana et al. \(2017\)](#) y con la solución actual de la compañía, obteniendo la técnica IACS-HCSP mejores resultados y permitiendo un ahorro económico estimado de 3.7 millones de euros anuales.

Palabras clave: Clustering, metaheurísticas, problemas de optimización combinatoria, optimización basada en colonias de hormigas, problemas de planificación de asistentes de atención domiciliaria.

Abstract

This research addresses the resolution of the family of problems called Home Care Scheduling Problem (HCSP) problems. The importance of solving this family of problems has been increasing in recent years, an importance that has also been reflected in the academic world with a growing number of publications [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#). This interest is motivated mainly by two factors, namely, the ageing of our population and the need to seek more efficient alternatives in the provision of home care services. The progressive ageing of our population is perhaps one of the most important challenges that we will face as a society, a challenge that has been identified both by European and international bodies [World Health Organization \(2015\)](#). In this demographic context, and with the growing demand for home-based care requests, home-based care service providers must look for more efficient and alternative ways to meet the growing demand, while preserving the quality of service and ensuring quality and sustainable home-based care services for our elderly.

This research addresses a real and easily transferable problem to the industry, faced by the company EULEN. The company provides home care services in the Community of Madrid and must provide around 1.5 million services per year, which amounts 2.1 million working hours, not including the inefficiencies inherent in the provision of services, such as travel and waiting times incurred by caregivers. The problem is addressed on a weekly basis, with 13344 services to be scheduled to each week. These services in turn consist of tasks, which must be provided at a specific time and at a particular location. Of these services, 80 percent must be provided Monday through Friday in morning shift (07:00 to 14:30) so that about 10,700 services must be planned and assigned at the same time. The size of the problem addressed is one order of magnitude higher than those addressed in the state of the art, along with the impossibility of dividing the problem into smaller instances has required the design and implementation of specific techniques for its resolution.

The problem is tackled from the perspective of clustering, consisting of resolving it by grouping services into groups of services or clusters, which will conform the timetable of each assistant. The main difficulty of tackling the problem from the perspective of clustering, in addition to its unusual size, are its restrictions and the concept of similarity or distance between clusters. This concept is defined to take into account the spatial and temporal components of the problem. Once defined, the problem is tackled with three novel techniques: the first is a hierarchical clustering method inspired by the Ward [Ward \(1963\)](#) method, while the other two approaches are based on the metaheuristics Ant Colony Optimization proposed by [\(Dorigo y Gambardella, 1997\)](#) and are called ACS-HCSP and IACS-HCSP.

All techniques are evaluated experimentally in a comprehensive manner, with a total of 96 different configurations adapted to different environments. First, the proposed techniques are compared with each other in order to determine their performance, both in terms of quality and execution time, and the relevant analyses of statistical significance are carried out. Once the techniques with the best performance have been determined, a comprehensive

comparison of the ACS-HCSP and IACS-HCSP techniques is made to determine whether the proposed modifications to the IACS-HCSP technique produce significant improvements, with a positive result. Finally, and with the aim of comparing the techniques proposed with existing state-of-the-art techniques, the proposed techniques are compared with the heuristic techniques proposed by [Quintana et al. \(2017\)](#) and with the company's current solution, obtaining the IACS-HCSP technique with better results and allowing an estimated economic yearly saving of 3.7 million euros.

Keywords: Clustering, metaheuristics, combinatorial optimization problems, ant colony optimization, home care scheduling problems.

Índice

Agradecimientos	IX
Resumen	XI
Abstract	XIII
1. Introducción	1
1.1. Introducción	2
1.2. Problemas de optimización	5
1.2.1. Tipos de problemas	6
1.2.2. Problemas de optimización combinatoria	7
1.2.3. Técnicas de resolución	7
1.3. Análisis clúster	8
1.4. Objetivos	12
1.5. Estructura	14
2. Estado del Arte	17
2.1. Problema del Viajante	18
2.1.1. Introducción	18
2.1.2. Evolución histórica	19
2.1.3. Formulación matemática	24
2.1.4. Tipos de problemas	26
2.1.4.1. Problema del Viajante simétrico.	26
2.1.4.2. Problema del Viajante asimétrico.	26
2.1.4.3. Problema del Viajante con Ventanas de Tiempo	26
2.1.4.4. Problema de m-vendedores de comercio	30
2.2. Problema de Enrutamiento de Vehículos	34
2.2.1. Introducción	34
2.2.2. Tipos de problemas	37
2.2.3. Problema de Enrutamiento de Vehículos con restricciones de Capacidad	39
2.2.3.1. Métodos de resolución	42
2.2.4. Problema de Enrutamiento de Vehículos con Ventanas de Tiempo	47
2.2.4.1. Métodos de resolución	49

2.3.	Problema de planificación de asistentes de atención domiciliaria	53
2.3.1.	Introducción	53
2.3.2.	Restricciones	55
2.3.2.1.	Restricciones relativas a los proveedores de servicios	56
2.3.2.2.	Restricciones de los clientes	58
2.3.2.3.	Restricciones de los asistentes	60
2.3.3.	Objetivos	62
2.3.3.1.	Minimizar el coste de la ruta	63
2.3.3.2.	Minimizar el número de clientes sin asignar	63
2.3.3.3.	Minimizar el número asistentes requeridos	63
2.3.3.4.	Maximizar la satisfacción	63
2.3.4.	Métodos de resolución	64
2.3.4.1.	Periodo único	64
2.3.4.2.	Periodo múltiple	67
2.4.	Optimización Basada en Colonias de Hormigas	70
2.4.1.	Principios biológicos	71
2.4.2.	Evolución histórica	74
2.4.3.	Ant System (AS)	74
2.4.4.	Elitismo en Ant System (AS)	76
2.4.5.	Ant Colony System (ACS)	76
2.4.6.	Max-Min Ant System (MMAS)	77
2.4.7.	Sistema mejor-peor hormiga (SMPH)	78
2.4.8.	Marco teórico ACO	78
2.4.9.	Aplicaciones ACO	79
3.	Caso de estudio: HCSP en la Comunidad de Madrid	81
3.1.	Introducción	82
3.2.	Descripción del problema	83
3.3.	Obtención de las instancias	85
3.3.1.	Geocodificación	86
3.3.2.	Distancias	88
3.3.3.	Particionamiento	90
3.4.	Formulación matemática	96
4.	Técnicas propuestas	99
4.1.	Introducción	100
4.2.	Clustering Jerárquico	100
4.2.1.	Similitud	102
4.2.1.1.	Tareas	102
4.2.1.2.	Servicios	104
4.2.2.	Método de Ward modificado	105
4.3.	Optimización basada en colonia de hormigas	107
4.3.1.	Creación del grafo	107

4.3.2.	ACS-HCSP	109
4.3.2.1.	Selección del vértice de inicio	110
4.3.2.2.	Regla de transición entre vértices	111
4.3.2.3.	Finalización clúster actual	112
4.3.2.4.	Actualización de feromona y elitismo	112
4.3.2.5.	Función de evaluación	113
4.3.2.6.	Pseudocódigo	113
4.3.3.	IACS-HCSP	115
4.3.3.1.	Construcción de soluciones	115
4.3.3.2.	Vértices alcanzables	117
4.3.3.3.	Criterio de parada	119
5.	Experimentación	123
5.1.	Introducción	124
5.1.1.	Objetivos y restricciones	136
5.1.2.	Parametrización	137
5.1.3.	Hardware utilizado	139
5.2.	Análisis del método de Ward modificado	139
5.2.1.	Parametrización	139
5.2.2.	Resultados	140
5.2.3.	Áreas de mejora	144
5.3.	Análisis ACS-HCSP	146
5.3.1.	Parametrización	146
5.3.2.	Resultados	147
5.3.2.1.	Función objetivo: F_{clust}	148
5.3.2.2.	Función objetivo: F_{cost}	152
5.4.	Análisis IACS-HCSP	156
5.4.1.	Parametrización	156
5.4.2.	Resultados	157
5.4.2.1.	Función objetivo: F_{clust}	157
5.4.2.2.	Función objetivo: F_{cost}	162
5.5.	Análisis comparativo	166
5.5.1.	Método de Ward modificado	166
5.5.2.	ACS-HCSP y IACS-HCSP	168
5.5.2.1.	Función objetivo: F_{clust}	168
5.5.2.2.	Función objetivo: F_{cost}	177
5.5.3.	Comparativa con (Quintana et al., 2017)	190
6.	Conclusiones	193
6.1.	Objetivos	194
6.2.	Aportaciones	196
6.3.	Limitaciones y líneas futuras	199

A. Resultados detallados	203
A.1. Ward	203
A.2. ACS-HCSP	204
A.3. IACS-HCSP	220
B. Formulación alternativa	235
B.1. Formulación matemática	235
B.1.1. Objetivos	235
B.1.2. Restricciones	235
Bibliografía	241

Índice de figuras

1.1. Comparativa de previsión por edades entre el año 2015 y el 2080 para la Europa de los 28, datos obtenidos de European Commission (2015)	3
1.2. Previsión de la composición por edades en la Europa de los 28 datos obtenidos de European Commission (2015)	3
1.3. Esquema adaptado de Talbi (2009) en el que se muestran distintas técnicas de optimización.	8
1.4. Como se puede apreciar ninguna de las dos agrupaciones es más exacta, ni mejor que la otra, simplemente se utilizan criterios distintos.	10
1.5. Dos ejemplos de resultados distintos al realizar un proceso de clustering, obteniéndose en cada uno de ellos dos clústers y cuatro clusters.	11
2.1. Ejemplo de ciclo o circuito Hamiltoniano para el grafo de un dodecaedro. El dodecaedro al igual que el resto de grafos platónicos contienen ciclos hamiltonianos.	19
2.2. Ejemplo de la instancia resuelta por (Dantzig et al., 1954) publicada en la revista Newsweek.	20
2.3. Panfleto publicitario de la compañía Procter Gamble en el que se ofrecía un premio de 10.000 \$ a quien encontrara la mejor solución para una instancia de 33 ciudades.	20
2.4. Ejemplo de la instancia resuelta por D. Applegate, R. Bixby, V. Chvátal, W. Cook, and K. Helsgaun. en 2004 para un total de 24978 ciudades de Suecia.	22
2.5. Ejemplo de la instancia resuelta por D. Applegate et al. para el diseño de un circuito electrónico integrado a gran escala con un total de 85.900 vértices.	23
2.6. Ejemplo de la instancia propuesta y resuelta por Robert Bosch para una instancia con un total de 100.000 ciudades en la que se representa la obra pictórica de Leonardo Da Vinci Mona Lisa.	24
2.7. Ejemplo en el que se muestran dos subciclos para una instancia del TSP con 7 nodos.	25
2.8. Ejemplo en el que se aprecian las diferencias entre utilizar una distancia Euclídea y la distancia real proporcionada por un sistema de información geográfica (Google Maps).	27
2.9. Ejemplo de tiempo total invertido en el <i>servicio_i</i> , como resultado del tiempo de espera más tiempo de servicio β_i , así como la venta ventana temporal $[a_i, b_i]$ el <i>servicio_i</i>	28

2.10. Ejemplo de la mejor solución para la instancia n20w120.002 propuesta por (Gendreau et al., 1998). El nodo inicial del que se debe iniciar la ruta es el nodo 1, se aprecia como el grafo resultante de la solución del mismo no es planar.	29
2.11. Ejemplo de la mejor solución para la instancia rat99 de la librería TSPLIB, adaptada al problema m-TSP compuesta por un total de 99 ciudades y 7 viajantes de comercio, resuelta por Necula et al. (2015a).	33
2.12. Ejemplo de la mejor solución para la instancia rat99 de la librería TSPLIB, adaptada al problema min-max m-TSP compuesta por un total de 99 ciudades y 7 viajantes de comercio, resuelta por Necula et al. (2015b).	33
2.13. Ejemplo de resolución para el problema CVRP, en el que existen cuatro vehículos con una capacidad de 10. La demanda de cada cliente viene denotada por d_i y los costes de transición entre nodos en cada uno de los arcos.	41
2.14. Solución óptima para la instancia X-n275-k28 de la librería CVRLIB propuesta por Uchoa et al. (2016). La solución óptima tiene un coste de 21245 y utiliza 28 vehículos cada uno de ellos con una capacidad $Q = 10$	44
2.15. Heurística de ahorro propuesta por Clarke y Wright (1964), en la parte izquierda se observan dos clientes i, j que son visitados en rutas distintas, estando el depósito representado por el nodo 0 en ambos casos.	46
2.16. Ejemplo de operador 2-opt, en el que las aristas (1,3) y (2,4) de la parte izquierda son sustituidas por las aristas (2,4) y (3,4) en la parte derecha. Caso particular con $\lambda = 2$ del operador $\lambda - opt$ propuesto por Lin (1965)	46
2.17. Solución óptima para la instancia R102 propuesta por Solomon en 1987 Solomon (1987).	50
2.18. Solución óptima para la instancia C102 propuesta por Solomon en 1987 Solomon (1987).	51
2.19. Experimento de puente binario con distintas longitudes de arcos (Goss et al., 1989).	72
2.20. Detalle puente binario con las mismas longitudes de arcos	73
2.21. Evolución del porcentaje de hormigas que eligen un camino u otro.	73
2.22. Pseudocódigo de Ant System	75
2.23. Comparativa de ACS respecto a otras técnicas para el problema del TSP. En negrita se muestra la mejor solución entera encontrada por el algoritmo, entre paréntesis la mejor solución real, y entre corchetes el número de tours necesarios para encontrarlos (Dorigo y Gambardella, 1997).	77
2.24. Aplicaciones de ACO a distintos problemas	80
3.1. Ejemplo de solución al problema mostrado en la tabla 3.1, la cual está formada por dos clústers, cada uno de ellos compuestos por dos y un servicio respectivamente.	85
3.2. Ejemplo de la respuesta obtenida al consultar el servicio web de Google de Geocodificación con la dirección “41 Calle de Casalarreina”, en formato JSON. 87	

3.3.	Captura de pantalla proporcionada por el Instituto de Estadística de la Comunidad de Madrid.	88
3.4.	Representación de los 13344 servicios geolocalizados utilizando la herramienta Google Maps.	90
3.5.	Representación de las 29034 tareas geolocalizadas utilizando la herramienta Google Maps y agrupadas de acuerdo a su proximidad.	92
3.6.	Ejemplo de la respuesta obtenida al consultar el servicio web de Google de direcciones con las coordenadas geográficas (40.421501, -3.7002308) y (40.4210003, -3.701397). Entre otros datos se muestra la distancia total en kilómetros y la duración estimada, teniendo en cuenta que la ruta se realizará a pie.	93
3.7.	Ruta a pie entre tres direcciones obtenidas y representada utilizando la herramienta Google Maps.	94
3.8.	Ilustración de como la distancia Manhattan puede ser una buena aproximación, dada la geometría de algunos distritos de la ciudad de Madrid y teniendo en cuenta que el recorrido se realiza a pie.	95
4.1.	Ejemplo de clustering jerárquico aglomerativo, se puede ver como todos los elementos A,B,C,D,E,F comienzan en su propio clúster, para posteriormente ir agrupándose de modo jerárquico hasta formar un clúster formado por todos los elementos iniciales.	101
4.2.	Ejemplo del concepto de similitud $D(t_1, t_2)$ para dos tareas t_1, t_2 , resultado de sumar el tiempo de desplazamiento, el tiempo de espera y el tiempo de realización de la tarea t_2 . Nótese que el valor h_2 que representa la hora de llegada a la tarea t_2 coincide con el comienzo del tiempo de espera.	104
4.3.	Ejemplo del concepto de similitud $D(S_1, S_2)$ para dos servicios S_1, S_2 , resultado de suma la distancia $D(t_i, t_j)$ dentro del horizonte de planificación. Nótese como el tiempo de desplazamiento entre las tareas del servicio S_1 y las tareas del servicio S_2 es constante y simétrico.	105
4.4.	Ejemplo de construcción de grafo con tres tareas T_1, T_2, T_3 , en el cual se ilustra cómo se crea el grafo para un conjunto de tareas, existiendo una arista entre cada par de tareas (t_i, t_j) siempre que $D(t_i, t_j) \neq \infty$	108
4.5.	Ejemplo de construcción de grafo con tres servicios S_1, S_2, S_3 , como se puede apreciar para cada par de servicios entre los que existe una distancia definida, se crea una arista. A modo de ejemplo se muestra el concepto de incompatibilidad entre dos servicios $D(S_1, S_3) = \infty$, al no respetarse el tiempo máximo de desplazamiento.	109
4.6.	Ejemplo en el que se muestran las distintas soluciones que se obtendrían para un problema con tres servicios S_1, S_2, S_3 , enumerando los posibles caminos recorridos por las hormigas.	116
4.7.	Ejemplo de recorrido detallado que realizaría una hormiga k si visitara los vértices en el siguiente orden S_1, S_2, S_3 , lo cual daría lugar a la solución $[C_1 = \{S_1, S_2, S_3\}]$	117

4.8.	Ejemplo de recorrido detallado que realizaría una hormiga k si visitara los vértices en el siguiente orden S_2, S_1, S_3 . Tal y como se puede apreciar al visitar el vértice S_1 , $N_{S_1}^k = \emptyset$ por lo que se comenzaría con la formación de otro clúster, obteniendo la solución $[C_1 = \{S_2, S_1\}, C_2 = \{S_3\}]$	118
4.9.	Ejemplo de función logística con valores $L = 1$, $x_0 = 0$ y valores de $k=[1, 0.5]$	121
5.1.	Gráfico en el que se representa el número total de tareas por grupo de servicios y el número total de servicios presentes en cada uno de dichos grupos.	125
5.2.	Gráfico en el que se representa la hora de comienzo de las tareas y la hora de fin por grupo de servicios.	126
5.3.	Diagrama de cajas en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo por cada turno de trabajo. Se observa una duración uniforme dentro de cada grupo de servicios, existiendo algunos valores atípicos representados con un rombo.	127
5.4.	128
5.5.	Gráfico de “violín” en el que se representa la duración de cada tarea, para cada día de la semana. Filtrándose todas aquellas tareas cuya duración sea superior de 150 minutos.	128
5.6.	Gráfico de “violín” en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo para cada grupo de servicios. Filtrándose todas aquellas tareas cuya duración sea superior de 150 minutos.	129
5.7.	Gráfico en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo por hora de inicio de la tarea y por cada grupo de servicios.	130
5.8.	Imagen en la que se muestra el total de 13.344 servicios, mostrándose en detalle una zona seleccionada.	131
5.9.	Mapa de calor con pesos en el que se representan todas las tareas de las distintas instancias, utilizando la librería Google Maps APIs (2016c)	133
5.10.	Mapa de calor con pesos para los distintos grupos de tareas.	134
5.11.	Histograma en 3D en el que los servicios se han agrupado espacialmente con un radio de 75m sumándose el total de horas requeridas.	135
5.12.	Valores medios obtenidos para cada uno de los objetivos Tot_{obj} y $Cluster_{obj}$ distinguiendo por tipo de agrupamiento y valor de $Wmax_k$. Se puede apreciar como el hecho de no limitar el tiempo máximo de espera provoca que la distancia total aumente y el número de asistentes requeridos disminuya. . .	142
5.13.	Valores medios obtenidos para cada uno de los objetivos $Wait_{obj}$ y $Travel_{obj}$ distinguiendo por tipo de agrupamiento y valor de $Wmax_k$. Nótese que los valores del eje Y no están en la misma escala.	143
5.14.	Histograma en el que se muestra el número de horas requeridas para cada tipo de agrupamiento y valor de tiempo máximo de espera $Wmax_k$	144
5.15.	Gráfico de dispersión segmentado por el valor de tiempo máximo de espera $Wmax_k$, en el que se muestran los resultados obtenidos por los 60 experimentos realizados para el método de Ward modificado.	145

5.16. Ejemplo de la problemática que ocasiona utilizar el método de Ward basado en estrategias “avariciosas”. La solución resultante contendrá dos clústers, el $C_1 = \{S_2, S_3\}$ y $C_2 = \{S_1, S_4\}$ que no será óptima, ya que la solución formada por los clústers $C_1 = \{S_1, S_2, S_3\}$ y $C_2 = \{S_4\}$ obtendría una menor distancia y requeriría el mismo número de asistentes.	146
5.17. Resultados resumidos obtenidos para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust}	150
5.18. Coste computacional (medido en segundos) para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust}	151
5.19. Resultados resumidos obtenidos para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost}	154
5.20. Coste computacional (medido en segundos) para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost}	155
5.21. Resultados resumidos obtenidos para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust}	160
5.22. Coste computacional (medido en segundos) para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust}	161
5.23. Resultados resumidos obtenidos para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost}	164
5.24. Coste computacional (medido en segundos) para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost}	165
5.25. Función de densidad para los tres métodos propuestos, tipo de agrupamiento B y función de coste F_{clust}	167
5.26. Función de densidad sobre el coste de las soluciones obtenidas en la figura 5.25.	168
5.27. Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento A.	171
5.28. Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento A.172	
5.29. Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento B.	173
5.30. Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento B.174	
5.31. Distribuciones de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP durante las 100 rondas de evolución para la función de evaluación F_{clust} , el tipo de agrupamiento B y el grupo de servicios 1.	177
5.32. Evolución de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} , el tipo de agrupamiento B y el grupo de servicios 1.	180
5.33. Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento A.	181
5.34. Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento A.182	

-
- 5.35. Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento B. 183
- 5.36. Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento B. 184
- 5.37. Distribuciones de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP durante las 100 rondas de evolución para la función de evaluación F_{cost} , el tipo de agrupamiento B y el grupo de servicios 1. 188
- 5.38. Evolución de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} , el tipo de agrupamiento B y el grupo de servicios 1. 189
- 5.39. Comparativa entre las técnicas propuestas Ward, ACS-HCSP y IACS-HCSP respecto a las heurísticas propuestas por [Quintana et al. \(2017\)](#) 192

Índice de Tablas

2.1.	Evolución histórica de los tamaños de las distintas instancias resueltas, donde la columna año indica el año en el que se encontró la solución óptima a dicha instancia Applegate et al. (2007)	21
2.2.	Evolución histórica de las diferentes técnicas usadas para la resolución del Problema del Viajante con Ventanas de Tiempo.	31
2.3.	Características que determinan las distintas variantes de los Problemas de Enrutamiento de Vehículos. Tabla elaborada a partir de Desrochers et al. (1990) ; Cherif-Khettaf et al. (2014)	36
2.4.	Tabla con las principales tipos de problemas pertenecientes a la familia del VRP, se recomienda consultar Desrochers et al. (1990) ; Cherif-Khettaf et al. (2014)	39
2.5.	Tabla resumen con las distintas características de las instancias utilizadas por otros investigadores para el HCSP.	70
3.1.	Ejemplo de problema con tres servicios y seis tareas.	84
3.2.	Estructura del fichero inicial proporcionado por la compañía en el que se detallan los servicios que se presta a cada usuario, la localización de dicho de usuario, el auxiliar que realiza dichos servicios y el horario de realización de los mismos.	86
3.3.	Estructura del fichero obtenido una vez realizado el proceso de geocodificación y limpieza del fichero proporcionado por la compañía. Cada registro de dicho fichero recibe el nombre de servicio y cada de las visitas que se realizan en un día concreto y a una hora determinada recibe el nombre de tarea	89
5.1.	Resultados resumidos obtenidos con el método de Ward para las distintas configuraciones. El valor medio viene definido por μ y la desviación estándar por σ , habiéndose obtenido dichos valores tras repetir 10 veces cada una de las configuraciones.	140
5.2.	Valor de los parámetros utilizados por la técnica ACS-HCSP.	147
5.3.	Resultados resumidos obtenidos para ACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{clust}	149
5.4.	Resultados resumidos obtenidos para ACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{cost}	153

5.5.	Valor de los parámetros utilizados por la técnica IACS-HCSP.	156
5.6.	Resultados resumidos obtenidos para IACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{clust}	159
5.7.	Resultados resumidos obtenidos para IACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{cost}	163
5.8.	Resultados del test de normalidad Shapiro-Wilk para distintas configuraciones de las tres técnicas propuestas.	167
5.9.	Resultados obtenidos al realizar un test ANOVA para cada una de las tres poblaciones y configuraciones descritas.	167
5.10.	Comparación de resultados entre ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust}	170
5.11.	Prueba t-Student para las poblaciones de ACS-HCSP y IACS-HCSP para el Grupo 1 de servicios y función de evaluación F_{clust}	176
5.12.	Comparación de resultados entre ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost}	179
5.13.	Prueba t-Student para las poblaciones de ACS-HCSP y IACS-HCSP para el Grupo 1 de servicios y función de evaluación F_{clust}	186
5.14.	Comparativa entre los resultados obtenidos por las técnicas propuestas Ward, ACS-HCSP y IACS-HCSP, y las heurísticas propuestas por Quintana et al. (2017)	191
A.1.	Resultados detallados obtenidos con el método de Ward para las distintas configuraciones.	204
A.2.	Resultados detallados obtenidos con ACS-HCSP para las distintas configuraciones.	219
A.3.	Resultados detallados obtenidos con el método de IACS para las distintas configuraciones.	234
B.1.	Ejemplo de tarea	235
B.2.	Ejemplo de servicio	236
B.3.	Parámetros.	237

Capítulo 1

Introducción

*Un hombre que no arriesga nada por sus ideas, o
no valen nada sus ideas, o no vale nada el
hombre.*

Platón

Resumen: En este capítulo se realiza una breve introducción a los conceptos fundamentales sobre los que está articulada la presente Tesis. El principal objetivo de la misma es la resolución de problemas de optimización combinatoria con restricciones espaciales y temporales. Dentro de la infinidad de problemas de optimización combinatoria existentes, la familia de problemas tratados en esta investigación se conoce como problemas de planificación de asistentes de atención domiciliaria, problemas cuyo interés social y económico ha ido creciendo durante los últimos años, debido en gran medida a la evolución demográfica de nuestra sociedad. La resolución de esta familia de problemas se tratará desde dos perspectivas novedosas y complementarias, como son las técnicas de clustering y las metaheurísticas. Una vez presentadas dichas técnicas, se exponen las distintas vías de mejora detectadas, las cuales determinan las motivaciones y objetivos del presente trabajo. Finalmente, se muestra la estructuración en capítulos, resaltándose de modo sucinto sus ideas principales.

1.1. Introducción

Entre los distintos retos que deberemos afrontar como sociedad este nuevo siglo, el envejecimiento de nuestra población sea quizás uno de los más importantes, al tratarse de un fenómeno global y con profundas repercusiones sociales, económicas y culturales. Por primera vez en la historia de nuestra especie, la esperanza de vida media en todo el planeta supera los 60 años [United Nations \(2007\)](#). Según la Organización de Naciones Unidas (ONU) a principios de siglo, la población mundial estimada era de 6.127 millones de habitantes, en el año 2011 la población mundial superó los 7.000 [Bloom \(2011\)](#), para el año 2050 se estima que dicha cifra alcance los 9.700 millones y según las mismas previsiones se llegará a los 11.213 millones para el año 2100, habiendo duplicado en solo un siglo la población mundial. Este crecimiento no será uniforme en todas las regiones ni países, por ejemplo, en el caso de Europa y España se espera una reducción significativa del número de habitantes, Europa pasará de un total 738 millones de habitantes en 2015 a 706 en 2050, llegando a 645 millones en 2100. En el caso de nuestro país pasaremos de 46 millones en 2015 a 44 millones en 2050, llegando a los 38 millones en 2100 [United Nations \(2015\)](#). De acuerdo con la Organización Mundial de la Salud (OMS), entre 2000 y 2050 la proporción de la población mundial de personas mayores de 60 años se duplicará, pasando del 11 % al 22 %. En términos absolutos, el número de personas mayores de 60 años aumentará de 900 millones en 2015 a 1.400 millones en 2030, llegando a 2.100 millones en 2050, pudiendo llegar a 3.200 millones en 2100. En el año 2050, en Europa, las personas mayores de 60 años representarán un 34 % de nuestra población, un 25 % de América Latina y el Caribe y Asia, y en África a pesar de tener la estructura demográfica más joven se pasará de 46 millones en 2015 a 147 millones [World Health Organization \(2015\)](#). A nivel europeo todos estos estudios apuntan a que se producirá una transición demográfica hacia poblaciones de mayor edad. En la figura 1.1 se puede ver una estimación sobre cuál será la evolución de la pirámide poblacional en la Europa de los 28 entre los años 2015 y 2080, en dicha figura se puede apreciar cómo se ha producido una disminución en la parte media de dicha pirámide, observándose una clara transición hacia una pirámide invertida. Este mismo hecho se puede apreciar en la figura 1.2 en la Europa de los 28 el porcentaje de personas mayores de 65 años pasará de representar el 18.9 % en 2015 al 28.7 % en 2050, lo cual provocará que la tasa de dependencia de personas mayores¹ llegue casi a duplicarse [European Commission \(2015\)](#).

A pesar de que dichos cambios demográficos suponen un gran reto para nuestra sociedad, cuentan con la importante ventaja de ser en gran medida previsibles, lo cual ha dado lugar a numerosos marcos de actuación. El problema del envejecimiento de la población se ha abordado desde distintos ámbitos tanto internacionales, como europeos y nacionales. En el año 2002, la Asamblea General de las Naciones Unidas aprobó la Declaración Política y el Plan de Acción Internacional de Madrid sobre el Envejecimiento [United Nations \(2002\)](#). Entre sus recomendaciones se reconocen tres prioridades para la acción: las personas mayores y el desarrollo; el fomento de la salud y el bienestar en la vejez, y la creación de un entorno propicio y favorable para las personas mayores. En dicho plan se señalan varias cuestiones clave que siguen siendo pertinentes a día de hoy, como son promover la salud y

¹Porcentaje definido como el número de habitantes mayores de 65 años, entre el número de habitantes cuya edad está entre los 15 y los 65 años.

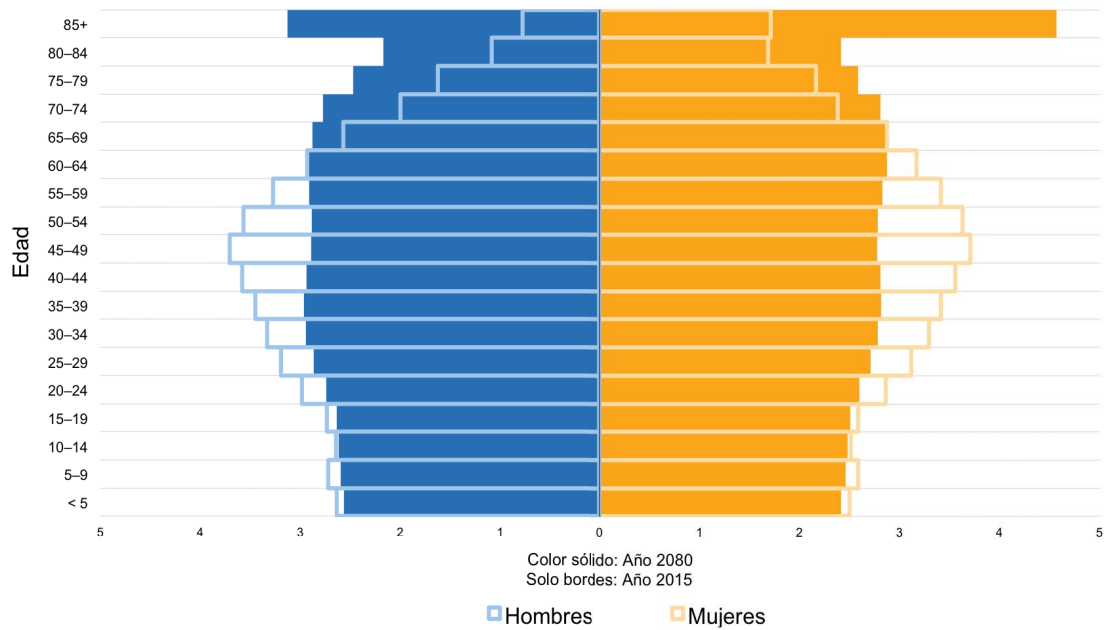


Figura 1.1: Comparativa de previsión por edades entre el año 2015 y el 2080 para la Europa de los 28, datos obtenidos de [European Commission \(2015\)](#)

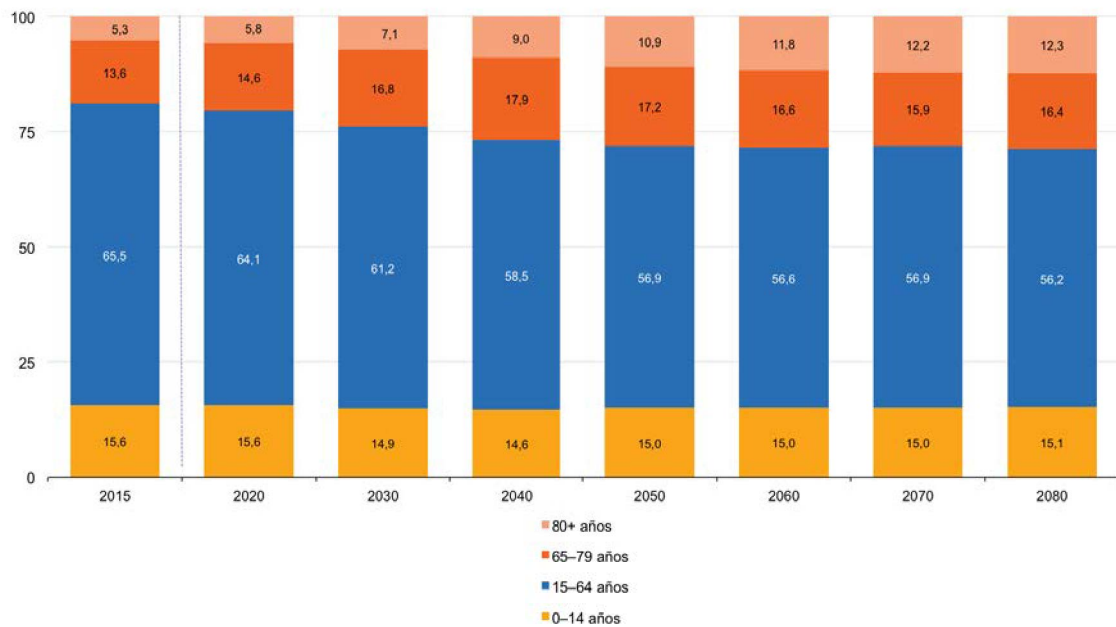


Figura 1.2: Previsión de la composición por edades en la Europa de los 28 datos obtenidos de [European Commission \(2015\)](#)

el bienestar durante toda la vida, garantizar el acceso universal y equitativo a los servicios de asistencia sanitaria, capacitar a los prestadores de cuidados y profesionales de la salud, proporcionar atención y apoyo a los cuidadores y prevenir la negligencia, la violencia y el maltrato contra las personas mayores. También en 2002 la OMS publicó un marco político en el que se detallan distintos planes de acción para conseguir un **envejecimiento activo**, de acuerdo con [World Health Organization \(2002\)](#) el envejecimiento activo se define como “el proceso en que se optimizan las oportunidades de salud, participación y seguridad a fin de mejorar la calidad de vida de las personas a medida que envejecen. El envejecimiento activo permite que las personas realicen su potencial de bienestar físico, social y se centra en las personas mayores y en la importancia de dar una imagen pública positiva de este colectivo.”

A nivel europeo, el envejecimiento activo ha estado muy presente tanto por parte de la Comisión Europea como por parte del Consejo Europeo. En el año 2009 el Consejo adoptó sus conclusiones sobre “Envejecimiento digno y saludable” entre las cuales se invitaba a la Comisión a que “desarrolle actividades de sensibilización a fin de fomentar el envejecimiento activo, en particular un posible Año Europeo del Envejecimiento Activo y la Solidaridad Intergeneracional en 2012”. En el año 2011 mediante la Decisión n° 940/2011 del Parlamento Europeo y del Consejo, se declaró el año 2012 como “Año Europeo del Envejecimiento Activo y Solidaridad Intergeneracional” para “promover la vitalidad y dignidad de todos”, de acuerdo con la definición descritos en dicha Decisión envejecimiento activo significa:

- Dar a las personas mayores la posibilidad de participar plenamente en la sociedad, fomentando sus oportunidades de empleo.
- Permitirles contribuir activamente a través del voluntariado y de programas intergeneracionales.
- Permitirles vivir con independencia adaptando la vivienda, las infraestructuras, la tecnología y el transporte.

Recientemente en abril de 2016, la OMS publicó la Estrategia y plan de acción mundiales sobre envejecimiento y la salud para el periodo 2016-2020 [World Health Organization \(2016\)](#). Entre otros muchos factores dicho plan resalta la importancia de envejecer en la comunidad donde se ha vivido, en lo que se denomina “Envejecimiento en el lugar” (Ageing in place). Las personas mayores suelen preferir envejecer en su hogar o al menos en la comunidad en que han vivido [Costa-Font et al. \(2009\)](#). Esto les permite mantener las relaciones y las redes comunitarias, las cuales les proporcionan bienestar y a las que recurrir cuando lo necesiten. Las TIC, en particular aquellas que fomentan la comunicación y la integración, las que ofrecen oportunidades de aprender y las que permiten vigilar y garantizar la seguridad de una persona mayor, pueden hacer que este objetivo sea más fácil de alcanzar en el futuro. El envejecimiento en el lugar puede fomentarse aún más mediante la creación de entornos adaptados a las personas mayores que permitan la movilidad y la participación de las personas mayores en actividades básicas, como ir de compras. Si bien el objetivo del envejecimiento en el lugar de residencia ha sido habitualmente garantizar una vivienda adecuada y asequible, unos entornos adaptados a las personas mayores, los servicios de salud también cumplen un papel importante al acercar la asistencia al lugar donde viven las personas. Por consiguiente, es necesario reorientar los modelos de atención

para priorizar la atención primaria y la atención en la comunidad. Esto abarca un cambio de la asistencia hospitalaria a la asistencia ambulatoria y las consultas externas, más intervenciones en el hogar, la participación de la comunidad y un sistema de derivación completamente integrado [World Health Organization \(2015\)](#).

Es precisamente este último punto señalado por la OMS en el que se encuentra enmarcado la presente investigación, la cual propone nuevas técnicas enfocadas a la resolución de problemas de planificación de asistentes de atención domiciliaria con una alta dimensionalidad y un conjunto de restricciones espaciales y temporales. La aplicación de dichas técnicas, debería permitir a las compañías prestadoras de servicios de atención domiciliaria la resolución de problemas de una mayor dimensionalidad, aumentando su eficiencia, reduciendo sus costes y posibilitando en última instancia una atención domiciliaria de calidad y sostenible para nuestros mayores.

1.2. Problemas de optimización

En nuestra vida cotidiana estamos acostumbrados a tomar multitud de decisiones, algunas de estas decisiones las tomamos tan a menudo que lo hacemos de un modo casi automático. Por el contrario, existen otras situaciones en las que el proceso de toma de decisiones es más complejo. En estos casos, el hecho de decidir implica generalmente evaluar un conjunto de alternativas y seleccionar aquella que mejor se adapte a nuestras necesidades. Esta misma problemática de decidir en entornos más complejos, está muy presente en el mundo empresarial, apareciendo normalmente en situaciones donde los recursos son limitados (personal, materias primas, tiempo), y donde existen ciertas restricciones que deben cumplirse (producción, horas de descanso), condicionando estos dos factores la selección de la alternativa adecuada, ya sea a nivel estratégico, táctico e incluso operativo. El proceso de toma de decisiones tiene por objetivo seleccionar aquellas alternativas que mejor se adaptan a una problemática concreta, usándose como criterios de selección minimizar costes, maximizar el beneficio, maximizar la satisfacción cliente u otros criterios similares. Desde la Segunda Guerra Mundial la toma de decisiones se ha abordado desde un punto de vista más formal, a través de disciplinas como la Investigación Operativa, la matemática aplicada y más recientemente por las ciencias de la computación, la cual ha contribuido sin duda a su automatización. La Investigación Operativa, también denominada Investigación de Operaciones, surge de los métodos científicos que se han aplicado en diversas organizaciones militares, industriales, económicas, sanitarias, administrativas, con el objetivo de mejorar la eficiencia y la efectividad de los resultados de dichas organizaciones. El propósito de dicha formalización es representar mediante modelos matemáticos el proceso de toma de decisiones presentes en las organizaciones, dichos modelos, a pesar de ser en la mayoría de los casos simplificaciones de la realidad, resultan de gran utilidad llegando a suponer ahorro de costes cientos de millones de dólares en algunas organizaciones [Hillier y Lieberman \(2015\)](#). El proceso de toma de decisiones está estrechamente relacionado con el concepto de optimización, por lo general cuando el proceso de toma de decisiones se aborda desde una perspectiva formal, el objetivo final es encontrar aquella decisión o conjunto de estas que optimizan cierta función objetivo. El concepto de optimización [Pardalos y Resende \(2002\)](#) es un término que está presente en diversas disciplinas, como son las matemáticas, la investigación operativa [Hillier y Lieberman \(2015\)](#) o las ciencias

de la computación. Optimizar significa hallar el valor máximo o mínimo de una función, definida en un dominio. En los problemas de optimización que generalmente se presentan en el mundo empresarial existen un conjunto de recursos escasos (presupuesto, personal, tiempo, materiales), un conjunto de requisitos o necesidades a cubrir (producción, tareas a realizar, clientes a satisfacer) y un conjunto de restricciones a respetar. Estos tres elementos condicionan la elección de una solución adecuada, ya sea a nivel operativo, táctico o estratégico. El objetivo perseguido es por lo general el de llevar a cabo un plan de manera óptima, es decir, minimizando o maximizando cierta función, que generalmente se conoce como función objetivo.

1.2.1. Tipos de problemas

Bajo el término problemas de optimización se aglutinan multitud de tipos de problemas, que resulta conveniente identificar. A continuación se muestra una clasificación no exhaustiva (consultar [Pardalos y Resende \(2002\)](#)) atendiendo a distintas características, como son el tipo de función objetivo, la naturaleza de las variables de decisión, el tipo de restricciones y finalmente la naturaleza de los datos.

- Atendiendo a la función objetivo. Se distinguen tres casos, problemas sin ninguna función objetivo, con un solo objetivo o con múltiples objetivos. En primer caso se habla de problemas de satisficibilidad, en los cuales es suficiente encontrar una solución que satisfaga todas las restricciones del problema sin optimizar ninguna función objetivo en particular. En el extremo opuesto se encuentran los problemas de optimización multiobjetivo [Censor \(1977\)](#), en este tipo de problemas existen varios objetivos a optimizar los cuales se encuentran enfrentados o en conflicto entre ellos, como por ejemplo optimizar el peso y la resistencia a la hora de desarrollar un nuevo material. Por último, encontramos los problemas con una única función objetivo, también conocidos como mono objetivo.
- Atendiendo a las variables de decisión. Si los valores de las variables de decisión del problema son continuos hablaremos de optimización continua. Por el contrario, si las variables de decisión son discretas, toman su valor de un conjunto discreto de valores, hablaremos de optimización discreta, destacando en este último tipo los problemas programación entera y los problemas de optimización combinatoria.
- Atendiendo a las restricciones. Los problemas en los que no existen restricciones se denominan problemas de optimización sin restricciones o no restringidos, entre sus métodos de resolución se encuentran métodos tan conocidos como método de Newton o el método del gradiente descendente. Del mismo modo cuando las variables de decisión tienen que cumplir ciertas restricciones hablamos de problemas de optimización con restricciones.
- Atendiendo a la naturaleza de los datos. En los problemas de optimización deterministas se asume que los datos proporcionados son exactos. Sin embargo, en numerosos problemas reales, dichos datos no se pueden conocer con exactitud debido a diferentes motivos. El primero de ellos es simple y está motivado por errores de medida, el segundo de ellos es que algunos de los datos del modelo hacen referencia a eventos

futuros y simplemente no se pueden conocer. Este tipo de problemas se conoce como problemas de optimización bajo incertidumbre o programación estocástica.

1.2.2. Problemas de optimización combinatoria

Uno de los temas fundamentales sobre los que se articula la presente Tesis, son problemas de optimización combinatoria. La resolución de este tipo de problemas ha tenido gran importancia en distintas áreas como son las telecomunicaciones, la planificación de la producción en diversos sectores industriales, el transporte de pasajeros y de mercancías, la distribución de tareas para su ejecución paralela en informática, el diseño de redes de transporte o telecomunicaciones, el diseño de campañas de marketing, la planificación de inversiones, la secuenciación de genes, la clasificación de plantas y animales, el diseño de nuevas moléculas, el posicionamiento de satélites, la asignación de trabajadores a tareas, la construcción de códigos seguros o el diseño de circuitos electrónico entre muchos otros Paschos (2014).

Los problemas de optimización combinatoria están incluidos dentro de los problemas de optimización, sin embargo y a diferencia de estos últimos, las variables de decisión son discretas. Las distintas combinaciones, ordenaciones, agrupaciones o asignaciones de dichas variables de decisión determinan el conjunto de soluciones candidatas. Para que una solución candidata pase a ser considerada una solución válida, debe cumplir un conjunto de restricciones. El subconjunto de soluciones candidatas que cumplen todas las restricciones del problema se denomina conjunto de soluciones o espacio de soluciones. El objetivo en este tipo de problemas, al igual que en el resto de problemas de optimización, es minimizar o maximizar una determinada función objetivo, es decir, encontrar dentro del espacio de soluciones aquella solución que minimiza o maximiza la función objetivo.

A priori se podría pensar qué, para encontrar la solución óptima a un problema cualquiera, basta con enumerar todas las soluciones candidatas, seleccionar aquellas que cumplen con las restricciones, evaluarlas mediante la función objetivo y quedarse con la mejor. El planteamiento anterior, aunque sencillo, tiene algunos inconvenientes. El principal de ellos, es la enumeración de todo el conjunto de soluciones candidatas, dicha enumeración resulta impracticable en la mayor parte de los casos, incluso para problemas de tamaño pequeño, debido a la naturaleza combinatoria de este tipo de problemas. Por ejemplo, supongamos que tenemos un conjunto de ciudades N que debemos visitar, tenemos la distancia entre cada par de ciudades i, j definida como $d_{ij} = d_{ji} \in \mathbb{R}^+$ y debemos visitar todas las ciudades minimizando la distancia total recorrida. Para dicho problema existen un total de $\frac{(n-1)!}{2}$ rutas posibles que pasen por todas las ciudades. El espacio de búsqueda para 11 ciudades, sería $10!/2$, lo que supone algo más de 1 millón y medio de combinaciones, sin embargo, para un problema con 100 objetos, tendríamos un conjunto de $100!/2$ soluciones candidatas lo que supondrían evaluar 10^{157} combinaciones, quedando patente la inconveniencia de este método. El problema anterior se conoce como *El problema del Viajante* y será tratado con mayor detalle en el capítulo 2.1.

1.2.3. Técnicas de resolución

Dentro de las distintas técnicas que se han desarrollado a lo largo de la historia, se pueden diferenciar dos familias de técnicas bien diferenciadas. Las técnicas exactas, las cuales garantizan la obtención de una solución óptima y las técnicas no exactas, que si bien no garantizan la obtención la solución óptima, obtienen soluciones de una calidad suficiente en un tiempo razonable y que por lo general permiten tratar problemas de una mayor dimensionalidad. Es precisamente por la explosión combinatoria que se produce en este tipo de problemas, por lo que en buena parte de los problemas reales las técnicas exactas no son viables, ya sea por su alto coste computacional en tiempo de ejecución o en cantidad de memoria requerida. En esta investigación se propondrán dos adaptaciones de la técnica metaheurística denominada optimización basada en colonias de hormigas propuesta por (Dorigo et al., 1996). A continuación se muestra un esquema con las distintas técnicas de resolución existentes.

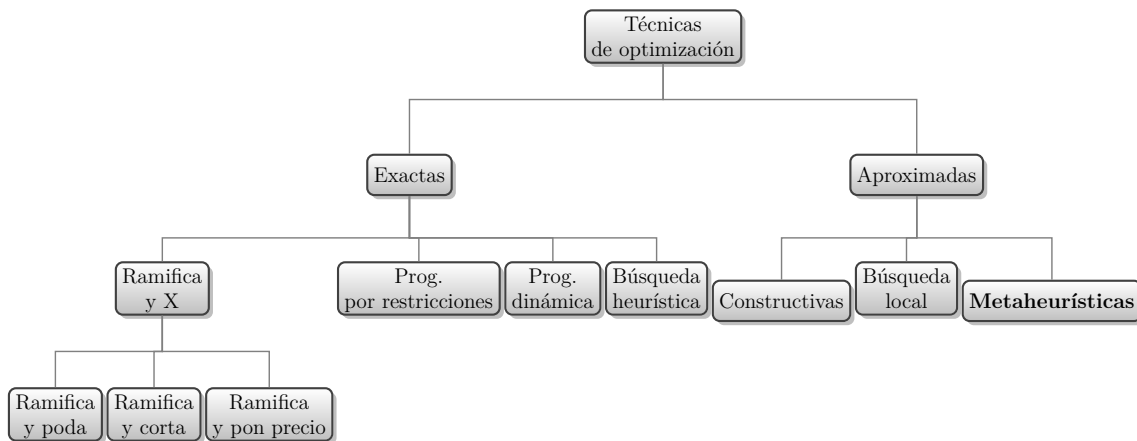


Figura 1.3: Esquema adaptado de Talbi (2009) en el que se muestran distintas técnicas de optimización.

1.3. Análisis clúster

Entre las capacidades que distinguen al ser humano del resto de seres vivos, destaca sin duda nuestra notable capacidad de abstraer las diferentes características de los elementos que nos rodean. Dicha capacidad de abstracción nos ha permitido, entre otras habilidades, agrupar, clasificar y categorizar los distintos objetos de nuestro entorno de un modo natural y en algunas ocasiones casi sin esfuerzo. Un buen ejemplo lo podemos apreciar en el proceso de aprendizaje que experimentan los niños, dada una colección de canicas de colores verde, azul y rojo cualquier niño es capaz de separarlas en tres grupos de objetos, utilizando como criterio de selección el color de las canicas. El ejemplo anterior, pone de manifiesto que la capacidad de agrupar objetos similares entre sí, atendiendo a algún criterio concreto de similitud, es sin duda una de las actividades más primitivas de los seres humanos (Anderberg, 1973). Esta capacidad, ha sido fundamental en diversos aspectos a lo largo de nuestra evolución, abarcando desde el desarrollo del propio del lenguaje que utilizamos

diariamente, hasta su utilización en disciplinas científicas tan diversas como la biología, la química, la genética, la meteorología o la astronomía, entre otras. Entendiendo dicha capacidad de agrupación en su sentido más amplio, se puede afirmar que ha sido necesaria e indispensable para el desarrollo del lenguaje. El lenguaje, está formado por un conjunto de palabras, las cuales nos ayudan a comunicarnos y nos permiten reconocer los distintos objetos y eventos que nos rodean. Cada nombre en un lenguaje, es básicamente una etiqueta utilizada para describir un tipo de objeto u objetos, que tienen en común una serie de características específicas; por ejemplo, los animales se pueden agrupar en mamíferos, reptiles, aves, peces y anfibios, las obras literarias, en líricas, épicas, dramáticas, narrativas, didácticas y ensayos. Además de ser una capacidad básica en la formación y evolución de los distintos lenguajes que utilizamos hoy en día, la agrupación o categorización es y ha sido una actividad fundamental en la mayor parte de disciplinas científicas, un buen ejemplo se puede encontrar en biología. La teoría y práctica sobre cómo agrupar y categorizar seres vivos se conoce como taxonomía. En sus orígenes, esta disciplina estuvo considerada más un arte que una disciplina científica. La primera agrupación de los seres vivos se debe a Aristóteles (Ross, 1923), quién en el siglo IV A.C diferenció a los seres vivos en dos reinos, el vegetal y el animal. El primero de ellos se diferenciaba por tener alma vegetativa, teniendo como características fundamentales la capacidad de reproducción, crecimiento y nutrición. El segundo de ellos, el animal, además de todo lo anterior poseía las capacidades de percepción, deseo y movimiento. Una vez realizada dicha división, Aristóteles volvió a dividir el grupo animal en dos subgrupos, a los que denominó géneros; el primero de ellos *anaíma*, se caracterizaba por no tener sangre y ser invertebrados, mientras que el segundo, *enáima*, poseía sangre y eran animales vertebrados. Después de Aristóteles, Teofrasto, quién sucediera a Aristóteles al frente de la escuela peripatética, realizó en el siglo IV A.C una clasificación sobre las plantas y su estructura, dividiéndolas en cuatro grupos distintos, árboles, arbustos, hierbas y matas. Su obra fue tan profunda y ofreció tanto detalles que constituyó la contribución más importante en botánica hasta el siglo XVII (Grene, 2004), es por ello que hoy en día es conocido con el sobrenombre de "padre la botánica". No fue hasta el siglo XVIII cuando Adanson (Adanson, 1763) desarrollara técnicas menos subjetivas tal y como se conocen hoy en día. La categorización de plantas y animales han sido fundamentales en disciplinas como la zoología o la botánica, dichas categorizaciones fueron fundamentales para el desarrollo de la teoría de la evolución de Darwin (Darwin, 1900). La clasificación también ha sido fundamental en otras áreas científicas, por ejemplo la clasificación de los elementos de la tabla periódica realizada por Mendeleev (Mendeleev et al., 1905) fue fundamental para la comprensión del átomo y su estructura atómica.

Todas las agrupaciones y clasificaciones expuestas anteriormente, fueron realizadas de un modo manual, es decir, no se utilizó ningún otro medio que no fuera la propia capacidad de abstracción y generalización del investigador. A partir de 1950, se empieza a plantear la posibilidad de realizar estas agrupaciones de un modo automático, había nacido lo que hoy conocemos como *técnicas de agrupamiento*, *análisis clúster* o simplemente *clustering*. En primer lugar, es conveniente aclarar que a lo largo de este capítulo y de los siguientes, se utilizará como sinónimos los términos análisis clúster y clustering. Aunque no es objetivo de este capítulo definir de un modo formal qué es el análisis clustering, es conveniente presentar una primera definición introductoria, que permita entender las motivaciones y objetivos del presente trabajo. En primer lugar, cabe destacar que no existe una definición

clara, formal, ni universalmente aceptada del mismo, este hecho está motivado por dos causas, el carácter multidisciplinar y cierto grado de subjetividad que se explicará más adelante. Aunque no existan definiciones formales del mismo, resulta conveniente exponer algunas de las definiciones propuestas por diferentes autores:

- Disciplina que tiene por objeto la organización de un conjunto de objetos sin etiquetar, en un conjunto de grupos, basándose para ello en cierto criterio de similitud. Cada grupo resultante, llamado *clúster*, debe contener objetos que sean similares entre sí y distintos de otros objetos que pertenezcan a otros grupos [Swagatam Das \(2009\)](#).
- Aquellos métodos enfocados a la identificación de grupos homogéneos de objetos ([Arabie et al., 1996](#))
- El análisis clúster está enfocado a la búsqueda de grupos dentro de un conjunto de datos. Estos grupos reciben el nombre de clúster. Un clúster es un conjunto de entidades que son parecidas entre sí, y las entidades de diferentes clústers no lo son ([Everitt et al., 2011](#)).

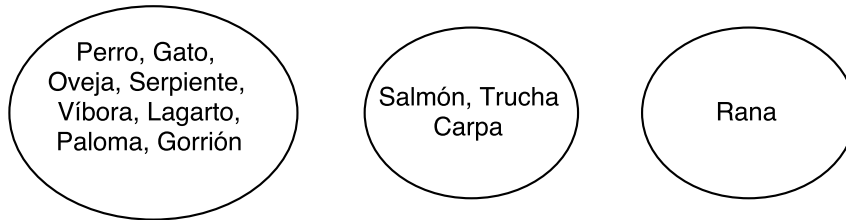
Uno de los aspectos a destacar dentro del análisis clúster, es el carácter inherentemente subjetivo en aspectos fundamentales del mismo, como son, los criterios utilizados a la hora de realizar el propio clustering, así como los distintos clústers resultantes del proceso. Para mostrar dicho carácter subjetivo a la hora de realizar un proceso de clustering, en la figura 1.4 se muestra un ejemplo con distintos tipos de animales vertebrados, utilizando distintos criterios. Como se puede ver en dicha figura, existen diversos criterios que pueden ser utilizados a la hora de realizar el proceso de clustering. En el ejemplo se han seleccionado dos tipos de criterios totalmente arbitrarios, de hecho, se podrían haber seleccionado otros criterios como la existencia de columna vertebral, en cuyo caso todos los animales estarían dentro del mismo clúster, el número de extremidades o cualquier otro factor.

Además del carácter subjetivo del proceso de agrupamiento, algunos autores como [Everitt et al. \(2011\)](#) no solo señalan la dificultad de realizar una definición formal lo suficiente amplia que incluya todos los matices, sino que además dicha definición puede estar incluso fuera de lugar. De acuerdo con el autor, el concepto de clúster es una generalización de lo que un humano percibe como grupo, en dos o tres dimensiones sobre un conjunto de datos, puesto que dicha percepción presenta una naturaleza inherentemente subjetiva, el hecho de intentarla representar mediante criterios únicos puede resultar incluso inadecuado. Otro ejemplo de dicha subjetividad lo podemos encontrar en la figura 1.5 donde se muestran distintos resultados tras realizar el proceso de clustering, el primero de ellos se muestra con líneas discontinuas y contiene un total de dos grupos, mientras que el segundo contiene cuatro clústers marcados por líneas continuas.

Una pregunta que surge al ver la imagen anterior es ¿cómo podemos saber qué resultado de los anteriores es el correcto? La respuesta es que no podemos, ambos resultados son válidos y a menos que obtengamos ambos resultados y se los mostremos a un experto sobre el conjunto de datos que estamos utilizando no existe una respuesta definitiva al respecto. Por lo tanto, a menos que sepamos de antemano el número de clústers, el problema de clustering no tiene una respuesta única para todos los casos, sino que es específica de cada problema, requiriendo en la mayor parte de los casos el conocimiento de humanos expertos en el dominio de aplicación para corroborar las hipótesis obtenidas de un modo automático.



(a) En este ejemplo se utiliza como criterio para la realización del clustering, la clase de animal vertebrado, mamíferos, peces, aves, reptiles o anfibios, a la que corresponde cada animal.



(b) En este ejemplo se utiliza como criterio para la realización del clustering el medio en el que vive el animal, si el animal vive en la tierra, en el agua o en ambos.

Figura 1.4: Como se puede apreciar ninguna de las dos agrupaciones es más exacta, ni mejor que la otra, simplemente se utilizan criterios distintos.

Dentro de las distintas técnicas de agrupamiento, la idea de incorporar conocimiento previo al proceso de clustering, ha creado todo un campo de investigación conocido como clustering con restricciones [Wagstaff et al. \(2008\)](#), existen numerosas situaciones en las que se posee cierto conocimiento previo que puede resultar muy importante a la hora de obtener un resultado satisfactorio. La idea principal es poder incorporar conocimiento previo y guiar el proceso de clustering. Dicho conocimiento se suele expresar en forma de relaciones dos a dos entre los datos, conocidas como restricciones, las cuales expresan la obligación o preferencia de que aquellos datos que intervienen en la relación deben o no deben pertenecer al mismo clúster. Dichas restricciones, conocidas en la literatura especializada como *must-link* y *Cannot link*, influyen en el proceso de clustering aunque siguen siendo los propios algoritmos quienes determinan el resultado final. Dicho conocimiento, no usado en las aproximaciones tradicionales de clustering, puede ser fácilmente incorporado con el objetivo de mejorar el proceso de clustering, haciéndolo más preciso, significativo y mejor adaptado a la propia visión que un usuario con conocimiento del dominio de aplicación tiene sobre los propios datos.

La idea anterior de incorporar restricciones al proceso de clustering, es especialmente útil cuando se trata la resolución de problemas de optimización combinatoria. En el capítulo 4 dicha idea es desarrollada y adaptada al objeto de estudio tratado en esta investigación,

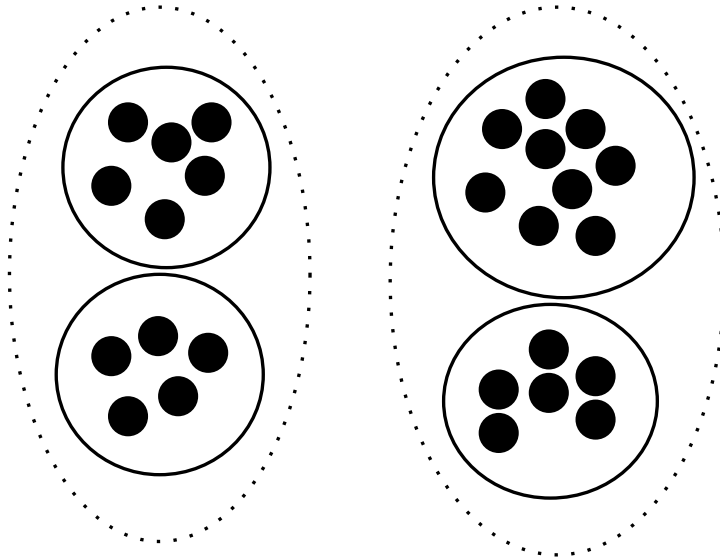


Figura 1.5: Dos ejemplos de resultados distintos al realizar un proceso de clustering, obteniéndose en cada uno de ellos dos clústers y cuatro clusters.

tomando como base la técnica de clustering jerárquica propuesta por (Ward, 1963). Dicha técnica, al igual que el resto de técnicas de clustering aglomerativas, comienza con tantos clústers como elementos se quieran agrupar, es decir, cada clúster está formado por un solo elemento. Posteriormente y de forma iterativa se une aquel par de clústers que minimizan cierta métrica de distancia. El proceso finaliza cuando solo existe un clúster que contiene todos los elementos.

1.4. Objetivos

Los problemas de optimización combinatoria están presentes en diversos sectores de nuestra sociedad, procesos industriales de fabricación, médica, farmacéutica, el transporte de mercancías, los servicios de mensajería, la planificación de proyectos o las redes de telecomunicaciones, entre muchos otros Du y Pardalos (2005). Dada su diversidad y complejidad, dichos problemas se han resuelto a lo largo de la historia, utilizando diversas técnicas. De entre todas las técnicas, pueden distinguirse dos grandes tipos de aproximaciones, las técnicas exactas y técnicas no exactas (ver figura 1.3). Las técnicas exactas garantizan la obtención de la solución óptima en una cantidad finita de tiempo, dicha cantidad de tiempo, crece de modo no lineal en los problemas de optimización combinatoria siendo de utilidad el uso de otro tipo de técnicas, denominadas no exactas o heurísticas. Dentro de las técnicas no exactas o heurísticas, destacan las técnicas metaheurísticas las cuales dada su naturaleza pueden ser aplicadas a diferentes dominios obteniendo resultados competitivos en una cantidad razonable de tiempo Dréo et al. (2006); Talbi (2009).

Las técnicas de clustering además de resultar eficaces en la identificación de grupos de

un modo automático [Everitt et al. \(2011\)](#), también han resultado ser adecuadas y eficaces en la resolución de problemas de optimización combinatoria, bien combinadas con otras técnicas ([Dondo y Cerdá, 2007](#)) o bien aplicadas por sí solas ([Chunyu y Xiaobo, 2006](#)). Sin embargo, para aquellos problemas de optimización combinatoria en los que se presentan restricciones espaciales y temporales, las técnicas de clustering no resultan eficaces, dada su imposibilidad de tratar dichas restricciones. La primera de las dificultades para aplicar técnicas del clustering a la resolución de problemas de optimización combinatoria, es la necesidad de definir una función de similitud o distancia entre los distintos elementos del clúster, en este tipo de problemas dos elementos podrían no ser compatibles entre sí al no respetar alguna restricción, y sin embargo los mismos elementos podrían ser compatibles si dentro del mismo clúster está presente un tercero. A pesar de existir numerosas investigaciones en las que se plantean técnicas de agrupamiento con restricciones ([Wagstaff y Cardie, 2000](#)), no existen publicaciones relevantes en las que las restricciones utilizadas deban ser evaluadas de un modo dinámico, ya que el cumplimiento de dichas restricciones depende de la propia composición del grupo. Por otro lado existen escasas publicaciones en las que se defina una función de similitud que contemple restricciones espaciales y temporales [Qi et al. \(2012\)](#), este tipo de restricciones son especialmente útiles para tratar problemas de planificación de asistentes de atención domiciliaría, en los cuales existen restricciones espaciales y temporales; se recomienda ver la sección 2.3 del capítulo 2.

Las técnicas metaheurísticas se han aplicado con éxito a multitud de problemas de optimización combinatoria ([Talbi, 2009](#)), aplicándose también a problemas de clustering ([Swagatam Das y Konar, 2009](#)). Dentro de las distintas técnicas existentes, una de las cuales se ha aplicado a una mayor diversidad de problemas, sea la técnica metaheurística conocida como optimización basada en colonias de hormigas, en adelante ACO, de su acrónimo inglés Ant Colony Optimization ([Dorigo y Stützle, 2010](#)). Las colonias de hormigas, además de haber servido como inspiración para la metaheurística ACO², el modo en el que las hormigas distribuyen sus cadáveres y mantienen limpio su hormiguero también ha servido como base para resolver ciertos problemas de clustering ([Machnik, 2006](#); [Lumer y Faieta, 1994](#)). Es preciso destacar qué, a pesar de estar ambas técnicas inspiradas en el comportamiento de las colonias de hormigas, las ecuaciones, heurísticas y funcionamiento general en las que se fundamentan difieren en numerosos aspectos.

El objetivo principal de esta Tesis es la resolución de problemas de optimización combinatoria, los cuales deberán presentar las siguientes características.

- Han de solucionarse instancias que pertenezcan a problemas reales de la industria, de modo que los resultados de esta investigación sean aplicables a dicha industria.
- Han de solucionarse instancias de alta dimensionalidad, entendiéndose por tal, instancias cuya dimensión sea de al menos un orden de magnitud superior a las instancias resueltas por las técnicas exactas en el estado del arte actual.
- Los problemas seleccionados han de presentar restricciones temporales y espaciales. Entendiéndose por estas, aquellas restricciones que tienen en cuenta el tiempo,

²La cual está inspirada fundamentalmente en como las colonias de hormigas encuentran y llevan comida hasta su hormiguero.

incluyéndose al menos los tiempos en los que se puede realizar una tarea o servicio, conocido como ventana temporal, el tiempo de desplazamiento y el tiempo de espera.

Dentro de los diversos problemas de optimización combinatoria, la familia de problemas que aborda esta investigación es la familia de problemas de planificación de asistentes de atención domiciliaria, que además de cumplir con las características enunciadas anteriormente, ha suscitado el interés de numerosos investigadores en los últimos años, en buena parte motivada por las previsiones demográficas y el envejecimiento de nuestra población [Fikar y Hirsch \(2017\)](#). El objetivo descrito anteriormente se tratará desde al menos dos perspectivas distintas y complementarias, como son las técnicas de clustering y la técnica metaheurística ACO.

- A pesar de que existen trabajos en los que se aborda la resolución de problemas de optimización combinatoria mediante técnicas de clustering, al utilizar este tipo de técnicas, existe una dificultad en definir funciones de similitud o distancia apropiadas para problemas que presenten restricciones espaciales y temporales [Qi et al. \(2012\)](#). A dicha dificultad, se debe sumar la dificultad de evaluar el cumplimiento de dichas restricciones de un modo dinámico, dependiente de la propia composición del grupo. Como ya se expuso anteriormente, en este tipo de problemas dos elementos pueden ser incompatibles entre sí al no respetar alguna de las restricciones temporales/espaciales y esos dos mismos elementos pueden ser compatibles entre sí dentro del clúster si existe un tercer elemento que los hagan compatibles.
- Respecto a la técnica metaheurística ACO, a pesar de haber sido utilizada para multitud de problemas de optimización combinatoria ([Dorigo y Stützle, 2010](#)), así como en diversos problemas de planificación [Neto y Filho \(2013\)](#), no existen publicaciones en el estado del arte actual en el que dichas técnicas se utilicen para la resolución de problemas de planificación de asistentes de atención domiciliaria [Fikar y Hirsch \(2017\)](#).

1.5. Estructura

Una vez presentados los objetivos y realizada una breve introducción a los temas fundamentales para el desarrollo de esta Tesis, se detalla la estructuración en capítulos de la misma.

- **Capítulo 2: Estado del arte.** En este capítulo se presenta el estado del arte relacionado y relevante para el desarrollo de esta investigación. En primer lugar, se presenta aquellas familias de problemas de optimización combinatoria que están relacionados con esta investigación, como son el problema del viajante (TSP) (sección 2.1), el problema de enrutamiento de vehículos (VRP)(sección 2.2) y el problema de planificación de asistentes de atención domiciliaria (HCSP) (sección 2.3), prestando especial atención a aquellos que presentan restricciones temporales y espaciales. Finalmente se hace revisión de la técnica metaheurística ACO, mostrando sus variaciones, su evolución histórica y las distintas aplicaciones en las que ha sido usada para resolver problemas de optimización combinatoria con restricciones.

- **Capítulo 3: Caso de estudio: HCSP en la Comunidad de Madrid.** En este capítulo se describe la problemática concreta para la resolución del problema seleccionado, el cual consiste en la planificación de asistentes de atención domiciliaria. En dicho capítulo, se expone el problema real al que se enfrenta una multinacional a la hora de planificar semanalmente a sus asistentes de atención domiciliaria. Las dos características que definen al problema son la alta dimensionalidad del mismo, se trata de planificar semanalmente algo más de 29.000 tareas, y la necesidad establecida por la compañía de garantizar la continuidad del servicio ³. El tamaño inusualmente grande de las instancias a resolver, junto con la imposibilidad de dividirlo en problemas más pequeños han requerido la elaboración de técnicas específicas para su resolución, las cuales se exponen en el capítulo 4.
- **Capítulo 4: Técnicas desarrolladas.** En este capítulo se describen las técnicas desarrolladas para tratar el problema descrito en el capítulo 3. En primer lugar se describe la técnica adaptada del método clásico de [Ward \(1963\)](#). En la segunda parte del capítulo se describen las dos técnicas propuestas para adaptar la metaheurística ACO al problema de planificación de asistentes de atención domiciliaria. En primer lugar se presenta la primera de las técnicas conocida como ACS-HCSP, la cual tiene como objetivo realizar una adaptación de la metaheurística ACO al problema HCSP realizando el número indispensable de modificaciones. Esta primera aproximación no solo servirá para evaluar las posibilidades de resolver el HCSP sino que también servirá de línea base a implementaciones más especializadas en el HCSP. Posteriormente y tras analizar las potenciales mejoras de ACS-HCSP se presenta la técnica IACS-HCSP cuyo objetivo fundamental es mejorar las debilidades detectadas en la técnica ACS-HCSP.
- **Capítulo 5: Experimentación y resultados.** En este capítulo se lleva a cabo la validación experimental de las técnicas propuestas en el capítulo 4, las cuales fueron diseñadas para resolver el problema objeto de estudio presentado en el capítulo 3. En primer lugar, se describen las características de las distintas instancias de problemas que serán objeto de experimentación. Posteriormente, se muestran los resultados obtenidos por cada una de las técnicas propuestas, en concreto para la adaptación del método de Ward, así como para las dos propuestas basadas ACO, ACS-HCSP y IACS-HCSP, realizándose análisis estadísticos de significación con el objetivo de medir la bondad de los métodos y técnicas propuestos en el Capítulo 4. Finalmente y con el objetivo de medir el rendimiento de las técnicas propuestas respecto a las existentes en el estado del arte, se realiza una comparativa con los resultados obtenidos por [Quintana et al. \(2017\)](#), investigación que también aborda el mismo problema con una serie de técnicas heurísticas.
- **Capítulo 6: Conclusiones y líneas futuras** Finalmente en este último capítulo se muestran las conclusiones obtenidas como resultado de esta investigación, analizando el cumplimiento de los objetivos marcados en este primer capítulo, destacándose

³Tal y como se mostrará en la sección 2.3.2.1 del capítulo 2 dicha característica hace referencia a la necesidad por parte de los prestadores de servicios de garantizar una regularidad en la asignación de asistentes y clientes.

posteriormente las aportaciones realizadas por esta investigación. Por último, se detallan las limitaciones de esta investigación, así como su posible aplicación a otros problemas de optimización combinatoria, los cuales conforman las posibles líneas de investigación que podrían ser llevadas a cabo por otros investigadores.

Capítulo 2

Estado del Arte

Las especies que sobreviven no son las más fuertes, ni las más rápidas, ni las más inteligentes; sino aquellas que se adaptan mejor al cambio.

El origen de las especies, capítulo V. Charles Darwin

Resumen: En este capítulo se hace una exposición detallada sobre el estado del arte de aquellos conceptos clave en los que enmarca la presente investigación, como son los problemas de optimización combinatoria con restricciones temporales y las técnicas metaheurísticas. Dentro de los problemas de optimización combinatoria, se presta especial atención a aquellas familias de problemas que presentan restricciones espaciales/temporales, como son la familia de problemas del Problema del Viajante y Problema de Enrutamiento de Vehículos al ser dichos problemas particularizaciones o generalizaciones del problema principal abordado en esta investigación Problema de planificación de asistentes de atención domiciliaria. Finalmente se hace una descripción detallada de la metaheurística Optimización Basada en Colonias de Hormigas y su aplicación a diferentes tipos de problemas de optimización combinatoria.

2.1. Problema del Viajante

2.1.1. Introducción

El Problema del Viajante de comercio, conocido por sus siglas en inglés *TSP* (*Traveling Salesman Problem*) es quizás uno de los problemas de optimización combinatoria más conocidos y estudiados, en lo sucesivo se hará referencia a dicho problema como TSP. A pesar de la aparente sencillez de su enunciado, es un problema que encierra una alta complejidad, presentando numerosas variantes. Dicha complejidad y dificultad ha atraído la atención de numerosos y prestigiosos investigadores, procedentes de diversas áreas del conocimiento como las matemáticas, la física, la biología o las ciencias de la computación. Una posible enunciación del mismo es la siguiente: *Dado un conjunto de ciudades y las distancias intermedias entre ellas, se debe encontrar la ruta más corta de tal modo que se visiten todas las ciudades exactamente una vez, regresando finalmente a la ciudad de origen.*

Dicho problema pertenece a la clase de complejidad \mathcal{NP} -difícil ¹, lo cual implica que hasta la fecha no se conoce un algoritmo que sea capaz de resolver cualquier instancia del mismo en tiempo polinomial. El mero hecho de probar si existe o no dicho algoritmo, es uno de los denominados problemas del milenio propuestos por el Instituto Clay de Matemáticas, cuya resolución está dotada con un premio de un millón de dólares [Clay Institute of Mathematics \(2000\)](#). Al igual que otros problemas de optimización combinatoria, el TSP crece de modo exponencial al aumentar el tamaño del mismo. Para un total de n ciudades, el número de rutas a evaluar es $\frac{(n-1)!}{2}$, nótese que distancia total no recorrida no depende del sentido del tour, de ahí que se divida entre 2. ²

Desde un punto de vista matemático, en concreto desde la perspectiva de teoría de grafos, el TSP es una generalización del problema descrito para encontrar un circuito Hamiltoniano en un grafo, el cual queda a su vez definido del siguiente modo. Dado un grafo $G = (V, A)$, donde $V = \{v_1, \dots, v_n\}$ define el conjunto de vértices y $A = \{(r, s) | r, s \in V\}$ define el conjunto de aristas, se debe encontrar la sucesión de aristas que visitan todos los vértices del grafo exactamente una vez, regresando al vértice de origen. Dicho problema fue estudiado durante siglo XIX por el matemático irlandés Sir William Rowan Hamilton, así como por el inglés Thomas Penyngton Kirkman. El problema de decisión asociado a determinar si existe un ciclo hamiltoniano para un grafo dado pertenece a la clase de complejidad \mathcal{NP} -completo [Karp \(1972\)](#). En la figura 2.1 puede encontrarse un ejemplo de un ciclo hamiltoniano para el grafo de un dodecaedro. Una discusión más detallada sobre los hallazgos de Hamilton y Kirkman puede encontrarse en [Biggs et al. \(1986\)](#).

¹El problema de decisión pertenece a la clase de complejidad \mathcal{NP} -completo se recomienda consultar ([Johnson, 2012](#); [Hartmanis, 1982](#)).

²Dicha fórmula sólo es aplicable a la versión simétrica del problema, la cual se describe en la sección 2.1.4.1.

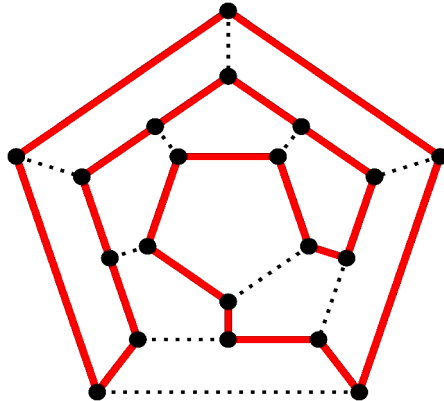


Figura 2.1: Ejemplo de ciclo o circuito Hamiltoniano para el grafo de un dodecaedro. El dodecaedro al igual que el resto de grafos platónicos contienen ciclos hamiltonianos.

En el caso del TSP, no sólo se debe encontrar un ciclo Hamiltoniano, sino que además se debe encontrar aquel que minimice el coste total de transitar por las aristas seleccionadas.

2.1.2. Evolución histórica

Los primeros estudios de este problema se atribuyen a Kalr Menger en las universidades de Viena y Harvard a principios de 1930, siendo estudiado con posterioridad en la universidad de Princeton por los investigadores Hassler Whitney y Merrill Flood, una descripción detallada sobre la conexión de estos investigadores y dichas universidades puede encontrarse en (Schrijver, 2005). Posteriormente, en 1954, (Dantzig et al., 1954) propusieron un método basado en programación lineal (LP ³) capaz de resolver una instancia con un total de 49 ciudades de los Estados Unidos, una por cada estado a la cual se añadió la ciudad de Washington D.C ⁴. Aunque hoy en día este tamaño de problema pueda parecer reducido, a finales de 1954 dicha instancia del problema era gigantesca, su resolución apareció incluso en revistas no científicas como *Newsweek* cuya imagen se puede apreciar en la Figura 2.2. El problema se popularizó entre la opinión pública en 1962, cuando la compañía Procter & Gamble ofreció un premio de 10.000 dólares, dinero suficiente para comprarse una casa en la época, a quien resolviese una instancia del problema con un total de 33 ciudades, en lo que se popularizó como el juego *Coche 54* ⁵ cuya imagen se muestra en la figura 2.3.

³Linear Programming.

⁴Hay que destacar que la instancia que resolvieron tenía 42 ya que se quitaron 7 ciudades de la instancia original, las comprendidas entre Whashington y Manchester ver figura 2.2.

⁵Los policías Toody y Muldoon, con su característico coche 54, eran personajes de una popular serie Americana.

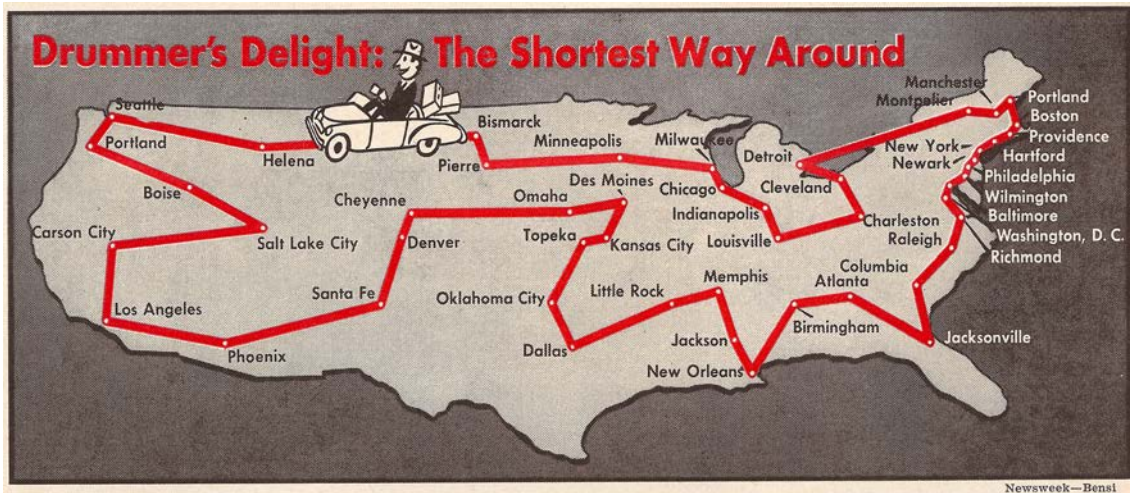


Figura 2.2: Ejemplo de la instancia resuelta por (Dantzig et al., 1954) publicada en la revista Newsweek.

HELP! WE'RE LOST!

HELP "CAR 54"... AND WIN CASH
 54...\$1,000 PRIZES
 ONE...\$10,000 GRAND PRIZE

Map by Rand McNally

Help Toody and Muldoon find the shortest round trip route to visit all 33 locations shown on the map. All you do is draw connecting straight lines from location to location to show the shortest round trip route.

HERE'S THE CORRECT START...
 Begin at Chicago, Illinois. From there, lines show correct route as far as Erie, Pennsylvania. Next, do you go to Carlisle, Pennsylvania or Wana, West Virginia? Check the easy instructions on back of this entry blank for details.

© PROCTER & GAMBLE 1962 OFFICIAL RULES ON REVERSE SIDE

Figura 2.3: Panfleto publicitario de la compañía Procter Gamble en el que se ofrecía un premio de 10.000 \$ a quien encontrara la mejor solución para una instancia de 33 ciudades.

Tal y como se aprecia en la tabla 2.1 la capacidad para resolver instancias de mayor tamaño ha evolucionado a lo largo de los años. No fue hasta 1971, 17 años después del trabajo de (Dantzig et al., 1954), cuando los investigadores de IBM Michael Held y Richard Karp resolvieron un problema con 64 ciudades. Posteriormente, en 1977, los matemáticos Martin Grötschel and Manfred Padberg resolvieron una instancia con un total 120 ciudades alemanas de modo óptimo. Posteriormente, Padberg empezó a trabajar con el investigador de IBM Harlan Crowder, siendo ambos investigadores capaces de encontrar la solución óptima para un ejemplo de 318 nodos presentes en una tarjeta de circuito impreso. Cincuenta años después de que (Dantzig et al., 1954) y su equipo resolvieran una instancia de 42 ciudades a mano, en 2004 se resolvió una instancia para las 24.978 ciudades (ver imagen 2.4) de Suecia. El problema se resolvió en algo más de un año aproximadamente, utilizando un sistema compuesto por 96 ordenadores, según los autores resolverlo en un solo ordenador hubiera llevado un total de 91.9 años Applegate et al. (2007).

Año	Investigadores	Tamaño de la instancia
1954	G. Dantzig, R. Fulkerson, and S. Johnson	49 ciudades
1971	M. Held and R.M. Karp	64 ciudades
1975	P.M. Camerini, L. Fratta, and F. Maffioli	67 ciudades
1977	M. Grötschel	120 ciudades
1980	H. Crowder and M.W. Padberg	318 ciudades
1987	M. Padberg and G. Rinaldi	532 ciudades
1987	M. Grötschel and O. Holland	666 ciudades
1987	M. Padberg and G. Rinaldi	2.392 ciudades
1994	D. Applegate, R. Bixby, V. Chvátal, and W. Cook	7.397 ciudades
1998	D. Applegate, R. Bixby, V. Chvátal, and W. Cook	13.509 ciudades
2001	D. Applegate, R. Bixby, V. Chvátal, and W. Cook	15.112 ciudades
2004	D. Applegate, R. Bixby, V. Chvátal, W. Cook, and K. Helsgaun	24.978 ciudades
2005	W. Cook, Espinoza and Goycoolea	33.810 ciudades
2006	D. Applegate, R. Bixby, V. Chvátal, W. Cook	85.900 ciudades

Tabla 2.1: Evolución histórica de los tamaños de las distintas instancias resueltas, donde la columna año indica el año en el que se encontró la solución óptima a dicha instancia Applegate et al. (2007).

Posteriormente, en 2006, se resolvió la mayor instancia resuelta hasta la fecha de modo óptimo, con un total de 85.900 “ciudades” Cook (2015). Como se puede ver en la imagen 2.5 la instancia no se corresponde con ningún mapa, sino que procede de una instancia generada a finales de 1980 en los laboratorios Bell para el diseño de circuitos electrónicos integrados a gran escala. En este caso, el objetivo es reducir la distancia total recorrida por un láser, que debe actuar en un conjunto de puntos conocidos a priori, al actuar, el láser desintegra aquellas conexiones entre puertas lógicas que no deben aparecer en el diseño final. La resolución de dicha instancia supuso un importante hito, ya que con su resolución todas las instancias incluidas en la librería TSPLIB quedaron resueltas Applegate et al. (2007). La instancia en cuestión fue incorporada a dicha librería en 1991 por el autor (Reinelt, 1991).

El aumento del tamaño de las instancias resueltas de modo óptimo que se recogen en la tabla 2.1 ha sido en buena parte gracias al software empleado desde 1992, conocido



Figura 2.4: Ejemplo de la instancia resuelta por D. Applegate, R. Bixby, V. Chvátal, W. Cook, and K. Helsgaun. en 2004 para un total de 24978 ciudades de Suecia.

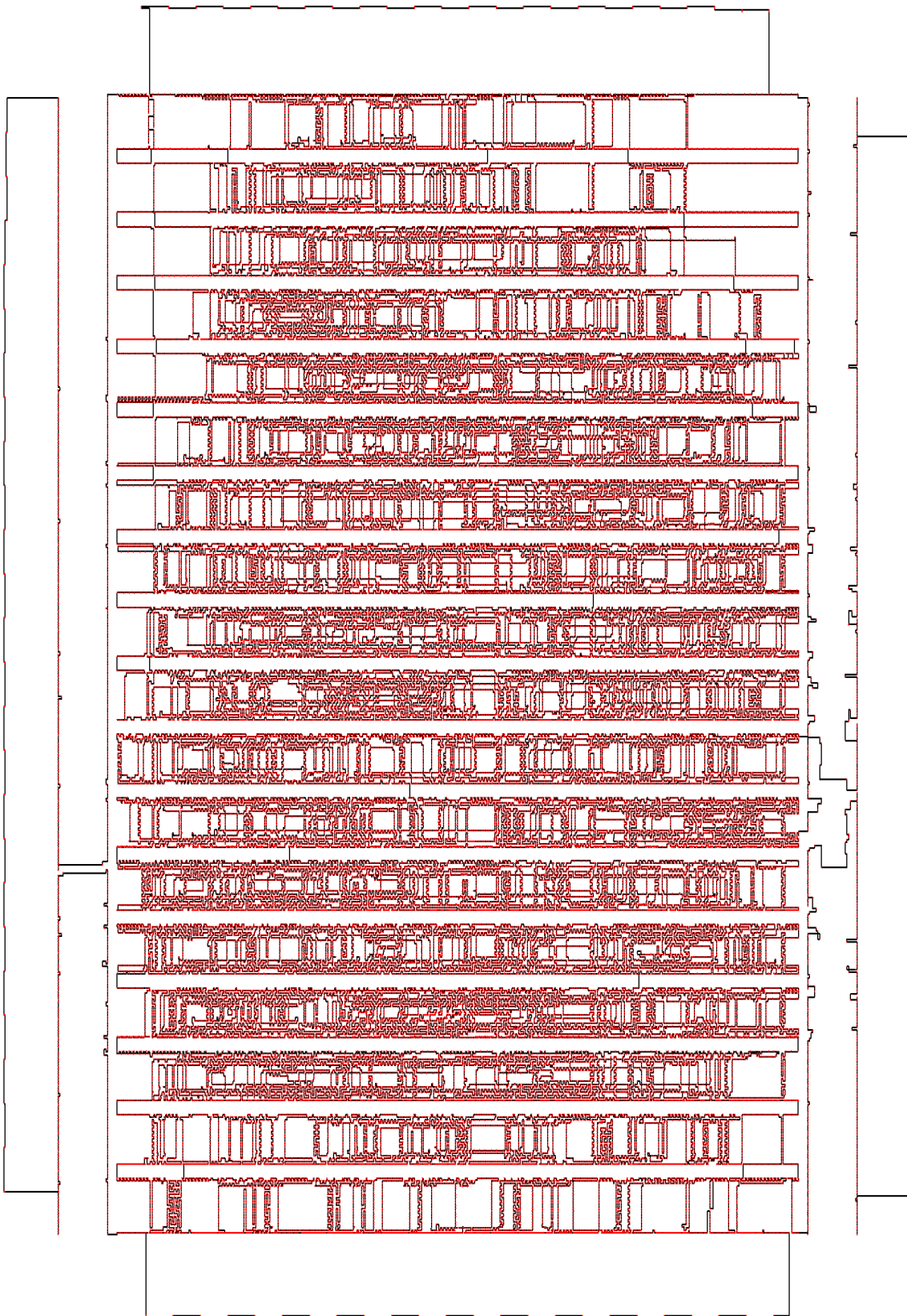


Figura 2.5: Ejemplo de la instancia resuelta por D. Applegate et al. para el diseño de un circuito electrónico integrado a gran escala con un total de 85.900 vértices.

como Concorde [D. Applegate, R. E. Bixby, V. Chvátal, and W. Cook \(2003\)](#). De las 110 instancias disponibles en la librería TSPLIB, 106 han sido resueltas de modo óptimo utilizando dicho software, formado por algo más 130000 líneas de código en lenguaje ANSI C. En su desarrollo y evolución trabajan matemáticos y otros profesionales incorporando los últimos avances y las mejores técnicas disponibles. En la actualidad existen instancias con cientos de miles de “ciudades” o incluso con millones de ellas. A pesar de no conocerse la solución óptima para dichas instancias, se conocen soluciones aproximadas cuya diferencia respecto de la solución óptima está por debajo del 1%. Un buen ejemplo son las instancias popularizadas por [\(Kaplan y Bosch, 2005\)](#) en las que se muestran distintas obras pictóricas universales, como el auto retrato de Vincent Van Gogh, el retrato de Juan de Pareja de Diego de Velázquez o la Mona Lisa de Leonardo Da Vinci. En 2009, el autor Robert Bosch creó una instancia con una representación de la Mona Lisa con un total de 100.000 ciudades tal y como se muestra en la figura 2.6. Ese mismo año, en 2009, el investigador Yuichi Nagata resolvió dicha instancia utilizando un algoritmo genético paralelo, proporcionando la mejor solución hasta la fecha, los detalles de dicha instancia y de otras obras pueden encontrarse en [Honda et al. \(2013\)](#). Trabajos posteriores del mismo investigador resolvieron instancias con hasta 200.000 ciudades, utilizando para ello un tipo de cruce conocido como EAX, proporcionando soluciones que mejoran las obtenidas por otras heurísticas presentes en el estado del arte [Nagata y Kobayashi \(2013\)](#).



Figura 2.6: Ejemplo de la instancia propuesta y resuelta por Robert Bosch para una instancia con un total de 100.000 ciudades en la que se representa la obra pictórica de Leonardo Da Vinci Mona Lisa.

2.1.3. Formulación matemática

Una vez descrita las principales características del TSP, así como una breve evolución histórica del mismo, se pasa a detallar la formulación matemática propuesta por (Dantzig et al., 1954). Sea $G = \{V, A\}$ un grafo finito, donde $V = \{v_1, \dots, v_n\}$ representa el conjunto de vértices, $A = \{(i, j) | i, j \in V\}$ el conjunto de aristas y $d_{ij} = d_{ji} \in \mathbb{R}^+$ la distancia de ir desde el vértice v_i al v_j o viceversa. Para todo $(i, j) \in A$ se definen un conjunto de variables binarias de decisión x_{ij} tal que:

$$x_{ij} = \begin{cases} 1, & \text{si } x_{ij} \text{ pertenece a la solución} \\ 0, & \text{en caso contrario} \end{cases} \quad (2.1)$$

La función objetivo del mismo queda definida del siguiente modo

$$\begin{aligned} &\text{minimizar} && \sum_{(i,j) \in A} d_{ij} x_{ij} \\ &\text{sujeto a} && \sum_{j \in V} x_{ij} = 2 \quad \forall i \in V \\ &&& i, j \in \{0, 1\} \end{aligned} \quad (2.2)$$

La restricción $\sum_{j \in V} x_{ij} = 2 \quad \forall i \in V$ garantiza que se debe visitar cada ciudad exactamente una vez, sin embargo, no impide que se formen subcircuitos o subtours, tal y como se muestra en la siguiente imagen 2.7.

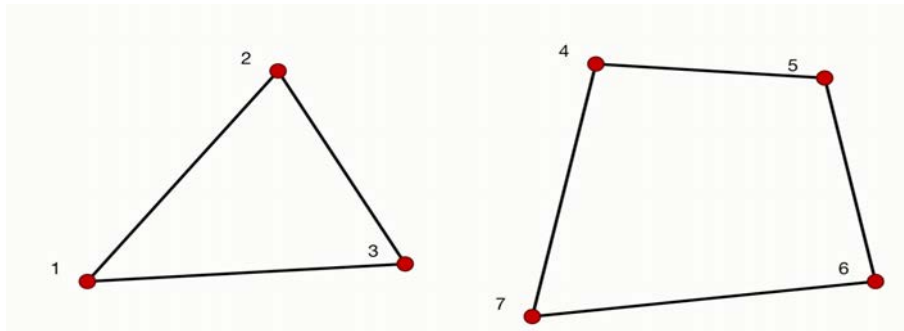


Figura 2.7: Ejemplo en el que se muestran dos subcircuitos para una instancia del TSP con 7 nodos.

Para eliminar dichos subtours se deben añadir nuevas restricciones, en concreto se debe cumplir:

$$\sum_{i,j \in S, i \neq j} x_{ij} \leq |S| - 1 \quad \forall S \subset V, S \neq \emptyset \quad (2.3)$$

De este modo para todo subconjunto $S \subset V$ el número de vértices presentes en la solución sea como máximo $|S| - 1$, en el caso de la figura 2.7 se puede comprobar como el conjunto formado por los vértices $S = \{1, 2, 3\}$ no cumple dicha restricción, ya

que $\sum_{i,j \in \{1,2,3\}, i \neq j} x_{ij} = 3 > 2 = |\{1, 2, 3\}| - 1$. Aunque la restricción anterior es suficiente, genera un nuevo problema, el número de subconjuntos crece de modo exponencial y consecuentemente el número de restricciones a crear. Si existen n ciudades en V existen un total de $2^n - 2$ subconjuntos $S \in V$ excluyendo $S = V$ y $S = \emptyset$, supondría insertar $2^n - 2$ restricciones siguiendo lo establecido en la ecuación 2.3, más las n restricciones especificadas en la ecuación 2.2. En lugar de incluir todas las restricciones de subtours en el modelo inicial, dichas restricciones se añaden un modo más sofisticado, en lo que se conoce como restricciones "perezosas" o planos de corte. Inicialmente dichas restricciones no son añadidas al modelo inicial, sino que una vez que se resuelve el modelo sin esas restricciones (modelo "relajado") se comprueba cual es la longitud del menor subtour existente, en adelante m , si $m = n$ se ha llegado a la solución, en caso contrario $m < n$ define un subtour y un subconjunto S con $|S| = m$. En este último caso se añaden la restricción definida para los subtours 2.3 y se continúa con la resolución del problema hasta que $m = n$. Este tipo de restricciones se pueden añadir en la mayoría de solvers modernos como ([Gurobi Optimization, 2015](#)) y ([ILOG CPLEX Optimization Studio, 2015](#)).

2.1.4. Tipos de problemas

Hasta el momento la descripción presentada del Problema del Viajante ha estado centrada en su descripción clásica, en lo que se conoce como Problema del Viajante simétrico. Como se describirá en esta sección, existen numerosas variantes del mismo, las cuales han ido evolucionando y adaptándose a las nuevas necesidades, quedando caracterizadas y diferenciadas especialmente por las restricciones que presentan cada una de ellas.

2.1.4.1. Problema del Viajante simétrico.

Sea $G = \{V, A\}$ un grafo finito, donde $V = \{v_1, \dots, v_n\}$ representa el conjunto de vértices, $A = \{(r, s) | r, s \in V\}$ el conjunto de aristas y $d_{rs} = d_{sr} \in \mathbb{R}^+$ el coste asociado al transitar por la arista $(r, s) \in A$, es decir ir del vértice v_r al v_s , se desea encontrar el ciclo Hamiltoniano que minimiza el coste total. En el caso de que los vértices vengan definidos por sus coordenadas x_i, y_i y la distancia $d_{rs} = \sqrt{(x_r - x_s)^2 + (y_r - y_s)^2}$ hablamos de Problema del Viajante Euclídeo. Se debe destacar que en algunas aplicaciones reales, como transporte por carretera, medir la distancia entre dos ciudades como una distancia Euclídea puede suponer una aproximación poco precisa y alejada de la realidad [Katragjini et al. \(2010\)](#); [Rodríguez y Ruiz \(2012\)](#). En la figura 2.8 se puede apreciar como dichas distancias pueden ser significativamente distintas.

2.1.4.2. Problema del Viajante asimétrico.

En el caso de no cumplirse la restricción de simetría descrita en el apartado anterior, $d_{rs} \neq d_{sr}$ el TSP se denomina Problema del Viajante asimétrico o no simétrico. La elección del tipo de distancia simétrica o no vendrá determinada por el tipo de problema a resolver, en ciertas aplicaciones puede ser razonable utilizar una distancia simétrica y euclídea, mientras que otras la elección de distancias simétricas puede suponer variaciones muy significativas y desviadas de la realidad, tanto en el esfuerzo para la resolución del

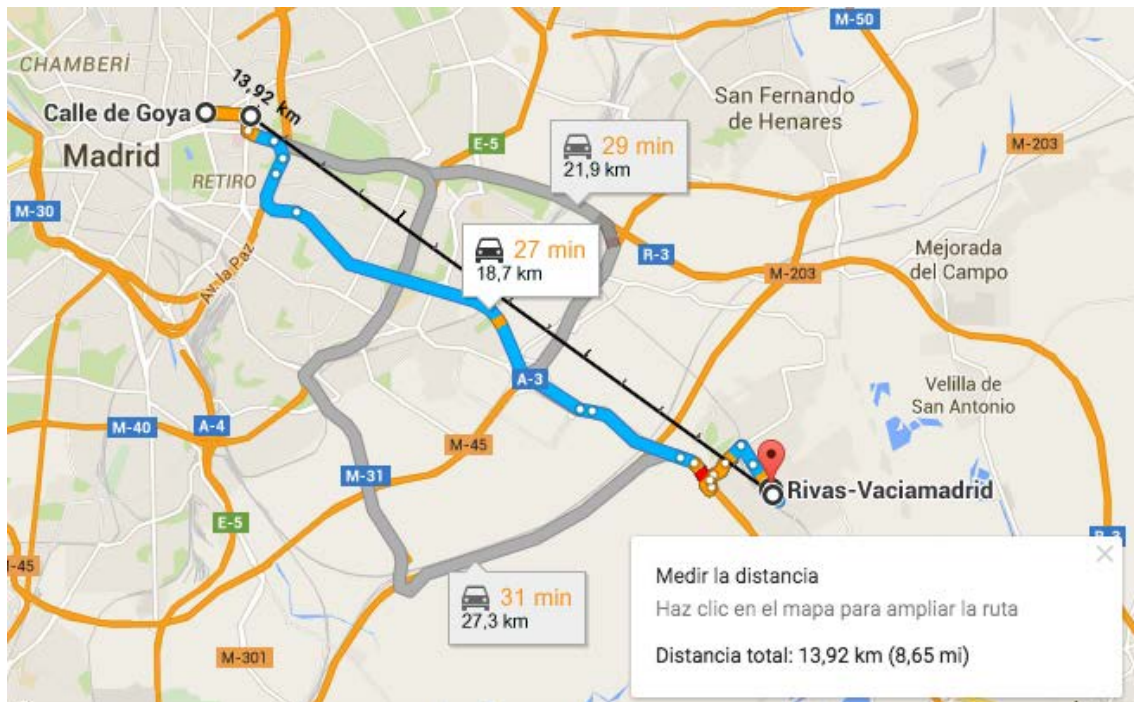


Figura 2.8: Ejemplo en el que se aprecian las diferencias entre utilizar una distancia Euclídea y la distancia real proporcionada por un sistema de información geográfica (Google Maps).

problema, como en la calidad de la solución obtenida [Rodríguez y Ruiz \(2012\)](#). Resulta obvio que cualquier técnica capaz de resolver el Problema del Viajante asimétrico, también es válida para la resolución del Problema del Viajante simétrico, de hecho existen algunas técnicas para transformar una instancia no simétrica en una simétrica, en concreto la transformación en tres nodos propuesta por [Karp \(1972\)](#) y la transformación basada en dos nodos propuesta por [Jonker y Volgenant \(1983\)](#).

2.1.4.3. Problema del Viajante con Ventanas de Tiempo

Hasta el momento, los distintos tipos de problemas planteados no han tenido en cuenta la componente temporal. En problemas más cercanos a la industria, es habitual que existan intervalos temporales en los cuales se debe producir cierta acción. El Problema del Viajante con Ventanas de Tiempo identificado por sus siglas en inglés *TSPTW* (*Traveling Salesman Problem with Time Windows*), es una de las variantes del Problema del Viajante más estudiadas y conocidas. Aplicado a un viajante de comercio, consiste en buscar la ruta de coste mínimo, empezando y volviendo a un mismo y único vértice inicial, denominado almacén. Se deben visitar todos los vértices una sola vez, dentro de una franja horaria especificada para cada vértice, conocidas como ventana temporal. En cada vértice visitado se debe prestar un servicio, el cual debe comenzar dentro de la ventana temporal, definida como el intervalo entre la fecha más temprana y la más tardía en la que está permitido el comienzo del servicio en dicho vértice. Está prohibido llegar al vértice después de la fecha u hora más tardía, en caso de llegar antes de la fecha u hora más temprana, está

permitido llegar antes de la fecha de inicio de inicio del servicio, en cuyo caso se incurrirá en un tiempo de espera.

En caso del problema que nos ocupa, el Problema del Viajante con Ventanas de Tiempo, dicha acción consiste en visitar un conjunto de clientes ⁶ denotado por $V = \{v_0, v_1, \dots, v_n\}$ a los que se debe visitar dentro de cierto intervalo temporal. Sea $G = \{V, A\}$ un grafo finito, donde $V = \{v_0, v_1, \dots, v_n\}$ es un conjunto compuesto por n vértices, más un vértice inicial (v_0) conocido como almacén, siendo $A = V \times V$ el conjunto de aristas que conectan los distintos vértices. Se define para cada arista $a_{ij} \in A$ un coste $c(a_{ij})$ que representa el tiempo de transitar entre los vértices i y j más un tiempo de servicio β_j en el vértice j . Además para cada vértice $v_i \in V$ existe una ventana temporal, $[a_i, b_i] \in \mathbb{R}^+$ siendo a_i el primer instante en el que se puede empezar el servicio en el vértice v_i y b_i el último instante, estando permitidos los tiempos de espera, es decir, si se visita el vértice v_i antes del instante a_i se incurrirá en un tiempo de espera, tal y como se muestra en la imagen 2.9. Una solución al problema es aquella que visita cada vértice exactamente una vez, partiendo y regresando al vértice inicial v_0 , dicho ciclo queda representado por $P = (p_0 = v_0, p_1, \dots, p_n, p_{n+1} = v_0)$ donde la secuencia $(p_1, \dots, p_k, \dots, p_n)$ es una permutación de los vértices de $V \setminus \{v_0\}$, donde p_k representa el k -ésimo vértice a visitar. Por último, los elementos p_0 y p_{n+1} representan el nodo inicial v_0 desde el que se parte y al que se debe regresar.

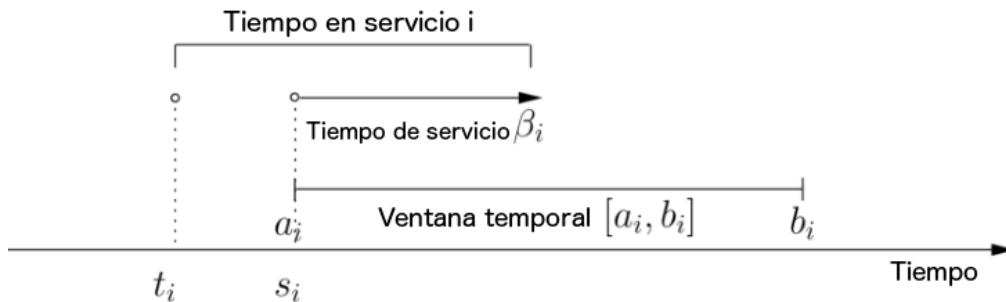


Figura 2.9: Ejemplo de tiempo total invertido en el *servicio* _{i} , como resultado del tiempo de espera más tiempo de servicio β_i , así como la venta ventana temporal $[a_i, b_i]$ el *servicio* _{i} .

Es importante destacar las implicaciones que resultan al añadir estas dos nuevas restricciones al Problema del Viajante con Ventanas de Tiempo, como son partir de un vértice inicial v_0 y la existencia de ventanas temporales para cada vértice, siendo esta última la de mayor impacto. En primer lugar y a diferencia del Problema del Viajante, la restricción de ventana temporal evita que cualquier permutación de los vértices de $V \setminus \{v_0\}$ sea una solución factible, el mero de decidir si para una instancia dada existe un permutación tal que se respeten todas las restricciones de ventana temporal, hace que el problema de decisión sea un problema \mathcal{NP} -completo. De modo análogo el problema de optimización pertenece a la clase de complejidad \mathcal{NP} -difícil Savelsbergh (1985). En segundo lugar, el hecho de añadir una ventana temporal en cada vértice, provoca que desde el punto de vista de teoría de grafos la solución óptima a dicho problema no debe ser

⁶Nótese que en la literatura especializada, a la hora de referirse al conjunto de nodos a visitar, no se habla por lo general de ciudades, sino de clientes a los que hay que visitar dentro de cierto horario.

necesariamente planar, es decir, pueden existir aristas de la solución óptima que se crucen entre sí, tal y como se aprecia en la figura 2.10. Finalmente, la función objetivo a optimizar pueden presentar distintos objetivos, aunque la mayor parte de la literatura especializada se centra en minimizar el tiempo invertido al transitar entre los distintos vértices, denominada TSPTW-TT, existe otra función objetivo que trata de minimizar el tiempo invertido en regresar al nodo inicial, denominada TSPTW. Los algoritmos y técnicas desarrollados para la variante TSPTW-TT se pueden aplicar con éxito a la segunda variante TSPTW-M mejorando en algunos las técnicas específicas presentes en el estado del arte [López-Ibáñez et al. \(2013\)](#).

n20w120.002 (218.0): 1 2 15 19 5 9 20 17 12 16 6 8 4 21 18 13 11 10 7 3 14

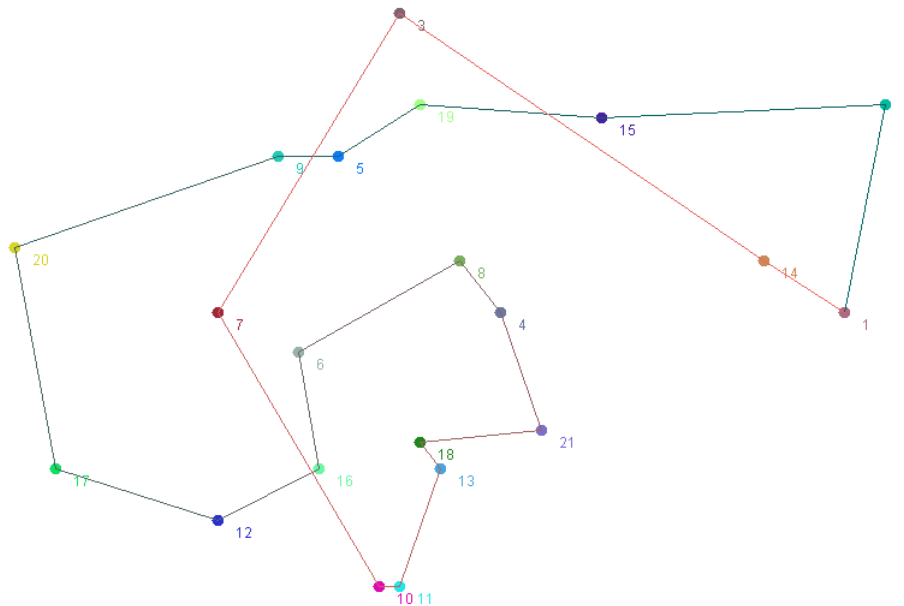


Figura 2.10: Ejemplo de la mejor solución para la instancia n20w120.002 propuesta por ([Gendreau et al., 1998](#)). El nodo inicial del que se debe iniciar la ruta es el nodo 1, se aprecia como el grafo resultante de la solución del mismo no es planar.

Desde una perspectiva histórica, los primeros trabajos para su resolución por métodos exactos proceden de ([Christofides et al., 1981a](#)) y ([Baker, 1983](#)), los autores proponen métodos de ramificación y acotación para resolver la variante TSPTW-M. El algoritmo propuesto en ([Christofides et al., 1981a](#)) se basaba en una técnica conocida como relajación del espacio de estados ([Christofides et al., 1981c](#)), resuelta mediante programación dinámica. El tamaño de las instancias resueltas por ambos autores es de un máximo de 50 vértices. Los autores ([Langevin et al., 1993](#)) también proponen un método de ramificación y acotación, pero basándose en una nueva formulación matemática, conocida como flujo de dos productos, más conocido por su nombre en inglés *Two-Commodity Flow Formulation*, resolviendo instancias con un máximo de 60 vértices. Distintos algoritmos basados en programación dinámica han sido propuestos por ([Dumas et al., 1995](#)), ([Mingozi et al., 1997](#)), ([Balas y](#)

Simonetti, 2001), resolviendo instancias de 200 y 120 respectivamente, Balas y Simonetti (2001) considera el caso particular en el que existen ciertas precedencias entre vértices. Focacci et al. (2002) propone un enfoque híbrido basado en programación por restricciones y técnicas de optimización para calcular el límite inferior de la solución óptima, resolviendo instancias de hasta 67 vértices.

Paralelamente a este tipo de técnicas exactas, se han ido desarrollando y surgiendo técnicas heurísticas que, a pesar de no garantizar la solución óptima objetivo que algunos casos logran, son capaces de resolver instancias de mayor tamaño y con un coste computacional menor. En Gendreau et al. (1992) se propone una heurística de inserción capaz de resolver instancias con hasta 200 vértices, Pesant et al. (1998) propone la utilización de programación lógica por restricciones proporcionando nuevas y mejores soluciones a las instancias propuestas por Solomon (1987). (Calvo, 2000) propone una heurística basada en dos fases, una primera fase constructiva en la que se crea la solución inicial y una segunda fase en la que dicha solución es mejorada por medio de búsqueda local, solucionando instancias de hasta 200 vértices. Ohlmann y Thomas (2007) propone una variante de la metaheurística Recocido Simulado ⁷ obteniendo mejores resultados que la metaheurística convencional. Cheng y Mao (2007) propone el uso de Optimización Basada en Colonias de Hormigas, en lo que denominan ACS-TSPTW obteniendo resultados competitivos respecto al algoritmo MACS-VRPTW propuesto por Gambardella et al. (1999). También basado en Optimización Basada en Colonias de Hormigas los autores (López-Ibáñez y Blum, 2010) proponen una técnica híbrida, resultante de combinación entre la búsqueda en haz y Optimización Basada en Colonias de Hormigas a la que denominan Beam-ACO, mostrándose competitiva en hasta siete conjuntos de pruebas diferentes, proporcionando nuevas mejores soluciones en hasta once de ellas y solucionando instancias con hasta 235 vértices. Los autores (da Silva y Urrutia, 2010) proponen la utilización de una búsqueda basada en vecindarios variables ⁸ solucionando instancias de hasta 400 vértices con ventanas de tiempo muy amplias, resultados que fueron mejorados posteriormente por los autores Mladenovic et al. (2013).

2.1.4.4. Problema de m-vendedores de comercio

De las variantes del Problema del Viajante presentadas hasta el momento, siempre se ha incluido de modo tácito la restricción de incluir un único vendedor de comercio. Sin embargo, en situaciones más cercanas a la problemática real, nadie se imaginaría a un solo vendedor visitando miles de ciudades, ni visitando a miles de clientes. De la necesidad de una mayor eficiencia, surge esta variante conocida por sus siglas en inglés m-TSP (Multiple Traveling Salesman Problem), en adelante m-TSP, es una generalización del Problema del Viajante en el cual se permiten múltiples vendedores, de tal modo que el tiempo total requerido para visitar a todos los clientes se reduzca. El problema queda definido del siguiente modo: Dado un conjunto de nodos $V = v_1 \dots v_n$, existen un total de $m | m < n$ vendedores situados en un nodo inicial $v_i \in V$, el resto de nodos a visitar queda definido por $V_e = V \setminus \{v_0\}$, el problema consiste en encontrar las m rutas, de tal modo que todos los vendedores regresen al nodo inicial y todo nodo $v \in V_e$ sea visitado exactamente una

⁷ *Compressed Annealing*, variante de la metaheurística *Simulated Annealing* Kirkpatrick et al. (1983).

⁸ *Variable neighborhood search*.

Año	Técnica	Número de vértices	Autores
2013	heurística	400	Mladenovic et al. (2013)
2010	heurística	400	da Silva y Urrutia (2010)
2010	heurística	235	López-Ibáñez y Blum (2010)
2007	heurística	46	Cheng y Mao (2007)
2007	heurística	200	Ohlmann y Thomas (2007)
2002	exacta	69	Focacci et al. (2002)
2000	heurística	200	Calvo (2000)
1998	heurística	100	Gendreau et al. (1998)
1998	heurística	46	Pesant et al. (1998)
1997	exacta	120	Mingozzi et al. (1997)
1996	heurística	N/A	Carlton y Barnes (1996)
1995	exacta	200	Dumas et al. (1995)
1993	exacta	60	Langevin et al. (1993)
1992	heurística	200	Gendreau et al. (1992)
1983	exacta	50	Baker (1983)
1981	exacta	50	Christofides et al. (1981a)

Tabla 2.2: Evolución histórica de las diferentes técnicas usadas para la resolución del Problema del Viajante con Ventanas de Tiempo.

vez, minimizando el coste total de visitar dichos nodos. De acuerdo con [Bektas \(2006\)](#) las posibles variaciones de este problema son las siguientes:

- Múltiples nodos iniciales. En lugar existir un solo nodo inicial v_i , existe un conjunto de nodos iniciales $V_i = v_1 \dots v_j$, la cantidad de vendedores que parten desde cada nodo inicial viene determinada por $M_i = m_1 \dots m_j$, dicha variante se conoce por sus siglas en inglés MDMTSP (Multi-depot Mutiple Traveling Salesman Problem), esta variante tiene múltiples aplicaciones en el campo de la robótica ver [Oberlin et al. \(2009\)](#). De este problema existen dos variantes, una primera en la que los vendedores deben volver a su nodo inicial ⁹y otra en la que dicha restricción es relajada. Dicha relajación consiste en permitir que los vendedores puedan regresar a cualquier nodo inicial, siempre y cuando el número de viajeros que regresen a los nodos iniciales v_j sea igual al número que había inicialmente m_j .
- Especificación del número de vendedores. Existen dos variantes, una primera en la que el número de vendedores es una variable fija m y una segunda en la que el número de vendedores se debe determinar al solucionar el problema. En esta segunda versión, es habitual que deba cumplirse la restricción $m \leq k$, siendo k el número máximo de vendedores posibles.

⁹Conocida por sus siglas en ingles m-TSP fixed destination.

- Costes fijos. Cuando el número de vendedores no es fijo, se define un coste por el hecho de utilizar o no a un vendedor, dicho coste se añade a la función objetivo del mismo, de tal modo que soluciones con distinto número de vendedores se puedan comparar.
- Ventana temporal. Al igual que sucede con el Problema del Viajante con Ventanas de Tiempo (ver 2.1.4.3), pueden existir ventanas de tiempo que limiten el intervalo temporal en el que se puede visitar un vértice. En el caso de los m -vendedores de comercio, dicha restricción se refiere al intervalo temporal $[t_{ni}, t_{nf}] \in \mathbb{R}^+$ en el que se debe visitar el nodo en cuestión, siendo t_{ni} y t_{nf} el primer y último instante en el que se puede visitar el vértice v_n .
- Otras restricciones. Además de las restricciones anteriores, existen otras restricciones para adaptarse a distintas problemáticas reales. Como limitar el número de vértices que cada viajante puede visitar o limitar la distancia mínima/máxima que estos pueden recorrer. El primer caso se conoce Bounded m -TSP, dicho límite suele estar comprendido entre dos variables que indican el número máximo y mínimo de ciudades que deben/pueden visitar cada viajante de comercio, estas restricciones son especialmente útiles en aplicaciones a problemas reales. En (Necula et al., 2015a) se propone la utilización de Optimización Basada en Colonias de Hormigas para resolución de esta variante concreta, para medir la bondad de las soluciones se utilizan distintas instancias de la librería TSPLIB (Reinelt, 1991), en la figura 2.11 se muestra la solución obtenida para la instancia *rat99* de dicha librería.

El problema del m -TSP es un caso particular del Problema de Enrutamiento de Vehículos con restricciones de Capacidad (ver sección 2.2.3) en el que no existe limitaciones de capacidad, es por ello, que los distintos métodos de resolución existentes para el Problema de Enrutamiento de Vehículos con restricciones de Capacidad se puedan utilizar para resolver instancias del m -TSP, basta con asignar una capacidad lo suficiente grande a cada vehículo. De igual modo, el m -TSP se puede reducir al TSP cuando solo existe un vehículo (Bektas, 2006). En la formulación tradicional del m -TSP, la función objetivo consiste en minimizar la suma de distancias recorridas por cada uno de los viajeros de comercio, en determinadas situaciones dicha función objetivo no cubre las necesidades reales. Existe una variante conocida como min-max m -TSP en la cual, el objetivo es minimizar la máxima distancia recorrida por los m -viajantes de comercio França et al. (1995). En Necula et al. (2015b) se propone la utilización de Optimización Basada en Colonias de Hormigas para la resolución de dicha variante del m -TSP llegando a solucionar instancias, de hasta 99 nodos y 7 viajeros de comercio, en la figura 2.12 puede apreciarse una de las soluciones obtenidas para dicha instancia.

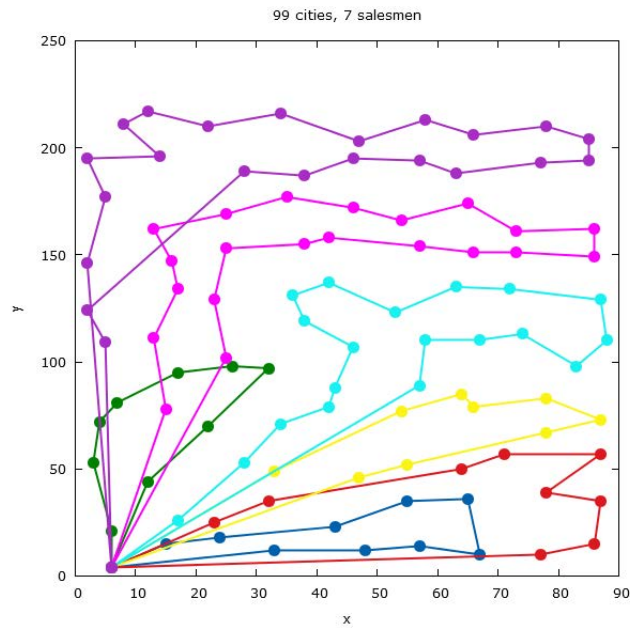


Figura 2.11: Ejemplo de la mejor solución para la instancia rat99 de la librería TSPLIB, adaptada al problema m-TSP compuesta por un total de 99 ciudades y 7 viajeros de comercio, resuelta por [Necula et al. \(2015a\)](#).

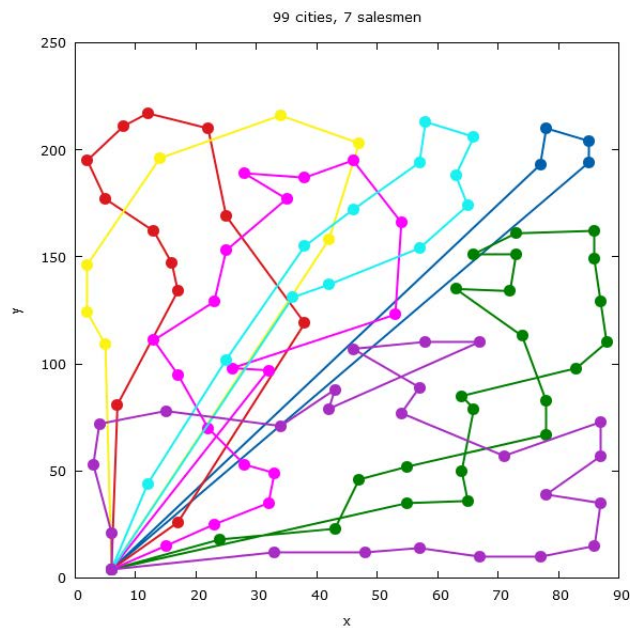


Figura 2.12: Ejemplo de la mejor solución para la instancia rat99 de la librería TSPLIB, adaptada al problema min-max m-TSP compuesta por un total de 99 ciudades y 7 viajeros de comercio, resuelta por [Necula et al. \(2015b\)](#).

2.2. Problema de Enrutamiento de Vehículos

En esta sección se presenta el conjunto de problemas denominados Problemas de Enrutamiento de Vehículos, describiendo sus variantes más representativas, las distintas técnicas de resolución, así como el tamaño de las distintas instancias resueltas. Prestando especial atención a aquellas variantes que presenten restricciones temporales y espaciales, por su fuerte relación con los problemas abordados en la presente Tesis.

2.2.1. Introducción

Conocido por sus siglas en inglés *VRP (Vehicle Routing Problems)*, en adelante *VRP*, este conjunto de problemas ha sido objeto de estudio intensivo durante los últimos 60 años. Desde que los autores [Dantzig y Ramser \(1959\)](#) lo propusieran como una generalización del TSP y lo aplicaran de modo satisfactorio a un problema de suministro de combustible, muchos han sido los esfuerzos y recursos que se han invertido en esta familia de problemas, tanto por su interés como problema de optimización combinatoria, como por su enorme interés práctico en diversos campos de aplicación como transporte, logística, comunicaciones y defensa entre muchos otros.

Una posible definición de esta familia de problemas es la siguiente:

Dado: Un conjunto de peticiones de transporte y una flota de vehículos.

El problema consiste en encontrar un plan para la siguiente tarea.

Tarea: Se debe determinar el conjunto de rutas que debe realizar cada vehículo, incurriendo en el mínimo coste posible; en particular se debe determinar qué vehículo estará encargado de gestionar cada petición y en qué orden, de tal modo que todas las rutas se puedan llevar a cabo, todas las peticiones de transporte sean atendidas y las distintas restricciones sean respetadas.

La definición proporcionada anteriormente no es en absoluto concisa, no se aportan detalles sobre cuestiones importantes como, ¿qué restricciones existen?, ¿desde dónde parten los vehículos?, ¿qué tipo vehículos existen?, ¿cómo es la demanda? o ¿cómo se mide el coste? entre muchas otras. Todas estas cuestiones responden a determinadas características de esta familia de problemas, dichas características han dado lugar a una cantidad casi ilimitada de variantes. En la tabla 2.4 se muestran las características más relevantes que caracterizan a esta familia de problemas, para una clasificación más detallada se recomienda consultar [Desrochers et al. \(1990\)](#); [Cherif-Khettaf et al. \(2014\)](#).

Características	Opciones
1. Grafo	No Orientado Orientado Mixto
2. Tiempo de desplazamiento entre vértices	Tiempo fijo Dependiente del horario Dependiente del vehículo Dependiente del horario y del vehículo

3. Tamaño de la flota	Un solo vehículo Número de vehículos limitado Número de vehículos ilimitado
4. Tipo de flota	Homogénea Heterogénea Compartimentado el vehículo por mercancías Sin compartimentar
5. Naturaleza de la demanda	Demanda determinada Demanda dinámica (dependiente del tiempo) Demanda estocástica (distribución de probabilidad) Demanda variable en función del cliente
6. Localización de la demanda	En nodos En arcos Mixta
7. Depósito	Depósito único Múltiples depósitos Depósitos intermedios de reaprovisionamiento
8. Tipo de operaciones realizadas por los vehículos	Reparto de mercancías Recogida de mercancías Recogida y reparto simultáneo de mercancías Primero recogida y después reparto o viceversa
9. Restricciones horarias	Sin restricciones Restricciones en inicio o en el final Restricciones en inicio y final Restricciones estrictas Restricciones no estrictas bajo penalización Ventana temporal única Múltiples ventanas temporales
10. Restricciones en la capacidad de los vehículos	Capacidad ilimitada Capacidad limitada e igual para todos los vehículos Capacidad limitada y diferente
11. Restricciones en la duración de las rutas	Duración ilimitada Duración limitada e igual para todos los vehículos Duración limitada y diferente Restricción estricta Restricción no estricta bajo penalización
12. Restricciones en el número de rutas	Múltiples rutas por vehículo, sin limitación Múltiples rutas por vehículo, con limitación

	Una única ruta por vehículo
13. Restricciones precedencias	Sin restricciones de precedencia Precedencias en la visita de clientes Precedencias en el uso de vehículos
14. Función objetivo	Minimizar los costes fijos Minimizar los costes fijos y variables Minimizar la distancia recorrida Minimizar el máximo de la distancia recorrida Minimizar la diferencia entre el máximo y el mínimo de las distancias recorridas Minimizar el número de vehículos Minimizar el tiempo empleado

Tabla 2.3: Características que determinan las distintas variantes de los Problemas de Enrutamiento de Vehículos. Tabla elaborada a partir de [Desrochers et al. \(1990\)](#); [Cherif-Khettaf et al. \(2014\)](#).

La gran variabilidad y versatilidad de esta familia de problemas, es quizás uno de los elementos que explican el enorme interés que suscitan tanto desde el punto de vista académico, como desde el punto de vista práctico y empresarial.

Desde el punto de vista de la investigación, la cantidad de publicaciones se cuenta por cientos de miles, de acuerdo con Google Scholar los términos de búsqueda “vehicle routing problem” dan como resultado más de 200.000 publicaciones; los mismos términos de búsqueda arrojan un total de 8003 publicaciones y más de 56800 patentes en la web Scopus ¹⁰ lo cual pone de manifiesto el enorme interés práctico e industrial que presenta la resolución de esta familia de problemas.

Centrándonos en el ámbito empresarial, son miles las compañías que cuentan con software específico para la resolución de este tipo de problemas, por su enorme interés práctico y por su impacto económico en ciertas organizaciones. De acuerdo con la encuesta de [Partyka y Hall \(2016\)](#), de las 17 compañías encuestadas que desarrollan productos de software específicos para esta familia de problemas, más del 30 % cuentan con más de 1.000 clientes, con unos costes que oscilan entre las decenas de miles de dólares. La mayoría de dicho software es capaz de resolver instancias de tamaño medio, entendiéndose por tal una flota compuesta por 50 vehículos y un total de 1.000 paradas, en aproximadamente 5 minutos, incluyendo restricciones con ventanas temporales de hasta dos horas de duración. Uno de los ámbitos de aplicación por excelencia de este tipo de problemas, son las compañías de transporte y mensajería, como la norteamericana UPS. Dicha compañía ha llevado a cabo importantes proyectos de I+D+i en la resolución de este tipo de problemas, al considerarlos una pieza estratégica en el desarrollo de su negocio. Destaca el proyecto ORION (On-Road Integrated Optimization and Navigation) con una duración aproximada de 10 años y con coste que ha superado los 250 millones de dólares. Gracias a dicho proyecto, la compañía ha reducido la distancia total recorrida por sus más de 55.000 conductores (sólo en EEUU), en 100 millones de millas, reduciendo al mismo tiempo la emisión de CO₂ en más 100.000 toneladas métricas, lo cual ha reportado (sólo en 2015) un ahorro total de

¹⁰Búsquedas realizadas en Julio del 2016.

más de 320 millones de dólares, proyecto por el cual le ha sido concedido premio Edelman entregado por la revista INFORMS (Horner, 2016).

2.2.2. Tipos de problemas

Tal y como se mostró en la tabla 2.3 existen numerosas características que determinan las distintas variantes dentro de esta familia de problemas, esta gran variedad ha dado lugar a una inmensa cantidad de problemas específicos, los cuales han sido objeto de estudio durante los últimos 50 años Laporte (2009). Cada uno de estos problemas específicos, están caracterizados por un conjunto de restricciones; en la tabla 2.4 se muestran los principales, detallando brevemente sus restricciones y características más relevantes, así como su acrónimo dentro de la literatura especializada. Posteriormente, se describen detalladamente aquellas variantes que por su características están relacionados con el problema de planificación de asistentes de atención domiciliaria 2.3 que será abordado en la presente Tesis.

Acrónimo	Denominación	Descripción
AVRP	Asymmetric Vehicle Routing Problem	La duración o coste entre dos vértices depende del sentido del trayecto
CVRP	Capacited Vehicle Routing Problem	Cada vehículo utilizado tiene una capacidad máxima que no puede ser superada
VRPLC	Vehicle Routing Problem with Length Constraint	La longitud de cada ruta no puede superar cierto límite
VRPM	Vehicle Routing Problem with Multiple use of vehicles	Cada vehículo puede ser asignado a más de una ruta dentro de un horizonte temporal
PVRP	Period Vehicle Routing Problem	Cada cliente ha de ser visitado un número de veces dentro de un horizonte temporal
FRP	Fixed Routes Problem	Se deben crear un conjunto de rutas que no pueden cambiar dentro de un horizonte temporal, mientras que las demandas de los usuarios pueden hacerlo
VRPSDV	Vehicle Routing Problem with Split Delivery	La demanda de cada cliente puede ser cubierta por varios vehículos
VRPTW	Vehicle Routing Problem with Time Windows	Cada vértice presenta un intervalo temporal en el que puede ser visitado, incluido el depósito
VRPSTW	Vehicle Routing Problem with Soft Time Windows	El intervalo definido anteriormente (VRPTW) puede ser incumplido mediante penalizaciones
VRPTD	Vehicle Routing Problem with Deadlines	Para cada vértice se define un tiempo final, antes del cual el vértice ha de ser visitado
FSMVRP	Fleet Size and Mix Vehicle Routing Problem	Centrado en minimizar tanto los costes de la ruta, como los costes fijos asociados a los distintos tipos de vehículos, existiendo una limitación en el uso de los mismos
VFMVRC	Vehicle Fleet Mix with Variable Unit Running Costs	Costes fijos y variables dependientes del tipo de vehículo, sin limitación en el número de los mismos

VRPHE	Vehicle Routing Problem with Heterogenous Fleet	Costes fijos y variables dependientes del tipo de vehículo, existiendo una limitación en el uso de los <u>mismos</u>
VRPB	Vehicle Routing Problem with Backhauls	No solo se llevan productos desde el almacén hasta los clientes, sino que también se recogen desde los clientes hacia al almacén. En el problema clásico, no se pueden recoger hasta que no se ha finalizado con las entregas
VRPDB	Vehicle Routing Problem with Deliveries and Backhauls	Variante del anterior (VRPB) en la que se permite la entrega y recogida simultánea siempre y cuando la restricción de capacidad sea respetada
PDP	Pickup and Delivery Problem	Se deben satisfacer un conjunto de peticiones de transporte, las cuales tienen un nodo origen y un <u>nodo destino</u>
MCVRP	Multi Compartment Vehicle Routing Problem	Se deben transportar varios productos, que deben estar separados unos de otros
min-max VRP	min-max Vehicle Routing Problem	Se debe minimizar la longitud de la ruta más larga
VRPPC	Vehicle Routing Problem with Precedence Constraints	Antes de visitar a cada cliente, pueden existir un conjunto clientes previos a los que se debe visitar
MDVRP	Multiple Depot Vehicle Routing Problem	Existen varios depósitos, cada uno tiene asignados varios vehículos los cuales deben partir y regresar al depósito asignado
VRPSF	Vehicle Routing Problem with Satellite Facilities	Existe un conjunto de depósitos intermedios en los que los vehículos <u>pueden reabastecerse</u>
OVRP	Open Vehicle Routing Problem	Se elimina la restricción de que los vehículos deban volver al depósito, aplicable en casos de <u>subcontratación</u>
LRP	Location Vehicle Routing Problem	Se debe determinar la localización del depósito o depósitos
DVRP	Dynamic Vehicle Routing Problem	Subfamilia de problemas en la que algunas variables varían a lo largo del tiempo
VRPVTT	Vehicle Routing Problem with Variable Travel Times	El tiempo de ir de un vértice a otro varía a lo largo del tiempo
SVRP	Stochastic Vehicle Routing Problem	Subfamilia de problemas en los que algunas de la variables son variables aleatorias
VRPST	Vehicle Routing Problem with Stochastic Travel Times	El tiempo de tránsito entre un vértice y otro está representado por una variable aleatoria
VRPSD	Vehicle Routing Problem with Stochastic Demands	La demanda de cada cliente está representada por <u>una variable aleatoria</u>
VRPSDC	Vehicle Routing Problem with Stochastic Demands and Customers	Tanto la existencia de los clientes, como su demanda está representado por una variable aleatoria

GVRP	Green Vehicle Routing Problem	Subfamilia de problemas centrada en aspectos medioambientales a lo largo del proceso de optimización, como emisión de gases, contaminación acústica y desperdicios
------	-------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 2.4: Tabla con las principales tipos de problemas pertenecientes a la familia del VRP, se recomienda consultar [Desrochers et al. \(1990\)](#); [Cherif-Khettaf et al. \(2014\)](#)

El número de variantes existente en esta familia de problemas es tan elevado, que existen numerosos trabajos centrados exclusivamente en construir una taxonomía de la misma, destacando los trabajos de [Desrochers et al. \(1990\)](#); [Eksioglu et al. \(2009\)](#); [Cherif-Khettaf et al. \(2014\)](#). A pesar de las numerosas variantes descritas, los problemas reales a los que se enfrentan las compañías suelen combinar varias de las características descritas con anterioridad, presentando un mayor grado de dificultad en su resolución. En el nivel más alejado de la realidad, encontramos los problemas clásicos o académicos, entre los que destacan el CVRP (Capacited VRP), VRPTW (VRP with Time Windows), HVRP (Heterogeneous fleet VRP), AVRP (Assymmetric VRP). Dichos problemas proporcionan entornos controlados, bajo los cuales es posible medir el rendimiento de nuevas técnicas, tanto exactas, como no exactas, permitiendo su comparación con las otras técnicas existentes en el estado del arte. En un segundo nivel, encontramos problemas que presentan un mayor grado de realismo, las cuales combinan varias características de las anteriores o bien añaden características nuevas como son formulaciones multiobjetivo, instancias de gran tamaño o la combinación con otros problemas de optimización combinatoria. Una gran parte de este tipo de problemas son resueltos mediante técnicas no exactas, siendo las técnicas metaheurísticas unas de las más utilizadas. Por último, en un tercer nivel más cercano a los problemas reales, tenemos problemas conocidos como RVRP (Rich VRP), a pesar de no existir una definición clara y ser un campo que se mantiene en constante evolución, algunas de las características que definen a esta familia de problemas son, presentar incertidumbre en las variables de decisión, dinamismo, diversidad de vehículos y clientes, multiperiodicidad a la hora de realizar las visitas, consideraciones espaciales a la hora de cargar las mercancías en los vehículos [Fuellerer et al. \(2009, 2010\)](#), así como aspectos medioambientales. Para la resolución de este tipo de problemas se han propuesto técnicas que combinan métodos exactos y metaheurísticas [Doerner y Schmid \(2010\)](#) e incluso algunos otros que combinan técnicas de simulación con metaheurísticas [Juan et al. \(2014\)](#).

2.2.3. Problema de Enrutamiento de Vehículos con restricciones de Capacidad

Conocido por sus siglas en inglés *CVRP (Capacited Vehicle Routing Problem)*, en lo sucesivo CVRP, es quizás una de las variantes más estudiadas de esta familia de problema. Propuesta por [Dantzig y Ramser \(1959\)](#) en 1959, los autores consideran una flota homogénea de vehículos, los cuales deben transportar un conjunto de bienes ¹¹ desde un único almacén o nodo central, hasta un conjunto de clientes. Todos los vehículos cuentan con la misma

¹¹El problema original fue planteado para un problema de real de suministro de combustible, el cual debía ser transportado desde un almacén central a un conjunto de estaciones de servicio.

capacidad de carga (flota homogénea), mientras que cada cliente deseada recibir una cantidad determinada de dichos bienes, demanda que debe ser satisfecha para todos los clientes. El coste incurrido al transitar desde un vértice a otro, viene dado por una matriz de coste, que usualmente mide la distancia, el tiempo de desplazamiento entre dos vértices o una combinación de ambos. El objetivo a resolver es encontrar el conjunto de rutas para todos los vehículos, tales que el coste total de visitar a todos los clientes respetando las restricciones de capacidad sea mínimo. En la figura 2.13 se muestra un ejemplo de solución óptima para un problema con 4 vehículos y 12 clientes.

Desde un punto de vista más formal el CVRP puede definirse del siguiente modo. Sea $G = \{N, A\}$ un grafo finito, donde $N = \{0\} \cup C$. Donde el nodo 0 representa el depósito y los nodos en $C = \{1, \dots, n\}$ representan el conjunto de clientes a visitar. Para cada cliente i existe una demanda no negativa d_i .

Sea $A = \{(i, j) | i, j \in V\}$ el conjunto de aristas y $c_{ij} = c_{ji} \in \mathbb{R}^+$ la distancia de ir desde el vértice v_i al v_j o viceversa, se asume que la desigualdad triangular es satisfecha, e.g $c_{ij} \leq c_{ik} + c_{kj} \forall i, k, j \in N$. Sea $K = \{1, \dots, m\}$ el conjunto de vehículos idénticos disponibles en el depósito con una capacidad q . Además para cada cliente en C existe una demanda denotada por $d_i | i \in \{1, \dots, n\}$.

Se define como ruta de un vehículo, una sucesión de vértices que empiezan y terminan en el depósito y que viene denotada por $r = \{v_0, v_1, \dots, v_n, v_{n+1}\}$ donde $v_0 = v_{n+1} = 0$ representa el depósito y $v_i \in \{1, \dots, n\} | i \in \{1, \dots, n\}$ representa a los clientes.

Una ruta factible, es aquella ruta que visita a los clientes como máximo una vez y para la cual no se excede la capacidad del vehículo, es decir $v_i \neq v_j \forall i, j \in \{1, \dots, n\} \wedge i \neq j$ y $\sum_{i=1}^n d_{v_i} \leq q$. El coste de la ruta viene dado por $c_r = \sum_{i=1}^n c_{v_i, v_{i+1}}$, mientras que el conjunto de clientes visitados por la ruta r se representa por C_r , de tal modo que el conjunto de clientes visitados en la ruta r sea un subconjunto del total de clientes a visitar ($C_r \subseteq C$).

Una solución factible para CVRP está compuesta por m rutas factibles, representado por $x = r_1, \dots, r_m$ cumpliéndose que $C_{r_k} \cap C_{r_l} = \emptyset, \forall k, l \in K$ y $\sum_{k=1}^m |C_{r_k}| = |C|$. Quedando definido el coste de una solución factible por $c_x = \sum_{k=1}^m c_{r_k}$.

Una solución óptima x^* es aquella que tiene el mínimo coste, $x^* = \arg \min_{x \in \chi} c_x$ donde χ representa el conjunto de todas las soluciones factibles al problema. En la figura 2.13 se ilustran los conceptos introducidos hasta el momento.

En 1985 Laporte et al. (1985) propone una formulación programación lineal entera mixta (MIP ¹²) para el CVRP donde la variable entera x_e indica el número de veces que una arista $e \in A$ es transitada en la solución. Sea $\delta(S) = \{(i, j) : i \in S, j \notin S \vee i \notin S, j \in S\}$ para $S \subseteq N$. Sea $v(S)$ el número mínimo de vehículos necesario para cubrir los nodos pertenecientes a S , la formulación queda como sigue:

$$\text{minimizar } \sum_{e \in A} c_e x_e \quad (2.4)$$

¹²Mixed Integer Programming

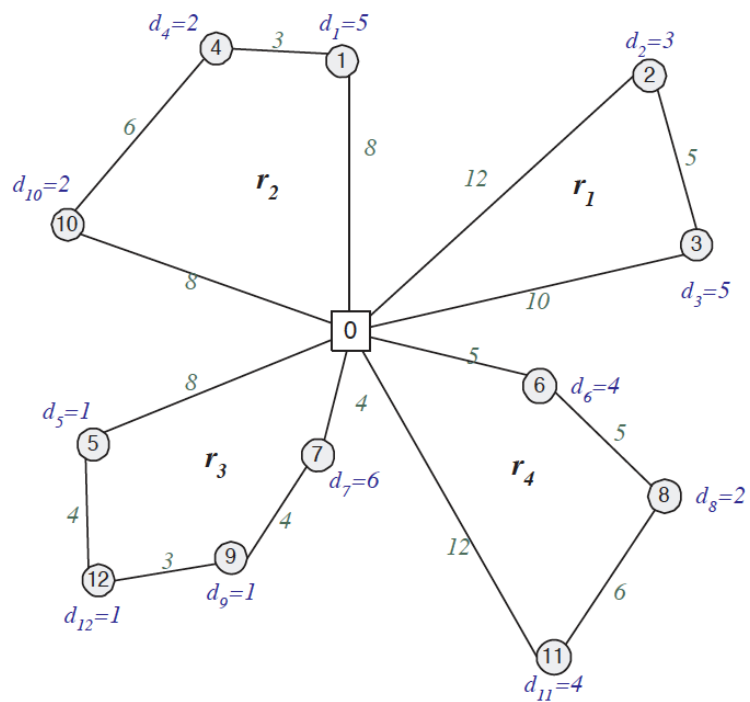


Figura 2.13: Ejemplo de resolución para el problema CVRP, en el que existen cuatro vehículos con una capacidad de 10. La demanda de cada cliente viene denotada por d_i y los costes de transición entre nodos en cada uno de los arcos.

$$\sum_{e \in \delta(i)} x_e = 2 \quad \forall i \in C \quad (2.5)$$

$$\sum_{e \in \delta(0)} x_e = 2m \quad (2.6)$$

$$\sum_{e \in \delta(S)} x_e \geq 2v(S) \quad S \subseteq N, S \neq \emptyset \quad (2.7)$$

$$x_e \in \{0, 1\} \quad e \notin \delta(0) \quad (2.8)$$

$$x_e \in \{0, 1, 2\} \quad e \in \delta(0) \quad (2.9)$$

Las ecuaciones (2.5) y (2.6), aseguran respectivamente, que cada cliente es visitado sólo una vez y que el número de rutas creadas es igual m . La restricción (2.7) impone tanto la conectividad de la solución como que la capacidad de los vehículos sea respetada al forzar que un número suficiente de aristas esté presente en cada subconjunto de nodos. Por último las restricciones (2.8) (2.8) aseguran respectivamente, que cada arista entre dos clientes es visitada como mucho una vez y que cada arista entre el depósito y cada cliente es visitada como máximo dos veces, considerando esta última restricción el caso en el que se visita un cliente y se vuelve al depósito, sin visitar a más clientes.

2.2.3.1. Métodos de resolución

Desde un punto de vista computacional la resolución del CVRP pertenece a la clase de complejidad \mathcal{NP} -difícil, ya que se trata de una generalización del m-TSP 2.1.4.4 [Lenstra y Kan \(1981\)](#). Dado su enorme interés práctico y al tratarse de un problema \mathcal{NP} -difícil su resolución ha sido abordada por numerosos investigadores durante estos últimos casi 60 años. Al igual que sucede con el resto de problemas de optimización combinatoria descritos hasta el momento, para su resolución existen dos grandes grupos de técnicas, las técnicas exactas y técnicas no exactas.

Técnicas exactas

Desde la perspectiva de las técnicas exactas, además de las formulaciones mostradas con anterioridad existen muchas otras cuyos detalles se recomienda consultar en [Achuthan y Caccetta \(1991\)](#); [Semet et al. \(2014\)](#); [Poggi y Uchoa \(2014\)](#), quizás las más extendidas son las propuestas por [Baldacci et al. \(2004\)](#) y por [Balinski y Quandt \(1964\)](#). Las técnicas exactas están basadas generalmente en la resolución de un problema de programación entera (Integer programming) en el que todas las variables de decisión toman valores enteros, o bien en programación entera mixta (Mixed Integer Programming) en el que algunas variables toman valores enteros y otras toman valores reales. Entre las técnicas de resolución destaca el algoritmo de ramificación y acotación, conocido como Branch & Bound (B&B) el cual sirve de base para otros algoritmos más sofisticados como son Branch & Cut (B&C) y Branch & Price (B&P). La idea principal en la que se basa el

método de ramificación y acotación es la dividir el espacio de búsqueda en subespacios, evaluando y acotando dentro de cada subespacio los límites inferior o superior dependiendo si el problema minimiza o maximiza la función objetivo. La evaluación de dichos límites permite determinar si el subespacio puede o no contener la solución óptima, en caso de que no la contenga el subespacio es podado, siendo ramificado y acotado nuevamente en caso contrario. Dependiendo de la técnica utilizada para estimar los límites inferiores, la técnica de ramificación y acotación (B & B) sirve de base para otras técnicas. Por ejemplo el método de Branch & Price utiliza una formulación en la que se debe seleccionar un subconjunto de todas las rutas factibles, utilizando la generación de columnas para estimar los límites inferiores dentro de cada nodo del árbol de búsqueda, ejemplos de esta aproximación se pueden encontrar en [Agarwal et al. \(1989\)](#); [Hadjiconstantinou et al. \(1995\)](#), resolviendo de modo exacto instancias con hasta 50 clientes. Otra aproximación es la conocida como Branch & Cut o planos de corte, en la cual los límites inferiores de cada nodo en el árbol de búsqueda son mejorados iterativamente al añadir un conjunto de restricciones que no son respetadas por la solución actual. Entre las formulaciones más sofisticadas destacan las propuestas de [Baldacci et al. \(2004\)](#); [Lysgaard et al. \(2004\)](#); [Baldacci et al. \(2008\)](#) capaces de resolver instancias con hasta 135 clientes, para conocer los detalles de dichas formulaciones se recomienda consultar [Baldacci et al. \(2012\)](#). Recientemente se ha publicado un nuevo conjunto de problemas de referencia conocido como CVRLIB¹³ la cual contiene instancias que van desde los 100 clientes hasta los 1.000. En [Pecin et al. \(2016\)](#) se propone una combinación de las dos técnicas anteriores, conocida como Branch& Bound & Price (B&B&P), la cual permite resolver 43 de las 100 instancias propuestas en CVRLIB. Dicha técnica permite resolver la mayoría de las instancias con hasta 275 clientes, de las 38 instancias en dicho rango no se encuentra la solución exacta para tres de ellas, empleando en el peor de los casos un total de más de 11 días para su resolución [Uchoa et al. \(2016\)](#). En la figura 2.14 se muestra un ejemplo que contiene la solución óptima para la instancia X-n275-k28 con un total de 275 vértices, nótese que dicha instancia tiene una particularidad a destacar, todos los clientes tienen exactamente la misma demanda que es igual a uno, los propios autores señalan que los tiempos de ejecución previstos para instancias con más de 300 clientes es de 17 días de ejecución.

Además del conjunto de problemas de la librería CVRLIB propuesta en 2014, existen otros problemas de referencia que han sido utilizados durante los últimos 50 años. La elaboración de estos problemas de referencia, permite a otros investigadores probar bajo un marco de referencia unificado nuevas técnicas e ideas, siendo útil conocer para cada instancia su solución óptima o el límite inferior de la misma. La resolución completa de un conjunto de problemas de referencia para el caso del CVRP ha durado en muchas ocasiones varias décadas, entendiéndose por tal la obtención de las soluciones óptimas para todas las instancias del mismo. [Christofides y Eilon \(1969\)](#) propusieron un conjunto de instancias con hasta 100 clientes cuya resolución no se completó hasta 2003, de modo análogo las instancias propuestas por [N. et al. \(1979\)](#) en 1979 con hasta 200 clientes, no se solucionaron de modo óptimo hasta 2013. Otros conjuntos de instancias como los propuestos por [Golden et al. \(1998\)](#) con hasta 420 clientes y por [Li et al. \(2005\)](#) con hasta 1200 clientes continúan sin haber sido resueltos de modo óptimo.

¹³Disponible en <http://vrp.atd-lab.inf.puc-rio.br/>

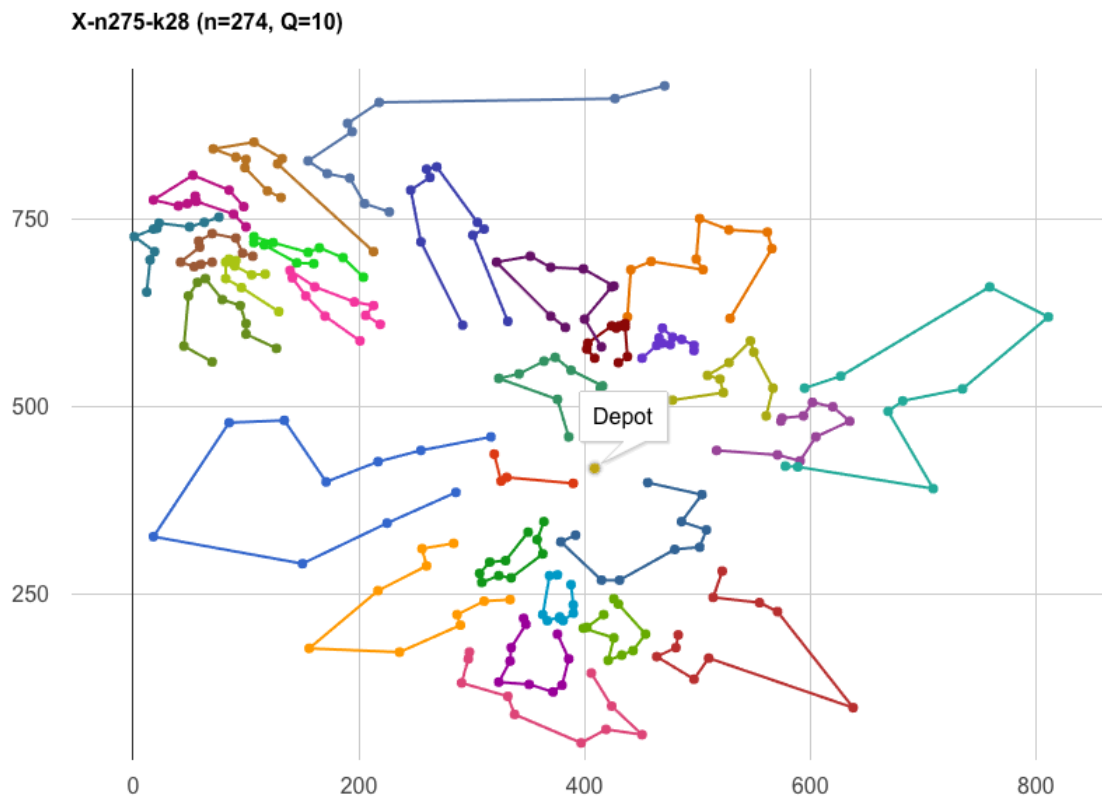


Figura 2.14: Solución óptima para la instancia X-n275-k28 de la librería CVRLIB propuesta por [Uchoa et al. \(2016\)](#). La solución óptima tiene un coste de 21245 y utiliza 28 vehículos cada uno de ellos con una capacidad $Q = 10$.

Técnicas no exactas

La principal ventaja de las técnicas exactas es que garantizan la obtención de la solución óptima, siendo su principal inconveniente los tiempos de ejecución requeridos, los cuales pueden resultar impracticables incluso para instancias de tamaño medio. En el caso del CVRP instancias con 300 clientes el tiempo requerido puede llegar hasta los 17 días ejecución (Uchoa et al., 2016), incluso utilizando las últimas técnicas exactas disponibles Pecin et al. (2016), con el objetivo de paliar estas deficiencias paralelamente a las técnicas exactas se han desarrollado diversas técnicas heurísticas y metaheurísticas. Dichas técnicas, a pesar de no garantizar la solución óptima proporcionan soluciones de buena calidad, pueden resolver instancias de mayor tamaño y requieren unos tiempos de ejecución menores. De acuerdo a la clasificación propuesta por Laporte y Semet (2002) las técnicas heurísticas se pueden clasificar en tres grandes grupos, heurísticas constructivas, heurísticas de mejora y metaheurísticas. Las heurísticas constructivas construyen una solución factible siguiendo ciertas reglas de construcción, una vez construida dicha solución no es mejorada. Este tipo de heurísticas son extremadamente rápidas, obteniendo generalmente una solución de una calidad inferior a otras técnicas. Las heurísticas de mejora parten de una solución completa al problema e intentan mejorarla de manera iterativa, utilizando para ello una serie de movimientos u operadores simples. Estas heurísticas de mejora solo aceptan movimientos que mejoren la solución de la que dispone, por lo que se pueden ver como procedimientos de intensificación guiados por la función objetivo. Por último, y como evolución de las heurísticas, encontramos las técnicas metaheurísticas, que combinan técnicas de búsqueda local con estrategias de un mayor nivel de abstracción permitiendo escapar de óptimos locales y realizando una búsqueda robusta en el espacio de soluciones.

Heurísticas constructivas

Las heurísticas constructivas fueron unas de las primeras propuestas para la resolución del CVRP, formando parte fundamental de muchas soluciones de software utilizadas hoy en día. Este tipo de heurísticas, se puede dividir a su vez en tres tipos, heurísticas de ahorros, de inserción y de dos fases. La primera heurísticas de ahorro fue propuesta por Clarke y Wright (1964), la idea subyacente en dicha heurística se basa en el concepto de ahorro, el cual se obtiene al visitar a dos clientes en la misma ruta, en lugar de servirlos en rutas, concepto que queda ilustrado en la figura 2.15. Sea i el último cliente de una ruta y j el primero de otra ruta, el ahorro asociado queda definido como $s_{ij} = c_{i0} + c_{0j} - c_{ij}$, si $s_{ij} > 0$ entonces visitar al cliente i e inmediatamente después al cliente j ahorra coste en la función objetivo. El algoritmo, construye inicialmente una solución con n rutas de un solo cliente, después se calculan todos los ahorros s_{ij} para cada par de clientes los cuales se ordenan de mayor a menor, examinando cada par de manera iterativa, conectando los vértices en caso de que se cumplan las restricciones. A pesar de su enorme velocidad para obtener soluciones factibles, el rendimiento de esta heurística es claramente mejorado por otras técnicas. La técnica original propuesta por Clarke y Wright ha sido objeto de numerosas investigaciones posteriores, las cuales han tratado de mejorar su eficiencia Golden et al. (1977); Nelson et al. (1985), la calidad de las soluciones obtenidas modificando la fórmula de ahorros Gaskell (1967) Paessens (1988) Altinel y Oncan (2004), así como algoritmos de emparejamientos centrados en realizar una mejor evaluación a la hora de evaluar los

ahorros [Altinkemer y Gavish \(1991\)](#); [Wark y Holt \(1994\)](#). Existiendo enfoques híbridos como el propuesto [Battarra et al. \(2007\)](#) por que combina la modificación de la fórmula de ahorros con algoritmos genéticos, estando estos últimos centrados en la optimización de los parámetros de dicha fórmula.

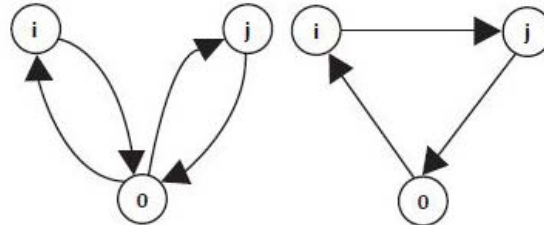


Figura 2.15: Heurística de ahorro propuesta por [Clarke y Wright \(1964\)](#), en la parte izquierda se observan dos clientes i, j que son visitados en rutas distintas, estando el depósito representado por el nodo 0 en ambos casos.

Heurísticas constructivas

Dada una solución factible para el CVRP, solución que puede ser generada con alguna de las técnicas constructivas expuestas con anterioridad, es posible realizar una modificación de tal modo que la calidad de su solución mejore y se sigan respetando las restricciones. Dicha modificación recibe el nombre de operador, movimiento o perturbación, de acuerdo al número de rutas que modifique se pueden distinguir dos tipos, operadores intra-ruta que afectan a una sola ruta y operadores inter-ruta los cuales afectan a varias rutas. Cabe destacar que los movimientos intra-ruta son movimientos enfocados a resolver el mismo problema que en el TSP, visitar todos los nodos con el menor coste posible ¹⁴, destacando los operadores 2-opt y 3-opt que son casos particulares del operador $\lambda - opt$ propuesto para el TSP por [Lin \(1965\)](#). Dicho operador elimina λ aristas dejando desconectados 2λ vértices que se vuelven a conectar de un modo nuevo, un ejemplo del operador 2-opt se puede encontrar en la figura 2.16. Considerar $\lambda > 3$ suele tener un coste computacional demasiado elevado, ya que existen $(\lambda - 1)!2^{\lambda-1}$ formas de volver a conectar los λ vértices.

Metaheurísticas

Paralelamente al desarrollo de las técnicas heurísticas expuestas con anterioridad, se han desarrollado toda una serie de técnicas denominadas metaheurísticas. De acuerdo con [Gendreau y Potvin \(2010\)](#), las técnicas metaheurísticas orquestan la interacción entre técnicas heurísticas y otras técnicas con un mayor nivel de abstracción, creando procesos de búsqueda en el espacio de soluciones, siendo dichos procesos robustos y capaces de escapar de óptimos locales. Para la resolución del CVRP muchas han sido las metaheurísticas propuestas, como son la búsqueda tabú (Tabu search) [Cordeau y Laporte \(2005\)](#), el recocido simulado (Simulated annealing) [Kirkpatrick et al. \(1983\)](#), la búsqueda local iterada (Iterated local search) [Baxter \(1984\)](#), los algoritmos genéticos (Genetic algorithms) [Prins \(2004\)](#); [Nagata \(2007a\)](#); [Nagata y Bräysy \(2009\)](#); [Vidal et al. \(2012\)](#), la búsqueda

¹⁴La restricción de capacidad es respetada ya que se parte de una solución válida y alterar el orden de visita de los clientes no afecta a la restricción de capacidad.

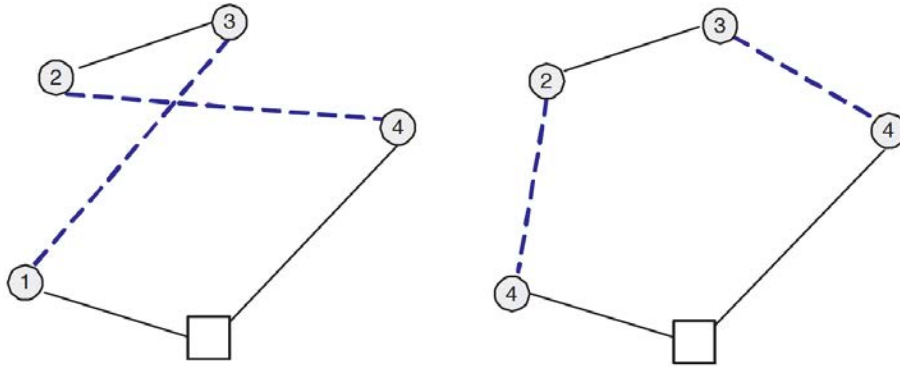


Figura 2.16: Ejemplo de operador 2-opt, en el que las aristas (1,3) y (2,4) de la parte izquierda son sustituidas por las aristas (2,4) y (3,4) en la parte derecha. Caso particular con $\lambda = 2$ del operador $\lambda - opt$ propuesto por [Lin \(1965\)](#)

dispersa (scatter search), la búsqueda en vecindad variable (Large neighborhood search) [Mladenović y Hansen \(1997\)](#); [Kytojoki et al. \(2007\)](#); [Hansen et al. \(2010\)](#) o la optimización basada en colonia de hormigas (Ant colony optimization) [Bullnheimer et al. \(1999\)](#); [Reimann et al. \(2002\)](#), entre otros. Todas estas metaheurísticas han sido aplicadas al CVRP pudiendo ser divididas en dos grandes grupos, las basadas en búsqueda local y las basadas en poblaciones, existiendo enfoques híbridos que pueden consultarse en [Laporte et al. \(2014\)](#).

2.2.4. Problema de Enrutamiento de Vehículos con Ventanas de Tiempo

El Problema de Enrutamiento de Vehículos con Ventanas de Tiempo conocido por sus siglas en inglés *VRPTW* (*Vehicle Routing Problem with Time windows*), en adelante VRPTW, es quizás, junto con el CVRP presentado en la sección 2.2.3, una de las variantes más estudiadas y conocidas dentro de esta familia de problemas. Dicha variante ha sido aplicada a numerosos problemas reales, entre los que cabe destacar la reposición de gasolineras [Cornillier et al. \(2009\)](#), la confección de rutas para autobuses escolares [Fügenschuh \(2009\)](#), la recolección de basuras [Kim et al. \(2006\)](#) o el reparto a domicilio [Braysy et al. \(2009\)](#) entre otros. De igual modo que el TSPTW (ver 2.1.4.3) es una extensión del TSP, el VRPTW es una extensión del CVRP. En dicha extensión, además de cumplir con las restricciones descritas para el CVRP, se incluyen una serie de restricciones conocidas como ventanas de tiempo o ventanas temporales. Para cada cliente se especifica una ventana temporal dentro de la cual debe ser visitado, si un vehículo llega antes incurrirá en un tiempo de espera hasta que pueda empezar a prestar el servicio en dicho cliente. La restricción de ventana temporal puede ser dura, es decir su incumplimiento invalidará la solución o puede ser blanda, es decir, su incumplimiento incurrirá en una penalización en la función objetivo, siendo predominante la primera opción. Respecto a la función objetivo generalmente las técnicas exactas tratan de minimizar la distancia total recorrida, mientras que las técnicas heurísticas minimizan en primer lugar el número de vehículos requeridos y en segundo lugar la distancia total recorrida.

De modo más formal el VRPTW puede definirse del siguiente modo. Sea $G = \{N, A\}$

un grafo finito, donde $N = \{0\} \cup C \cup \{n+1\}$ representa el conjunto de vértices y $A = \{(i, j) : i, j \in N, i \neq j\}$ representa el conjunto de aristas. Los nodos 0 y $n+1$ representan el depósito, que serán el inicio y el final de cada ruta y los nodos en $C = \{1, \dots, n\}$ representan el conjunto de clientes a visitar. El coste de transitar por cada arista $(i, j) \in A$ viene definido por c_{ij} y representa el coste incurrido al transitar del nodo i al j , de igual modo se define un t_{ij} que representa el tiempo de desplazamiento entre i y j . Al igual que en el CVRP 2.2.3, se define $K = \{1, \dots, m\}$ el cual representa el conjunto de vehículos idénticos disponibles en el depósito con una capacidad q , definiéndose para cada cliente $i \in C$ una demanda no negativa d_i , siendo $d_0 = d_{n+1} = 0$. Para cada cliente $i \in C$ se define una ventana temporal denotada por $[a_i, b_i]$, cualquier vehículo que visite al cliente i antes de a_i incurrirá en un tiempo de espera, no estando permitido visitar a dichos clientes después de b_i . La ventana temporal para el depósito queda definida del siguiente modo $[a_0, b_0] = [a_{n+1}, b_{n+1}]$, para que existan soluciones factibles al problema planteado se debe cumplir que $a_0 \leq \min_{i \in C} \{b_i - t_{0i}\}$ y $b_0 \geq \max_{i \in C} \{\max\{a_0 + t_{0i}, a_i\} + s_i + t_{i, n+1}\}$. Por último, para cada cliente $i \in C$ se define un tiempo de servicio s_i que representa el tiempo incurrido al visitar al cliente i siendo $s_0 = s_{n+1} = 0$. Como se puede comprobar existen ciertas aristas $(i, j) \in A$ que pueden ser eliminados al no cumplir con las restricciones de ventana temporal $a_i + s_i + t_{ij} > b_j$ o de capacidad $q_i + q_j > q$.

A continuación se presenta una formulación para la resolución de un problema de programación lineal entera mixta para el VRPTW, en el cual se describen dos tipos de variables: variables binarias $x_{ij}^k(i, j) \in A$ que toman el valor 1 si la arista (i, j) es visitada por el vehículo k y el valor 0 en caso contrario y variables continuas w_i^k que determinan el tiempo en el que el vehículo k comenzará el servicio en el cliente i , quedando la formulación como sigue:

$$\text{minimizar} \quad \sum_{k \in K} \sum_{(i,j) \in A} c_{ij} x_{ij}^k \quad (2.10)$$

$$\sum_{k \in K} \sum_{j \in N \setminus \{i\}} x_{ij}^k = 1 \quad \forall i \in C \quad (2.11)$$

$$\sum_{(i,j) \in A} d_i x_{ij}^k \leq q \quad \forall k \in K \quad (2.12)$$

$$\sum_{j \in N \setminus \{0\}} x_{0j}^k = 1 \quad \forall k \in K \quad (2.13)$$

$$\sum_{i \in N \setminus \{i\}} x_{ih}^k - \sum_{j \in N \setminus \{i\}} x_{hj}^k = 0 \quad \forall h \in C \forall k \in K \quad (2.14)$$

$$\sum_{i \in N \setminus \{n+1\}} x_{i, n+1}^k = 1 \quad \forall k \in K \quad (2.15)$$

$$w_j^k \geq w_i^k + s_i + t_{ij} - M(1 - x_{ij}^k) \quad \forall i, j \in N, k \in K \quad (2.16)$$

$$a_i \leq w_i^k \leq b_i \quad \forall i, j \in N, k \in K \quad (2.17)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j) \in A, k \in K \quad (2.18)$$

$$w_i^k \geq 0 \quad \forall i \in N, k \in K \quad (2.19)$$

La función objetivo definida en (2.10) minimiza el coste de las aristas seleccionadas para una solución, La restricción (2.11) asegura que cada cliente es visitado una única vez por un solo vehículo. La restricción (2.12) garantiza que ningún vehículo excede la carga máxima permitida. Las restricciones (2.13) - (2.15) garantizan que todo vehículo empieza y termina en el depósito y que el flujo es conservado en cada cliente,, es decir que un vehículo no sale de un cliente, si dicho cliente no ha sido visitado anteriormente. Con un M lo suficientemente grande, el cual se puede determinar del siguiente $M = \max\{b_i + s_i + t_{ij} - a_j, 0\} (i, j) \in A$, la restricción (2.16) establece que el vehículo k no comience el servicio j antes del instante $w_i^k + s_i + t_{ij}$ si dicho vehículo transita del nodo i al j . Por último la restricción (2.17) establece que la ventana temporal debe ser respetada en cada nodo y las restricciones (2.18) y (2.19) definen la variables binarias y continuas. Otra formulación alternativa para el VRPTW es aquella que se basa en la partición de todo el conjunto de rutas factibles, de las cuales se conoce su coste de antemano (Desaulniers et al., 2014). Otras formulaciones alternativas a las aquí expuestas se pueden encontrar en Kallehauge (2008). Sea R el conjunto de todas las rutas factibles, cada una de las cuales cumple con las restricciones del VRPTW, es decir respeta las restricciones de capacidad y de ventana temporal. Se define el coste de cada ruta $r \in R$ como c_r , entendiéndose por tal el coste mínimo de visitar a todos los clientes de la ruta r . Además, se define una variable auxiliar a_i^r que define si el cliente i es visitado por la ruta r y finalmente la variable binaria z_r será igual a 1 si la ruta r está presente en la solución y 0 en caso contrario, quedando la formulación del siguiente modo:

$$\text{minimizar } \sum_{r \in R} c_r z_r \quad (2.20)$$

$$\sum_{r \in R} a_i^r z_r = 1 \quad (2.21)$$

$$\sum_{r \in R} z_r \leq m \quad (2.22)$$

$$z \in \{0, 1\} \quad (2.23)$$

2.2.4.1. Métodos de resolución

La inclusión de las restricciones de ventana temporal, tiene varias implicaciones. En primer lugar, desde el punto de vista de complejidad computacional, el decidir si para una instancia concreta del VRPTW existe una solución es un problema \mathcal{NP} -completo, el problema de optimización es por definición un problema \mathcal{NP} -difícil al ser una extensión del CVRP (Savelsbergh, 1985). Además tal y como sucediera con el TSPTW 2.1.4.3, cuyas soluciones son no planares, las soluciones del VRPTW varían respecto a las obtenidas para el CVRP, tal y como se puede apreciar en la figura 2.17 las soluciones óptimas pueden presentar aristas que se crucen para una misma ruta. Un problema estrechamente

relacionada con el VRPTW es el m-TSPTW 2.1.4.4, de hecho, si se elimina la restricción de capacidad de los camiones o esta se supone infinita ambos problemas son equivalentes.

Dichas características han hecho del VRPTW uno de los problemas más estudiados dentro de los VRP. Al igual que para el resto de problemas expuestos, a lo largo de los años se han propuesto técnicas exactas y técnicas no exactas, de las cuales se muestra un breve resumen a continuación. Se recomienda consultar (Baldacci et al., 2012; Desaulniers et al., 2014) como referencia básica de para los métodos de resolución exactos y Potvin (2009a); Desaulniers et al. (2014); Labadie et al. (2016) para las técnicas de resolución no exactas.

Técnicas exactas

Durante los últimos 30 años han sido numerosas las técnicas exactas propuestas para la resolución del VRPTW, entre las distintas técnicas exactas se pueden diferenciar tres grandes grupos, a saber, ramificación y acotación (Branch & Bound), ramificación y poda (Branch & Cut) y ramificación, poda y poner precio (Branch & Cut & Price). En lo sucesivo se hará referencia a dichas técnicas por sus letras iniciales, a saber, B&B (Branch & Bound), B&C (Branch & Cut) y B&C&P (Branch & Cut & Price). La evolución de dichas técnicas ha estado estrechamente relacionada con la resolución del conjunto de instancias de referencia propuesta por Solomon (1987). Dicho conjunto, está compuesto por un total de 56 instancias todas ellas con 100 clientes, divididas en dos clases (C1, C2) cada una de las cuales contiene tres series (C, RC, R). Las clases hacen referencia a la amplitud de la ventana temporal, presentando ventanas temporales ajustadas las instancias de C1 y ventanas temporales más amplias las de la clase C2. Las series, hacen referencia a la distribución de los clientes, donde C significa que los clientes se encuentran en pequeños grupos (ver figura 2.18), R clientes que están distribuidos al azar (ver figura 2.17) y RC una mezcla de los dos anteriores, en ambas figuras para cada nodo i se muestra $[a_i|w_i|b_i]$, donde a_i y b_i representan los límites inferior y superior de la ventana temporal para el cliente i y w_i es el tiempo en el que es visitado dicho cliente de acuerdo a la solución. Aunque dicho conjunto de instancias fue propuesto en 1987 su resolución completa mediante técnicas exactas no se completó hasta 2012 Desaulniers et al. (2014).

Los primeros algoritmos exactos para la resolución del VRPTW pueden atribuirse a Kolen et al. (1987) en dicho trabajo se propone el uso de programación dinámica para computar los límites inferiores en un algoritmo de B&B, con dicha aproximación se resuelven instancias con hasta 15 clientes. A pesar de que existen numerosas formulaciones y trabajos Cordeau et al. (2002), las técnicas exactas que han obtenido mejores resultados son aquellas que utilizan la técnica de Generación de Columnas Dantzig y Wolfe (1960). Dicha técnica consiste en un proceso iterativo que es aplicable para resolver ciertos problemas de programación lineal entera mixta en los que existe un gran número de variables, un ejemplo de dicho problema es la relajación lineal del problema planteado en las ecuaciones (2.20)-(2.23). Destacan los trabajos de Jepsen et al. (2008) en el que se propone una serie cortes para mejorar un algoritmo de B&C&P, resolviendo un total de 45 de las 56 instancias propuestas por Solomon, el trabajo propuesto por Desaulniers et al. (2008) en el que se utiliza la búsqueda tabú como generador de columnas para un algoritmo de B&C&P resolviendo un total de 51 instancias y finalmente el trabajo de Baldacci et al. (2011) en el que se resuelven todas las instancias de Solomon excepto una por falta

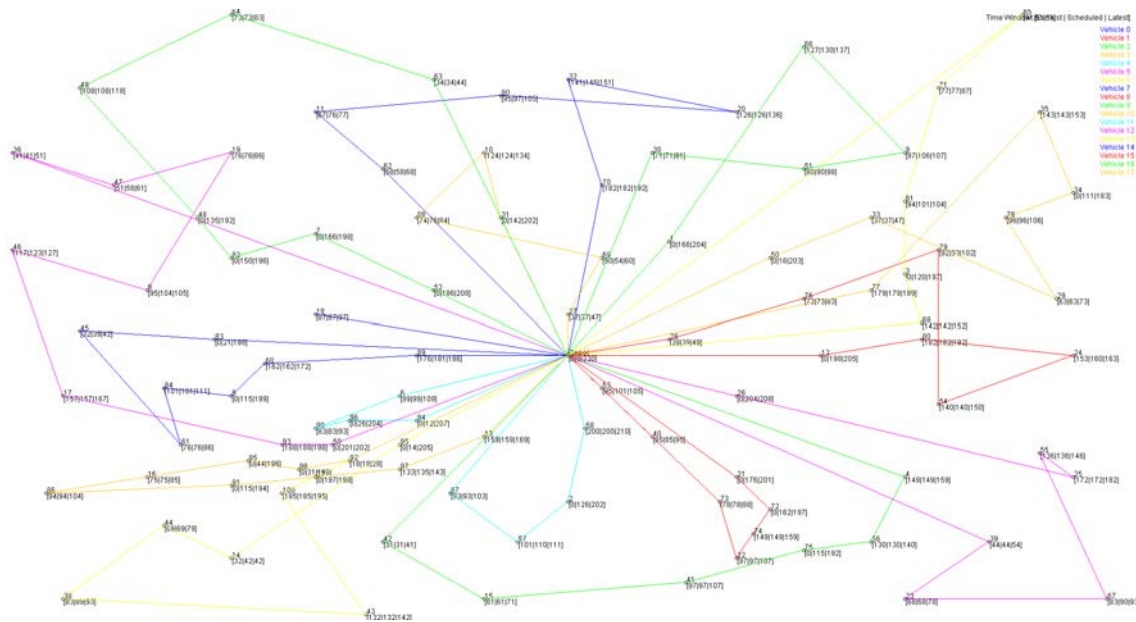


Figura 2.17: Solución óptima para la instancia R102 propuesta por Solomon en 1987 Solomon (1987).

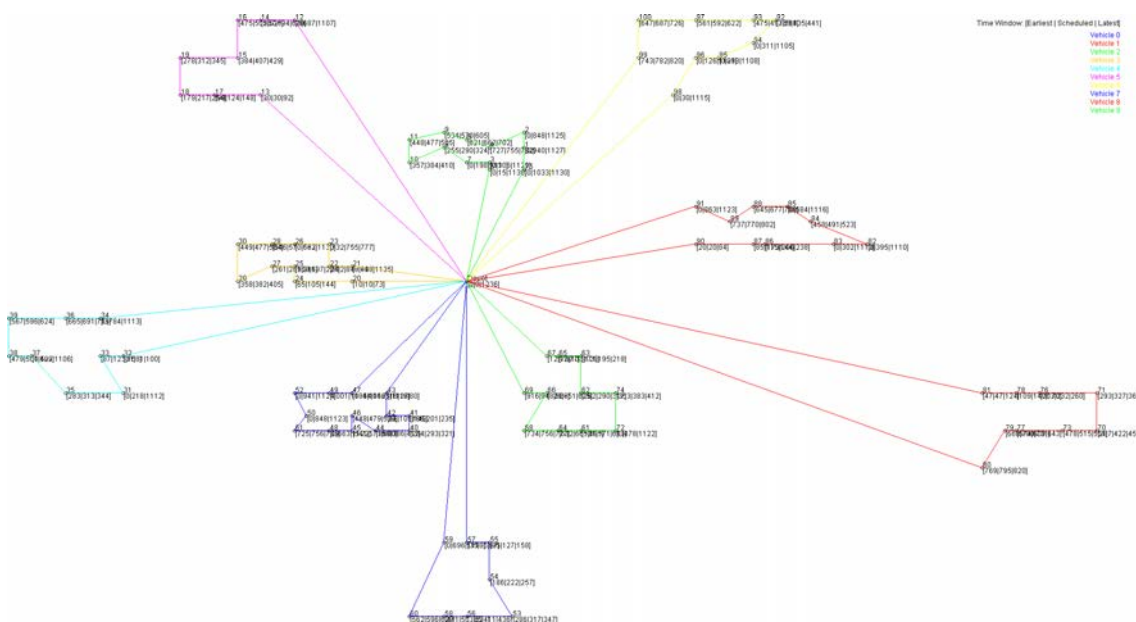


Figura 2.18: Solución óptima para la instancia C102 propuesta por Solomon en 1987 Solomon (1987).

de memoria. Reciente en 2012, el autor Stefan Ropke reportó haber resuelto todas las instancias de Solomon [Ropke \(2012\)](#) utilizando un algoritmo de B&C&P con una estrategia de ramificación jerárquica conocida como "strong branching".

Técnicas no exactas

Al igual que las técnicas exactas, las técnicas heurísticas han evolucionado a lo largo de estos últimos 30 años, dicha evolución ha permitido resolver instancias de mayor tamaño, de forma más eficaz y eficiente. Tal y como se describió en la sección 2.2.3.1, las técnicas heurísticas se pueden dividir en tres grandes grupos, heurísticas constructivas, de mejora y metaheurísticas [Potvin \(2009b\)](#). Desde el año 2000, la mayor parte de los trabajos sobre técnicas no exactas para el VRPTW, han estado centrados en la utilización de metaheurísticas, existiendo cierta escasez de trabajos que propongan nuevas heurísticas constructivas [Desaulniers et al. \(2014\)](#).

La primera diferencia que debe resaltarse respecto de las técnicas exactas es la función objetivo del VRPTW. Generalmente cuando se utilizan técnicas no exactas, ya sean heurísticas o metaheurísticas, la función objetivo está jerarquizada, es decir, la primera prioridad de la función objetivo es minimizar el número de vehículos, siendo la segunda prioridad minimizar la distancia total recorrida. Cabe destacar que la dimensionalidad de las instancias abordadas con técnicas heurísticas es en ocasiones un orden de magnitud superior a las instancias resueltas mediante técnicas exactas. El conjunto de instancias propuesto por [Gehring y Homberger \(2001\)](#) es el utilizado generalmente como referencia. En dicho conjunto y basándose en las clases (C1, C2) y las series (C,R, RC) propuestas por [Solomon \(1987\)](#), se añade un tercer atributo que representa el número de clientes a visitar existiendo instancias con 200, 400, 600, 800 y 1.000 clientes. Para cada posible combinación de clase (C1, C2), serie (C,R,CR) y número de clientes (200, 400, 600, 800 y 1.000) existen 10 instancias, estando formado el conjunto de instancias de referencia por un total de 300 instancias.

En términos generales las técnicas heurísticas, tanto constructivas de mejora o metaheurísticas, expuestas en la sección 2.2.3.1 para el CVRP, son también aplicables a la resolución del VRPTW, para ello basta con tener en cuenta las restricciones de ventanas de temporales. Por ello, en esta sección no se detallaran las distintas técnicas heurísticas, sino que simplemente se mencionaran las propuestas más recientes para la resolución del VRPTW, para una visión más detallada sobre dicha evolución histórica se recomienda consultar [Desaulniers et al. \(2014\)](#).

Metaheurísticas

Durante las dos últimas décadas, la mayor parte de los trabajos realizados con técnicas heurísticas para la resolución del VRPTW han estado centrados en la mejora o elaboración de nuevas metaheurísticas [Desaulniers et al. \(2014\)](#). Tal y como se expuso en la sección 2.2.3.1 las técnicas metaheurísticas orquestan la interacción entre técnicas heurísticas y otras técnicas con un mayor nivel de abstracción, las cuales permiten buscar en el espacio soluciones de un modo robusto, siendo capaces de escapar de óptimos locales. Dado que el VRPTW es una extensión del CVRP, las metaheurísticas propuestas para el segundo han sido adaptadas al primero. Entre todas las metaheurísticas propuestas para el VRPTW

destacan la búsqueda tabú (Tabu search) [Semet y Taillard \(1993\)](#); [Potvin et al. \(1996\)](#); [Taillard et al. \(1997\)](#), el recocido simulado (Simulated annealing), la búsqueda local iterada (Iterated local search), los algoritmos genéticos (Genetic algorithms) [Thangiah et al. \(1991\)](#); [Thangiah \(1995\)](#); [Nagata \(2007b\)](#); [Nagata y Braysy \(2009\)](#); [Nagata et al. \(2010\)](#) y meméticos [Moscatto y Cotta \(2010\)](#), la búsqueda en grandes vecindarios (Large neighborhood search) o la optimización basada en colonia de hormigas (Ant colony optimization) [Gambardella et al. \(1999\)](#); [S. et al. \(2014\)](#), entre otras. Todas estas metaheurísticas han sido aplicadas al VRPTW pudiendo ser divididas en dos grandes grupos, las basadas en búsqueda local y las basadas en poblaciones, existiendo enfoques híbridos que pueden consultarse en [Laporte et al. \(2014\)](#). En [Sintef \(2017\)](#) se pueden consultar las mejores soluciones conocidas para las instancias propuestas por [Gehring y Homberger \(2001\)](#), algunas de dichas soluciones han sido conseguidas por la empresa Quintiq, la cual combina un método híbrido [Quintiq \(2017\)](#).

2.3. Problema de planificación de asistentes de atención domiciliaria

En esta sección se presenta el conjunto de problemas denominados problemas de planificación de asistentes de atención domiciliaria, describiendo los trabajos previos a esta investigación, las distintas técnicas de resolución, sus restricciones y objetivos, así como el tamaño de las distintas instancias resueltas.

2.3.1. Introducción

La familia de problemas de planificación de asistentes de atención domiciliaria ha suscitado el interés de numerosos investigadores en los últimos años, interés que ha quedado reflejado en el creciente número de publicaciones existentes [Castillo-Salazar et al. \(2016\)](#); [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#). Este creciente interés está motivado por dos razones fundamentales, como son el envejecimiento progresivo de nuestra población y la necesidad de proveer servicios sanitarios o de asistencia en el lugar de residencia de nuestros mayores. Este tipo de servicios, tiene por objetivo atender las necesidades de nuestros mayores en su lugar de residencia, necesidades que pueden abarcar tareas sanitarias, de asistencia en el hogar, actuaciones de carácter preventivo (fisioterapia, terapia ocupacional), atención personal o de atención doméstica, entre otras, siendo este tipo de asistencia, la domiciliaria la preferida por la mayoría de nuestros mayores [Costa-Font et al. \(2009\)](#). Desde el punto de vista económico, en Europa entre el 1% y el 5% del presupuesto sanitario se destina a este tipo de servicios [World Health Organization \(2012\)](#), en otros países como EEUU el número de personas que recibieron este tipo de asistencia ascendió a 9 millones en 2014, con un total de más de 67000 proveedores y con un gasto estimado de entre 210.000 y 317.000 millones de dólares [L et al. \(2016\)](#). Entre los distintos retos que afrontan las compañías que proporcionan este tipo de servicios [Rest et al. \(2012a\)](#), está la necesidad proveerse de soluciones eficientes, que sean capaces de satisfacer la creciente demanda, manteniendo la satisfacción de los clientes y la de los propios empleados, siendo viables desde el punto de vista económico.

Las compañías que proporcionan este tipo de servicios, deben tomar decisiones en distintos ámbitos cada uno de los cuales se corresponden con diferentes problemas de optimización combinatoria, como son la planificación de horarios, la asignación de personal y la planificación de rutas de dicho personal [Gutierrez et al. \(2013\)](#). Otros autores ([Cisse et al., 2017](#)) señalan que dichas decisiones se toman dentro de tres niveles de decisión como son el nivel estratégico, el táctico y el operacional. A nivel estratégico las decisiones a tomar están relacionadas en cómo particionar el territorio en el que prestan servicios este tipo de compañías, creando clúster de clientes en diferentes distritos o regiones [Blais et al. \(2003\)](#). A nivel táctico el objetivo de las compañías que proporcionan este tipo de servicios es el dimensionamiento de los recursos, es decir una vez que se han creado un conjunto de distritos se debe identificar el conjunto de recursos necesarios para proveer el nivel de servicio adecuado en cada uno de ellos, estos recursos pueden ser humanos o materiales [MASc et al. \(2006\)](#). Por último a nivel operativo nos encontramos con dos tipos de problemas, el primero de ellos consiste en decidir qué asistente visitará a cada cliente [Lanzarone et al. \(2012\)](#), mientras que el segundo de ellos se debe especificar una

planificación temporal para cada paciente, es decir no solo se debe determinar qué asistente visitará a qué cliente, sino que además se debe determinar el orden de las visitas [Lanzarone et al. \(2012\)](#), pudiéndose resolver dichos problemas independientemente o de modo conjunto [Semih et al. \(2016a,b\)](#).

En la mayoría de los casos un conjunto de asistentes de atención domiciliaria, cuya cualificación profesional es variable, abarcando enfermeras, especialistas en terapia ocupacional, personal para la realización de tareas en el hogar o de cuidado personal, ha de ser asignado a un conjunto de clientes que requieren un conjunto de servicios heterogéneos, encontrándose cada cliente en su lugar de residencia, siendo habitual que cada asistente visite a lo largo de su jornada laboral a distintos clientes. A partir de la descripción anterior, se puede deducir que existirán un conjunto de restricciones que permitirán decidir qué asistente puede ser asignado a cada cliente, teniendo en cuenta la cualificación del asistente y el tipo de servicio requerido por el cliente, sin embargo, en el mundo real la problemática y restricciones a las que se enfrentan los proveedores de este tipo de servicios suele ser mucho más compleja. Por ejemplo, se debe tener en cuenta la legislación aplicable en materia laboral, la equidad a la hora de elaborar los horarios de sus trabajadores, la disponibilidad, lo cual puede incluir diferentes tipos de contratos, vacaciones, asistencia a jornadas de formación, baja por enfermedad, entre otros. Además los clientes requieren que se les preste el servicio a ciertas horas, como ya se presentó en las secciones 2.1.4.3 y 2.2.4 este tipo de restricción se conoce como ventana temporal, siendo su definición idéntica a la proporcionada dichos apartados. Finalmente existen todo un conjunto de restricciones asociadas al desplazamiento que debe realizar cada asistente entre las distintas visitas a cada cliente, en primer lugar hay que destacar que pueden existir distintos medios de transporte como puede el vehículo propio de los asistentes, el transporte público, en bicicleta, a pie o una combinación de los anteriores. A pesar de los retos que supone la toma de decisiones en los distintos niveles y de las distintas restricciones que deben cumplirse, la mayoría de las compañías siguen realizando las planificaciones de su personal de un modo manual [Eveborn et al. \(2006\)](#); [Braekers et al. \(2016\)](#); [Rest y Hirsch \(2016\)](#), este también es el caso de la compañía EULEN [Quintana et al. \(2017\)](#), cuya problemática será presentada en los siguientes capítulos.

Al tratarse de una familia de problemas relativamente nueva, y cuyo interés se ha visto incrementando en los últimos años, no existe una denominación única que englobe a esta familia de problemas habiendo recibido las siguientes denominaciones en la literatura especializada.

- Home health care scheduling problem [Begur \(1997\)](#)
- Home health care problem [Bertels y Fahle \(2006\)](#)
- Home care worker scheduling [Akjiratikar et al. \(2007a\)](#)
- Home help staff scheduling [Ikegami y Uno \(2007\)](#)
- Home care crew scheduling problem [Rasmussen et al. \(2012\)](#)
- Home health care routing and scheduling problem [Mankowska et al. \(2014\)](#)
- Therapist routing and scheduling problem [Bard et al. \(2014\)](#)

A pesar de las diversas denominaciones en este texto nos referiremos a esta familia de problemas como *Home care scheduling problems*, en adelante HCSP. Algunos autores como [Castillo-Salazar et al. \(2016\)](#) enmarcan a esta familia de problemas dentro de otra a la que denominan problemas de planificación y enrutamiento de la fuerza de trabajo ¹⁵, la cual además del HCSP abarca a otros problemas como la planificación de técnicos de mantenimiento o la planificación y enrutamiento de vigilantes de seguridad, estando todos ellos estrechamente relacionados con los problemas presentados en las secciones 2.1 y 2.2. El problema a resolver en el HCSP puede resumirse del siguiente modo:

Dado: Un conjunto de clientes distribuidos dentro de un área geográfica y un conjunto de asistentes.

El problema consiste encontrar un plan para la siguiente tarea.

Tarea: Se debe determinar el conjunto de visitas (tareas) que deberá realizar cada asistente, tratando de minimizar alguna métrica especificada previamente, como la distancia total recorrida o el tiempo de espera de cada cliente. En concreto, se debe determinar para cada asistente qué clientes debe visitar y en qué orden, de tal modo que todas los clientes sean atendidos y las distintas restricciones para clientes y asistentes sean respetadas.

Algunos autores como ([Kergosien et al., 2009a](#)) han modelado la resolución de esta familia de problemas como un generalización del m-TSP utilizando para ello una serie de restricciones que no estaban presentes en el problema original. Por otra parte, existe una mayoría de investigadores que ha modelado su resolución como una generalización del VRP a la cual se han añadido un conjunto de restricciones propias del problema ([Marcon y Chaabane, 2017](#)). La definición presentada anteriormente tiene bastante similitudes con la proporcionada en el apartado 2.2.1, en última instancia el HCSP consiste en el diseño de un conjunto de rutas sobre un horizonte de planificación, de tal forma que se preste el servicio de atención domiciliaria para cada cliente, a la vez que se minimiza o maximiza cierta métrica de interés y respetando las distintas restricciones existentes. Sin embargo, los estudios que abordan el HCSP se diferencian del VRP ya que presentan ciertas características que dificultan su resolución y que deben ser consideradas:

- Continuidad en la prestación de tareas. Este tipo de restricciones aseguran que cada cliente sea asignado a un conjunto reducido de asistentes de atención domiciliaria. Cuando dicho conjunto se reduce a un único asistente sobre todo el horizonte de planificación se habla de "continuidad total".
- Dependencia temporal y compatibilidad entre tareas. Por ejemplo, una tarea debe ser realizada inmediatamente después o dos tareas no pueden realizarse al mismo tiempo.
- Características de los clientes y de los asistentes. Cada asistente puede tener un conjunto de características o habilidades las cuales se deben tener en cuenta a la hora de ser asignado a un cliente. Del mismo modo cada cliente puede tener un conjunto de preferencias que limitan el conjunto de asistentes que les pueden visitar.

¹⁵Workforce scheduling and routing problems

2.3.2. Restricciones

Las diversas características de cada prestador de servicios y su problemática concreta se han visto reflejadas en la diversidad de publicaciones existentes hasta la fecha. Dichas características se pueden reflejar dentro de la formulación del HCSP de dos modos distintos, o bien se reflejan como restricciones que han de cumplirse o bien se presentan dentro de la función objetivo como criterio a optimizar. Estas restricciones pueden ser clasificadas de distintos modos, en primer lugar y atendiendo a cada uno de los actores del problema a los que afectan, se pueden distinguir restricciones que afectan a la empresa proveedora de servicios, a los asistentes de atención domiciliaria o a los clientes. A su vez dentro de cada uno de ellos podemos distinguir restricciones y agruparlas de acuerdo a su naturaleza, a saber, restricciones temporales, restricciones relativas a la asignación y restricciones geográficas [Cisse et al. \(2017\)](#).

2.3.2.1. Restricciones relativas a los proveedores de servicios

Restricciones temporales

Para los proveedores de servicios de atención domiciliaria, quizás una de las restricciones temporales más importantes es lo que se conoce como horizonte de planificación. Este término hace referencia al periodo temporal dentro del cual la organización debe resolver el HCSP, planificando sus asignaciones y las rutas de sus asistentes. De acuerdo con [\(Fikar y Hirsch, 2017\)](#) se pueden distinguir dos grandes grupos de problemas, los problemas de periodo único en los cuales el horizonte de planificación se centra en un solo día y los de periodo múltiple que se centran en la planificación de varios días, normalmente semanas y muy pocos casos meses. La utilización de uno u otro tipo de periodo depende de diversos factores, como son la calidad de la información, su disponibilidad y la frecuencia con la que esta se ve modificada a lo largo del tiempo [Cisse et al. \(2017\)](#). La mayor parte de las investigaciones resuelven problemas de un único día [Bertels y Fahle \(2006\)](#); [Akjratikar et al. \(2007b\)](#); [Bredström y Rönnqvist \(2008\)](#); [A. y P \(2011\)](#); [Rasmussen et al. \(2012\)](#); [Mankowska et al. \(2014\)](#) o una semana [Bräaysy et al. \(2009\)](#); [Bard et al. \(2012, 2014\)](#). En algunos casos dicho horizonte se extiende a más de una semana, como en [Wirnitzer et al. \(2016\)](#) donde el horizonte de planificación es extendido a 4 semanas. En la mayoría de los casos la variabilidad de servicios se produce dentro de la propia semana, este hecho facilita la planificación mensual, ya que al no existir variación entre los servicios de las distintas semanas el problema mensual es fácilmente particionable. En [Hewitt et al. \(2016\)](#) se señalan las ventajas de utilizar horizontes de planificación más extensos entre dos y tres meses. En concreto los autores señalan las mejoras alcanzadas al no partir el problema de varios meses en distintas semanas y resolver dichos problemas semanales de forma individual.

La otra restricción temporal referida a los proveedores de servicios de atención domiciliaria, es la frecuencia con la que se planifican las rutas y asignaciones dentro del horizonte temporal. Por ejemplo, dentro una planificación semanal pueden existir casos en los que las decisiones para cada día se puedan tomar de modo independiente, es decir, las rutas y asignaciones generadas para el lunes, no influyen en las rutas generadas para los siguientes días. Este es el caso de [Mankowska et al. \(2014\)](#) donde las decisiones para cada día son tomadas de modo independiente. Del mismo modo, existen situaciones en las que existe

cierta dependencia entre los servicios de distintos días por lo que las decisiones sobre las asignaciones y rutas se deben realizar de modo conjunto al principio del horizonte de planificación [Bard et al. \(2013\)](#).

Restricciones relativas a la asignación

Este tipo de restricciones modelan y limitan la posibilidad de asignar los asistentes de atención domiciliaria a los distintos clientes, limitando qué asistente puede ser asignado a cada cliente, sin tener en cuenta el aspecto temporal, las preferencias de los usuarios, ni las cualificaciones de los asistentes. En la bibliografía especializada este tipo de restricciones han recibido distintos nombres como “continuidad del servicio en la prestación de tareas”, “lealtad entre cliente-asistente” [Nickel et al. \(2012\)](#) o “regularidad de los empleados” [Gamst y Jensen \(2012\)](#). A lo largo del texto nos referiremos a este tipo de restricciones como “continuidad del servicio”. Cabe destacar como este tipo de restricciones también aparecen en los problemas VRP, denominándose ConVRP (consistent VRP) [Kovacs et al. \(2015\)](#) debiéndose garantizar que los clientes son visitados por el mismo vehículo y a una hora aproximadamente similar. En el ámbito del HCSP, dicha característica se puede prestar en distintos grados: no considerándose, considerándose parcialmente dentro la función objetivo o estableciéndose como restricción de tal modo que se asegure que los servicios prestados a un cliente son solo realizados por uno y solo de los asistentes de atención domiciliaria, denominándose esta última como continuidad total.

La continuidad total en la prestación del servicio es importante tanto para los clientes como para las propias organizaciones. Al existir un único asistente asignado a cada cliente, las organizaciones se aseguran que la pérdida de información entre distintos asistentes sea mínima. Por otro lado, se establecen una serie de relaciones interpersonales entre asistente y cliente, que son percibidos por los clientes como uno de los factores más importantes a la hora de valorar la calidad del servicio prestado. Consiguiéndose una mejora de desempeño de los asistentes al encontrarse estos en un entorno que conocen y en el que se desenvuelven con facilidad [Freeman y Hughes \(2010\)](#). La continuidad total también es especialmente deseable en algunos contextos como la maternidad [Bowers et al. \(2015\)](#) en los que la misma matrona debe aplicar cuidados prenatales y post natales a la madre y al recién nacido. Esta característica de continuidad total también está presente en el problema de la compañía EULEN, el cual será expuesto en el capítulo 3. La continuidad parcial se observa normalmente en situaciones en las que los clientes deben recibir varios servicios en el mismo día, pudiéndose establecer asistentes de referencia que coordinan la prestación de los distintos servicios. Finalmente, en el caso de no proporcionar continuidad de servicio, las empresas proveedoras no necesitan considerar la información existente de otros días dentro del horizonte de planificación. Cabe destacar, que mantener al mismo asistente a lo largo de todo el horizonte de planificación, respetando las distintas restricciones complica la resolución del problema. Por ejemplo, en el caso de tener un horizonte de planificación de varios días, la resolución de los distintos días no puede ser independiente, ya que las decisiones y asignaciones realizadas en un día del horizonte afectan al resto de días. Es por ello, que habitualmente dicha característica se incluye como una preferencia y no como una restricción [Duque et al. \(2015\)](#), existiendo algunos trabajos en los que se considera como una restricción a satisfacer [Hewitt et al. \(2016\)](#); [Quintana et al. \(2017\)](#). Por otro lado,

pueden existir situaciones en los que un cliente necesite un número de servicios, tal que un solo asistente no le pueda atender por restricciones horarias, en tales casos se intenta minimizar el número de asistentes distintos que son asignados a cada cliente.

Restricciones geográficas

Es habitual que los proveedores de servicios de asistencia domiciliaria agrupen a los miembros de su equipo en distintos distritos, teniendo en cuenta restricciones geográficas, las habilidades de los asistentes y las necesidades de los clientes. Este particionamiento previo hace que la gestión de los distintos equipos sea más sencilla, especialmente cuando no se cuenta con herramientas de software que den soporte a la toma de decisiones. Una de las aproximaciones más sencillas es la de no considerar distritos [Mankowska et al. \(2014\)](#). Otros autores como [Eveborn et al. \(2006\)](#) consideran que cada distrito puede ser resuelto de modo independiente. En dicho trabajo se presenta un sistema de soporte a la toma de decisiones llamado *Laps-Care*, dicho sistema es probado en el municipio Sueco de Danderyd ayudando a resolver instancias con 28 asistentes de atención domiciliaria y hasta 1560 clientes, de acuerdo a los autores su uso supuso un ahorro en tiempo de desplazamiento de hasta el 20% y un 7% en lo referente al tiempo total trabajado por cada asistente, ya que gracias al sistema se eliminaron las reuniones en las que se planificaba el trabajo a realizar.

Además de particionar los asistentes en diversos distritos, otra característica es la distribución geográfica de los centros y el tipo de servicio que prestan los proveedores de servicio. Los autores [Bräaysy et al. \(2009\)](#) consideran el envío de comida a personas dependientes desde una cocina central, así como la recogida de los materiales de empaquetado en el caso de no ser desechables. En [Liu et al. \(2013\)](#) los autores abordan los problemas logísticos de una farmacéutica que opera en Francia y que debe enviar dispositivos médicos y medicinas a los pacientes, siendo además responsable de la recolección de muestras orgánicas para su análisis clínico. Desde el punto de vista de modelización ambos problemas se pueden considerar un caso especial del problema “Pick up and Delivery Problem” variante del VRP que fue expuesta en la sección anterior 2.4.

2.3.2.2. Restricciones de los clientes

Existen otro conjunto de restricciones que hacen referencia a los clientes, entre este tipo de restricciones se pueden identificar restricciones temporales, restricciones en la asignación y restricciones geográficas.

Restricciones temporales

Quizás una de las restricciones temporales más característica que pueden establecer los clientes, es el horario de visita dentro del cual se les debe proporcionar la asistencia domiciliaria. Dicho horario se conoce como ventana temporal, cuya representación se puede ver en la figura 2.9. Al igual que en el caso del Problema del Viajante con Ventanas de Tiempo y del Problema de Enrutamiento de Vehículos con Ventanas de Tiempo dichas ventanas temporales pueden ser fijas [Akjiratikar et al. \(2007b\)](#); [Bard et al. \(2014\)](#) o flexibles [Hiemann et al. \(2015\)](#); [Mankowska et al. \(2014\)](#); [Nickel et al. \(2012\)](#); [Trautsamwieser](#)

y Hirsch (2011). En el caso de ventanas temporales fijas, el prestador de servicios está obligado a respetar dicha restricción, de hecho, un conjunto de asignaciones entre pacientes y clientes sólo será válido, y por lo tanto considerado como solución al problema, si respeta dicha restricción. En caso de ventanas temporales flexibles, la restricción de visitar al cliente dentro de cierto horario sigue existiendo, aunque en este caso al ser flexible no se trata como una restricción, sino como una preferencia que es penalizada en caso de incumplimiento dentro de la función objetivo. Además de las restricciones de ventana temporal, existen otras investigaciones Di Gaspero y Urli (2014) en las que existen un conjunto de restricciones que limitan la hora a la que se debe comenzar a prestar el servicio, en dicho trabajo se definen un conjunto discreto de horas de comienzo permitidas entre las cuales se debe elegir una de ellas.

Otra restricción temporal relacionada con los clientes, es lo que se conoce como periodicidad. La mayoría de los trabajos asumen una periodicidad regular, es decir, las visitas se deben realizar a la misma hora y el mismo día de la semana, cada semana. Por ejemplo “todos los lunes a las 09:00 AM”, existiendo variaciones en los que se dicha periodicidad puede referirse a varios días, como es el caso de EULEN el cual será presentado en el capítulo 3. Aunque es habitual que el horario en el que se debe realizar las visitas y el día de la semana esté predefinido, existen trabajos en los día de la semana y el horario ha de ser determinado como parte del problema, esta es la problemática expuesta por Duque et al. (2015). En dicho trabajo, cada cliente cuenta con un número de visitas a realizar a lo largo de la semana (entre uno y cinco) además se establece la restricción de no poder visitar a un cliente dos días consecutivos a menos que el número de visitas requeridas sea igual a cinco. En Begur (1997) se resalta la necesidad de algunos proveedores de respetar ciertos patrones a la hora de determinar los horarios de visitas a los clientes, por ejemplo un paciente que debe ser al menos dos veces por semana existiendo al menos dos días entre cada visita. Finalmente otros proveedores cuentan con una serie de patrones de asignación predefinidos Bard et al. (2014), entre los cuales se debe elegir lo cual aporta flexibilidad desde el punto de vista del cliente y del proveedor. Por ejemplo un paciente que requiere dos visitas semanales puede ser visitado Lunes y Martes (patrón 1) o Jueves y Viernes (patrón 2), otros autores como (Duque et al., 2015) además de incluir el día de la semana también incluyen la franja horaria lo cual redundante en un mayor número de patrones entre los que se debe elegir, elevando la dificultad de resolución.

La dependencia temporal entre tareas es otra de las restricciones temporales relacionadas con los clientes. Estas dependencias pueden establecer restricciones entre visitas asignadas a distintos asistentes, es decir entre distintas rutas o bien restricciones entre las visitas realizadas por el mismo asistente. El primer caso se da en aquellas situaciones donde los clientes requieren la participación de dos o más asistentes, como es el caso de levantar a personas con movilidad reducida, lo cual requiere una sincronización entre las rutas de dos o más asistente. El segundo caso, dependencia entre tareas del mismo asistente, se refiere a aquellas situaciones en los servicios prestados a un mismo paciente deben realizarse respetando ciertas precedencias, por ejemplo, después de administrar cierto medicamento a un cliente ha de transcurrir cierto tiempo hasta que se le preste el siguiente servicio. En (Bredström y Rönnqvist, 2008) los autores destacan la importancia y dificultad de incluir restricciones temporales de precedencia y sincronización en esta familia de problemas. En concreto los autores experimentan con instancias de problemas en las que se requiere

la presencia simultánea de varios asistentes, como levantar a personas con movilidad reducida o tareas en las que se requiere que un asistente visite a un cliente después de que otro asistente lo haya visitado, como por ejemplo la administración de fármacos después/antes de la comida. Los autores proporcionan una formulación MIP, la cual es modelado como una generalización del VRPTW (ver 2.2.4) a la que añaden las restricciones citadas anteriormente. Utilizando un enfoque híbrido llegan a solucionar instancias con hasta 16 asistentes y 80 clientes, de los cuales solamente 8 requieren visitas sincronizadas. En [Rasmussen et al. \(2012\)](#) se modela la resolución del HCSP como un problema de particionamiento de conjuntos (Set Partitioning Problem), considerándose hasta un total de cinco tipos de restricciones de precedencia. Para la resolución del problema los autores proponen una solución exacta basada en Branch & Price (B&P) y una formulación basada en el VRPTW.

Restricciones relativas a la asignación

Los clientes pueden tener una serie de preferencia a la hora de elegir a su asistente, algunos ejemplos de dichas preferencias se pueden encontrar en [Wirnitzer et al. \(2016\)](#)

Restricciones geográficas

El conjunto de clientes y su localización geográfica es otra de las características importantes a la hora de resolver el HCSP. La localización de los clientes da lugar a una red o un grafo, la distribución espacial del conjunto de clientes representa una de las características más importantes a considerar, por ejemplo, en el caso de zonas rurales las distancias entre distintos pacientes serán más cortas que en zonas urbanas. De hecho algunos autores [Rest et al. \(2012b\)](#) reportan que el ratio entre tiempo efectivo en el que se visita al paciente y tiempo de desplazamiento es significativamente distinto en zonas rurales que en zonas urbanas.

Dicha localización de los clientes da lugar a quizás otro de los aspectos claves en la resolución del HCSP, que es la estimación de los tiempos de desplazamiento entre clientes. En primer lugar, cabe destacar que la mayor parte de los trabajos consideran la distancia Euclídea como medida de distancia para estimar el tiempo de desplazamiento invertido por un asistente en moverse desde el cliente A al cliente B [Cisse et al. \(2017\)](#), con algunas excepciones en los que se utilizan sistemas de información geográfica [Begur \(1997\)](#). En segundo lugar, otro aspecto a tener en cuenta son los modos de transporte existentes entre distintos clientes, siendo los más habituales desplazamiento en coche, utilizando el transporte público, a pie o una combinación de los anteriores. En el caso de utilizarse varios modos de transporte, se habla de transporte multimodal existiendo escasos trabajos que aborden el HCSP desde dicha perspectiva [Hiermann et al. \(2015\)](#); [Rendl et al. \(2012\)](#). Finalmente, otra dificultad añadida a la hora de realizar estimaciones sobre los tiempos de desplazamiento, es la dependencia de dichos tiempos y la hora en la que se realizan. Por ejemplo, el tiempo de desplazamiento entre dos puntos dentro de una ciudad varía mucho si se realiza en "hora punta" o si se realiza en "hora valle". En [Rest y Hirsch \(2016\)](#) los autores proponen un modelo en el que el tiempo de desplazamiento es dependiente del tiempo, utilizando el transporte público como modo de desplazamiento, el cual y de

acuerdo a los autores es especialmente sensible a los efectos citados con anterioridad.

2.3.2.3. Restricciones de los asistentes

Finalmente y tras haber expuesto las restricciones relativas a los proveedores de servicios de HCSP y a los clientes, en esta sección se exponen las restricciones relacionadas con los asistentes a de atención domiciliaria.

Restricciones temporales

La primera restricción temporal en la que podemos pensar referida a los asistentes de atención domiciliaria, es la cantidad de horas que estos están disponibles, esta disponibilidad dependerá del tipo de contrato, existiendo contratos de tiempo completo y tiempo parcial o de fines de semana. Normalmente los empleados han de cumplir con su horario y solo en determinadas ocasiones se puede incurrir en horas extras. La mayoría de los trabajos asumen que cada asistente tiene un conjunto de horas de trabajo predefinidas y no se contempla la inclusión de horas extras, pudiendo ser a su vez este tipo de restricción una restricción "dura" [Hiermann et al. \(2015\)](#); [Redjem y Marcon \(2016\)](#); [Braekers et al. \(2016\)](#) o "blanda" permitiendo la realización de horas extras [Trautsamwieser y Hirsch \(2011\)](#).

Además de la ventana temporal otro aspecto temporal a considerar por parte de los asistentes es la carga de trabajo que pueden realizar, aquí es importante diferenciar la disponibilidad del empleado tratada anteriormente del número máximo de horas que este puede realizar. Es habitual hacer que ambas coincidan de tal modo que la restricción de horas máximas trabajadas se respete automáticamente, creándose una variable auxiliar que determina la hora de comienzo del turno de cada asistente. Cabe destacar que esta limitación es dependiente de cada país, por ejemplo en Reino Unido el máximo tiempo diario de trabajo es de 7.5 horas [Akjiratikar et al. \(2007b\)](#) mientras que en otros países como Finlandia [Braysy et al. \(2009\)](#) o España [Quintana et al. \(2017\)](#) dicha limitación es de 8 horas. Este tipo de limitaciones también pueden estar presentes en periodos temporales más largos, como semanas [Duque et al. \(2015\)](#), en este caso cada asistente tiene un tipo de contrato que determina el máximo de horas semanales que pueden trabajar, penalizándose las horas extras en la función objetivo.

Finalmente, otra restricción temporal de los asistentes es la relacionada con los descansos que estos deben realizar. Se pueden distinguir dos grandes tipos de restricciones, aquellas que fuerzan a descansar dentro de un intervalo temporal [Trautsamwieser y Hirsch \(2011\)](#), y aquellas que fuerzan a descansar después de que la ruta del asistente haya alcanzado cierta duración [Begur \(1997\)](#). Desde el punto de vista de la modelización, algunos autores han optado por añadir un cliente ficticio que represente dicha tarea de descanso [Kergosien et al. \(2009b\)](#), mientras que otros [Bard et al. \(2012\)](#); [Liu et al. \(2017\)](#) proponen añadir un tiempo de espera entre las visitas a clientes.

Restricciones relativas a la asignación

Una de las restricciones más frecuentes relativas a la asignación de los asistentes hace referencia a la cualificación o habilidades de los mismos. Este tipo de restricción establece

que para que un asistente pueda prestar cierto servicio y por tanto ser asignado a un cliente, debe poseer ciertas habilidades o características [Braysy et al. \(2009\)](#); [Duque et al. \(2015\)](#); [Braekers et al. \(2016\)](#). Este tipo de características pueden hacer referencia a la formación del asistente, como ser médico, enfermera, fisioterapeuta o terapeuta ocupacional, pudiendo hacer referencia a otras habilidades como hablar cierto idioma [Eveborn et al. \(2009\)](#). Este tipo de restricciones se pueden modelar de tres modos distintos. La primera aproximación se puede aplicar siempre y cuando las tareas de los clientes solo requieran una cualificación y cada asistente solo tenga una cualificación, en tal caso, el problema original se puede descomponer y resolver de modo independiente en distintos problemas, uno por cada cualificación [Kergosien et al. \(2009b\)](#). Una segunda aproximación consiste en asignar a cada asistente un conjunto de cualificaciones [Bertels y Fahle \(2006\)](#) siendo esta una generalización de la primera aproximación. Por último, la tercera aproximación consiste en asignar un nivel de cualificación a cada asistente para cada una de las posibles cualificaciones, el cual incluye a las dos aproximaciones previas. En esta tercera aproximación cada asistente cuenta con un nivel para cada una de las cualificaciones posibles, a su vez cada cliente requiere un conjunto de cualificaciones con ciertos niveles mínimos a partir de los cuales puede ser asignado a un asistente [Cire y Hooker \(2012\)](#).

El segundo tipo de restricciones relativas a los asistentes hace referencia al equilibrio o equidad en la asignación de los horarios, dicha equidad está dirigida a garantizar un reparto de horas y festivos equitativo para todos los trabajadores, lo cual es fundamental para mantener motivado a los asistentes y reducir la rotación laboral. Puesto que un equilibrio perfecto puede resultar muy complejo de encontrar, pudiendo no existir, este tipo de restricción es usualmente incluida como una preferencia dentro de la función objetivo.

Restricciones geográficas

Una de las restricciones geográficas relacionadas con los asistentes tiene que con los puntos geográficos de inicio y fin de cada una de las rutas asignadas a los asistentes. Existe una amplia mayoría de estudios en los cuales un único punto de inicio y de fin es considerado, dicho punto geográfico suele representar la sede del proveedor de servicios. Al igual que en el caso del VRP, en el caso del HCSP también existen aproximaciones en las se utilizan múltiples puntos de inicio y fin como es el caso expuesto por [Akjiratikar et al. \(2007b\)](#). De acuerdo a [Trautsamwieser y Hirsch \(2011\)](#) existen tres tipos de asistentes de acuerdo a sus localizaciones de inicio y fin, a saber. Aquellos que comienzan/terminan sus rutas en la sede del proveedor de servicios, pudiendo existir una o varias sedes. En segundo lugar, tenemos a aquellos asistentes comienzan y terminan sus rutas en la casa del propio asistente. Finalmente, y como variante del anterior, aquellos que pese a comenzar y terminar sus rutas en la casa del propio asistente, el desplazamiento entre el primer y el último cliente al lugar de residencia del asistente no es computado como parte de la jornada laboral de este.

2.3.3. Objetivos

En el estado del arte y atendiendo al número de objetivos considerados por la función objetivo, existen dos tipos de aproximaciones. En primer lugar están aquellos trabajos que

han utilizado funciones monobjetivo existiendo un gran número de trabajos que utilizan dicha aproximación [Fikar y Hirsch \(2017\)](#). Existen trabajos en los se debe tomar un enfoque multiobjetivo, variando el número de objetivos a considerar, desde 2 objetivos [Akjiratikar et al. \(2007b\)](#), hasta 13 [Hiermann et al. \(2015\)](#), es importante destacar que la mayoría de trabajos [Bertels y Fahle \(2006\)](#); [Mankowska et al. \(2014\)](#); [Quintana et al. \(2017\)](#) realizan una suma ponderada en la función objetivo, incluso cuando cada componente de dicha suma está en distintas unidades. En [Duque et al. \(2015\)](#) los autores proponen un objetivo lexicográfico para un total de dos objetivos, el primero de ellos trata de satisfacer las preferencias de asistentes y clientes, mientras que el segundo minimiza la distancia total recorrida por los asistentes. Al tratarse de un método lexicográfico, el valor del primero de los objetivos prevalece sobre el segundo, sin importar el valor de este, solo en caso de que dos soluciones obtengan el mismo valor en el primer objetivo, el segundo será tenido en cuenta. A diferencia de las investigaciones anteriores, en [Braekers et al. \(2016\)](#) se propone una enumeración del frente de Pareto para dos objetivos enfrentados como son minimizar el coste total (compuesto por el tiempo de desplazamiento y las horas extras) y minimizar a su vez el incumplimiento de preferencias por parte de clientes y asistente. A pesar de que existen numerosos trabajos y cada uno de ellos establece un conjunto de objetivos específico y adaptado a las necesidades de cada compañía, existen al menos cuatro objetivos bien diferenciados que se presentan de modo recurrente en la literatura especializada, los cuales se detallan a continuación.

2.3.3.1. Minimizar el coste de la ruta

Minimizar el coste total de las rutas asignadas a los asistentes es un criterio bastante estándar. Dicho coste puede medirse como una medida de distancia o bien como una medida de tiempo, a su vez dicho coste se puede referir a coste incurrido entre las distintas visitas a clientes o al coste total, estando todas estas medidas fuertemente relacionadas. Aproximadamente el 90 % de las investigaciones utilizan la suma de distancia recorridas por los asistentes como un criterio a minimizar [Nickel et al. \(2012\)](#); [Di Gaspero y Urli \(2014\)](#); [Braekers et al. \(2016\)](#)

2.3.3.2. Minimizar el número de clientes sin asignar

En algunas situaciones los proveedores de servicios no cuentan con personal suficiente para abastecer la creciente demanda de sus servicios, en dichas situaciones es habitual que se recurra a empresas colaboradoras que son subcontratadas para prestar el servicio a los clientes. En dicho contexto, el proveedor de servicios puede plantear como función objetivo, minimizar el número de servicios sin prestar o minimizar el número de clientes sin atender [Akjiratikar et al. \(2007b\)](#); [Nickel et al. \(2012\)](#); [Hiermann et al. \(2015\)](#); [Liu et al. \(2017\)](#). De modo análogo, otro criterio también utilizado en este tipo de contextos es maximizar el número de servicios prestados o el número de clientes atendidos [Bennett y Erera \(2011\)](#). Este tipo de objetivos tienen un impacto importante en la estructura del problema, la restricción de tener que visitar a todos los pacientes se relaja y es sustituida por los costes anteriormente descritos dentro de la función objetivo.

2.3.3.3. Minimizar el número asistentes requeridos

Al igual que sucediera con el CVRP (ver 2.2.3) en el cual uno de los objetivos a minimizar es el número de vehículos necesarios, en el caso del HCSP también existen investigaciones donde uno de los objetivos a minimizar es el número de asistentes necesarios [Ikegami y Uno \(2007\)](#); [Hewitt et al. \(2016\)](#). Siendo también uno de los objetivos de la compañía EULEN cuya problemática será presentada en el capítulo siguiente [Quintana et al. \(2017\)](#).

2.3.3.4. Maximizar la satisfacción

A la hora maximizar la satisfacción existen dos grupos de objetivos bien diferenciados, aquellos que maximizan la satisfacción de los clientes y aquellos que maximizan la satisfacción de los asistentes. En el primer caso, maximizar la satisfacción de los clientes, se logra al respetar las distintas preferencias que estos tienen acerca de distintos aspectos como son, la ventana temporal en la que prefieren ser visitados, el conjunto de días que mejor se adapta a sus preferencias o aquellos asistentes que prefieren que les atiendan. Puesto que resulta complicado poder satisfacer todas estas preferencias para todos los clientes, es habitual que estas sean relajadas, incluyéndose dentro de la función objetivo. Por ejemplo en [Mankowska et al. \(2014\)](#) cada vez que se asigna a un cliente fuera de su ventana temporal preferida dicha asignación es penalizada en la función objetivo. En [Wirnitzer et al. \(2016\)](#) los autores consideran hasta un total de 6 preferencias de los clientes relacionadas con la continuidad del servicio, entre las que destacan : Número total de asistentes por cliente, número total de asistentes por grupo de clientes, número total de cambios de asistentes por paciente (nótese que no es igual al número de asistentes, ya que mide el número de cambios).

El segundo tipo de restricciones relativas a los asistentes, hacen referencia al equilibrio o equidad en la asignación de los horarios, dicha equidad está dirigida a garantizar un reparto de horas y festivos equitativo para todos los trabajadores, lo cual es fundamental para mantener motivado a los asistentes y reducir la rotación laboral. Para garantizar dicho equilibrio se pueden tomar dos aproximaciones distintas, la primera está enfocada a equilibrar el número de horas que realiza cada asistente [Begur \(1997\)](#) , mientras que la segunda trata de equilibrar el número de servicios que realiza cada agente sin importar la duración total de los mismos [Begur \(1997\)](#) . Puesto que a medida que aumenta el número de asistentes resulta más complicado garantizar una equidad entre todos ellos, es habitual que dicha equidad se deba garantizar solo entre aquellos empleados que tengan las mismas características, por ejemplo, comparando sólo aquellos empleados que tienen el mismo tipo de contrato o cuyas horas contratadas se encuentran dentro de cierto rango.

En [Duque et al. \(2015\)](#) los autores proponen un enfoque jerárquico. En primer lugar las preferencias de clientes y asistentes son maximizadas, sin tener en cuenta el segundo objetivo. Posteriormente y una vez fijado un nivel de preferencias como restricción, es minimizada la distancia total recorrida por cada uno de los asistentes.

2.3.4. Métodos de resolución

Los distintos métodos de resolución aplicados a la resolución del HCSP se clasifican de distintos modos dependiendo del criterio seleccionado. Atendiendo al horizonte de

planificación, se pueden distinguir métodos que se centran en la resolución de problemas de un día (periodo único) o aquellos que están enfocados a la resolución de varios días, incluso meses (periodo múltiple). Atendiendo a las técnicas, se pueden distinguir métodos exactos y métodos aproximados, a lo largo de esta sección se abordarán las características principales de cada uno.

2.3.4.1. Periodo único

De acuerdo con [Fikar y Hirsch \(2017\)](#) la mayor parte de los trabajos existentes se centran en la resolución de esta subfamilia de problemas. En primer lugar se presentan las técnicas exactas para posteriormente presentar las técnicas no exactas.

Técnicas exactas

Uno de los primeros trabajos en este campo fue publicado por ([Fernandez et al., 1974](#)), en dicho trabajo se investiga una comunidad de enfermeras rurales en el Reino Unido, para la cual los autores proponen distintos modos de asignar enfermeras a las distintas zonas rurales, realizando estimaciones sobre los tiempos de desplazamiento y el número total de servicios que una enfermera podría realizar, no utilizándose ningún tipo de técnica de optimización. Uno de los primeros trabajos que abordó esta subfamilia de problemas desde el punto de vista de la optimización, fue propuesto por [Begur \(1997\)](#) en el cual se integra la resolución de este tipo de problemas dentro de un sistema de soporte a la toma de decisiones, para una compañía de Estados Unidos, cabe destacar que dicha aproximación no tenía en cuenta restricciones de ventana temporal y su objetivo era minimizar el tiempo total de desplazamiento, así como la carga de trabajo entre las distintas enfermeras. Posteriormente [Cheng y Rich \(1998\)](#) propone un método basado en MIP así como una solución heurística planteada en varias fases, enfocada a minimizar las horas extras de las enfermeras, consiguiendo resolver de forma heurística instancias de hasta 300 enfermeras y 900 pacientes. También enmarcado dentro un sistema de ayuda a la toma de decisiones se encuentra el trabajo realizado por [Bertels y Fahle \(2006\)](#), en el cual y desde una aproximación en la cual se combinan técnicas exactas y no exactas (híbrida), se presenta el sistema *PARPAP* basada en la combinación de programación lineal (LP), programación por restricciones (CP) combinadas con metaheurísticas. En dicho trabajo se consideran instancias para un solo día, con número de entre 20 y 50 enfermeras y un número total de entre 111 y 326 tareas. Los autores usan la combinación de LP y CP para generar un conjunto de soluciones iniciales, las cuales son mejoradas de modo iterativo utilizando las basadas en búsqueda tabú y recocido simulado.

A diferencia de los trabajos anteriores los autores ([Bredström y Rönnqvist, 2008](#)) destacan la importancia y dificultad de incluir restricciones temporales de precedencia y sincronización en esta familia de problemas. En concreto los autores experimentan con instancias de problemas en las que se requiere la presencia simultánea de varios asistentes, como levantar a personas con movilidad reducida o tareas en las que se requiere que un asistente visite a un cliente después de que otro asistente lo haya visitado, como por ejemplo la administración de fármacos antes o después de la comida. Los autores proporcionan una formulación MIP, la cual es una generalización del VRPTW (ver 2.2.4) a la que

se añaden las restricciones citadas anteriormente. Utilizando un enfoque híbrido llegan a solucionar instancias con hasta 16 asistentes y 80 clientes, de los cuales solamente 8 requieren visitas sincronizadas. En [Kergosien et al. \(2009a\)](#) esta familia de problemas HCSP es modelada como una extensión del TSP, en concreto como una variación del problema de los m-vendedores de comercio con ventanas de tiempo, problema que fue descrito en la sección 2.1.4.4. Para ello los autores proponen una serie de restricciones adicionales y proporcionan una formulación exacta que puede ser resuelta mediante la resolución de un problema MIP, llegando a resolver instancias con 40 clientes. Los autores [A. y P \(2011\)](#) proponen el uso de la técnica VNS ([Mladenović y Hansen, 1997](#)) para la planificación de un conjunto de enfermeras, bajo la premisa de que cada enfermera usará un vehículo determinado.

En [Rasmussen et al. \(2012\)](#) proponen el uso de una aproximación basada en generación de columnas dinámica incluida dentro de un esquema de Branch & Price (B& P) para resolver un HCSP con dependencias temporales, las cuales afectan al inicio de las tareas que se deben realizar. Dichos autores modelan la resolución del HCSP como un problema de particionamiento de conjuntos (Set Partitioning Problem), considerándose hasta un total de cinco tipos de restricciones de precedencia. Dicha aproximación es probada con distintas instancias reales (7-15 cuidadores, 60-150 tareas), instancias generadas de modo aleatorio (7-15 cuidadores, 150 tareas) e instancias reportadas en el estado del arte [Bredström y Rönnqvist \(2007\)](#). Los resultados reportados por dichos autores obtienen mejores resultados que los propuestos por [Bredström y Rönnqvist \(2007\)](#) obteniendo mejores soluciones en menor tiempo de cómputo.

En [Hiermann et al. \(2015\)](#) los autores presentan la resolución de un HCSP multimodal basado en dos etapas. La primera de ellas está enfocada a obtener una primera solución inicial que cumpla con las restricciones del problema, para tal fin utilizan la técnica de programación por restricciones (CP), dicha solución es mejorada con una de serie metaheurísticas entre las que destaca la búsqueda Tabú.

En [Liu et al. \(2017\)](#) se presenta un algoritmo basado en Branch & Price (B& P) para resolver un HCSP de una empresa China en la que destaca la necesidad de incluir los descansos para comer en cada de las rutas generadas para cada asistentes. Además, los autores proponen numerosas novedades a en distintas partes del algoritmo, como en la generación de columnas o un esquema de ramificación jerárquico. Los autores utilizan el solver de CPLEX y dos tipos de instancias, la primera de ellas es una adaptación de las instancias propuestas por ([Solomon, 1987](#)) mientras que la segunda procede de un conjunto de datos reales facilitados por una compañía China.

Técnicas no exactas

En [Eveborn et al. \(2006\)](#) se presenta un sistema de soporte a la toma de decisiones llamado *Laps-Care*, dicho sistema es probado en el municipio Sueco de Danderyd ayudando a resolver instancias con 20 asistentes de atención domiciliaria, 123 clientes y hasta 1560 servicios. El problema se resuelve utilizando un algoritmo iterativo que va asignando a los distintos a los asistentes, teniendo la posibilidad de dividir rutas en ciertos momentos. Los criterios considerados por dicho algoritmo son varios, a saber, el tiempo de desplazamiento, las preferencias de los clientes y de los asistentes. De acuerdo a los autores su uso supuso

un ahorro en tiempo de desplazamiento de hasta el 20 % y un 7 % en lo referente al tiempo total trabajado por cada asistente, ya que gracias al sistema se eliminaron las reuniones en las que se planificaba el trabajo a realizar.

En [Akjratikar et al. \(2007a\)](#) se propone el uso de la metaheurística PSO (Particle Swarm Optimization) para planificar a un conjunto de asistentes de atención domiciliaria en el Reino Unido, en total se planifican más de 100 tareas diarias para un total de 50 clientes y que deben ser asignadas a 12 asistentes de atención domiciliaria, proporcionando ahorros de hasta el 31 % en lo referente a la distancia total recorrida. En [Braysy et al. \(2009\)](#) se presentan un conjunto de casos reales para distintas ciudades de Finlandia, en concreto los autores analizan los márgenes de mejora existentes para tres problemas distintos, como son la planificación de rutas de un conjunto de enfermeras que prestan servicios de atención domiciliaria, el transporte de personas mayores o el reparto de comida a personas dependientes, usándose para la resolución de todos ellos el software propietario ([Solutions, 2016](#)). Todos los problemas anteriores son tratados como ciertas variantes del VRP ver 2.2.4. Para el problema de planificación de rutas se planifican un total de 38 enfermeras y algo más 100 clientes diarios, lográndose ahorros de hasta el 52 % en la distancia total para ciertos escenarios. Finalmente para el problema de reparto de comida a personas dependientes, los autores solucionan instancias con una media de hasta 260 clientes diarios, reportándose mejoras de más del 30 % en la distancia total recorrida y del 30 % en el número de vehículos requeridos.

Los autores ([Mankowska et al., 2014](#)) proponen la resolución del HCSP utilizando una búsqueda basada en vecindarios variable, VNS (variable neighborhood search). En dicho problema los pacientes necesitan ser atendidos dentro de una ventana temporal blanda y requieren una cualificación por parte de los asistentes. El algoritmo comienza con una heurística de inicialización avariciosa, donde los pacientes de mayor prioridad son asignados en primer lugar, el resto de pacientes son asignados a aquellos asistentes que pueden llegar en primer lugar. Dicha solución inicial es mejorada utilizando un total de hasta 8 vecindarios distintos, los cuales se pueden agrupar en 4 tipo de movimientos, a saber, retrasar la visita a un paciente sin cambiar el asistente, intercambiar dos pacientes que pertenezcan a la misma ruta, cambiar dos pacientes que pertenezcan a distintas rutas y finalmente retrasar la visita de un paciente cambiando de asistente. La técnica propuesta es objeto de experimentación utilizando instancias generadas de forma aleatoria, las cuales tiene entre 10-300 clientes y entre 3-40 asistentes de atención domiciliaria. Las soluciones obtenidas son comparadas con una solución exacta utilizando una formulación básica en CPLEX, la cual es capaz de conseguir soluciones óptimas para instancias muy pequeñas, 10 pacientes y 3 asistentes, requiriendo un total de 10 horas para la resolución de cada una. Para instancias medianas CPLEX es capaz de encontrar una solución aunque se queda bastante alejado del límite inferior, siendo incapaz de encontrar solución alguna para las instancias de mayor tamaño. La técnica propuesta por los autores es capaz de encontrar soluciones en menos de 6 segundos para instancias de hasta 100 clientes.

En [Liu et al. \(2013\)](#) los autores se enfrentan a un problema HCSP en el cual no solo se deben visitar a los pacientes, sino que además se les debe suministrar ciertos fármacos y equipamiento médico, debiéndose recoger al mismo tiempo muestras biológicas de alguno de los pacientes. Dicho problema es similar al VRPDB (Vehicle routing problem with deliveries and backhauls) expuesto en la sección 2.2.2. Los autores proponen dos tipos

de soluciones, la primera de ellas está basada en una búsqueda tabú y la segunda en un algoritmo genético. La búsqueda tabú utiliza como solución inicial, la mejor solución encontrada por el algoritmo genético. El vecindario de dicha solución inicial es explorado a través de dos movimientos, el primero de ellos mueve un vértice (cliente) de una ruta a otra, mientras que el segundo intercambia clientes de distintas rutas. Posteriormente y después de dichos movimientos se aplican una serie de operadores de búsqueda local. Para el caso del algoritmo genético (AG), los autores utilizan una codificación que consiste en una permutación de los clientes a visitar. La inicialización de los individuos del AG es realizada utilizando tres heurísticas de ahorros y una de vecino más próximo. Ambos algoritmos, búsqueda tabú y el algoritmo genético son probados utilizando dos tipos distintos de instancias, en concreto las instancias propuestas por [Solomon \(1987\)](#) con hasta 120 tareas y 20 asistentes, y las propuestas por [Gehring y Homberger \(2002\)](#) las cuales alcanzan un total de 200 tareas y 50 asistentes. Los resultados de ambas técnicas son comparados con los resultados obtenidos por CPLEX, obteniendo ambas técnicas una mejora media del 17% para el primer conjunto de instancias y de un 6.35% para el segundo conjunto de instancias.

2.3.4.2. Periodo múltiple

Dentro de esta segunda subfamilia se incluyen todos aquellos problemas HCSP en los que los clientes requieren ser visitados a lo largo de varios días, semanas y en algunos casos incluso meses. La dificultad en la resolución este tipo de problemas de estriba en la necesidad de considerar una serie de factores como son restricciones en materia laboral, descansos y tiempos máximos de trabajo, continuidad en el servicio [Nickel et al. \(2012\)](#). Al igual que en la sección anterior, en primer lugar se presentan las técnicas exactas para posteriormente presentar las técnicas no exactas.

Técnicas exactas

En [Andrea y Patrick \(2014\)](#) los autores proponen un algoritmo de B&C&P para resolver un HCSP en el que los clientes deben ser visitados a lo largo de la semana, dentro de una ventana temporal requiriendo la visita de asistentes de atención domiciliaria con ciertas habilidades y características. El número total de horas que puede trabajar cada asistente está limitado, tanto a nivel diario como a nivel semanal, además se deben realizar ciertos descansos a lo largo de la jornada laboral. Con dicha aproximación resuelven una instancia real que involucra a un total de 9 asistentes, 45 clientes y un total de 203 tareas de una hora duración distribuidas a lo largo de una semana.

Utilizando también una formulación entera [Cappanera y Scutellá \(2015\)](#) resuelve un HCSP semanal, donde la utilización de patrones cobra un especial interés a la hora de generar las planificaciones de cada asistente. Dichos patrones que son generados inicialmente, tienen un doble objetivo, en primer lugar aseguran la continuidad del servicio y por otro garantizan la compatibilidad entre asistentes y tareas. Dichos autores también consideran la obtención de soluciones balanceadas en lo referente al número de horas trabajadas por cada asistente, solucionando instancias con 4-11 asistentes y 40-162 tareas.

Un ejemplo de planificación mensual ha sido propuesto recientemente por [Wirnitzer et al.](#)

(2016), en dicho estudio los autores abordan la resolución de un total de 45 instancias reales en las cuales se tienen en cuenta restricciones de continuidad en el servicio, disponibilidad, compatibilidad y capacidad. Dichas instancias son resueltas utilizando un conjunto de planes maestros semanales, siendo estos las variables de decisión del problema MIP a resolver. Con dicha aproximación los autores resuelven instancias de hasta 37 trabajadores con 143 pacientes y con un total de 1114 tareas semanales. Los resultados muestran una mejora clara en la continuidad del servicio cuando dicha métrica es comparada con la solución manual.

Técnicas no exactas

Puesto que la dificultad de las instancias del HCSP aumenta al aumentar el horizonte de planificación, resulta esperable que existan un mayor número de estudios que traten de abordar este tipo de problemas con técnicas no exactas. En [Bard et al. \(2014\)](#) los autores proponen la utilización de una técnica GRASP para la resolución de un HCSP semanal en el cual se distingue entre asistentes de atención domiciliaria y fisioterapeutas. Los autores proponen dos aproximaciones una paralela y otra secuencial, las cuales son probadas en instancias reales con 4-12 asistentes y con un total de 93-238 tareas, obteniendo la aproximación secuencial mejores resultados.

En [Duque et al. \(2015\)](#) los autores proponen un algoritmo de dos etapas en el que se tratan de optimizar dos objetivos de modo jerárquico, como son maximizar el nivel de servicio, una vez fijado dicho nivel de servicio como restricciones adicionales al problema original se pasa a minimizar de la distancia total recorrida. En primer lugar las preferencias de clientes y asistentes son maximizadas utilizando un conjunto de patrones; posteriormente y una vez fijado dicho nivel de satisfacción de preferencias como restricción, es minimizada la distancia total recorrida por cada uno de los asistentes. Los autores prueban su algoritmo con un total de 30 instancias generadas de modo aleatorio con un total de 26-109 clientes y un total de 3-25 asistentes.

Utilizando como base teórica una extensión del PVRP (ver sección 2.2.2) en el que se consideran ventanas temporales, los autores [Liu et al. \(2014\)](#) tres tipos de demandas, a saber, el envío de medicamentos o equipos médicos, el envío de medicamentos especiales que por su naturaleza deben transportarse bajo ciertas condiciones y la recolección de muestras de sangre. Cada paciente está definido por un número de visitas que deben ser satisfechas a lo largo del horizonte de planificación, teniendo además un conjunto de patrones de días en los que debe ser visitado. En dicho estudio los autores proponen la utilización de búsqueda tabú junto con técnicas de búsqueda local, obteniéndose la solución inicial a través de la utilización de una heurística de inserción, la cual solo garantiza la satisfacción de las restricciones de precedencia.

En [Nickel et al. \(2012\)](#) los autores también proponen una solución basada en dos fases, combinando programación por restricciones y una búsqueda por vecindad a gran escala para la resolución de un HCSP con un horizonte de planificación semanal. Los clientes quedan caracterizados por su ventana temporal, la cualificación que requieren para ser atendidos, la frecuencia de las visitas y una combinación de turnos de visitas. En la primera fase se utiliza la programación por restricciones a fin de obtener una solución factible, la cual es mejorada en la segunda fase utilizando una búsqueda por vecindad a gran escala.

Esta aproximación es capaz de resolver instancias reales con una semana de duración, con 11-12 enfermeras y un total de 287-361 tareas.

Finalmente, en la tabla 2.5 se muestra a modo de resumen algunas de las distintas técnicas utilizadas en el estado del arte, prestando especial atención al tamaño de las instancias utilizadas por cada autor. En dicha tabla se puede apreciar como en muy pocas ocasiones el tamaño de las instancias a resolver supera las 1.000 tareas. Tal y como se mostrará en el capítulo 3 el tamaño de las instancias abordadas en esta investigación supera dicha cifra de modo muy considerable.

Tabla 2.5: Tabla resumen con las distintas características de las instancias utilizadas por otros investigadores para el HCSP.

Autores(s)	Tipo de instancia	Horizonte de planificación	Max. Núm. asistentes	Max. Núm. clientes	Max. Núm. tareas
Akjiratikar et al. (2007a)	R	Corto	12	100	?
Bertels y Fahle (2006)	A	Corto	50	600	?
Braekers et al. (2016)	R	Corto	89	300	?
Bredström y Rönnqvist (2007)	A	Corto	16	90	?
Eveborn et al. (2006)	R	Corto	20	123	?
Hiermann et al. (2015)	R	Corto	518	717	?
Liu et al. (2013)	L	Corto	50	200	?
Liu et al. (2017)	R/L	Corto	12	100	?
Mankowska et al. (2014)	R	Corto	40	?	300
Rasmussen et al. (2012)	R/A/Le	Corto	15/15/16	150/150/80	?
Rest y Hirsch (2016)	R	Corto	46	202	?
Andrea y Patrick (2014)	R/A	Corto	75/20	512/100	?
A. y P (2011)	R/A	Corto	75/20	512/100	?
Bard et al. (2013)	A	Largo	20	?	200
Bard et al. (2014)	R/A	Largo	12/16	278/?	?/160
Begur (1997)	R	Largo	7	?	200
Bowers et al. (2015)	A	Largo	5	?	138
Braysy et al. (2009)	R	Largo	9	?	324
Cappanera y Scutellá (2015)	R/L	Largo	11/11	?	162/36
Hewitt et al. (2016)	R	Largo	75	400	?
Liu et al. (2014)	R/L	Largo	2/50	?/200	58/?
Nickel et al. (2012)	R	Largo	12	361	?
Andrea y Patrick (2014)	R	Largo	9	203	?
Wirmitzer et al. (2016)	R	Largo	37	143	1114

A: Instancias generadas de modo aleatorio. R: Instancias reales. L: Instancias reutilizadas de la literatura

2.4. Optimización Basada en Colonias de Hormigas

En esta sección se presentan los principios biológicos que explican algunos comportamientos interesantes presentes en ciertas especies de hormigas, como son: total autonomía de los individuos, control totalmente distribuido, tolerancia a fallos, estrategias colectivas y cooperativas, y auto organización. Posteriormente, se explicarán cuáles fueron los trabajos que permitieron desarrollar un marco teórico sobre el comportamiento de las colonias

hormigas reales, para finalizar exponiendo la evolución histórica de la Optimización Basada en Colonias de Hormigas y un breve resumen de sus aplicaciones más comunes.

Los insectos con aproximadamente 1 millón especies descritas¹⁶, y con estimaciones de hasta 30 millones¹⁷ de especies no descritas (Erwin, 1982), representan potencialmente más del 90 de las formas de vida del planeta. Su aparición se remonta a hace aproximadamente 400 millones de años, nuestra especie lo hizo hace aproximadamente 200.000 años (Utah, 2005). Algunos insectos viven en sociedades extremadamente organizadas, presentando comportamientos eusociales, término procedente de Batra (1966); Wilson (1971). También conocidos como insectos sociales, el comportamiento de termitas, abejas, avispas y hormigas, ha fascinado siempre a los entomólogos, destacando especialmente el comportamiento de las hormigas (Holldobler y Wilson, 1990). Estas últimas, han evolucionado desde los últimos 130 millones de años, con un total de 12.623 especies en la actualidad (Ohio, 2011), convirtiéndoles probablemente, en el grupo de mayor éxito dentro de los insectos sociales.

Debido a sus características, en los últimos 20 años también han captado la atención de ingenieros, especialmente de aquellos relacionados con las Telecomunicaciones y las Ciencias de la Computación. Entre algunas de estas características destacan la total autonomía de los individuos, control totalmente distribuido, tolerancia a fallos, comunicación entre individuos y a través del entorno, emergencia de comportamientos complejos respecto a los comportamientos simples de cada hormiga, estrategias colectivas y cooperativas, y auto organización. Estas características han convertido a las colonias de hormigas en fuentes de inspiración para desarrollar nuevos algoritmos y nuevos sistemas multi-agente basados en colonias de hormigas. Posiblemente la línea de investigación más fructífera y popular entre los investigadores en estos últimos 20 años, han sido los algoritmos basados en colonias para problemas de optimización combinatoria, los cuales se conocen bajo el nombre de Optimización Basada en Colonias de Hormigas (Dorigo y Stützle, 2004; Dorigo et al., 1999, 1996; Dorigo, 1992).

2.4.1. Principios biológicos

Las hormigas constituyen la familia Formicidae, perteneciente al orden Hymenoptera, destacando por su socialización en la mayoría de las especies que la componen. A diario millones de hormigas colaboran resolviendo diversos problemas, de un modo descentralizado, robusto, fiable y autoorganizado utilizando para ello reglas sencillas y métodos de comunicación basados en señales químicas conocidas como feromonas. Sin ánimo de ser exhaustivos, el funcionamiento básico de las colonias de hormigas reales es el siguiente (Wilson, 2002). Existen una o varias reinas que son fértiles y cuya vida, oscila entre 15 y 20 años. La función de la reina es poner huevos, generando nuevos individuos que pueden ser hembras estériles, hembras fértiles o machos. Las hormigas hembras que son estériles se dedican a realizar diversas tareas, que son necesarias para la supervivencia del hormiguero, su vida es de aproximadamente un año. La misión de los machos y de las hembras fértiles es permitir la formación de nuevos hormigueros, los machos una vez fecundan a la nueva reina mueren, la nueva reina fecundada se entierra en la tierra y forma un nuevo hormiguero.

¹⁶Más que todos los otros grupos de animales juntos

¹⁷Estudios más recientes (Reaka-Kudla M.L., 1997; Novotny et al., 2002) rebajan la cifra de especies no descritas 6 y 10 millones.

Ninguna hormiga le dice a otra hormiga lo que tiene que hacer, sin embargo, la colonia es capaz de variar automáticamente y en función del entorno, el número de hormigas que realizan una u otra tarea. Las tareas básicas de un hormiguero son, patrullar alrededor del hormiguero, recolectar comida (encontrando el camino más corto), mantener la estructura interna del hormiguero, y mantener limpio los alrededores del hormiguero.

Las hormigas son capaces de comunicarse por medio de señales químicas, las feromonas. Esta comunicación se puede realizar de dos modos, a través del medio físico (estigmergia) o través del contacto entre las antenas de unas hormigas con otras, este segundo tipo de comunicación permite a las hormigas detectar, las tareas que realizan las demás, y en función de esta decidir que tarea realizar. Las feromonas son utilizadas por infinidad de especies para su comunicación cumpliendo un importante papel, ya que representa un sistema de comunicación química entre individuos de la misma especie sobre el estado fisiológico, reproductivo o social, dichas señales son recibidas e interpretadas por el conjunto de la especie. Estas pueden ser liberadas de forma activa por la especie, para indicar ciertas conductas específicas en momentos determinados (emisión de feromona ocasionado por el daño producido en una colonia de hormigas por un ataque externo), o por el contrario de una forma pasiva cuando se dan ciertas circunstancias y el individuo emisor no participa de manera activa (marca de feromona realizada por la hormiga por el simple hecho de desplazarse).

Una de las tareas importantes para la supervivencia de una colonia es buscar alimento, las colonias de hormigas lo hacen siguiendo la ruta más corta, entre el alimento y el hormiguero. La hormiga, en su tránsito es capaz de ir realizando el depósito de feromona en el medio, efectuando, una marca del recorrido elegido. Este hecho es el que origina el aprovechamiento de la información por el resto de la comunidad. La detección de la feromona depositada por el conjunto de individuos de un hormiguero es la fuente de información que utiliza un individuo para su movimiento. Esto hace que dependiendo de la intensidad de feromona depositada la hormiga decida el camino a recorrer, a mayor cantidad más probabilidad tiene de elegir ese rastro. Este comportamiento queda demostrado en los experimentos que se detallarán a continuación, constituye uno de los ejemplos más utilizados a la hora de hablar de comportamientos emergentes.

Entre las décadas de 1980 y 1990, un grupo de investigadores de la Universidad “Université Libre de Bruxelles de Bélgica” llevaron a cabo numerosos experimentos que ponían de manifiesto una relación entre la feromona y el comportamiento de las hormigas. En (Goss et al., 1989) se llevaron a cabo experimentos con una especie concreta “*Linepithema humile*”, también conocida como hormiga Argentina, con el objetivo de comprobar cómo se comportaban las hormigas, ante la selección de dos rutas que tenían distintas longitudes. Para el experimento se disponen de un hormiguero, y una fuente de comida, desde el hormiguero se puede llegar a la fuente de comida de varios modos, tal y como muestra en la Figura 2.19.

En Goss et al. (1989) se demuestra que la probabilidad de elegir el camino más corto entre el hormiguero y la fuente de comida, para un conjunto de hormigas, está relacionada directamente con el ratio existente entre el camino más largo y el más corto, a mayor ratio mayor será dicha probabilidad. Este experimento se puede explicar del siguiente modo: en un principio las hormigas tienen las mismas opciones de repartirse entre ambos caminos, ya que la cantidad depositada de feromona es nula, dejando la posibilidad de exploración al

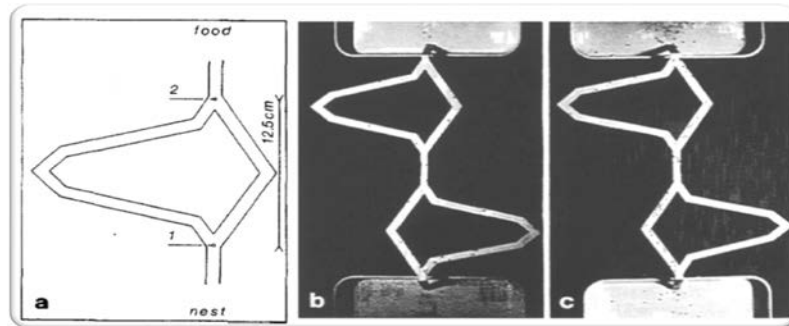


Figura 2.19: Experimento de puente binario con distintas longitudes de arcos (Goss et al., 1989).

conjunto de hormigas para la búsqueda de cualquier posible camino. Una vez detectada la comida, las hormigas se desplazan de nuevo al hormiguero. Las hormigas con un recorrido más corto lo realizarán en un tiempo menor, dando la posibilidad de reforzar con feromona el camino más rápidamente que otras hormigas que han elegido una peor ruta. Esto origina una retroalimentación o refuerzo positivo, el cual es el causante de encontrar la ruta más corta entre la fuente del alimento y el hormiguero. Transcurrido un tiempo, las hormigas tienden a confluír por la ruta más corta, existiendo una mínima posibilidad de que alguna hormiga de modo aislado elija otra ruta.

Posteriormente a estos experimentos en (Deneubourg et al., 1990), se llevó a cabo un experimento en el que ambos caminos tenían la misma longitud, en ausencia de una fuente de comida, sorprendentemente el comportamiento fue el mismo que el reportado por Goss et al. (1989). En la Figura 2.21 se puede apreciar como el porcentaje de hormigas que eligen uno u otro camino varía con el tiempo, y como de modo emergente, y siguiendo una serie de reglas sencillas, la mayor parte de las hormigas terminan transitando por uno de los dos arcos, incluso cuando estos tienen exactamente la misma longitud, tal y como puede verse en la Figura 2.20.

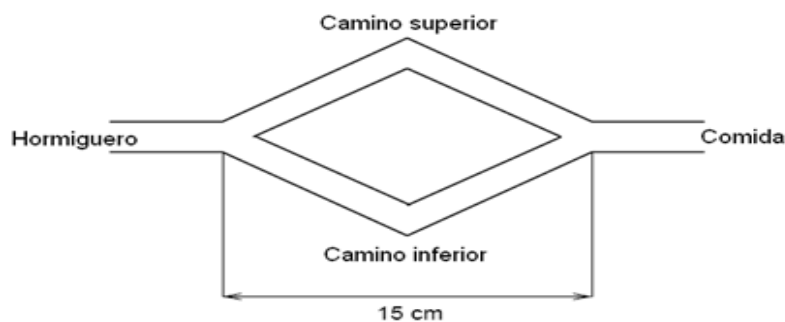


Figura 2.20: Detalle puente binario con las mismas longitudes de arcos

A partir de estos experimentos, se pudo elaborar un modelo matemático que reflejaba cuál era la probabilidad de que una hormiga eligiera un camino u otro. La probabilidad de

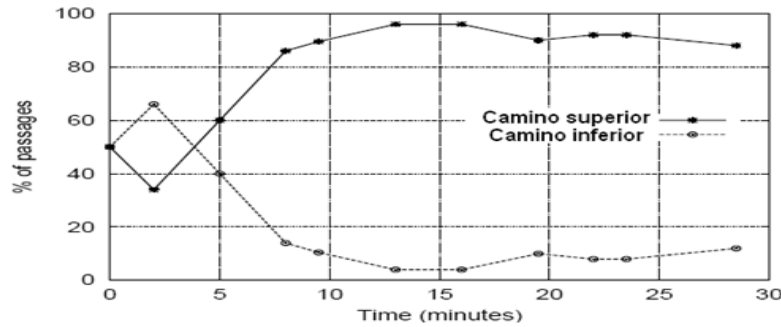


Figura 2.21: Evolución del porcentaje de hormigas que eligen un camino u otro.

que una hormiga elija la ruta 1, viene definida por

$$P_1(m+1) = \frac{(m_1 + k)^h}{(m_1 + k)^h + (m_2 + k)^h} \quad (2.24)$$

de modo análogo la probabilidad de elegir la segunda ruta está definida por

$$P_2(m+1) = 1 - P_1(m+1) \quad (2.25)$$

donde k y h son parámetros del modelo que sirven de ajuste a los datos del experimento. Se debe indicar que la feromona en un entorno natural, sufre una evaporación con el paso del tiempo, así los recorridos menos valiosos presentan una pérdida progresiva de feromona que les hace cada vez menos interesantes para la colonia. Estos trabajos permitieron el desarrollo de los primeros sistemas computacionales basados en colonias de hormigas tal y como se mostrará en los siguientes apartados.

2.4.2. Evolución histórica

En este apartado se presentará cual ha sido la evolución de la meta heurística ACO, desde sus inicios hasta la actualidad. De modo general los trabajos sobre ACO se pueden dividir en dos grandes grupos, los que se centran en proponer nuevas modificaciones sobre el algoritmo ACO y los que se centran en analizar la convergencia y complejidad de dicho algoritmo. Durante la década de los 90 y hasta el año 2000 la mayor parte de los trabajos se centran en proponer nuevos algoritmos, a partir del año 2000 comienzan a aparecer las primeras publicaciones centradas en analizar la convergencia, complejidad de ACO así como otros aspectos como son la utilización de múltiples colonias o la resolución de problemas multiobjetivo.

2.4.3. Ant System (AS)

La primera propuesta de optimización basada en colonias de hormigas fue denominado por sus autor Ant System (AS), en concreto en la tesis doctoral de [Dorigo \(1992\)](#), en la cual se proponen hasta tres alternativas denominadas ant-cycle, ant-density y ant-quantity¹⁸. Para probar la eficacia del método propuesto el autor abordó la resolución del TSP,

¹⁸Anteriormente fueron publicados en [Dorigo et al. \(1996\)](#)

cada una de las variantes propuestas se caracterizaba por los distintos modos en los que se producía el depósito de feromona.

- Densidad (ant-density): El depósito de feromona se realizaba durante el transcurso del recorrido, además la cantidad de feromona depositada era siempre constante.
- Cantidad (ant-quantity): El depósito de feromona se realizaba durante el transcurso del recorrido, además la cantidad de feromona depositada estaba determinada por la deseabilidad del tramo.
- Ciclo (ant-cycle): El depósito de feromona se realizaba una vez encontrada una solución.

Esta última propuesta (ant-cycle) es la que mejor resultados presenta, y la que se conoce como AS. Caracterizada por el hecho de que la actualización de feromona se realiza una vez que todas las hormigas han completado sus soluciones, el pseudocódigo de AS se puede apreciar en la Figura 2.22.

```

Procedure Ant System
Inicializar
Para t=1 hasta el número de ciclos haga
Inicio
  Para k = 1 hasta m haga
    Inicio
      repetir
        seleccionar la siguiente ciudad a visitar con probabilidad  $P_{ij}^k$ 
      hasta hormiga  $k$  complete su tour
      calcular largo  $L_k$  del tour  $d$  la hormiga  $k$ 
    fin para
  Salvar la mejor solución encontrada hasta el momento
  Modifique los niveles  $\tau_{ij}$  de feromona
Fin

```

Figura 2.22: Pseudocódigo de Ant System

En un primer momento, todos los arcos presentan la misma probabilidad. En el algoritmo, existen dos procesos fundamentales en su ejecución, movimiento de la hormiga y actualización de feromona. Una hormiga k , situada en el nodo i elige ir al nodo j con una probabilidad que viene dada por

$$P_{ij}^k = \left\{ \begin{array}{ll} \frac{[\tau_{ij}(t)]^\alpha \times [\eta_{ij}(t)]^\beta}{\sum_{u \in N_i^k} [\tau_{iu}(t)]^\alpha \times [\eta_{iu}(t)]^\beta} & \text{si } j \in N_i^k \\ 0 & \text{si } j \notin N_i^k \end{array} \right\} \quad (2.26)$$

donde N_i^k son el conjunto de nodos alcanzables por la hormiga k desde el nodo i , α y β son parámetros que ponderan la importancia de la heurística utilizada y los valores de feromona detectados, τ_{ij} representa el valor de feromona y $\eta_{ij}(t)$ representa el valor heurístico de transitar entre los nodos i y j . Cada hormiga k almacena el recorrido realizado para no

repetir visitas al mismo nodo. La modificación de feromona, presenta dos subprocesos: la evaporación y la deposición. En la evaporación, los rastros de feromona se reducen un valor constante. Los diferentes arcos sufren una disminución de su valor de feromona que viene dado por la expresión.

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) \quad (2.27)$$

Donde ρ es la tasa de evaporación, con valores iguales a 1 se realizaría una búsqueda totalmente aleatoria. En el proceso de deposición cada hormiga actualiza su tour en función de la calidad de la solución encontrada, dicha deposición viene dada por

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (2.28)$$

Donde $\Delta\tau_{ij}(t)$ indicia la calidad de la solución encontrada, que en este problema viene dado por el inverso de la distancia. Este primer modelo propuesto, no era competitivo con otros algoritmos existentes para el TSP en la época, lo que motivó algunas modificaciones tal y como se mostrará en el siguiente apartado. Los principales problemas con los que contaba eran convergencia lenta, bajo rendimiento para problemas de más 75 ciudades y estancamiento.

2.4.4. Elitismo en Ant System (AS)

Una de las primeras mejoras introducidas sobre Ant System fue el elitismo. Se basa en el modelo explicado anteriormente, pero añadiendo una característica que conseguía mejores resultados. La extensión consistía en que la deposición de feromona solo la realizan aquellas hormigas con mejores resultados hasta el momento. Este aporte era función de la solución considerada y de e , que representa el número de hormigas elitistas tenidas en cuenta,

$$\Delta\tau_{ij}^{gb}(t) = \left\{ \begin{array}{ll} \frac{e}{L^{gb}(t)} & \text{si } i, j \in T^{gb} \\ 0 & \text{en otro caso} \end{array} \right\} \quad (2.29)$$

2.4.5. Ant Colony System (ACS)

Continuando con las mejoras anteriores, y utilizando técnicas de aprendizaje automático por refuerzo [Gambardella y Dorigo \(1995\)](#); [Dorigo y Gambardella \(1996\)](#) elaboran el sistema conocido como Ant-Q. Este nuevo enfoque junto con las mejoras anteriores concluyen con el algoritmo Ant Colony System [Dorigo y Gambardella \(1997\)](#), donde se realiza la modificación del algoritmo original Ant System en cuatro facetas principalmente.

1. La transición de estados: Regla de transición distinta, nuevo parámetro q_0 que permite controlar la exploración-explotación.

$$s = \begin{cases} \arg \max_{u \in N_i^k} \left\{ \left[\tau_{uj}(t) \right]^\alpha \times \left[\eta_{uj}(t) \right]^\beta \right\} & \text{si } q \leq q_0 \\ P_{ij}^k & \text{si } q > q_0 \end{cases} \quad (2.30)$$

Cada vez que una hormiga k , debe transitar a otro nodo, elige una variable aleatoria q distribuida uniformemente en $[0,1]$. Si $q > q_0$ entonces la hormiga usa la función definida en (ecuación 2.26), en caso contrario, elige aquel camino que maximice $\left[\tau_{uj}(t) \right]^\alpha \times \left[\eta_{uj}(t) \right]^\beta$.

2. La actualización global de feromona: Se realiza una actualización global sobre la mejor ruta encontrada en cada iteración. Además se incluye otro nuevo parámetro α conocido como parámetro de decaimiento.

$$\tau_{ij}(t+1) = (1 - \alpha) \cdot \tau_{ij}(t) + \alpha \cdot \Delta\tau_{ij}(t) \quad (2.31)$$

$$\Delta\tau_{ij}^{gb}(t) = \begin{cases} \frac{1}{L^{gb}(t)} & \text{si } i, j \in T^{gb} \\ 0 & \text{en otro caso} \end{cases} \quad (2.32)$$

Para llevar a cabo la actualización, el sistema sólo considera una hormiga concreta, la que generó la mejor solución global, T^{gb} . Cabe destacar que en algunos trabajos iniciales se propuso un tipo de actualización basada en la mejor hormiga de la iteración.

3. La actualización local de feromona: Las hormigas al ir transitando por los arcos, disminuyen la feromona, favoreciendo por lo tanto la exploración. El objetivo es hacer menos atractivos los arcos visitados, buscando evitar la convergencia prematura en la misma ruta. Lo hacen utilizando la siguiente formula.

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \tau_0 \quad (2.33)$$

donde ρ es la tasa de actualización local y τ_0 el nivel de feromona mínimo. Como puede verse, la regla de actualización local incluye tanto la evaporación de feromona como la deposición de la misma. Puesto que la cantidad de feromona depositada es muy pequeña la aplicación de esta regla hace que los rastros de feromona entre las conexiones recorridas por muchas hormigas disminuyan favoreciendo que las demás hormigas recorran distintos arcos (exploración).

4. Uso de una lista candidata: Lista donde se tienen ordenadas las ciudades por preferencia de visita.

Tras las modificaciones mencionadas anteriormente, los resultados que obtuvo este algoritmo para el problema del TSP fueron mejores que las técnicas existentes en el estado del arte, tal y como se muestra en la Figura 2.23.

Problem name	ACS	GA	EP	SA	Optimum
Eil50 (50-city problem)	425 (427.96) [1,830]	428 (N/A) [25,000]	426 (427.86) [100,000]	443 (N/A) [68,512]	425 (N/A)
Eil75 (75-city problem)	535 (542.37) [3,480]	545 (N/A) [80,000]	542 (549.18) [325,000]	580 (N/A) [173,250]	535 (N/A)
KroA100 (100-city problem)	21,282 (21,285.44) [4,820]	21,761 (N/A) [103,000]	N/A (N/A) [N/A]	N/A (N/A) [N/A]	21,282 (N/A)

Figura 2.23: Comparativa de ACS respecto a otras técnicas para el problema del TSP. En negrita se muestra la mejor solución entera encontrada por el algoritmo, entre paréntesis la mejor solución real, y entre corchetes el número de tours necesarios para encontrarlos (Dorigo y Gambardella, 1997).

2.4.6. Max-Min Ant System (MMAS)

Este algoritmo, modifica la estrategia a la hora de realizar el depósito de feromonas y solamente permite la actualización a las hormigas que alcanzan buenos resultados (Stützle y Hoos, 1997). Esta estrategia puede dar origen a una convergencia prematura en un resultado no óptimo, pero se consideraba oportuno que los mejores resultados ejercieran una influencia mayor en el algoritmo utilizado. Para evitar el aumento desmesurado del valor de feromona en rutas no óptimas y por lo tanto la convergencia en estas rutas, se introdujo el concepto de máximo y mínimo, que se referían a los valores que podían alcanzar las rutas en cuanto a cantidad de feromona depositada. El nivel mínimo pone límite al nivel de exploración, para que este siempre tenga posibilidades de ejecución, el nivel de feromona cumplirá por tanto con

$$\tau_t^{min} \leq \tau_{t+1} \leq \tau_t^{max} \quad (2.34)$$

2.4.7. Sistema mejor-peor hormiga (SMPH)

Es un algoritmo de sistema de hormigas que incorpora conceptos de computación evolutiva. El SMPH (Cordon et al., 2000) utiliza la misma regla de transición de estados que el Sistema de Hormigas, así como la misma regla de evaporación de feromona que se aplica a todas las transiciones. Además, tal como hace el MMAS, el SMPH siempre considera la explotación sistemática de optimizadores locales para mejorar las soluciones de las hormigas. Formando el núcleo del SMPH podemos encontrar las siguientes tres acciones durante la ejecución:

1. La regla mejor-peor de actualización de rastros de feromona, basada en la regla de actualización del vector de probabilidades, refuerza las aristas que se encuentran en la mejor solución global. Además, penaliza cada conexión de la peor solución generada

hasta el momento, Speor-actual, que no se encuentre en la mejor global realizando una evaporación de feromona adicional de esos rastros.

2. Se realiza una mutación de los rastros de feromona para introducir diversidad en el proceso de búsqueda. Para llevarla a cabo, el rastro de feromona asociado a cada una de las transiciones desde cada nodo (por ejemplo, cada fila de la matriz de rastros de feromona) se muta con una probabilidad P_m , utilizando cualquier operador de mutación con codificación real.
3. La propuesta original del SMPH aplicaba un operador que alteraba los rastros de feromona de cada transición mutada añadiendo o restando la misma cantidad en cada iteración.

2.4.8. Marco teórico ACO

A partir del año 2000 el marco teórico sobre el que se sustenta ACO se comienza a desarrollar. En una fase inicial, se intenta responder a la pregunta de ¿cuánto tiempo es necesario para que un algoritmo ACO converja a la solución óptima? La primera demostración de convergencia la podemos encontrar en [Gutjahr \(2000\)](#), para el algoritmo Graph Based Ant System (GBAS) con probabilidad $1 - \epsilon$. Para los algoritmos ACS y MMAS se pueden encontrar demostraciones similares en [\(Stützle y Dorigo, 2006\)](#). Sin embargo, dichos trabajos no se centran en estudiar la velocidad con la que convergen los algoritmos, por lo tanto las preguntas importantes para desarrollar el marco teórico sobre ACO, está relacionadas con la velocidad de convergencia y su complejidad desde un punto de vista computacional. El primer resultado respecto a la velocidad de convergencia fue publicado en [Gutjahr \(2006, 2007\)](#). Respecto a la complejidad computacional, los autores [\(Zhou et al., 2009\)](#) obtienen que el orden de complejidad para una instancia concreta del TSP es $\mathcal{O}(n^6 + \frac{1}{\rho} \cdot n \cdot \ln n)$ sin utilizar heurística y $\mathcal{O}(n^5 + \frac{1}{\rho} \cdot n \cdot \ln n)$ utilizándola.

2.4.9. Aplicaciones ACO

Desde su primera aparición hace más de 20 años, ACO se ha aplicado a multitud de problemas. Debido a sus orígenes, la Optimización Basada en Colonias de Hormigas se ha aplicado tradicionalmente a problemas de ruteo, por ejemplo el TSP. En este tipo de problemas el objetivo es encontrar un camino entre dos puntos, minimizando alguna métrica y cumpliendo las restricciones del problema, existiendo entornos en los que las métricas varíen con el tiempo [\(Caro et al., 2005\)](#). Una revisión sobre las aplicaciones presentes y futuras de ACO se puede encontrar en [Dorigo y Stützle \(2009\)](#); [Monmarché et al. \(2010\)](#). En la Figura 2.24 podemos encontrar un cuadro resumen con las principales aplicaciones de ACO.

<i>Problem type</i>	<i>Problem name</i>	<i>Authors</i>	<i>Year</i>
Routing	Traveling salesman	Dorigo et al.	1991, 1996
		Dorigo & Gambardella	1997
		Stützle & Hoos	1997, 2000
	TSP with time windows	López Ibáñez et al.	2009
	Sequential ordering	Gambardella & Dorigo	2000
	Vehicle routing	Gambardella et al.	1999
		Reimann et al.	2004
		Favoretto et al.	2007
		Fuellerer et al.	2009
	Multicasting	Hernández & Blum	2009
Assignment	Quadratic assignment	Maniezzo	1999
		Stützle & Hoos	2000
	Frequency assignment	Maniezzo & Carbonaro	2000
	Course timetabling	Socha et al.	2002,2003
	Graph coloring	Costa & Hertz	1997
Scheduling	Project scheduling	Merkle et al.	2002
	Weighted tardiness	den Besten et al.	2000
		Merkle & Middendorf	2000
		Stützle	1997
	Flow shop	Rajendran, Ziegler	2004
	Open shop	Blum	2005
	Car sequencing	Solnon	2008
Subset	Set covering	Lessing et al.	2004
	l -cardinality trees	Blum & Blesa	2005
	Multiple knapsack	Leguizamón & Michalewicz	1999
	Maximum clique	Solnon, Fenet	2006
Machine learning	Classification rules	Parpinelli et al.	2002
		Martens et al.	2006
		Otero et al.	2008
	Bayesian networks	Campos, Fernández-Luna	2002
	Neural networks	Socha, Blum	2007
Bioinformatics	Protein folding	Shmygelska & Hoos	2005
	Docking	Korb et al.	2006
	DNA Sequencing	Blum et al.	2008
	Haplotype Inference	Benedettini et al.	2008

Figura 2.24: Aplicaciones de ACO a distintos problemas

Capítulo 3

Caso de estudio: HCSP en la Comunidad de Madrid

*Aquellos que pueden imaginar cualquier cosa,
pueden crear lo imposible*

Alan Turing

Resumen:

En este capítulo se realiza una exposición detallada del problema que será objeto de experimentación en capítulos posteriores. En primer lugar, se realiza una introducción al problema abordado por una multinacional que presta servicios de atención domiciliaria en la Comunidad de Madrid. Dicha compañía atiende anualmente a alrededor de 1.5 millones de clientes, los cuales requieren más de 2,1 millones de horas de trabajo al año; no incluyendo dicho tiempo las ineficiencias inherentes a la prestación de servicios, como son, los tiempos de espera y de desplazamiento de los asistentes de atención domiciliaria. Una vez expuestas las particularidades de la compañía, se detallan las distintas instancias que serán objeto de experimentación en el capítulo 5, describiéndose de modo detallado el proceso de obtención de las mismas, así como sus características más relevantes. Finalmente, se presenta una formulación matemática del problema a resolver, formulación que será usada en capítulos posteriores a fin de explicar algunos conceptos clave de las técnicas propuestas por esta investigación.

3.1. Introducción

El problema de planificación de asistentes de atención domiciliaria, es un problema de creciente interés y cuya resolución exige de técnicas especializadas, las cuales fueron mostradas en la sección 2.3 del capítulo 2. Las previsiones demográficas, junto con las recomendaciones de distintas organizaciones como la OMS o la UE, hacen que la asistencia domiciliaria de nuestros mayores en su domicilio, se haya convertido en una actividad cuya demanda va en aumento y cuyas previsiones apuntan a un aumento todavía más significativo durante los próximos años.

En dicho contexto se desarrolla la actividad del Grupo EULEN, multinacional presente en más de 8 países y con más de 85.000 empleados. Dentro de los distintos servicios que presta el grupo, la asistencia domiciliaria está englobada dentro del grupo de servicios sociosanitarios, que incluyen entre otros servicios de teleasistencia, gestión de centros de atención diurna, formación. Dentro del término asistencia domiciliaria se engloban los siguientes servicios [EULEN Group \(2016\)](#):

- Atención personal.
- Atención doméstica.
- Acompañamientos dentro y fuera del domicilio.
- Actuaciones de carácter preventivo (fisioterapia, terapia ocupacional, psicología, enfermería, etc.).
- Educación en hábitos saludables.
- Alimentación y nutrición.
- Lavado y repaso de ropa.

El problema de planificación de asistentes de atención domiciliaría al que se enfrenta la compañía, tiene una enunciación aparentemente sencilla, consiste en asignar a cada persona que demanda uno de los servicios anteriores, aquel profesional que mejor se adapte a sus necesidades, respetando las numerosas restricciones que se detallarán en la sección 3.4, como son respetar el máximo de horas semanales y diarias que puede trabajar cada asistente, asegurar que el servicio se realiza a la hora acordada, etc. Entre las características que diferencian a este problema de planificación de asistentes de atención domiciliaria de los expuestos en la sección 2.3 destacan las siguientes características:

- Su alta dimensionalidad. Quizás una de las características más relevantes del problema al que se enfrenta la compañía es la alta dimensionalidad del problema a resolver. En la Comunidad de Madrid se deben planificar y realizar casi 30.000 tareas semanalmente, tareas que se realizan entre las 07:00 y las 22:00 de lunes a domingo. Esta cifra está hasta dos órdenes de magnitud por encima de las instancias abordadas en el estado del arte mostradas en la tabla 2.5.
- La continuidad en la prestación de servicios. Tal y como se mostró en la sección 2.3 la continuidad en el servicio, es una necesidad que tienen los prestadores de servicios

por la cual una vez asignado un asistente a una tarea de un cliente, el resto de tareas de la semana deberán ser asignadas al mismo asistente siempre que sea posible. Por ejemplo, algunos clientes requieren servicios tanto por la mañana como por la tarde, no siendo posible en algunos casos cumplir con dicha restricción.

Las dos características anteriores hacen de modo conjunto que el problema no se pueda dividir en instancias más pequeñas para facilitar su resolución, estrategia que ha sido usada por otros investigadores en el pasado [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#). Dichas aproximaciones se basan en resolver el problema semanal de modo iterativo, resolviendo un problema distinto para cada día de la semana. Sin embargo, esta aproximación no es posible dado que existe la restricción de continuidad en la prestación de servicios y las soluciones encontradas para un día pueden dar como resultado una solución infactible en otros días de la semana. Otra característica a destacar es que una vez que un asistente realiza su primera visita asignada, el resto de los desplazamientos para visitar a los demás clientes que le han sido asignados, son realizados a pie. Esto implica una serie de restricciones que obligan a conocer o al menos a tener una estimación de la distancia entre los distintos clientes. Finalmente, los clientes han de ser visitados dentro de un horario fijo, que se debe respetar y que no puede ser modificado. En la actualidad la empresa no cuenta con ningún tipo de herramienta automatizada que realice dichas asignaciones, sino que estas se realizan de modo manual por un equipo humano que se apoya en una herramienta informática cuya principal tarea es la gestión y almacenamiento de las asignaciones creadas manualmente. Cabe destacar que dicho equipo está altamente especializado y que cuenta con numerosos años de experiencia realizando dicha labor. Después de algunos cambios en la regulación laboral y dado el crecimiento en el número de peticiones de este tipo de servicios, la compañía ha comenzado a experimentar algunos problemas con la escalabilidad del sistema y la calidad de las soluciones obtenidas de un modo manual. Es por ello que ha comenzado a explorar nuevas alternativas para realizar las asignaciones de un modo automático con el objetivo de asegurar la calidad de los servicios prestados y la eficiencia en la asignación de los mismos. Dada la importancia del problema a resolver, la compañía considera muy importante que se sigan las siguientes directrices.

- Todo cliente deberá ser visitado dentro del horario acordado de modo regular y de acuerdo a una evaluación de sus necesidades. Debiendo ser visitados por el mismo equipo, de tal modo que se fomente la confianza entre los asistentes.
- Las restricciones laborales de los asistentes se deben tener en cuenta. En concreto los asistentes deben trabajar dentro de su turno de trabajo, siendo su límite de trabajo semanal de 40 horas.
- El coste del servicio, subvencionado en parte por el ayuntamiento, debe ser minimizado de tal modo que se pueda atender a la mayor parte de clientes de un modo sostenible.

3.2. Descripción del problema

El problema al que se enfrenta la compañía se puede representar utilizando dos conceptos clave, que serán utilizados en los capítulos siguientes, los cuales reciben el nombre de **tarea**

y **servicio**. A lo largo de la semana cada cliente puede requerir una serie de visitas a su domicilio, a lo largo del texto no referiremos a ellas como **tarea**. La compañía con el objetivo de garantizar la continuidad en el servicio, agrupa las tareas de cada cliente bajo una unidad de asignación indivisible a la que nos referiremos como **servicio**. El problema de la compañía consiste en agrupar dichos servicios de tal forma que se minimicen los objetivos expuestos en la sección 3.4 así como las restricciones.

Servicio	Tarea	Día	Hora comienzo	Duración	Ventana temporal	Localización
2801	1	lunes	10:30	60'	0'	(40.42 -3.70)
	2	Martes	11:30	60'	0'	(40.42, -3.70)
2802	1	viernes	11:30	90'	0'	(40.422 -3.71)
2803	1	lunes	10:30	60'	0'	(40.45, -3.75)
	2	Martes	9:30	60'	0'	(40.45, -3.75)
	3	viernes	8:30	60'	0'	(40.45, -3.75)

Tabla 3.1: Ejemplo de problema con tres servicios y seis tareas.

Un ejemplo simple de problema, con tres servicios y seis tareas se muestra en la tabla 3.1. Cada servicio está definido por un identificador y por una colección de tareas. A su vez, cada tarea está definida por un identificador, la localización del domicilio del cliente, definida por una latitud y una longitud, el día de la semana en la que se debe realizar, la hora de comienzo, la duración y una ventana temporal que describe cierto margen en el cual los asistentes pueden comenzar a realizar la tareas.

A diferencia de otros problemas existentes en la literatura, donde una solución consiste en una asignación explícita entre clientes y asistentes, en el caso objeto de estudio, la solución consistirá en un clustering/agrupamientos de todos los servicios, de tal forma que todas las restricciones sean respetadas y todos los clientes sean atendidos. En este punto es importante destacar que la compañía está explorando nuevas alternativas para la automatización de las asignaciones con independencia del personal con el que cuenta actualmente. Utilizando la tabla 3.1 como ejemplo de problema, en la figura 3.1 se muestra un ejemplo de solución. Dicha solución está formada por dos clústers, el primero (clúster 1) de ellos contiene dos servicios, en concreto los servicios 2803 y 2802, mientras que el segundo (clúster 2) está formado solo por el servicio 2801. De un modo más formal la solución está formada por dos clústers $S = \{C_1, C_2\}$ y cada clúster está formada por uno o más servicios, $C_1 = \{S_{2803}, S_{2802}\}$ y $C_2 = \{S_{2801}\}$. En este punto, es importante resaltar que la factibilidad de una solución se comprueba a lo largo de toda la semana, en el ejemplo de la solución mostrada en la figura 3.1, los servicios 2801 y 2803 no son compatibles ya que sus tareas se solapan completamente el lunes a las 10:30 horas. Puesto que los servicios son la unidad mínima de asignación cualquier intento por disgregar las tareas de cada servicio a fin de reducir la dimensionalidad del problema resulta infructuoso.

La notación utilizada para evaluar una solución a una instancia concreta es mostrada a continuación.

- **Número total de asistentes** (Clust_{obj}) requeridos para visitar a todos los clientes

	Hora	Lunes	Martes	Viernes
Cluster 1	08:30			Servicio 2803 Tarea 3
	09:00			
	09:30		Servicio 2803 Tarea 2	Desplazamiento
	10:00			Espera
	10:30	Servicio 2803 Tarea 1		
	11:00			
	11:30			Servicio 2802 Tarea 1
	12:00			
	12:30			

	Hora	Lunes	Martes
Cluster 2	08:30		
	09:00		
	09:30		
	10:00		
	10:30	Servicio 2801 Tarea 1	
	11:00		
	11:30		Servicio 2801 Tarea 2
	12:00		
	12:30		

Figura 3.1: Ejemplo de solución al problema mostrado en la tabla 3.1, la cual está formada por dos clústers, cada uno de ellos compuesto por dos y un servicio respectivamente.

a lo largo del horizonte de planificación.

- **Tiempo total (Tot_{obj})** requerido para visitar a todos los clientes a lo largo del horizonte de planificación. Este tiempo incluye el tiempo de desplazamiento, el tiempo de espera y el tiempo productivo, los cuales se detallan a continuación.
- **Tiempo de desplazamiento (T)** como la suma del tiempo empleado por cada asistente al desplazarse entre las residencias de cada cliente a lo largo de su jornada laboral.
- **Tiempo de espera (W)** que representa la suma del tiempo empleado por cada asistente antes de comenzar a realizar las distintas tareas, es decir tiempo en el que no realiza ningún trabajo efectivo, no se desplaza ni realiza ningún de tareas.
- **Tiempo productivo (P)** calculado como la suma de tiempos en la que los asistentes están realizando una tarea de algún cliente. Nótese que para cualquier solución válida dicho valor permanecerá constante.

3.3. Obtención de las instancias

Como se expuso en la introducción, la compañía no cuenta con ningún tipo de herramienta específica para la realización de las asignaciones de modo automático, pero si cuenta con herramientas de software que dan soporte a la operativa diaria y en las que se almacenan las asignaciones entre clientes y asistentes. En este apartado se expone cómo a partir de las fuentes de datos manejadas por la compañía se llegan a construir las instancias reales que serán objeto de experimentación en el capítulo 5, proceso que está compuesto de tres fases, a saber, geocodificación, cálculo de distancias entre clientes y particionamiento en turnos de trabajo. Como fuente de datos inicial para la generación de

Campo	Descripción	Ejemplo
AUX-ID	Identificador del auxiliar	0001
USR-ID	Identificador del usuario	0001
PROV	Provincia	Madrid
DIST	Distrito	Centro
VIA	Tipo de vía	Plaza
CALLE	Nombre de la vía	Goya
NUM	Número de la vía	15
C.P	Código postal	28523
SER	Servicios requeridos	AP
HOR-L	Horario lunes	09:00-11:00
HOR-M	Horario Martes	11:10-12:00
HOR-X	Horario Miércoles	–
HOR-J	Horario Jueves	–
HOR-V	Horario viernes	09:00-11:00
HOR-S	Horario sábado	–
HOR-D	Horario domingo	–

Tabla 3.2: Estructura del fichero inicial proporcionado por la compañía en el que se detallan los servicios que se presta a cada usuario, la localización de dicho de usuario, el auxiliar que realiza dichos servicios y el horario de realización de los mismos.

las distintas instancias se cuenta con una exportación de la base de datos de la compañía la cual presenta la estructura descrita en la tabla 3.2.

Tal y como puede observarse en la tabla 3.2, la estructura del fichero proporcionado, no solo aporta información sobre los servicios que se deben prestar, la dirección y el horario de los mismos, sino que además proporciona la solución actual que utiliza la compañía, ya que para cada cliente (AUX-ID) se detalla el identificador del auxiliar que le visita (USR-ID).

3.3.1. Geocodificación

El primer paso para la obtención de las instancias es obtener las coordenadas geográficas (latitud-longitud) a partir de las direcciones de cada usuario, este proceso recibe el nombre de geocodificación o geocoding. La ubicación de los usuarios a visitar viene dada por una descripción textual de distintos campos, como son el tipo de vía, el nombre y número de la vía y el código postal, los cuales forman una dirección. Para extraer las coordenadas geográficas a partir de la dirección de cada usuario se optó por la utilización de los servicios web de Google Maps [Google Maps APIs \(2016b\)](#). Dicha API proporciona, entre otros, los servicios de geocodificación necesarios para llevar a cabo dicha conversión. En la figura 3.2 se muestra un ejemplo del resultado obtenido para una dirección concreta, como puede verse en dicha figura en el campo “location” se muestran las coordenadas geográficas para la dirección “41 Calle de Casalarreina” siendo el formato de la respuesta JSON.

A pesar de que el proceso de geocodificación puede parecer sencillo, la obtención de resultados para todas las direcciones ha presentado algunas dificultades que conviene mencionar. En primer lugar, existen direcciones que son erróneas y que provocan un

```
{
  "results" : [
    {
      "address_components" : [
        {
          "long_name" : "41",
          "short_name" : "41",
          "types" : [ "street_number" ]
        },
        {
          "long_name" : "28032",
          "short_name" : "28032",
          "types" : [ "postal_code" ]
        }
      ],
      "formatted_address" : "Calle De Casalarreina, 41,
        28032 Madrid, España",
      "geometry" : {
        "location" : {
          "lat" : 40.4069284,
          "lng" : -3.6159015
        }
      },
      "place_id" : "ChIJAxtaPHolQgOR2bq3vvyHpbE",
      "types" : [ "street_address" ]
    }
  ],
  "status" : "OK"
}
```

Figura 3.2: Ejemplo de la respuesta obtenida al consultar el servicio web de Google de Geocodificación con la dirección “41 Calle de Casalarreina”, en formato JSON.

funcionamiento anómalo del servicio web de Google Maps, estos errores se pueden deber a distintas causas como pueden ser códigos postales que no se correspondan con la dirección, números de vía inexistentes, nombres de vía que no existen o abreviaturas que no existen. El proceso de filtrado y limpieza de los datos ha requerido de un esfuerzo considerable, ya que para algunos se ha debido realizar de modo manual, como por ejemplo detectar abreviaturas erróneas. Para otros casos, como el de los códigos postales se ha podido automatizar. En este último caso, el de los códigos postales erróneos, se produce por distintos motivos, hay calles que están ubicadas en el límite de varios códigos postales tal y como se muestra en la imagen 3.3. En dicha imagen se aprecia como una misma calle puede pertenecer a distintos códigos postales, por ejemplo, el número 88 de la calle Zurbano pertenece al código postal 28010, mientras que el número 90 pertenece al 28003, lo cual puede provocar que los usuarios no sepan exactamente su código postal. Con el fin de solventar y automatizar la resolución de este problema, se creó un grafo con los distintos

códigos postales de Madrid capital, de tal forma que los códigos postales adyacentes se encuentren unidos por un arco, de este modo si al obtener las coordenadas una dirección, el servicio web devuelve algún error se vuelve a intentar con uno de los códigos postales adyacentes.

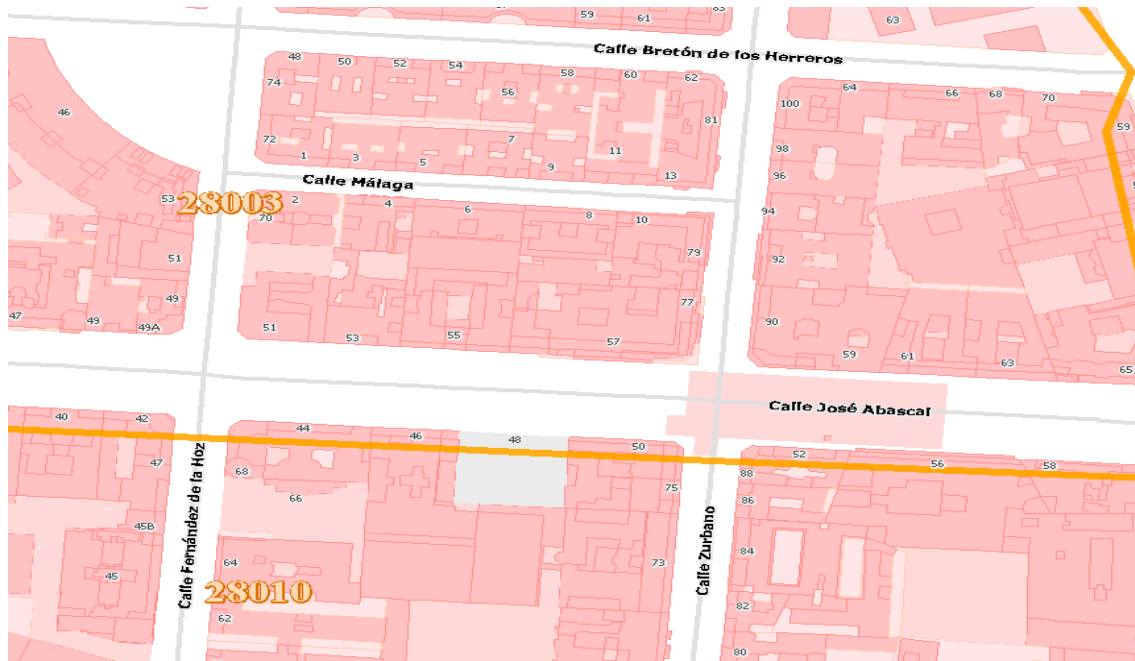


Figura 3.3: Captura de pantalla proporcionada por el Instituto de Estadística de la Comunidad de Madrid.

Una vez realizado el proceso de limpieza de datos y geocodificación se obtienen un total de 13344 **servicios**. La estructura de cada servicio queda definida en la tabla 3.3. Como se puede apreciar en dicha tabla, cada servicio está compuesto por una serie de tareas que se deben realizar en un día concreto de la semana y en un horario concreto. El número total de tareas a realizar a lo largo de la semana es de 29034, el resultado de obtener las coordenadas geográficas de dichos servicios se puede ver la figura 3.4, mientras que en la figura 3.5 se muestra el conjunto total de tareas, agrupadas de acuerdo a su proximidad.

3.3.2. Distancias

Una vez obtenidas las coordenadas geográficas de cada cliente a visitar y las tareas a realizar en cada día de la semana, es necesario poder calcular la distancia existente entre cada cliente. Como se expuso en la introducción 3.1 el desplazamiento entre los distintos servicios es realizado a pie, por lo que es necesario conocer la distancia entre cada servicio a fin de poder respetar las restricciones de distancia máxima recorrido y distancia máxima entre servicios que se expondrán en la sección 3.4. Además de los servicios de geocodificación, los servicios web de Google Maps [Google Maps APIs \(2016a\)](#) ofrecen la posibilidad de calcular las distancias entre un conjunto de coordenadas geográficas, así como las rutas para desplazarse entre dicho conjunto de coordenadas, en la figura 3.6 se muestra un ejemplo de

Campo	Descripción	Ejemplo
USR-ID	Identificador del usuario	0001
LAT	Latitud	40.4069284
LNG	Longitud	-3.6159015
HOR-L	Horario lunes	09:00-11:00
HOR-M	Horario Martes	11:10-12:00
HOR-X	Horario Miércoles	-
HOR-J	Horario Jueves	-
HOR-V	Horario viernes	09:00-11:00
HOR-S	Horario sábado	-
HOR-D	Horario domingo	-

Tabla 3.3: Estructura del fichero obtenido una vez realizado el proceso de geocodificación y limpieza del fichero proporcionado por la compañía. Cada registro de dicho fichero recibe el nombre de **servicio** y cada de las visitas que se realizan en un día concreto y a una hora determinada recibe el nombre de **tarea**.

respuesta obtenido al consultar dos coordenadas geográficas. Este servicio permite obtener rutas entre dos puntos utilizando distintos modos de desplazamiento, como son a pie, en coche, en bicicleta o utilizando el transporte público, en la figura 3.7 se muestra la ruta entre tres puntos yendo a pie.

Para calcular la matriz de distancias entre los 13.344 servicios, se deberían realizar aproximadamente un total de $13344^2 = 178.062.336$ llamadas al servicio web de Google Maps, el cual cuenta con un límite diario de llamadas máximas de 100.000 por licencia y con un coste de 0.50€ cada 1.000 llamadas [Google Maps APIs \(2018\)](#), el coste total ascendería a los 89.031 €. Suponiendo distancias simétricas, se deberían realizar $\frac{13,343^2 - 13,342}{2} = 89.024.496$ llamadas cuyo coste ascendería a 44.512 €¹. Otro factor a tener en cuenta además del coste, es el tiempo en completar dichos cálculos, con unos límites de 100.000 llamadas diarias, para el caso asimétrico se necesitarían aproximadamente 1.790 días y en el caso asimétrico 896 días, dicho proceso se podría ejecutar de modo paralelo aumentando el coste del mismo.

Dado el coste y el tiempo de obtención de la matriz de distancias utilizando los servicios web de Google Maps, se decidió calcular la distancia entre dos coordenadas geográficas utilizando la distancia Manhattan, cuya formulación viene dada por la ecuación (3.1) donde lat_1, lat_2 representan las latitudes en grados, lng_1, lng_2 las longitudes y C es una constante que mide la longitud en metros de un grado.

$$F(lat_1, lng_1, lat_2, lng_2) = C * (|lng_1 - lng_2| * \cos(\frac{lat_1 - lat_2}{2}) + |lat_1 - lat_2|) \quad (3.1)$$

Tal y como puede apreciarse en la figura 3.8 la distancia Manhattan puede ser una buena aproximación en ciertos distritos de Madrid como el distrito de Salamanca, en dicha

¹El número de cálculos para n servicios suponiendo distancias simétricas viene dada por $\frac{(n-1)^2 - (n-1)}{2}$, en el caso simétrico por $n^2 - n$.

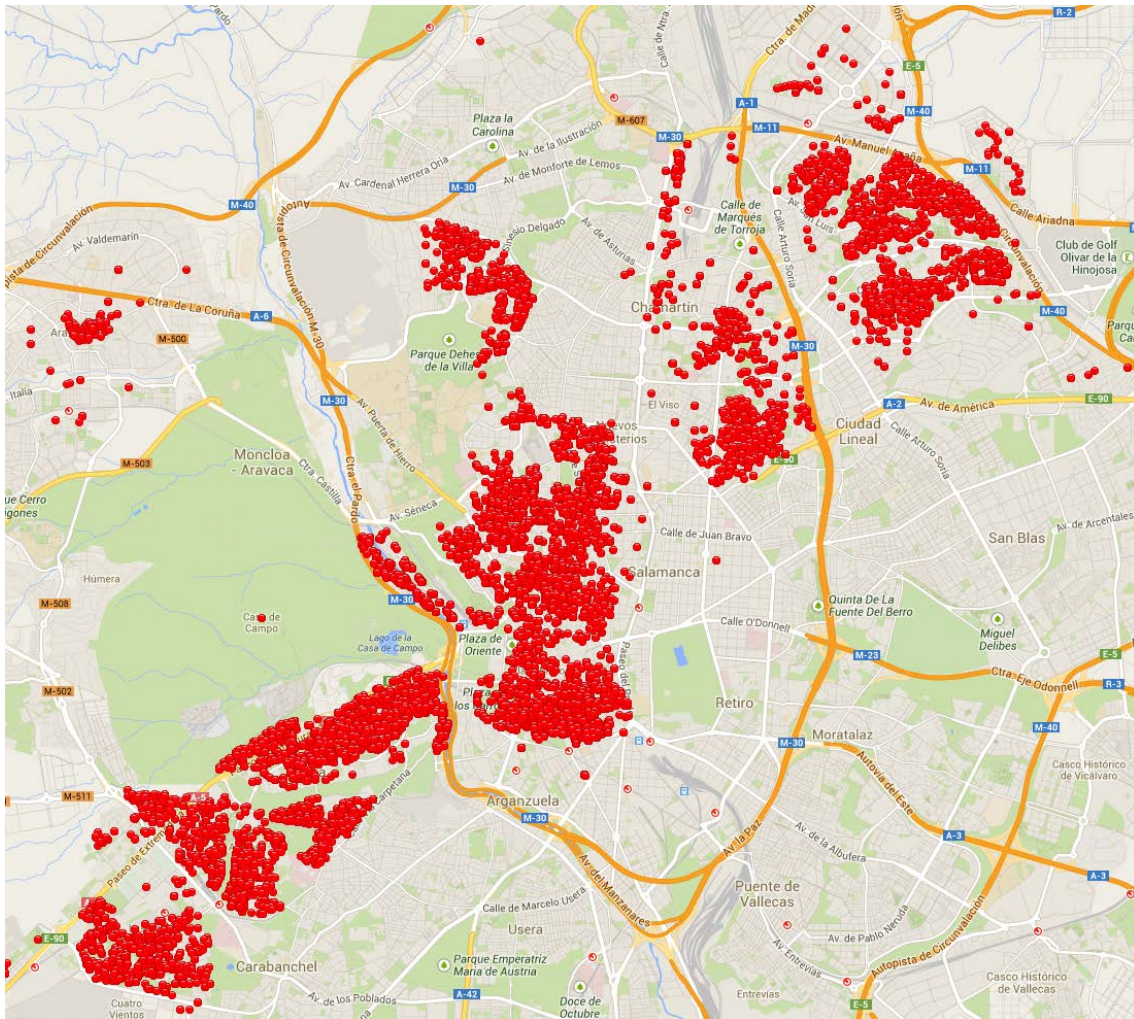


Figura 3.4: Representación de los 13344 servicios geolocalizados utilizando la herramienta Google Maps.

imagen se muestra la ruta a pie entre dos direcciones obtenidas y representada utilizando la herramienta Google Maps, se puede apreciar como las rutas alternativas sugeridas por el SIG presentan la misma distancia de 1,1Km y la misma estimación en tiempo de 15 minutos.

3.3.3. Particionamiento

El último proceso una vez realizada la geocodificación y el cálculo de distancias, ha sido el de dividir los 13344 servicios en cuatro grupos de servicios atendiendo al horario y a los días de la semana en el que se realizan dichos servicios.

- Grupo 1: Servicios realizados de lunes a viernes entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 10.709 servicios.

- Grupo 2: Servicios realizados de lunes a viernes entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 1.159 servicios.
- Grupo 3: Servicios realizados de sábado a domingo entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 1.070 servicios.
- Grupo 4: Servicios realizados de sábado a domingo entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 406 servicios.

Cada uno de los grupos de servicios anteriores forman una instancia del problema de planificación de asistentes de atención domiciliaria, las cuales serán objeto de experimentación en el capítulo 5.

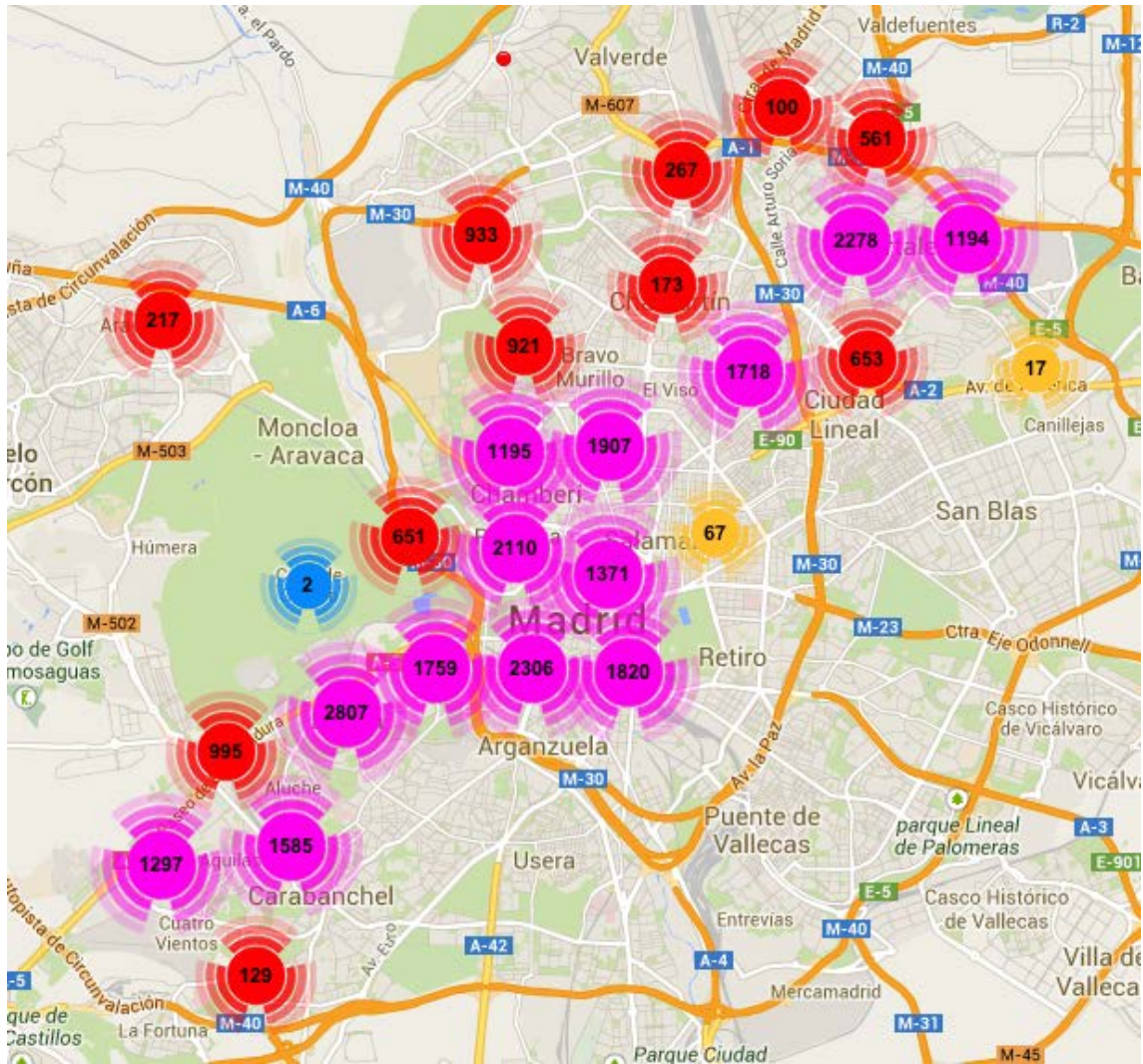


Figura 3.5: Representación de las 29034 tareas geolocalizadas utilizando la herramienta Google Maps y agrupadas de acuerdo a su proximidad.


```
{
  "routes" : [
    {
      "bounds" : {
        "northeast" : {
          "lat" : 40.421501,
          "lng" : -3.7002308
        },
        "southwest" : {
          "lat" : 40.4210003,
          "lng" : -3.701397
        }
      },
      "legs" : [
        {
          "distance" : {
            "text" : "0,2 km",
          },
          "duration" : {
            "text" : "2 min",
          }
        }
      ]
    },
    "status" : "OK"
  ]
}
```

Figura 3.6: Ejemplo de la respuesta obtenida al consultar el servicio web de Google de direcciones con las coordenadas geográficas (40.421501, -3.7002308) y (40.4210003, -3.701397). Entre otros datos se muestra la distancia total en kilómetros y la duración estimada, teniendo en cuenta que la ruta se realizará a pie.

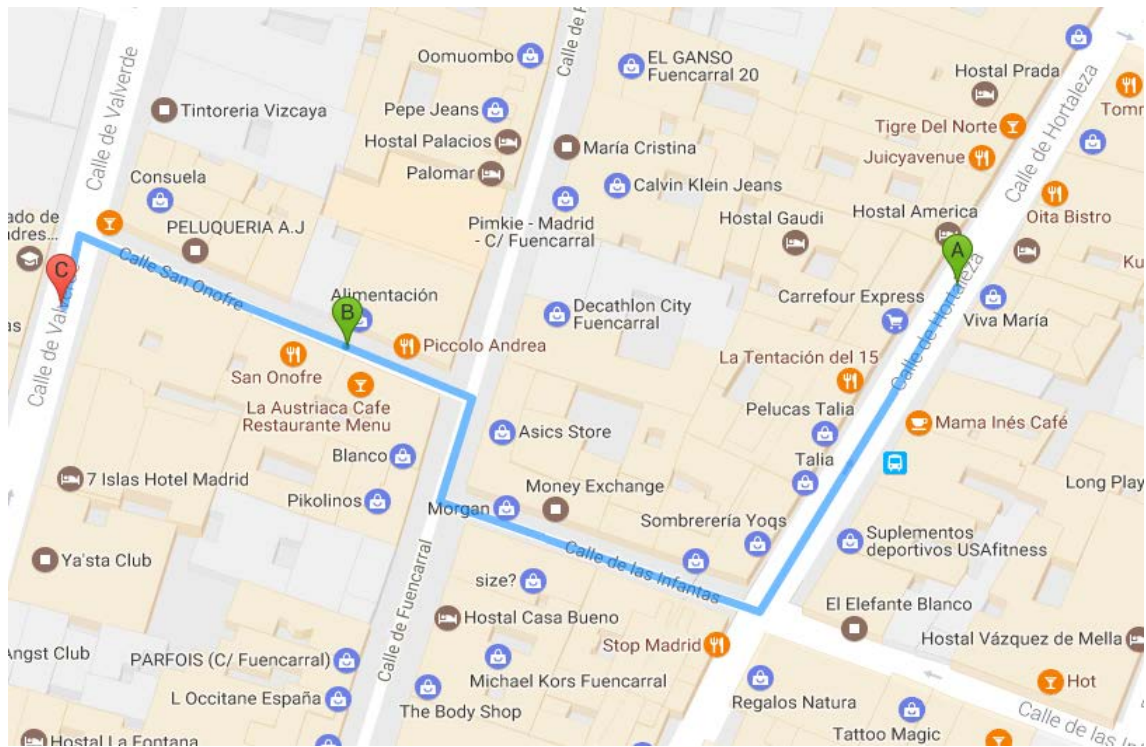


Figura 3.7: Ruta a pie entre tres direcciones obtenidas y representada utilizando la herramienta Google Maps.

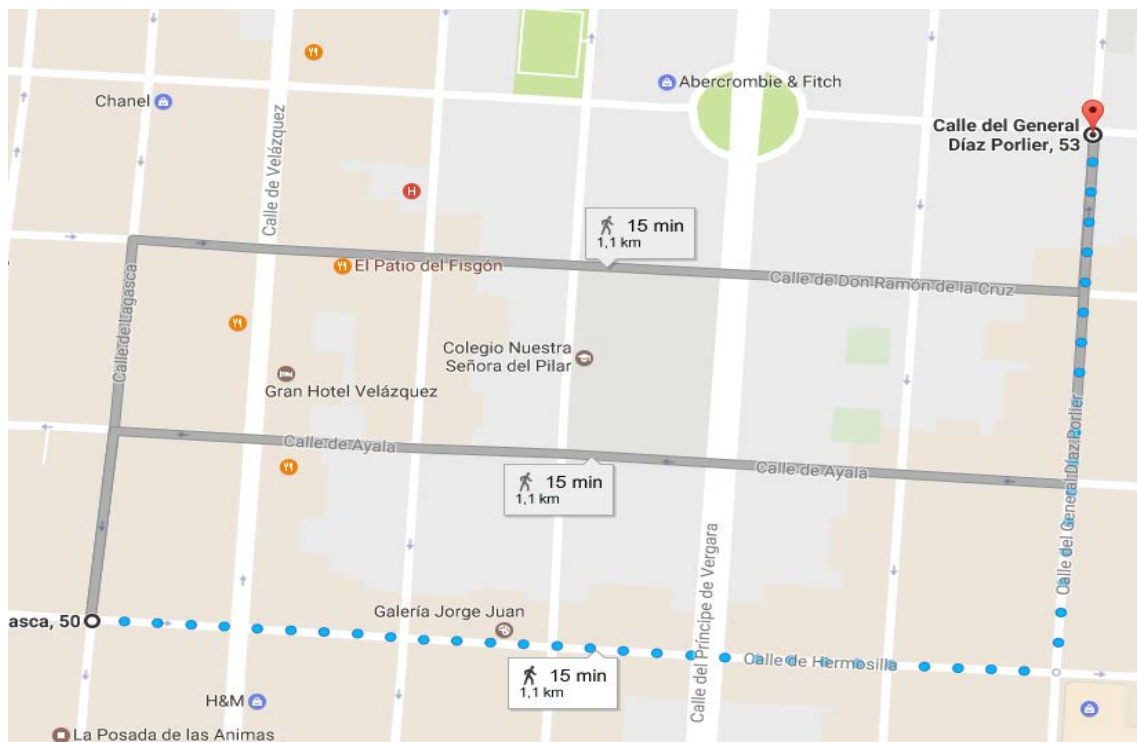


Figura 3.8: Ilustración de como la distancia Manhattan puede ser una buena aproximación, dada la geometría de algunos distritos de la ciudad de Madrid y teniendo en cuenta que el recorrido se realiza a pie.

3.4. Formulación matemática

Tal y como se expuso anteriormente, el problema objeto de estudio consiste en dado un conjunto de servicios, se debe encontrar un agrupamiento de dichos servicios, de tal modo que cada servicio pertenezca exactamente a único grupo a la vez que se respeten todas las restricciones. Mas formalmente el problema puede ser enunciado del siguiente modo: dado un conjunto de servicios $S = \{s_1, \dots, s_j, \dots, s_n\}$, el objetivo es encontrar una partición de los mismos $C = \{c_1, \dots, c_k\} (k \leq n)$, de tal modo que:

$$\bigcup_{i=1}^k c_i = S \quad (3.2)$$

$$c_i \neq \emptyset \quad \forall i \in [0, \dots, k] \quad (3.3)$$

$$c_i \cap c_j = \emptyset \quad \forall i \neq j \in [0, \dots, k] \quad (3.4)$$

Dicho problema tiene dos objetivos que se encuentran en conflicto y que deben ser minimizados:

1. Minimizar el número de asistentes requeridos para atender todas las tareas. En lo sucesivo a dicho objetivo será representado como Clust_{obj} .

$$\text{Minimizar } \text{Clust}_{obj} = |C| \quad (3.5)$$

2. Minimizar el tiempo total requerido para visitar a todos los clientes a lo largo del horizonte de planificación. En lo sucesivo dicho objetivo será representado por Tot_{obj} .

$$\text{Minimizar } \text{Tot}_{obj} = \sum_{i=1}^k W_i + T_i + P_i \quad (3.6)$$

Donde W_i es el tiempo total de espera, T_i es el tiempo total de desplazamiento y P_i es el tiempo total productivo, donde todos los subíndices i hacen referencia al i -ésimo clúster.

Puesto que $\sum_{i=1}^k P_i$ es una constante para cualquier instancia dada del problema, el objetivo anterior es equivalente a minimizar la suma de los tiempos de espera y de desplazamiento de los asistentes de atención domiciliaria. En lo sucesivo dicho tiempo será denominado tiempo no productivo y estará representado por U_i para el i -ésimo clúster.

El primero de los objetivos, Clust_{obj} minimizar el número total de asistentes requeridos, está enfrentado con el segundo objetivo, minimizar el tiempo total requerido Tot_{obj} . Dicho objetivo trata a su vez de minimizar dos factores como son el tiempo de desplazamiento y el tiempo de espera de los asistentes, ya que la tercera componente, el tiempo productivo es una constante y no se puede modificar. Que dichos objetivos se encuentren enfrentados, se traduce en que reducir el número de asistentes utilizados, requiere de modo general ampliar el tiempo empleado por cada uno de ellos. De modo análogo, aumentar el número total

de asistentes requeridos, implica reducir el tiempo total requerido por cada asistente para visitar a todos los clientes, ya que al utilizar un mayor número de asistentes los tiempos de espera y desplazamiento se reducen. En el extremo, si para cada cliente se tuviera asignado en exclusiva un asistente, los tiempos de espera y desplazamiento desaparecerían. Desafortunadamente este tipo de soluciones no son viables, ni desde un punto de vista económico, ni logístico, ni práctico.

Utilizando el problema de ejemplo mostrado en la tabla 3.1 y la posible solución mostrada en la figura 3.1, la compañía podría considerar el coste de contratar un nuevo asistente para asignarlo al servicio 2802 reduciendo el tiempo no productivo a cero y obteniendo un tiempo total requerido de $Tot_{obj} = 390$ minutos. Sin embargo, aunque el tiempo total requerido sería menor que el de la solución mostrada en la figura 3.1 el coste económico de dicha solución podría ser mayor. Todo ello dependerá del coste marginal de contratar a un nuevo asistente. Una situación similar surge cuando los costes marginales de contratar a un nuevo empleado aumentan, en dicha situación puede resultar beneficioso para la empresa obtener soluciones que, a pesar de tener un mayor tiempo no productivo, requieran una menor cantidad de asistentes para satisfacer la demanda. En esta investigación utilizaremos la siguiente función para estimar el coste monetario de una solución, la cual fue propuesta por (Quintana et al., 2017):

$$F_{cost} = 14,0 * \frac{Tot_{obj}}{60} + \frac{Clust_{obj} * 554,64}{52} \quad (3.7)$$

La solución propuesta en la figura 3.1 tendría un coste total de $14,0 * \frac{510}{60} + \frac{3 * 554,64}{52} \approx 151€$. De acuerdo con los autores, dicho coste presenta dos componentes. La primera de ellas representa el coste en el que incurre la compañía por hora trabajada, el cual incluye el salario del trabajador, así como el coste asociado a la cotización de la seguridad social. La segunda componente, representa el coste fijo en el que se incurre por contrato, el cual ha sido extraído de la encuesta de coste laboral (Instituto Nacional de Estadística, 2017) entre los que se incluyen gastos de carácter social, indemnizaciones por despido, gastos en formación profesional, transporte, indemnizaciones por fin de contrato, pagos compensatorios, pequeño utillaje, ropa de trabajo o selección de personal.

Como se mencionó anteriormente el objetivo consiste en encontrar una partición para un conjunto de servicios dados, de tal modo que las siguientes restricciones sean respetadas

1. Todo servicio debe ser atendido durante la semana y debe ser realizado dentro de su ventana temporal. Cualquier solapamiento entre tareas está expresamente prohibido.
2. El número máximo de horas de trabajo que un asistente puede desempeñar a lo largo de la semana debe ser respetado. En sucesivo dicho parámetro será representado por $max_w = 40$ horas

$$W_i + T_i + P_i < max_w \quad \forall i \in [0, \dots, k] \quad (3.8)$$

3. Puesto que los asistentes se desplazan de un cliente a otro a pie, la distancia máxima que pueden recorrer entre clientes está limitada. Dicho parámetro será representado por $Dmax_k$.

4. El tiempo de espera en el que un asistente puede incurrir antes de iniciar la realización de una tarea no deberá exceder cierto límite. Dicho límite será representado por $Wmax_k$. Esta restricción fue añadida para asegurar que las soluciones obtenidas están alienadas con los objetivos de negocio de la compañía.

Capítulo 4

Técnicas propuestas

Los que aseguran que es imposible no deberían interrumpir a los que estamos intentándolo.

Thomas Alva Edison

Resumen:

En este capítulo se realiza una exposición detallada del conjunto de técnicas propuestas para la resolución de problemas de optimización combinatoria con restricciones temporales y espaciales, en concreto para la resolución de problemas conocidos como problemas de planificación de asistentes de atención domiciliaria, que fueron presentados en el capítulo 3. En primer lugar, se presenta una técnica inspirada en el método clustering jerárquico propuesto por [Ward \(1963\)](#) adaptándolo a las restricciones espaciales y temporales presentes en el problema objeto de estudio. A tal efecto se define el concepto de similitud entre grupos de servicios, que como se mostrará está enfocada a capturar las componentes espaciales y temporales del problema estudiado, haciéndolo abordable mediante técnicas de clustering jerárquicas. En la segunda parte del capítulo se describen dos adaptaciones de la metaheurística ACO, propuesta por [Dorigo y Gambardella \(1997\)](#), al problema de planificación de asistentes de atención domiciliaria. La primera de las técnicas conocida como ACS-HCSP, tiene como objetivo realizar una adaptación de la metaheurística ACO al problema HCSP realizando el número indispensable de modificaciones. Esta primera aproximación no solo servirá para evaluar las posibilidades de resolver el HCSP, sino que también servirá de línea base a implementaciones más especializadas en el HCSP. Posteriormente y tras analizar las potenciales mejoras de ACS-HCSP se presenta la técnica IACS-HCSP cuyo objetivo fundamental es mejorar la técnica ACS-HCSP.

4.1. Introducción

Como se mostró en el capítulo 2 los problemas de planificación de asistentes de atención domiciliaria son problemas cuyo interés se ha visto acrecentado en los últimos años. Las previsiones demográficas, junto con las recomendaciones de distintas organizaciones como la Organización Mundial de la Salud o la Unión Europea, hacen que la asistencia domiciliaria de nuestros mayores en su domicilio, se haya convertido en una actividad cuya demanda va en aumento y cuyas previsiones apuntan a un aumento todavía más significativo durante los próximos años.

En el capítulo 3 se presentó el caso de estudio de una multinacional que opera en la Comunidad de Madrid cuya problemática y tamaño de las instancias, algunas con más de 10.000 servicios, a resolver se encuentran un orden de magnitud por encima del tamaño de instancias presentes en el estado del arte (ver tabla 2.5). En dicho capítulo se mostró como el problema a resolver puede plantearse como un problema de agrupamiento o clustering (ver sección 3.4). El tamaño de dichas instancias junto con la existencia de restricciones temporales y especiales requieren de la utilización de nuevas técnicas o de la adaptación de técnicas ya existentes, las cuales sean capaces de encontrar soluciones de un modo eficaz y que permitan en un futuro a la compañía resolver instancias de mayor tamaño. En este capítulo se presentan dos tipos de técnicas; la primera de ellas es una adaptación de la técnica de clustering aglomerativa propuesta por (Ward, 1963), mientras que la segunda es una adaptación de la metaheurística ACO, propuesta por Dorigo y Gambardella (1997).

4.2. Clustering Jerárquico

En esta sección se expone como abordar el problema presentado en la sección 3.4 utilizando una técnica de clustering jerárquico. Las técnicas de clustering se pueden dividir en dos grandes grupos, técnicas particionales y técnicas jerárquicas. Las técnicas particionales dividen un conjunto de n objetos con d dimensiones, en k grupos, de tal modo que sea minimizada la desviación total de cada objeto desde el centro de su clúster o desde una distribución de clústers. Las técnicas jerárquicas, como su propio nombre indica, buscan agrupar un conjunto de n objetos de tal modo que se cree una jerarquía, existiendo a su vez dos aproximaciones, las técnicas divisivas y las aglomerativas. En las primeras, todos los objetos comienzan en un mismo clúster, el cual se va dividiendo a medida que se descende en la jerarquía, por su parte las técnicas aglomerativas actúan de modo inverso. Cada objeto comienza dentro de su propio grupo, mientras que los pares de grupos se van juntando a medida que se asciende en la jerarquía.

Quizás una de las técnicas aglomerativas más conocidas es el método de Ward (1963). Este método, al igual que el resto de técnicas aglomerativas, comienza con tantos clústers como elementos se quieran agrupar, es decir, cada clúster está formado por un solo elemento. En cada etapa del proceso, se calcula una métrica que indica la distancia para cada par de clústers, y aquel par de clústers con la menor distancia es unido componiendo un nuevo clúster. El proceso finaliza cuando solo existe un clúster que contiene todos los elementos, teniendo como resultado una jerarquía que queda definida por el orden en el que se produjeron las uniones de los distintos clústers, en la figura 4.1 se puede ver un ejemplo de este tipo de técnicas aglomerativas.

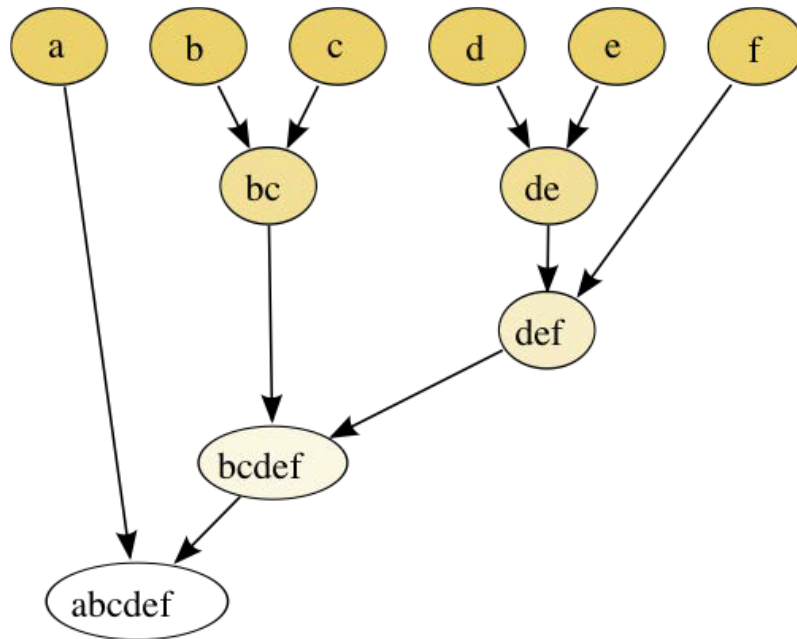


Figura 4.1: Ejemplo de clustering jerárquico aglomerativo, se puede ver como todos los elementos A,B,C,D,E,F comienzan en su propio clúster, para posteriormente ir agrupándose de modo jerárquico hasta formar un clúster formado por todos los elementos iniciales.

El método propuesto por Ward mide la distancia entre dos clústers, A y B , como el incremento en la suma de las distancias desde los elementos $\vec{x}_i \in A \cup B$ al nuevo centroide $\vec{m}_{A \cup B}$, respecto de los clústers originales A y B ; en el caso de un método aglomerativo se selecciona en cada paso aquellos pares de clústers que minimizan dicho incremento. De un modo más formal, se define dicho incremento como:

$$\Delta(A, B) = \sum_{i \in A \cup B} \|\vec{x}_i - \vec{m}_{A \cup B}\|^2 - \sum_{i \in A} \|\vec{x}_i - \vec{m}_A\|^2 - \sum_{i \in B} \|\vec{x}_i - \vec{m}_B\|^2 = \frac{n_a n_b}{n_a + n_b} \|\vec{m}_A - \vec{m}_B\|^2 \quad (4.1)$$

de modo genérico \vec{m}_i es el centro del clúster j y n_j es el número de elementos que pertenecen dicho clúster j .

Dicha idea de agrupar aquellos elementos que produzcan el menor incremento en cierta medida de distancia, puede ser reutilizada con el objetivo de adaptarlo al problema al problema de planificación de asistentes de atención domiciliaria presentado en el capítulo 3. En primer lugar, es conveniente destacar que el concepto de centroide utilizado en la ecuación (4.1), no es fácilmente trasladable al caso que nos ocupa, no es sencillo hallar una definición de centroide para dos servicios que están formados a su vez por tareas y sobre los cuales se deben construir turnos de trabajo que cumpla con ciertas restricciones. Además, hay que tener en cuenta que existen dos componentes, la componente espacial, que indica donde debe realizarse la tarea y la componente temporal que indica el instante o rango temporal en el que debe realizarse. Si bien para el cálculo de la componente espacial, bastaría con promediar sus coordenadas, existiendo trabajos en esa dirección [Shin](#)

y Han (2011), no se han encontrado trabajos relevantes en los que se tengan en cuenta las restricciones temporales, menos aun cuando el conjunto de tareas a planificar debe realizarse dentro de un horizonte de planificación de varios días, como es el caso que nos ocupa. Finalmente, y a diferencia del método original propuesto por Ward, en la técnica propuesta no obtendremos un único clúster que incluya a todos los servicios, ya que los clústers obtenidos como solución al problema deben respetar las restricciones del problema. Por todo lo anteriormente expuesto, se decide utilizar otra métrica de similitud no basada en centroides. A continuación se expone como se calcula dicha similitud para un conjunto de tareas, no teniéndose en cuenta el horizonte de planificación y posteriormente se expone como se generaliza dicha distancia en un horizonte de planificación de varios días.

4.2.1. Similitud

En esta sección se presenta uno de los conceptos centrales de esta investigación, el cual se centra en cómo medir la similitud/distancia entre servicios que deben de ser prestados semanalmente y que presentan tanto restricciones temporales como espaciales, en lo sucesivo nos referiremos a dicha medida como similitud o distancia. Como se introdujo en el capítulo 3, cada uno de estos servicios están formados por un conjunto de tareas que deben prestarse de modo conjunto dentro de un horario y en una localización concreta. En dicho capítulo también se presentaron las diversas restricciones que deben cumplirse a la hora de agrupar servicios, como son la limitación de distancia máxima recorrida, el número máximo de horas semanales/diarias de trabajo y el tiempo máximo de espera de espera en el que puede incurrir un asistente de atención domiciliaria a la hora de visitar a un cliente. Estas restricciones, no solo tienen una componente temporal, sino que además presentan una componente espacial. Ambas componentes propician que el concepto usual de distancia o similitud utilizado por las técnicas tradicionales de clustering tenga que ser replanteado a fin de incluir ambas componentes. En primer lugar, se presenta como incluir ambas componentes para un conjunto de tareas que deben prestarse el mismo día de la semana y a partir de dicha definición se presenta el mismo concepto para un conjunto de servicios.

4.2.1.1. Tareas

Como se presentó en el apartado 3.4 cada uno de los servicios a agrupar con el objetivo de formar un horario de trabajo, está formado por una o varias tareas. Una tarea t_i queda caracterizada por las siguientes propiedades, $t_i = \{lat, lon, \alpha_i, h_i, d_i\}$ donde

- lat, lon Indican las coordenadas geográficas donde la tarea debe ser realizada.
- $[e_i, l_i]$ Indica la hora mínima e_i y máxima l_i a la que debe comenzar dicha tarea.
- h_i Indica la hora a la que se llegará a dicha tarea.
- d_i Indica la duración de dicha tarea.

A partir de la definición anterior, la similitud para dos tareas t_i, t_j queda definida del siguiente modo.

$$D(t_i, t_j) = \begin{cases} T(lat_i, lon_i, lat_j, lon_j) + d_j + \frac{(e_j - h_j) + |e_j - h_j|}{2} & \text{si } e_j \leq h_j \leq l_j \\ \infty & \text{en otro caso} \end{cases} \quad (4.2)$$

Dicha medida de similitud, es en realidad una medida del tiempo empleado desde la conclusión de la primera tarea hasta la finalización de la segunda, el cual está compuesto por tres sumandos. El primer sumando $T(lat_i, lon_i, lat_j, lon_j)$ se refiere al tiempo de desplazamiento entre las dos tareas, el cual queda definido por el cociente entre la distancia para dos pares de coordenadas y una constante μ que mide la velocidad de desplazamiento de los asistentes, es decir.

$$T(lat_i, lon_i, lat_j, lon_j) = \frac{F(lat_i, lon_i, lat_j, lon_j)}{\mu} \quad (4.3)$$

Tal y como se definió en la ecuación (3.1) del capítulo 3 para medir la distancia entre dos coordenadas geográficas se utiliza la distancia Manhattan, por lo que la ecuación anterior se puede expresar del siguiente modo

$$T(lat_i, lon_i, lat_j, lon_j) = \frac{C * (|lng_1 - lng_2| * \cos(\frac{lat_1 - lat_2}{2}) + |lat_1 - lat_2|)}{\mu} \quad (4.4)$$

El segundo sumando, d_j de la ecuación 4.2, indica la duración de la segunda tarea, mientras que el tercero indica el tiempo de espera entre la llegada a la segunda tarea y el inicio de la misma. Cabe destacar que a pesar de haber definido como se calcularán la similitud entre dos tareas, ha sido necesario la utilización del concepto de h_i que representa el tiempo de llegada a la tarea t_i , dicho cálculo no es constante y depende de la tarea que se haya visitado con anterioridad. Dicho cálculo de h_i se va realizando de modo incremental a medida que se añaden tareas al clúster. En el estado inicial, el clúster está compuesto por una sola tarea t_1 , por lo que $h_1 = 0$, al añadir una nueva tarea t_2 el cálculo de la hora de llegada a dicha tarea queda determinado por:

$$h_2 = h_1 + \frac{(e_1 - h_1) - |e_1 - h_1|}{2} + d_1 + T(lat_1, lon_1, lat_2, lon_2) \quad (4.5)$$

De modo genérico se tiene:

$$h_{i+1} = h_i + \frac{(e_i - h_i) - |e_i - h_i|}{2} + d_i + T(lat_i, lon_i, lat_{i+1}, lon_{i+1}) \quad (4.6)$$

En la figura 4.2 se ilustra el concepto de similitud entre dos tareas $D(t_1, t_2)$, tiempo de desplazamiento y tiempo de espera. Tal y como se puede apreciar el concepto de similitud entre dos tareas no es simétrico, ni tiene un valor absoluto, sino que depende del orden en el que dichas distancias son calculadas. En general $D(t_i, t_j) \neq D(t_j, t_i)$, en el ejemplo de la figura 4.2 la distancia entre t_1, t_2 es igual $D(t_1, t_2) = 2$ horas valor obtenido al sumar los tiempos de desplazamiento 30 minutos, espera 1 hora y realización de la tarea t_2 30 minutos, mientras que la distancia $D(t_2, t_1) = \infty$ ya que es imposible realizar primero la tarea t_2 y posteriormente la tarea t_1 . Tal y como se expone en la ecuación (4.2), en el caso de no llegar a tiempo para la realización de una tarea t_j al ser la hora de llegada $h_j > e_j$ posterior al inicio de la tarea, la similitud con dicha tarea sea ∞ , además existen otras restricciones como las enunciadas en el apartado 3.4 que hacen que la similitud entre dos tareas sea ∞ por ejemplo que la distancia entre ellas sea superior a cierto valor $Dmax_k$.

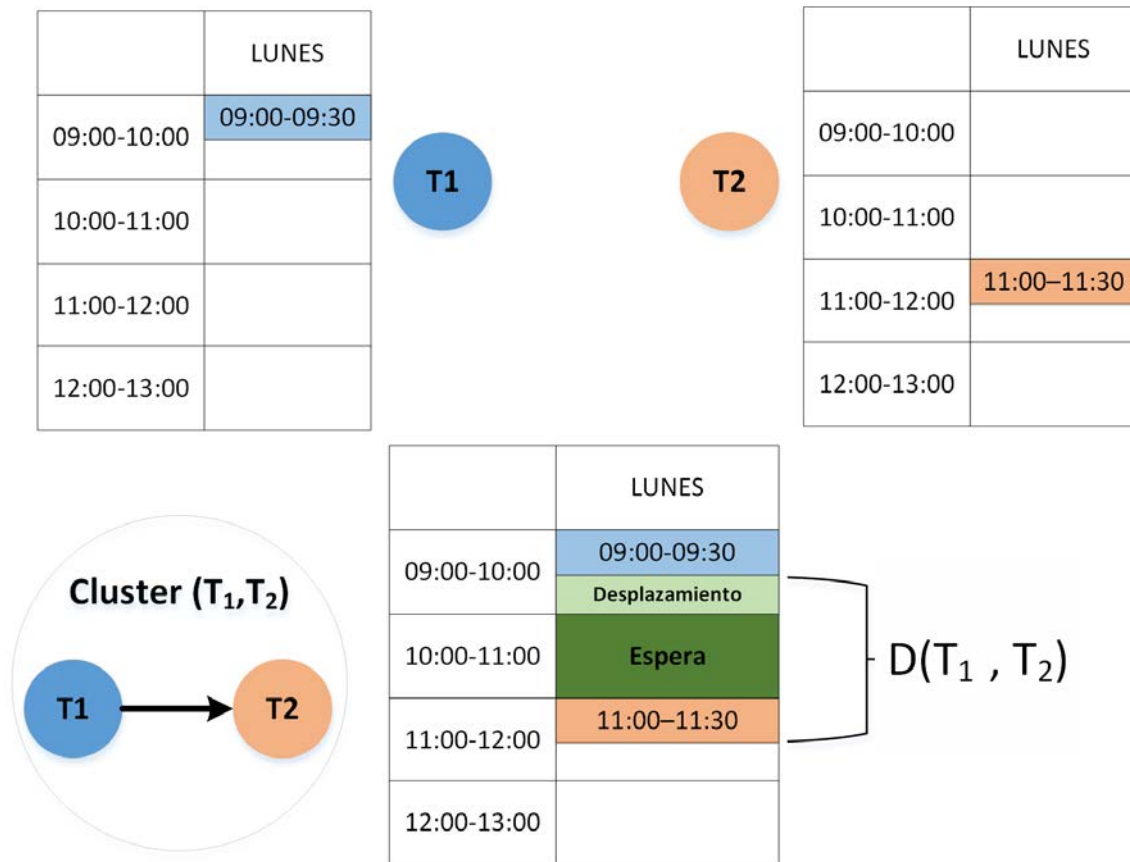


Figura 4.2: Ejemplo del concepto de similitud $D(t_1, t_2)$ para dos tareas t_1, t_2 , resultado de sumar el tiempo de desplazamiento, el tiempo de espera y el tiempo de realización de la tarea t_2 . Nótese que el valor h_2 que representa la hora de llegada a la tarea t_2 coincide con el comienzo del tiempo de espera.

4.2.1.2. Servicios

Una vez definido el concepto de similitud para un par de tareas, resulta sencillo extender dicho concepto a los servicios. Tal y como se mostró en la tabla 3.1, un servicio es un conjunto de tareas que deben realizarse dentro de un horizonte de planificación, las cuales han de ser prestadas por el mismo asistente de atención domiciliaria. En realidad agrupar clientes o servicios es equivalente, por lo que la similitud entre dos servicios o clientes, no es más que la suma de similitudes para cada uno de los días d del horizonte de planificación, en la figura 4.3 se ilustra dicho concepto.

Como se puede apreciar en la figura, la similitud entre dos servicios $D(S_i, S_j)$ es igual a $D(S_i, S_j)$, ya que en este caso y a diferencia de la distancia entre tareas, el orden de visita se determina para cada día del horizonte de planificación. Al igual que sucediera con las tareas la distancia $D(S_i, S_j) = \infty$ si no se cumplen con las restricciones detalladas en 3.4.

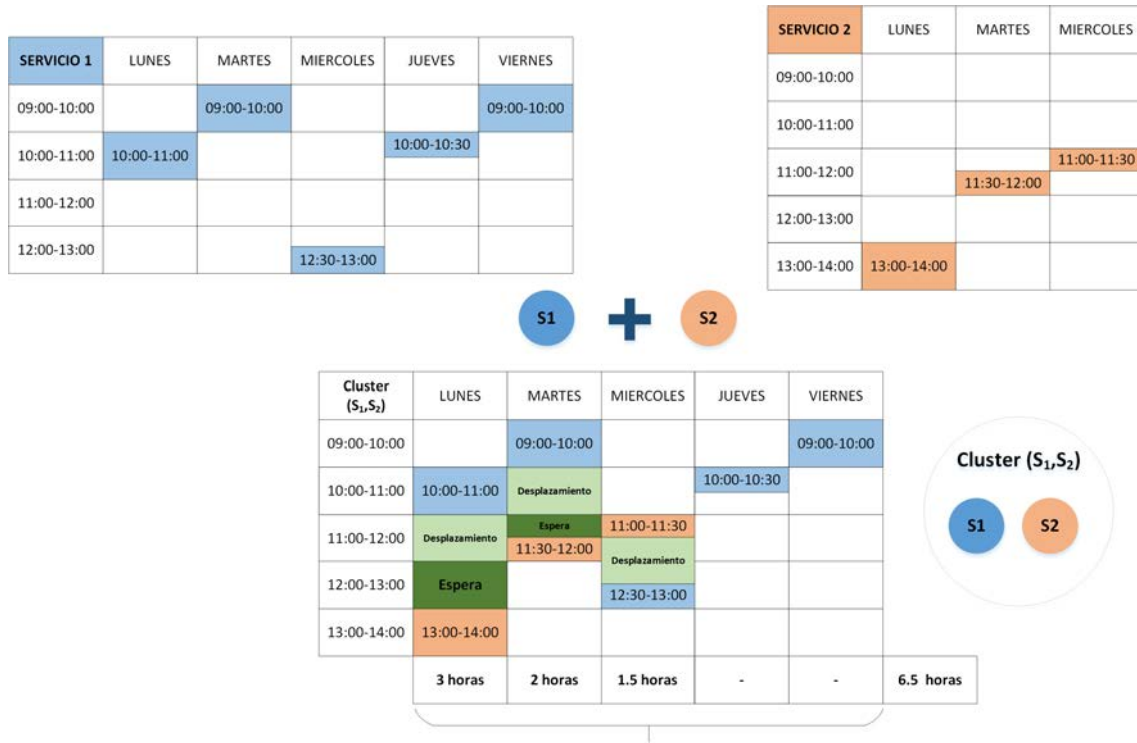


Figura 4.3: Ejemplo del concepto de similitud $D(S_1, S_2)$ para dos servicios S_1, S_2 , resultado de suma la distancia $D(t_i, t_j)$ dentro del horizonte de planificación. Nótese como el tiempo de desplazamiento entre las tareas del servicio S_1 y las tareas del servicio S_2 es constante y simétrico.

4.2.2. Método de Ward modificado

Una vez definido el concepto de similitud entre tareas y servicios, se pasa a detallar el algoritmo de clustering jerárquico que dado un conjunto de servicios S , es capaz de agruparlos en clústers C que minimiza la medida de distancia expuesta con anterioridad. El algoritmo comienza asignando a cada servicio S_i un clúster C_i distinto. En cada iteración del algoritmo se calculan las similitudes entre cada par de clústers, uniéndose aquel par de clústers que obtuvieron en menor valor en el cálculo de la distancia. El proceso finaliza cuando se satisface la condición de parada. En el método original propuesto por Ward (1963), al igual que en el resto de técnicas aglomerativas, dicha condición se alcanza cuando solamente existe un clúster que incluye a todos los clústers iniciales. En nuestro caso y dadas las restricciones del problema (ver 3.4), la condición de parada se cumple cuando la distancia entre todo par de clústers es ∞ . El pseudocódigo del algoritmo propuesto se puede ver en 4.1.

El algoritmo anterior al estar inspirada en una técnica de clustering aglomerativa presenta una complejidad computacional $O(n^3)$, lo cual puede suponer un problema para problemas que presenten una alta dimensionalidad, como es nuestro caso. Con el objetivo de reducir los tiempos de cálculo se implementa un mecanismo de caché, de tal modo que distancias calculadas entre pares de servicios en iteraciones anteriores son reutilizadas,

Algoritmo 4.1: Algoritmo de Ward Modificado

Datos: $S = \{S_1 \dots S_n\}$ Conjunto de servicios
Resultado: $C = \{C_1 \dots C_m\}$ Conjunto de clústers

```

1 para  $i \leftarrow 0$  a  $n$  hacer
  // Asignamos a cada clúster un servicio;
2    $C_i = S_i$ ;
3  $stop \leftarrow false$ ;
4 mientras  $\neg stop$  hacer
5    $bestDistance \leftarrow \infty$ ;
6   para  $k \leftarrow 0$  a  $m$  hacer
7      $C_k \leftarrow getCluster(k)$ ;
8     para  $l \leftarrow k + 1$  a  $m$  hacer
9        $C_l \leftarrow getCluster(l)$ ;
10       $d_{kl} \leftarrow D(C_k, C_l)$ ;
11      si  $d_{kl} < \infty$  entonces
12        si  $d_{kl} < bestDistance$  entonces
13           $bestDistance \leftarrow d_{ij}$ ;
14           $l' \leftarrow l$ ;
15           $k' \leftarrow k$ ;
        // Si no es posible unir ningún clúster más;
16      si  $bestDistance = \infty$  entonces
17         $stop \leftarrow true$ ;
        // Se junta aquel par de clústers con la menor distancia;
18      en otro caso
19         $C' \leftarrow Merge(k', l')$ ;
20         $C \leftarrow C \cup C'$ ;
21         $C \leftarrow C \setminus C_{k'} \setminus C_{l'}$ ;

```

reduciendo los tiempos de cálculo.

4.3. Optimización basada en colonia de hormigas

Tal y como se expuso en la sección 2.4, la Optimización basada en colonias de hormigas, ACO, ha sido aplicada a multitud de problemas de optimización combinatoria TSP (Dorigo y Gambardella, 1997), TSPTW (Cheng y Mao, 2007; López-Ibáñez y Blum, 2010) VRPTW (Gambardella et al. (1999), PVRPTW (Yu y Yang, 2011) en la sección 2.4.9 se pueden encontrar otras aplicaciones de ACO. A pesar de que ACO se ha aplicado a problemas clásicos de optimización de rutas como el TSP (ver sección 2.1) y el VRP (ver sección 2.2) y a sus variantes con ventanas temporales, hasta la fecha no se han encontrado publicaciones relevantes que intenten abordar el problema de planificación de asistentes de atención domiciliaria con este tipo de metaheurística. En esta sección se proponen dos adaptaciones de ACO, una primera denominada ACS-HCSP (Ant Colony System for HCSP) en la que se proponen una serie de cambios mínimos para adaptarla al problema tratado y una segunda, denominada IACS-HCSP (Improved Ant Colony System for HCSP), en la que se realizan un mayor número de cambios enfocados a mejorar el rendimiento y adaptarlo a las características particulares del problema abordado.

4.3.1. Creación del grafo

La optimización basada en colonias de hormigas, ACO ver 2.4, fue propuesta inicialmente por Dorigo et al. (1996) para resolver el problema del TSP, por lo tanto es una técnica que trabaja sobre un grafo, completo y no dirigido, al considerarse las distancias simétricas en la mayoría de los casos. En dicho grafo, el espacio de soluciones es explorado de un modo probabilístico, de tal modo que las mejores soluciones encontradas hasta el momento, tengan una mayor probabilidad de ser visitadas, estando dicha técnica inspirada en el depósito de feromona que realizan las hormigas reales. Por lo tanto, para abordar la resolución de problemas de planificación de asistentes de atención domiciliaria mediante esta metaheurística, el primer paso es crear un grafo que represente el problema a resolver. En primer lugar, se plantea la creación del grafo para un conjunto de tareas que han de prestarse en el mismo día, para posteriormente extender dicha idea a un conjunto de servicios.

En el caso del problema de planificación de asistentes de atención domiciliaria, cada tarea estará representada por un vértice, se creará una arista entre todo par de tareas/vértices (t_i, t_j) , siempre que la distancia definida $D(t_1, t_2) \neq \infty$, estando dicha distancia definida en la ecuación (4.2). Es decir, se creará una arista entre dos tareas, siempre que la realización de las dos tareas por parte de un mismo asistente de atención domiciliaria respete las restricciones del problema, en particular el tiempo de desplazamiento. En la figura 4.4 se muestra un ejemplo de grafo para tres tareas, mostrándose también la incompatibilidad entre las mismas; en concreto se ilustra el concepto de incompatibilidad entre dos tareas $D(T_1, T_3) = \infty$, al no respetarse el tiempo máximo de desplazamiento.

Para el caso de servicios, el concepto de similitud/distancia es idéntico al empleado en la sección 4.2.1.2, siendo esta la suma de distancias para cada día del horizonte de planificación. De modo análogo a la creación de un grafo con tareas, por cada servicio existente se creará un vértice, y para todo par de vértices/servicios cuya distancia $D(S_i, S_j) \neq \infty$ se creará una arista que contendrá la distancia entre dicho par de servicios. En la figura 4.4 se muestra

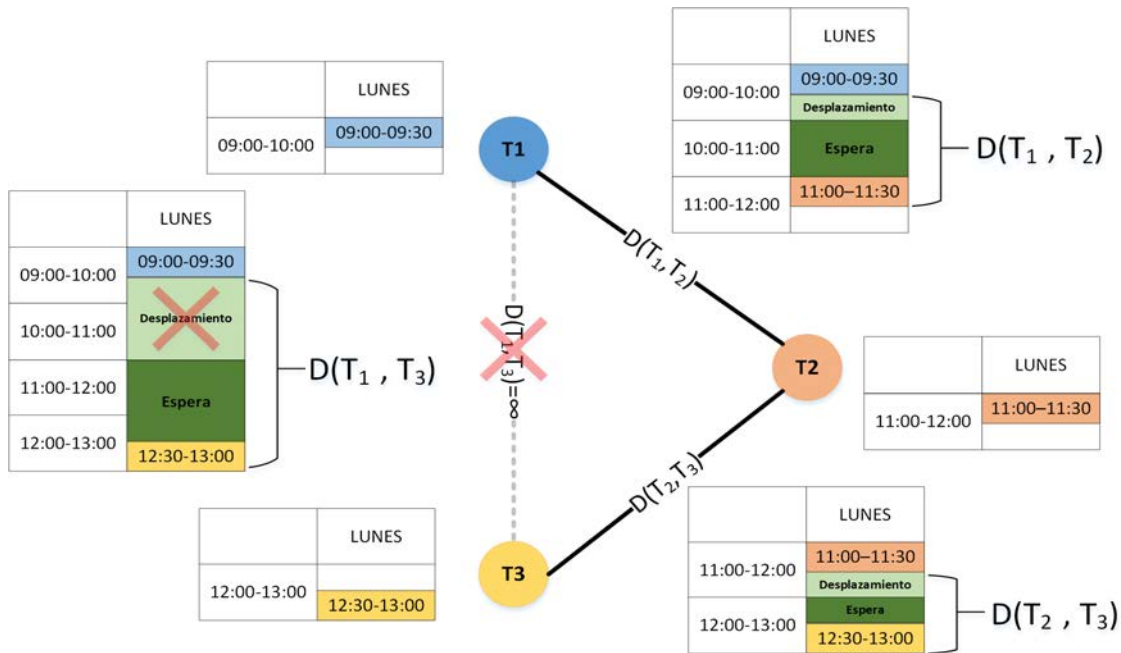


Figura 4.4: Ejemplo de construcción de grafo con tres tareas T_1, T_2, T_3 , en el cual se ilustra cómo se crea el grafo para un conjunto de tareas, existiendo una arista entre cada par de tareas (t_i, t_j) siempre que $D(t_i, t_j) \neq \infty$.

un ejemplo de grafo para tres tareas, mostrándose también la incompatibilidad entre las mismas.

De un modo más formal, sea $G = \{V, A\}$ un grafo finito no dirigido, donde $V = \{v_1 \dots v_n\}$ representa un conjunto de nodos o vértices, $A = \{(i, j) | i, j \in V\}$ un conjunto de aristas y $d_{ij} = d_{ji} \in \mathbb{R}^+$ una medida de distancia entre el nodo v_i y v_j o viceversa. Para cada **servicio** existente en el problema se añade un **nodo** al grafo, dicho nodo incluye toda la información referente al conjunto de tareas que incluye como son, el tiempo de inicio de la tarea, la duración, la localización, el día de la semana y la ventana temporal. El otro componente fundamental de un grafo son las **aristas**, tradicionalmente dichas aristas han representado distancias entre pares de nodos, representado distancias entre ciudades como en el caso del TSP o distancias entre clientes a visitar en el caso del VRP. En nuestro caso, para la resolución de problemas de planificación de asistentes de atención domiciliaria, las aristas del grafo tienen una doble función, ya que representan la similitud/distancia entre pares de servicios y al mismo tiempo garantizan que si se incluyesen dichos nodos en un único clúster se respetarían las restricciones presentadas en la sección 3.4, en concreto se deben garantizar:

- Ninguna tarea se puede solapar con otra.
- Se debe respetar que la distancia recorrida por los asistentes entre dos visitas no supere cierto umbral $Dmax_k$.
- Se debe respetar que el tiempo máximo de espera en el que un asistente puede incurrir

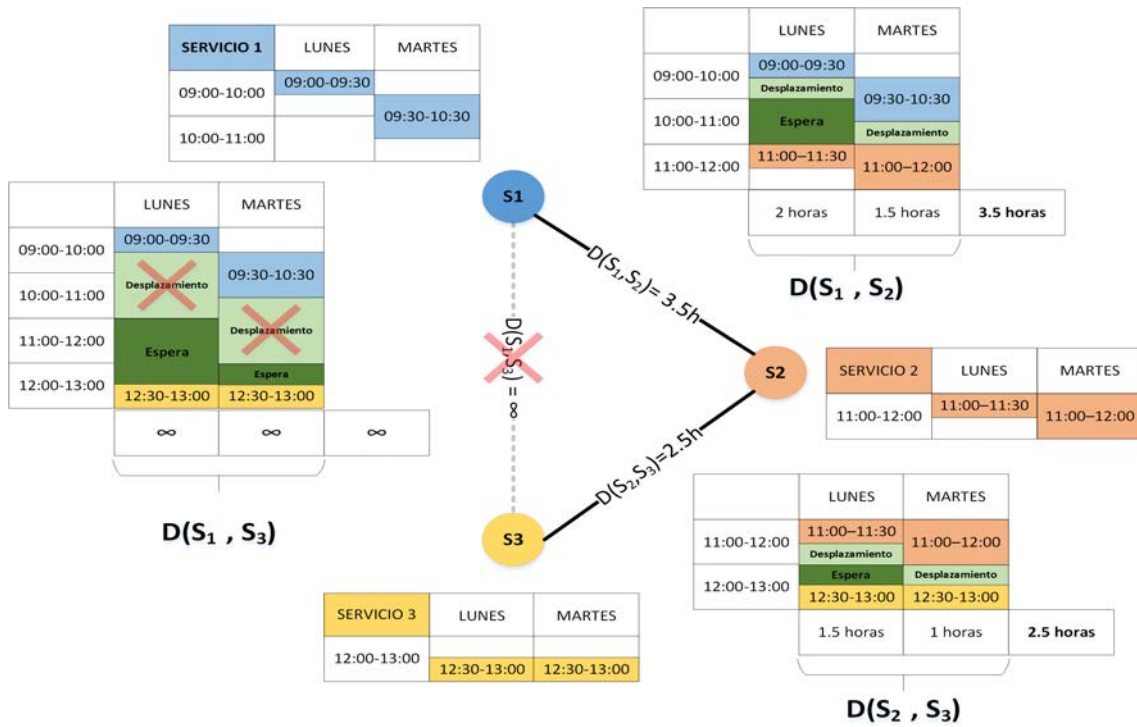


Figura 4.5: Ejemplo de construcción de grafo con tres servicios S_1, S_2, S_3 , como se puede apreciar para cada par de servicios entre los que existe una distancia definida, se crea una arista. A modo de ejemplo se muestra el concepto de incompatibilidad entre dos servicios $D(S_1, S_3) = \infty$, al no respetarse el tiempo máximo de desplazamiento.

antes de comenzar cada tarea no supere cierto umbral $Wmax_k$.

En la figura 4.5 se muestra como la restricción de máximo desplazamiento puede ser modelada de modo explícito a la hora de crear el grafo. En dicha figura, se muestra como los servicios S_1, S_3 son incompatibles entre sí al no respetar la restricción de distancia máxima recorrida, para garantizar que la restricción de distancia máxima recorrida es satisfecha, basta con no añadir dicha arista al grafo. Al no añadirse dicha arista se garantiza que ninguna hormiga transite entre dichos vértices y por lo tanto se garantiza respetar dicha restricción. Como se mostrará en los siguientes apartados dicha concepción estática entre la compatibilidad de dos servicios puede resultar en algunos casos demasiado restrictiva (ver apartado 4.3.3) y demasiado laxa en otros ya que a medida que cada hormiga construye su solución dicha compatibilidad inicial puede variar.

4.3.2. ACS-HCSP

A pesar de que ACO se ha aplicado con éxito a numerosos problemas de enrutamiento de vehículos, muchos de los cuales presentaban diferentes restricciones como capacidad de carga, distancia máxima recorrida o ventanas temporales (VRPTW) no existe un modo formal de aplicar dicha metaheurística a un problema de optimización concreto, ni tan

quiera para el mismo problema de optimización. Por ejemplo en el caso del VRPTW, Gambardella et al. (1999) propone la utilización de dos colonias de hormigas cada una de las cuales minimiza el número de vehículos y la distancia total recorrida, mientras que en Shi y Weise (2013) se propone el uso de una sola colonia, obteniendo resultados competitivos.

A la hora diseñar un algoritmo basado en ACO y aplicarlo al problema de planificación de asistentes de atención domiciliaria, hay varias cuestiones que deben ser consideradas. En primer lugar, hay que detallar cómo se crean las soluciones y cómo los agentes, denominados hormigas, llegan a dichas soluciones. Cada hormiga representa una solución completa y válida al problema, en nuestro caso será un conjunto de clústers $C = \{c_1, \dots, c_k\}$, cada uno de los cuales estará formado por un conjunto de servicios $c_k = \{s_1, \dots, s_m\}$. Las soluciones se obtienen de un modo constructivo a medida que cada hormiga se va desplazando por los vértices del grafo, el orden de visita determina el conjunto de clústers obtenidos. Inicialmente cada hormiga selecciona un vértice inicial, de acuerdo a una política concreta 4.3.2.1 y lo incorpora a un clúster c_0 que inicialmente está vacío. Posteriormente y de acuerdo a una regla de transición probabilística, se selecciona cual será el siguiente vértice a visitar, el cual pasará a formar parte del clúster en construcción. Dicho proceso de incorporación de vértices al clúster actual finalizará cuando se cumpla cierta condición de parada. Una vez se cumpla dicha condición se habrá formado el primer clúster c_0 , el proceso de formación de clústers se reiniciará y repetirá formando tantos clústers como sean necesarios, hasta que todos los vértices del grafo hayan sido visitados. En segundo lugar, hay que tener en cuenta como se garantiza la obtención de soluciones válidas que respeten las restricciones expuestas en 3.4. La obtención de soluciones válidas se consigue a través de dos elementos, el grafo y los propios agentes. A través del grafo, y gracias a las aristas, se pueden modelar algunas restricciones como qué servicios son incompatibles entre sí, ya sea por requerir tareas al mismo tiempo o por encontrarse a una distancia máxima superior a la que puede recorrer un asistente entre dos pacientes. Como se mostrará en el apartado 4.3.3, utilizar de modo exclusivo las aristas del grafo puede resultar o demasiado restrictivo o demasiado laxo. A continuación se exponen de modo detallado cada una de las fases de las que consta el algoritmo.

4.3.2.1. Selección del vértice de inicio

Cada vez que una hormiga comienza a buscar una nueva solución, el primer paso a realizar es la selección de un vértice inicial, dicho vértice será incluido en el clúster actual, y determinará qué otros vértices podrán ser visitados. Dicha política será utilizada cada vez que sea necesario formar un nuevo clúster. Dicha selección es realizada teniendo en cuenta la cardinalidad de cada nodo y una regla de selección probabilística. La probabilidad del nodo v_n de ser seleccionado como nodo inicial para un clúster viene dada por

$$P(v_n) = \begin{cases} \frac{|v_n|}{\sum_{i=0}^n |v_i|} & \text{if } q^i \leq q_0^i \\ \operatorname{argmax}_x \{|v_x|\} & \text{if } q^i > q_0^i \end{cases} \quad (4.7)$$

Donde $q_0^i \in [0, 1]$ es un parámetro y q^i es una variable aleatoria uniformemente

distribuida entre $[0, 1]$. La idea subyacente en la ecuación anterior es la comenzar por aquellos servicios que tengan mayor posibilidad de poder agruparse con otros servicios.

4.3.2.2. Regla de transición entre vértices

Una vez elegido el vértice inicial por el cual se comenzará a formar el clúster, se debe especificar cuál es la política de transición entre un vértice dado v_i y el siguiente v_j , en la sección 4.3.2.6 dicha política es referida como una función cuyo nombre es *selectNextService* (c_{rnk}, Ant_{rn}), la cual dado el k -ésimo clúster por la n -ésima hormiga en la r -ésima ronda y la propia hormiga es capaz de decidir cuál es el siguiente vértice a visitar. Dicha función de transición es una adaptación de la regla proporcional pseudoaleatoria propuesta por [Dorigo y Gambardella \(1997\)](#), en nuestro caso la probabilidad de transición de una hormiga k situada en el vértice i visite el vértice j queda definida del siguiente modo.

$$j = \begin{cases} \arg \max_{j \in N_i^k} \{[\tau_{ij}] \cdot [\eta_{ij}]\} & \text{si } q \leq q_0 \\ P_{ij}^k & \text{si } q > q_0 \end{cases} \quad (4.8)$$

en dicha propuesta cada vez que una hormiga situada en el vértice i desea transitar a otro vértice, elige una variable aleatoria q uniformemente distribuida entre $[0, 1]$. Si $q \leq q_0$ la hormiga selecciona aquel vértice que maximice la función $[\tau_{ij}^x(t)] \times [\eta_{ij}^x(t)]$, utilizando en caso contrario la función definida por P_{ij}^k y cuya definición se muestra a continuación.

$$P_{ij}^k = \begin{cases} \frac{[\tau_{ij}] \cdot [\eta_{ij}]}{\sum_{u \in N_i^k} [\tau_{iu}] \cdot [\eta_{iu}]} & \text{si } j \in N_i^k \\ 0 & \text{si } j \notin N_i^k \end{cases} \quad (4.9)$$

Donde N_i^k es el conjunto de nodos alcanzables desde el vértice i por la hormiga k , τ_{ij}^x representa el valor de feromona y η_{ij}^x representa el valor heurístico de transitar entre los nodos i y j .

Para el problema planteado de planificación de asistentes de atención domiciliaria existen varias adaptaciones que deben ser resaltadas. En primer lugar, es conveniente aclarar que se entiende por conjunto de nodos alcanzables desde el vértice i por la hormiga k representado por N_i^k . En el caso del TSP, la única restricción existente es la de visitar todos los vértices exactamente una sola vez, por lo tanto dicho conjunto incluye a todos los vértices del grafo (al ser un grafo completo) exceptuando aquellos que ya han sido visitados con anterioridad por la hormiga k . En el caso objeto de estudio, el grafo no es un grafo completo (ver sección 4.3.1) al no definirse aristas entre aquellos vértices que no son compatibles, además existen numerosas restricciones que deben satisfacerse, por lo tanto, el conjunto de vértices alcanzables desde el vértice i para la hormiga k , estará definido por el subconjunto de vértices directamente conectados al vértice i que hayan sido visitados por anterioridad por dicha hormiga. Que un vértice j sea alcanzable desde otro vértice i no significa que dichos vértices siempre puedan estar dentro del mismo clúster, a medida que cada hormiga construye su solución al problema la compatibilidad entre servicios varía, debiéndose comprobar en cada paso. Por otra parte, y dada la definición de

distancia expuesta en 4.2.1.2, cualquier transición hacia un vértice que no cumpla con las restricciones definidas en la sección 3.4 no estará permitida, siendo su probabilidad igual a 0. La función heurística η_{ij} queda definida del siguiente modo

$$\eta_{ij} = \frac{1}{D(c_{rkn}, v_j)} \quad (4.10)$$

Donde $D(c_{rkn}, v_j)$ es la medida de distancia de acuerdo a lo definido en la sección 4.2.1 al incluir el vértice v_j dentro del clúster actual c_{rkn} que representa el clúster n encontrado por la hormiga k en la ronda r .

4.3.2.3. Finalización clúster actual

Tal y como se expuso con anterioridad, una vez que se alcanza cierta condición de parada es necesario finalizar la construcción del clúster actual c_{rkn} y comenzar con la construcción de otro nuevo. En este caso se ha optado por dar por finalizada la construcción del clúster actual, cuando desde el vértice i no existan vértices compatibles, es decir cuando $N_i^k = \emptyset$. Cuando se produce dicha situación, se crea un nuevo clúster $c_{rk(n+1)}$ guardando la información del clúster anterior c_{rkn} y realizándose la selección del nuevo vértice de inicio se realiza de acuerdo a la

4.3.2.4. Actualización de feromona y elitismo

Al igual que en la propuesta de ACS [Dorigo y Gambardella \(1997\)](#) el depósito de feromona se realiza en dos etapas, depósito local y depósito global. Esta función actualiza el depósito de feromona de las aristas existentes entre los distintos vértices, dicha función es referida en la sección 4.3.2.6 como `updatePheromone`.

Cada vez que una hormiga transita desde un vértice i al vértice j realiza una actualización local de feromona, disminuyendo la feromona, favoreciendo por lo tanto la exploración. El objetivo es hacer menos atractivos los vértices ya visitados, buscando evitar la convergencia prematura en la misma ruta, para lo cual se utiliza la siguiente ecuación.

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \tau_0 \quad (4.11)$$

donde ρ es la tasa de actualización local y τ_0 el nivel de feromona mínimo. Adicionalmente, en cada iteración, el camino de la mejor solución encontrada hasta el momento es reforzada utilizando para ello la siguiente ecuación

$$\tau_{ij}(t+1) = (1 - \alpha) \cdot \tau_{ij}(t) + \alpha \cdot \Delta\tau_{ij}(t) \quad (4.12)$$

$$\Delta\tau_{ij}(t) = \begin{cases} Q(c_{rkn}) & \text{if } c_{rkn} \in S_{bi} \\ 0 & \text{otherwise} \end{cases} \quad (4.13)$$

Donde $Q(c_{rkn})$ indica la calidad de la solución del clúster k encontrado por la n -ésima hormiga y S_{bi} representa la mejor solución encontrada hasta el momento por la colonia de

hormigas. La calidad de cada clúster se define por siguiente ecuación:

$$Q(c_{rnk}) = \frac{P_{rnk}}{W_{rnk} + T_{rnk} + P_{rnk}} * \min\left(1, \frac{W_{rnk} + T_{rnk} + P_{rnk}}{C}\right) \quad (4.14)$$

Donde C es una constante creada con el objetivo de penalizar aquellos clústers cuya duración total sea menor de cierto límite ¹. La duración total de cada clúster está definida por la suma $W_{rnk} + T_{rnk} + P_{rnk}$, donde W_{rnk} representa el tiempo total de espera, T_{rnk} tiempo total de desplazamiento y P_{rnk} el tiempo total productivo en el que se incurre a lo largo de la semana para dicho clúster. Con esta función de actualización el depósito de feromona será menor en aquellos clústers que cuya duración sea menor que dicha constante C .

4.3.2.5. Función de evaluación

Una vez que cada hormiga ha construido su solución al problema, es necesario evaluar la bondad de cada una de las soluciones encontradas. En la sección 4.3.2.6 la función “selectBest” es la encargada de medir la calidad de cada una de las soluciones encontradas y devolver aquella que minimice el coste de dicha solución. Existen numerosas formas de medir dicho coste, para el problema que nos ocupa se han definido dos. La primera de ellas solamente tiene en cuenta el número de asistentes necesarios para visitar a todos los clientes durante el periodo de planificación, mientras que la segunda realiza una combinación lineal entre el tiempo total requerido y el número de asistentes requeridos. En el primer caso la bondad de la solución vendrá dada por la ecuación (3.5), mientras que en segundo viene dado por la ecuación (3.7) las cuales se describen a continuación con el fin de facilitar la lectura.

$$F_{clust} = Clust_{obj} = |C|$$

$$F_{cost} = 14,0 * \frac{Tot_{obj}}{60} + \frac{Clust_{obj} * 554,64}{52}$$

4.3.2.6. Pseudocódigo

Una vez presentadas las distintas partes en las que se puede dividir la técnica ACS-HCSP, se detalla el pseudocódigo de la misma.

Donde C_{rn} es la partición encontrada por la hormiga n-ésima en la r-ésima ronda, la cual representa una agrupación de los servicios originales, en los cuales se respetan las restricciones del problema, quedando definida del siguiente modo.

$$C_{rn} = \bigcup_{k=0}^K C_{rnk} = S \quad (4.15)$$

De modo similar C_{rnk} se corresponde con el k-ésimo clúster encontrado por la n-ésima hormiga en la r-ésima ronda.

¹El cual toma el valor 20 para los conjuntos de servicios $S1$ y $S2$ (L-V Mañana-Tarde) y 10 para los servicios de fin de semana, expresadas ambas cantidades en horas.

Algoritmo 4.2: Pseudocódigo para ACS-HCSP y IACS-HCSP

Datos: $S = \{s_1, \dots, s_j, \dots, s_n\}$ Conjunto de servicios
Resultado: $C = \{C_1, \dots, C_K\}$ Conjunto de clústers

```

1 para  $r \leftarrow 0$  a Rounds hacer
2    $AntList \leftarrow createAntList()$  ;
3   para  $n \leftarrow 0$  a NumberOfAnts hacer
4      $Ant_{rn} \leftarrow newAnt()$  ;
5      $c \leftarrow selectInit(Ant_{rn})$  ;
6      $C_{rn} = \emptyset$  ;
7     mientras  $C_{rn} \neq S$  hacer
8        $c_{rnk} = c_{rnk} \cup c$  ;
9       mientras  $selectNextService(c_{rnk}, Ant_{rn}) \neq \emptyset$  hacer
10         $c \leftarrow selectNextService(c_{rnk}, Ant_{rn})$ ;
11         $c_{rnk} = c_{rnk} \cup c$  ;
12         $C_{rn} = C_{rn} \cup c_{rnk}$  ;
13         $c \leftarrow selectNewCluster(Ant_{rn})$  ;
14         $c_{rnk} = \emptyset$  ;
15         $k++$  ;
16      $AntList \leftarrow Ant_{rn}$ 
17  $bestAnt \leftarrow selectBest(AntList)$ ;
18  $updatePheromone(bestAnt)$ ;
19  $C \leftarrow getSolution(bestAnt)$  ;

```

4.3.3. IACS-HCSP

En la sección anterior se presentó una adaptación de la metaheurística ACO a la resolución de problemas de planificación de asistentes de atención domiciliaria, denominada ACS-HCSP, realizando solamente aquellos cambios que resultaran imprescindibles para el correcto funcionamiento de la misma. Dicha adaptación básica, presenta algunas características que pueden ser mejoradas, lo cual impide encontrar soluciones de buena calidad para las instancias de mayor tamaño tal y como se mostrará en el capítulo 5. Son varias las características que contribuyen a obtener dichos resultados, en primer lugar como se expuso en la sección 4.3.1 el grafo construido no es completo, es decir, solo se añaden aristas entre aquellos pares de vértices que sean compatibles, permaneciendo dicha definición inalterable durante todo el proceso de búsqueda. Dicha característica, junto con la definición de vértices alcanzables vecindario N_i^k expuesta en la sección 4.3.2.2, disminuyen la probabilidad de encontrar soluciones de buena calidad. Por último, el criterio de parada presentado en 4.3.2.3 induce a que las soluciones encontradas minimicen el número de clústers (ver ecuación (3.5)), con independencia de la función de evaluación utilizada. Esta característica supone un problema cuando se utilizan funciones de evaluación que tengan en cuenta otros aspectos como la propuesta por (Quintana et al., 2017) definida en la ecuación (3.7).

4.3.3.1. Construcción de soluciones

Como se mostró anteriormente, cada hormiga construye una solución válida al problema a través de su recorrido del grafo. Partiendo del grafo que se mostró en la figura 4.5, en la figura 4.6 se enumeran los posibles recorridos que una hormiga podría realizar y las soluciones que se obtendrían. Como se puede apreciar en dicha imagen, siempre que el recorrido de la hormiga comienza por el vértice que representa al servicio S_2 la solución obtenida requiere un total de dos clústers, ya que entre los vértices S_1 y S_3 no existe arista en el grafo que se creó inicialmente, permaneciendo dicha definición inmutable a lo largo del proceso de búsqueda. Es precisamente en dicha inmutabilidad donde reside el problema como se mostrará en esta sección.

Si la construcción de la solución comienza por los vértices S_1 o S_3 , la solución obtenida será $[C_1 = \{S_1, S_2, S_3\}]$ requiriendo únicamente la utilización de un clúster, en el cual los tres servicios serán realizados en el mismo clúster. En la figura 4.7 se puede apreciar el recorrido detallado de una hormiga que comienza la construcción de su solución por el vértice S_1 .

Si por el contrario, el recorrido de la hormiga comienza en S_2 , la solución obtenida requeriría dos clústers y por lo tanto dos asistentes de atención domiciliaria. En la figura 4.8 se muestra el recorrido detallado que realizaría una hormiga si como vértice inicial eligiera el servicio S_2 , el cual se examina a continuación. Inicialmente, la memoria de trabajo de la hormiga k no contiene ningún vértice seleccionado, al visitar en primer lugar el vértice S_2 , se comienza con la construcción del primer clúster C_1 , al cual se añade el servicio S_2 . Una vez que la hormiga k se encuentra en S_2 , su vecindario queda definido por $N_{S_2}^k = \{S_1, S_3\}$, suponemos que de estos dos vértices la hormiga selecciona de modo probabilístico el vértice S_1 , el cual vuelve a incorporar al clúster de trabajo actual C_1 quedando este $C_1 = \{S_2, S_1\}$. Una vez que la hormiga se encuentra en S_1 y de acuerdo a lo

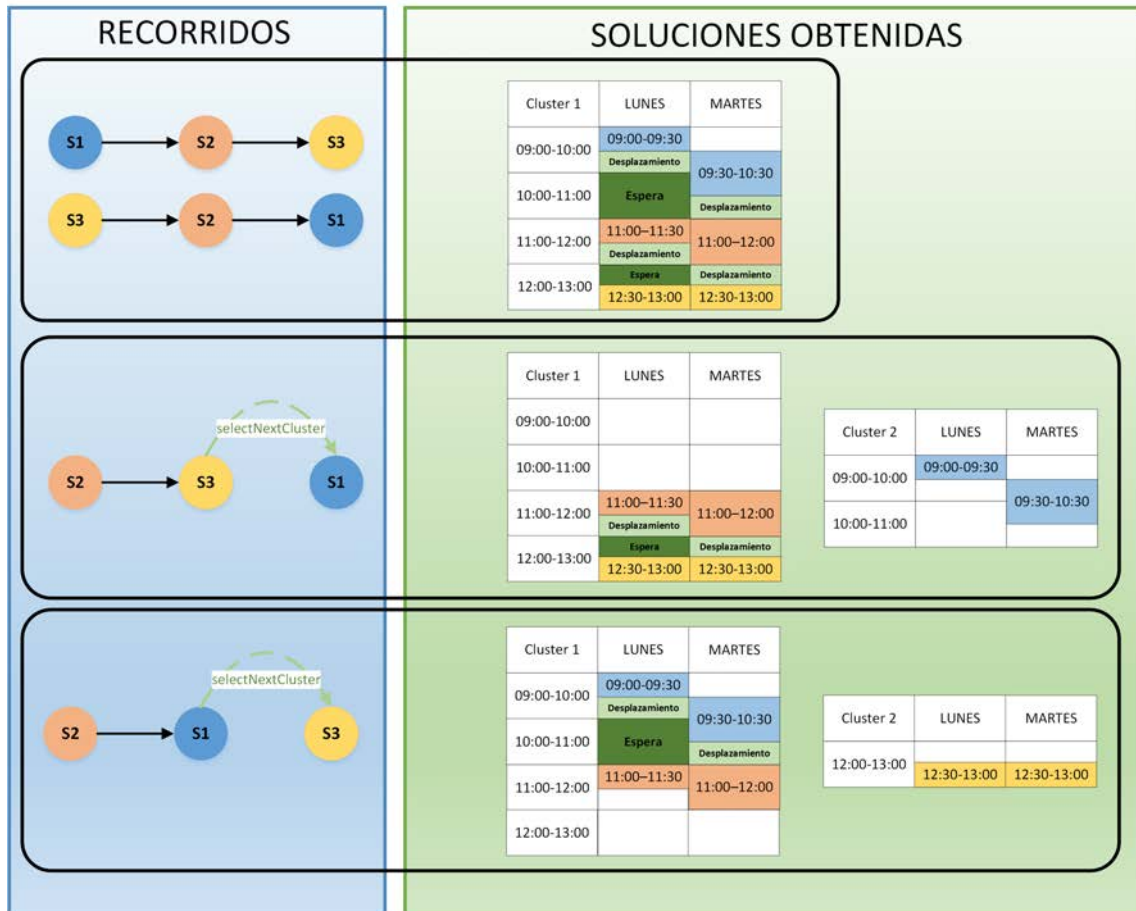
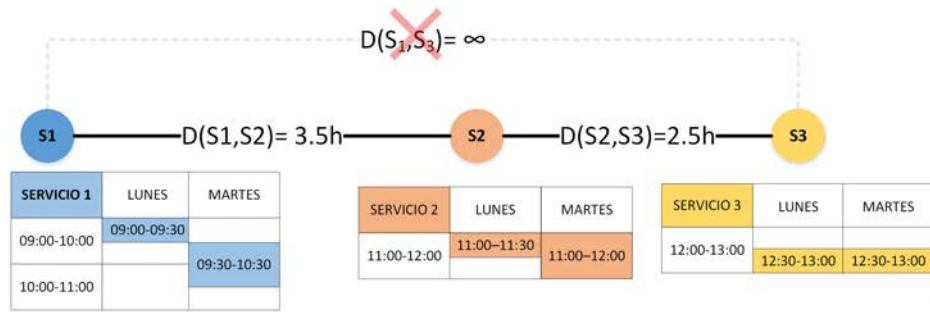


Figura 4.6: Ejemplo en el que se muestran las distintas soluciones que se obtendrían para un problema con tres servicios S_1, S_2, S_3 , enumerando los posibles caminos recorridos por las hormigas.

expuesto en la sección 4.3.2.2 el conjunto de nodos alcanzables desde el vértice S_1 , $N_{S_1}^k = \emptyset$, ya que S_2 ya ha sido visitado y S_3 no es alcanzable desde S_1 . Por lo tanto y de acuerdo a lo expuesto en 4.3.2.3 se comenzará un nuevo clúster C_2 , seleccionando finalmente el único vértice que queda sin visitar S_3 y añadiéndolo al clúster actual $C_2 = \{S_3\}$, obteniendo la solución $[C_1 = \{S_2, S_1\}, C_2 = \{S_3\}]$. Es importante resaltar, que si en lugar de visitar el

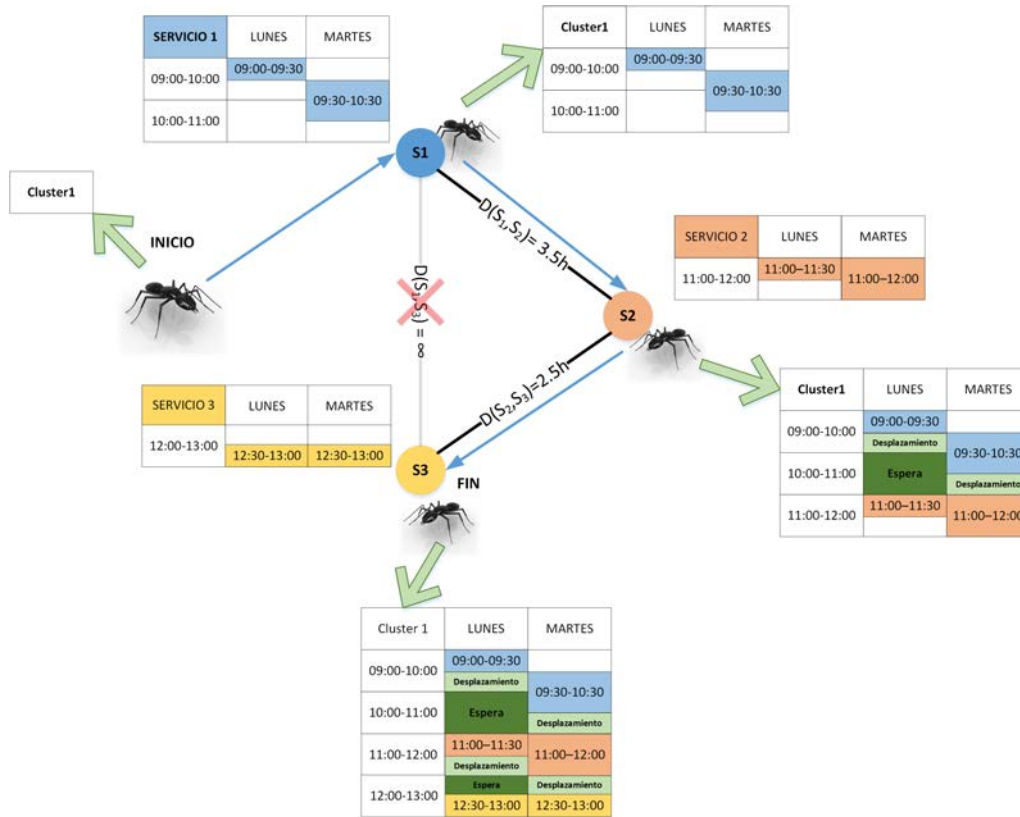


Figura 4.7: Ejemplo de recorrido detallado que realizaría una hormiga k si visitara los vértices en el siguiente orden S_1, S_2, S_3 , lo cual daría lugar a la solución $[C_1 = \{S_1, S_2, S_3\}]$.

vértice S_1 después de S_2 se hubiera visitado S_3 , el resultado final habría sido muy similar $[C_1 = \{S_2, S_3\}, C_2 = \{S_1\}]$, no variando el número de clústers requeridos.

4.3.3.2. Vértices alcanzables

En la sección 4.3.2.2 se definió el conjunto de vértices alcanzables desde un vértice i para una hormiga k , como todos aquellos vértices directamente conectados con el vértice i , que no hayan sido visitados anteriormente por la hormiga k , estando representado por N_i^k . Dicha definición hace que en determinadas situaciones, como la mostrada en la figura 4.8, las soluciones obtenidas presenten una peor calidad, por ello se propone ampliar dicha definición de vecindario.

El problema de la definición original reside en cómo se crean las aristas entre vértices. Dichas aristas, las cuales representan la compatibilidad entre servicios, se definen inicialmente junto con la creación del grafo y permanecen inalteradas durante todo el proceso de búsqueda de soluciones. Dicha definición, solo tiene en cuenta si dos servicios son compatibles entre sí, es decir, si por sí mismos podrían un clúster, el cual estuviera formado solamente por los dos vértices, de tal modo que el clúster resultante respetase las restricciones. Está aproximación, aunque válida, no tiene en cuenta el camino que ha realizado la hormiga k para la formación del clúster actual, el cual puede alterar la

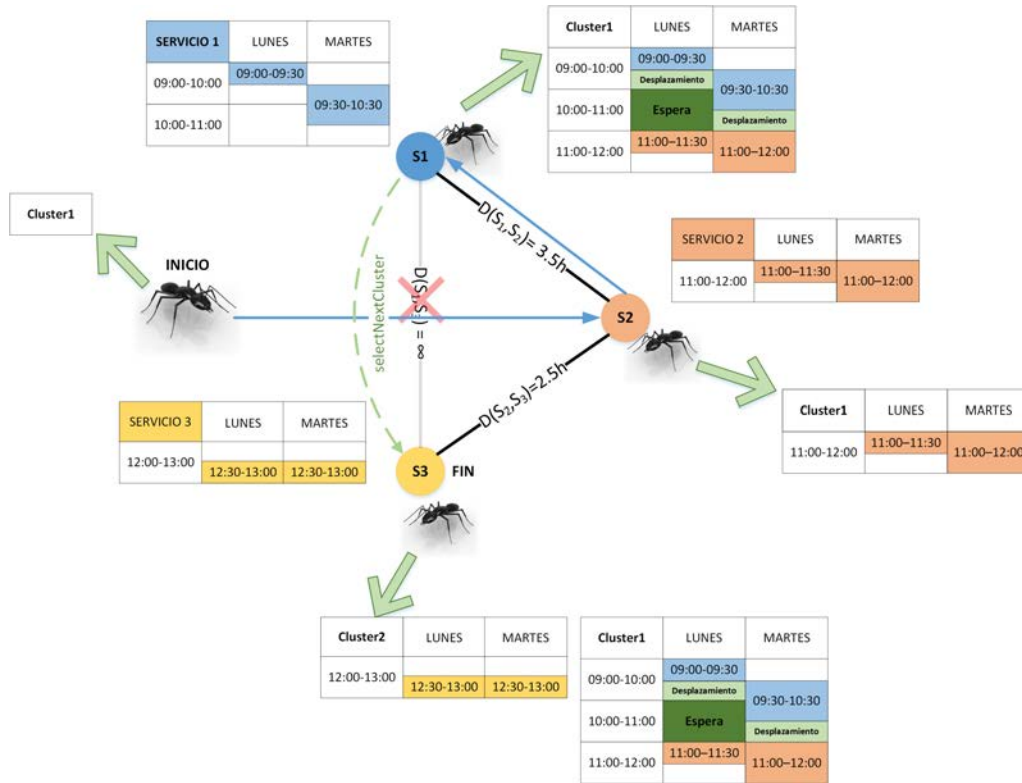


Figura 4.8: Ejemplo de recorrido detallado que realizaría una hormiga k si visitara los vértices en el siguiente orden S_2, S_1, S_3 . Tal y como se puede apreciar al visitar el vértice S_1 , $N_{S_1}^k = \emptyset$ por lo que se comenzaría con la formación de otro clúster, obteniendo la solución $[C_1 = \{S_2, S_1\}, C_2 = \{S_3\}]$.

definición original. En el caso de la figura 4.8, cuando la hormiga visita el vértice S_1 el nodo S_3 no es directamente alcanzable, ya que por sí solos los vértices S_1 y S_3 no podrían formar un clúster válido. Sin embargo, puesto que la hormiga antes de visitar S_1 visitó el vértice S_2 , el cual si está directamente conectado con S_3 , S_3 debería pertenecer al conjunto de vértices alcanzables desde S_1 para dicha hormiga. Hay que remarcar que tal y como se expuso en la sección 4.3.2.2 pertenecer al conjunto de vértices alcanzables simplemente implica que la distancia a dicho vértice desde el clúster actual será evaluada y en caso de que cumpla con las restricciones tendrá cierta posibilidad de ser visitado de acuerdo a la ecuación (4.9).

Existen varias posibilidades para conseguir que el vértice S_3 sea alcanzable desde S_1 , la primera y más sencilla sería crear un grafo completo, esta aproximación es idéntica a la utilizada en el TSP por [Dorigo y Gambardella \(1997\)](#). Dicha aproximación, aunque sencilla, presenta varios inconvenientes ambos originados por la alta dimensionalidad de las instancias a resolver, algunas con más de 10.700 servicios ver 3.3.3. El primero de ellos es el tiempo de evaluación para transitar de un vértice a otro, si en las primeras iteraciones cada hormiga ha de elegir entre un número elevado de vértices el coste computacional aumenta. El segundo, es la cantidad de memoria requerida, para la instancia de mayor

tamaño con 10.709 servicios el número total de aristas asciende a casi 60 millones. Además de la propia arista, se debería almacenar información sobre los niveles de feromona. En lugar de tomar esta aproximación, se propone ampliar el concepto de nodos alcanzables del siguiente modo.

Sea $c_{rnk} = \{S_0 \dots S_n\}$ el k -ésimo clúster de la n -ésima hormiga en la r -ésima ronda en su estado, donde $\{S_0 \dots S_n\}$ representan el conjunto de servicios que han sido visitados con anterioridad para la formación de dicho clúster, el conjunto extendido de vértices alcanzables desde el vértice i queda definido del siguiente modo.

$$\hat{N}_i^k = \bigcup_{i=0}^n N_{S_i}^k \cup N_i^k \setminus c_{rnk} \quad \forall S_i \in c_{rnk} \quad (4.16)$$

donde c_{rnk} es el clúster actual para la hormiga k , el cual contiene el conjunto de vértices visitados con anterioridad. Es decir, el conjunto extendido de vértices alcanzables para el vértice i representado por \hat{N}_i^k , se define como las uniones de los distintos conjuntos de vértices alcanzables para todos los vértices visitados por la hormiga k en la formación del clúster actual c_{rnk} , excluyendo aquellos vértices que ya pertenecen a dicho clúster. Con esta nueva definición la calidad de las soluciones obtenidas mejora sensiblemente. Continuando con el ejemplo del recorrido ilustrado en la figura 4.8, supongamos que la hormiga se halla acaba de visitar el servicio S_1 habiendo visitado previamente el servicio S_2 , con lo que su clúster actual será $C_1 = S_2, S_1$ con dicha configuración se puede observar como $\hat{N}_{S_1}^k = S_3$ ya que

$$\hat{N}_{S_1}^k = N_{S_2}^k \cup N_{S_1}^k \setminus C_1 = \{S_1, S_3\} \cup \{S_2\} \setminus \{S_2, S_1\} = S_3$$

Con esta nueva definición de \hat{N}_i^k , las soluciones obtenidas partiendo desde cualquiera de los tres vértices S_1, S_2, S_3 sería $[C_1 = \{S_1, S_2, S_3\}]$ aumentando la calidad de las soluciones obtenidas tal y como se mostrará en el capítulo 5.

4.3.3.3. Criterio de parada

Tal y como se expuso en la sección 4.3.2.3 el criterio de parada para dar por concluido la construcción del clúster actual c_{rnk} y comenzar con la construcción de uno nuevo, consiste en la no existencia de más vértices alcanzables desde el vértice actual i , es decir, cuando $N_i^k = \emptyset$. Este criterio de parada provoca que las soluciones encontradas se centren solo en el objetivo de minimizar el número de asistentes requeridos definido por la ecuación (3.5). Dependiendo del negocio y del contexto, existen casos en los que el objetivo Clust_{obj} es priorizado sobre el tiempo total requerido Tot_{obj} , este es el caso de la ecuación expuesta en (3.7) la cual utiliza una combinación lineal de ambos objetivos [Quintana et al. \(2017\)](#). En este caso, el utilizar como criterio de parada $N_i^k = \emptyset$ hace que las soluciones encontradas presenten una peor calidad, al no existir mecanismos mediante los cuales la construcción del clúster actual se pueda detener al descender de cierto nivel de eficiencia. Cuando se trata de minimizar el primer objetivo Clust_{obj} de forma aislada, el tiempo total requerido Tot_{obj} no es relevante, por lo que la eficiencia de cada ruta pasa a un segundo plano siempre y cuando se reduzca el número de asistentes requeridos. La eficiencia de un clúster cada

una de las soluciones encontradas viene dada por la ecuación

$$E_{rnk} = 1 - \min \left(1, \frac{W_{rnk} + T_{rnk}}{P_{rnk}} \right) \quad (4.17)$$

Donde P_{rnk} es el tiempo total productivo incurrido a lo largo de la semana, T_{rnk} representa el tiempo total de desplazamiento y W_{rnk} el tiempo total de espera, estando referidos todos estos parámetros al k -ésimo clúster encontrado por la n -ésima hormiga durante la r -ésima ronda.

Si se utiliza el criterio de parada expuesto en 4.3.2.3, la distancia total Tot_{obj} tenderá a aumentar, mientras que el número de asistentes requeridos, al igual que la eficiencia de las soluciones encontradas, tenderán a disminuir con total independencia de la función de evaluación utilizada. Es decir, aun cuando el número de asistentes no es tenido en la función de evaluación, el número de clústers tenderá a disminuir debido al criterio de parada. Por lo tanto, dicho criterio de parada debe ser modificado con el objetivo de permitir obtener soluciones de buena calidad con independencia de la función de evaluación definida.

Idealmente y dependiendo exclusivamente de la función de evaluación, la colonia de hormigas debería ser capaz de minimizar distintos tipos de objetivos, ya sea minimizar el número de asistentes o una combinación entre el número de asistentes y la distancia total recorrida, es en este punto en el que la definición de eficiencia dada para una solución completa (4.17) juega un papel determinante. Dicha definición se puede utilizar para medir la eficiencia de cada uno de los k clústers construidos por cada hormiga n en cada ronda r c_{rnk} . Inicialmente cuando c_{rnk} está formado por un único servicio, la eficiencia será $E(c_{rnk}) = 1$ ya que no existen tiempos de desplazamiento ni de espera. A medida que cada hormiga va visitando vértices en el grafo, dicha eficiencia decrecerá hasta que no existan más vértices compatibles con el clúster actual, momento en el que comenzará con la construcción de otro clúster 4.3.2.3. La propuesta de esta investigación a fin de prevenir dicho comportamiento, es definir una eficiencia objetivo para cada hormiga n en cada ronda r , representada por E_{rn}^{obj} , de tal modo que cuando la eficiencia clúster en construcción, $E(c_{rnk})$ esté cerca de dicha eficiencia objetivo la probabilidad de finalizar el clúster actual aumente. Dicha probabilidad vendrá dada por la siguiente función logística.

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}} \quad (4.18)$$

Donde x_0 es el punto donde la función alcanza su valor medio, L es el valor máximo de la función y k la pendiente de la curva, en la figura 4.9 puede verse una representación de dicha función con distintos valores de k .

Para adaptar dicha función logística y poder utilizarla como criterio de parada, el valor de x vendrá dado por la diferencia entre la eficiencia objetivo de la hormiga y la eficiencia del clúster en construcción, es decir $\Delta E = E_{rn}^{obj} - E(c_{rnk}) * 100$, quedando la probabilidad de parada definida del siguiente modo

$$f(\Delta E) = \frac{1}{1 + e^{-0,5(\Delta E)}} \quad (4.19)$$

Tal y como se puede ver en la imagen 4.9 cuando $\Delta E \leq -10$, es decir cuando la eficiencia del clúster sea un 10 % más eficiente que la eficiencia objetivo, la probabilidad de

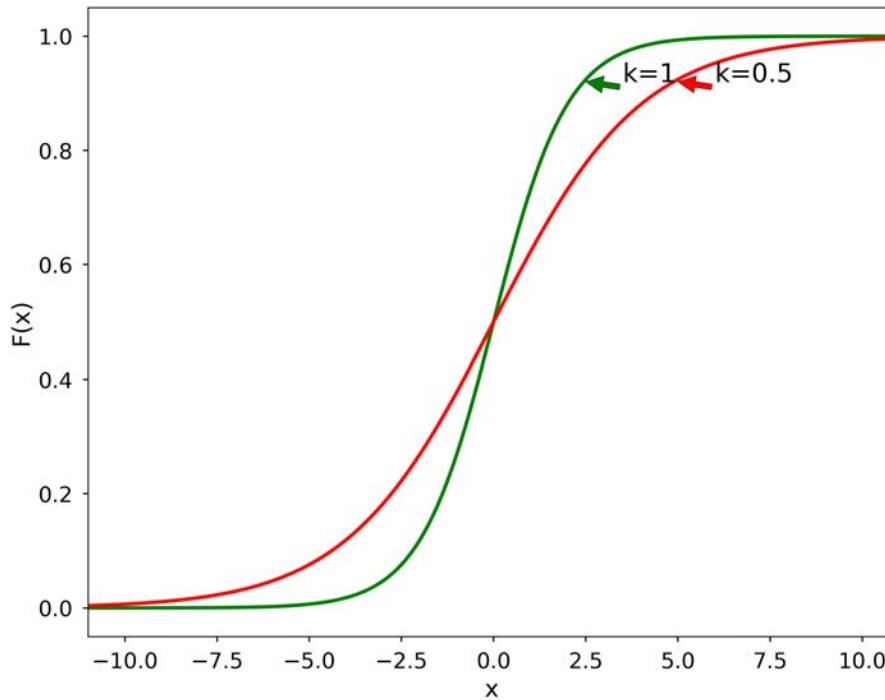


Figura 4.9: Ejemplo de función logística con valores $L = 1$, $x_0 = 0$ y valores de $k=[1, 0.5]$

parada es casi nula, de modo análogo con valores $\Delta E \geq 10$ la probabilidad de parada es cercana a 1. En cada iteración, cada hormiga utilizará dicha probabilidad para determinar si debe continuar con la construcción del clúster actual, o si por el contrario debe finalizarlo y comenzar la construcción de uno nuevo. Otro aspecto no mencionado hasta el momento, es cómo se calcula el valor de la eficiencia objetivo para cada hormiga n en cada una de las iteraciones r , valor que vendrá dado por la ecuación

$$E_{rn}^{obj} = E_{r-1}^{obj*} + \gamma' \quad (4.20)$$

Donde E_{r-1}^{obj*} representa la eficiencia objetivo de la hormiga que ha obtenido mejores soluciones hasta la iteración $r - 1$ y γ' es una variable aleatoria uniformemente distribuida entre $[-\gamma, \gamma]$. Cabe destacar que en la primera iteración el valor de E_{0n}^{obj} tomará un valor aleatorio distribuido entre $[0, 1]$.

Capítulo 5

Experimentación

Ninguna cantidad de experimentación puede probar definitivamente que tengo razón; pero un solo experimento puede probar que estoy equivocado.

Albert Einstein

Resumen: En este capítulo se realiza una exposición detallada de los resultados obtenidos por las técnicas propuestas en el capítulo 4. En primer lugar, se presentan las distintas instancias que serán objeto de experimentación, exponiendo sus características principales, así como un análisis descriptivo de las mismas. Posteriormente se presentan las distintas técnicas que serán objeto de experimentación, la configuración de las mismas, así como los resultados obtenidos por cada técnica. En primer lugar, se presentan los resultados obtenidos con el método jerárquico de Ward adaptado al problema de planificación de asistentes de atención domiciliaria, el cual ha servido como línea base con la que comparar las dos técnicas basadas en ACO. Posteriormente se presentan la configuración y resultados obtenidos tanto por la técnica ACS-HCSP como por la versión mejorada IACS-HCSP. Finalmente se realiza un estudio comparativo entre las distintas técnicas presentadas y se analiza su rendimiento respecto a otras soluciones existentes como son la solución utilizada por la compañía y las heurísticas propuestas por [Quintana et al. \(2017\)](#).

5.1. Introducción

Una de las características más significativas del conjunto de datos utilizados para validar las técnicas propuestas en el capítulo 4, es sin duda la alta dimensionalidad de las instancias. Como se expusiera en el capítulo 2, las instancias presentes en el estado del arte y utilizadas en la resolución de problemas de planificación de asistentes de atención domiciliaría, en pocas ocasiones superan los cientos de clientes [Braekers et al. \(2016\)](#), no habiéndose encontrado hasta la fecha publicaciones relevantes en las que se llegue a resolver instancias con más de 1000 clientes. En nuestro caso y como se mostrará a continuación existen instancias con hasta 10.709 servicios que deben ser planificadas semanalmente. Tal y como se presentó en la sección 3.3.3, existen un total de 13.344 servicios, los cuales deben ser planificados de lunes a domingo entre las 07:00 y las 22:00. Dichos servicios han de ser agrupados dentro de ciertos horarios preestablecidos por la compañía. Atendiendo a la hora del día, se distinguen turnos en horario de mañana o de tarde. De modo análogo, atendiendo al día de la semana, se distinguen horarios de fines de semana y horarios de días de diario. Estas características hacen que el total de los 13.344 servicios y las más de 29.000 tareas hayan sido divididas en distintos turnos de trabajo o grupos de servicios del siguiente modo.

La primera agrupación de servicios atendiendo al horario en el que se desarrollan los servicios será denominada agrupamiento tipo A y está subdividida en los siguientes grupos de instancias que serán resueltas de modo independiente.

- Grupo 1: Servicios y tareas realizados de lunes a viernes entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 22.982 tareas y agrupadas en 10.709 servicios. Esta es sin duda la instancia de mayor dimensionalidad, presentando mayor dificultad en su resolución y requiriendo un coste computacional más elevado.
- Grupo 2: Servicios y tareas realizados de lunes a viernes entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 3.812 tareas y agrupadas 1.159 servicios.
- Grupo 3: Servicios y tareas realizados de sábado a domingo entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 1.576 tareas distribuidas en 1.070 servicios.
- Grupo 4: Servicios y tareas realizados de sábado a domingo entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 664 tareas, distribuidas en 406 servicios.

Una variante del agrupamiento anterior y que también será utilizado en esta fase experimental consiste en considerar los servicios prestados durante el fin de semana como uno solo, refiriéndonos a dicho agrupamiento como B, el cual queda compuesto del siguiente modo:

- Grupo 1: Servicios y tareas realizados de lunes a viernes entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 22.982 tareas y agrupadas en 10.709 servicios. Esta es sin duda la instancia de mayor dimensionalidad, presentando mayor dificultad en su resolución y requiriendo un coste computacional más elevado.

- Grupo 2: Servicios y tareas realizados de lunes a viernes entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 3.812 tareas y agrupadas 1.159 servicios.
- Grupo 3*: Servicios y tareas realizados de sábado a domingo entre las 07:00 y las 22:00 horas ambas incluidas, formado por un total de 2.240 tareas distribuidas en 1.476 servicios.

Cada uno de estos grupos será resuelto de manera independiente, constituyendo por sí mismo una única instancia del problema a resolver. Tal y como se puede apreciar en la imagen 5.1 el Grupo 1 representan más del 80 % de los servicios y algo más del 79 % de las tareas, siendo la instancia que presenta una mayor dificultad en su resolución.

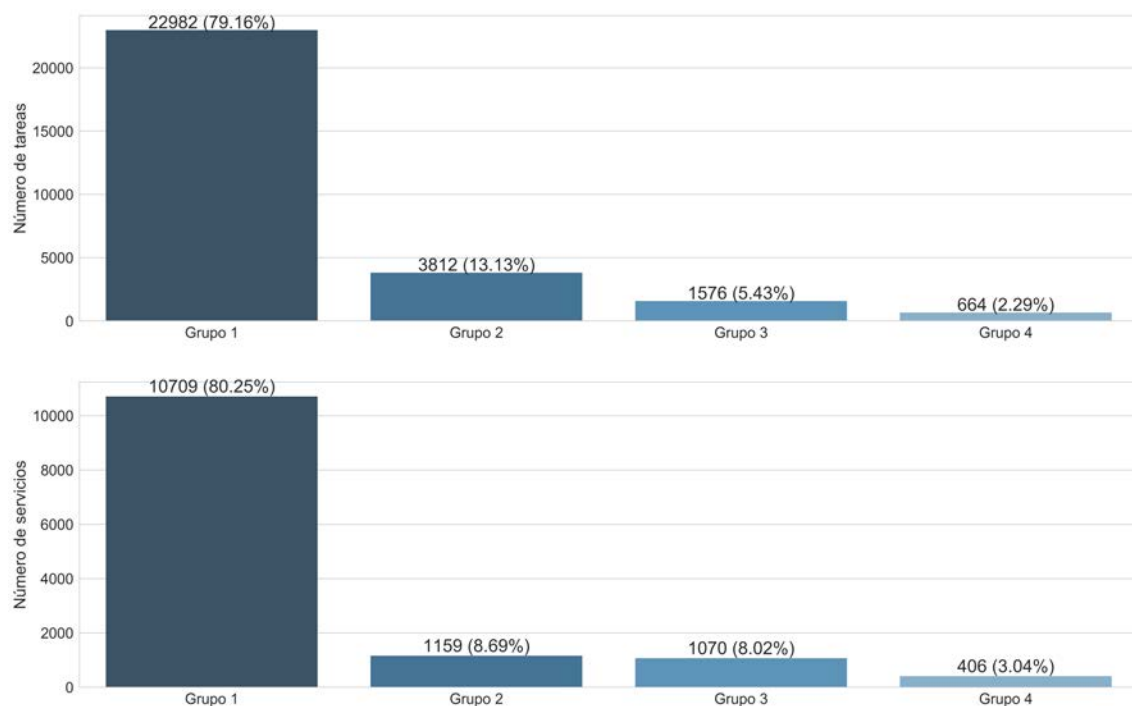


Figura 5.1: Gráfico en el que se representa el número total de tareas por grupo de servicios y el número total de servicios presentes en cada uno de dichos grupos.

Además del número de tareas y servicios, otro aspecto relevante a estudiar es la distribución horaria de las tareas, así como la duración de las mismas, ya que dichas características junto con las restricciones del problema determinaran en gran medida las características de las soluciones encontrada por cada una de las técnicas de resolución propuestas. En la figura 5.2 se muestran sendos histogramas con las horas de inicio y fin de todas las tareas. Tal y como se aprecia existe una mayor demanda de tareas durante las primeras horas del día, de hecho, entre las 08:00 y las 10:00 comienzan más del 45 % del total de tareas requiriendo un mayor número de asistentes en dichas horas.

Otro aspecto que caracteriza cada una de las instancias, es la duración de las tareas. En la figura 5.3 se muestra un diagrama de cajas con la duración de las mismas para cada

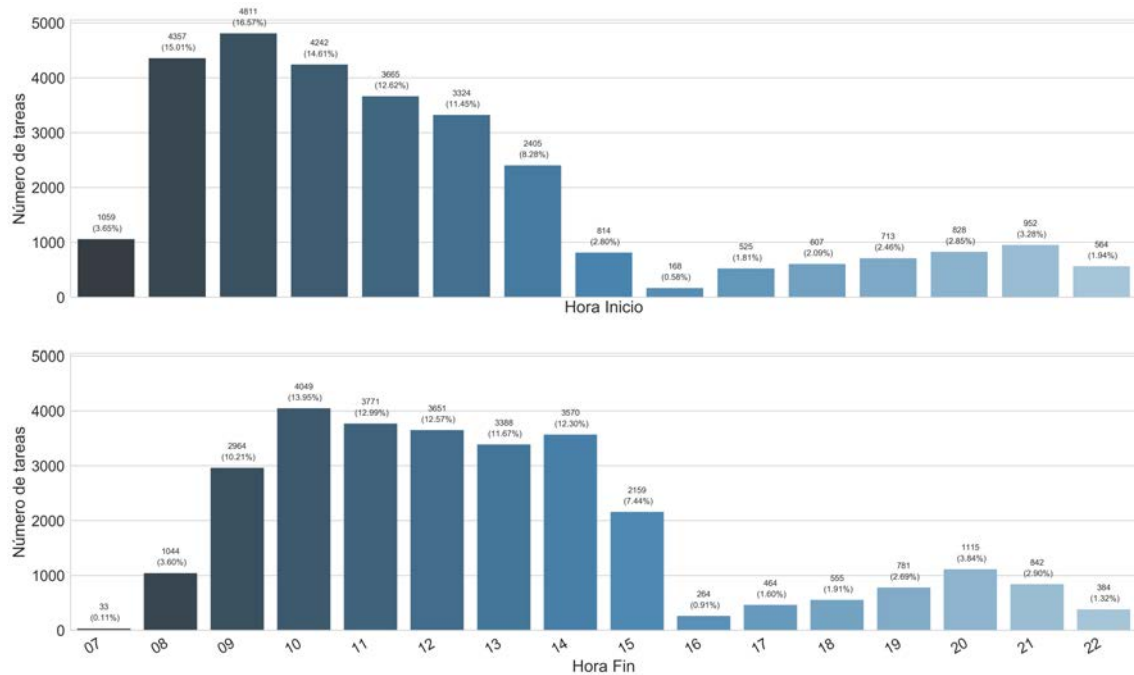


Figura 5.2: Gráfico en el que se representa la hora de comienzo de las tareas y la hora de fin por grupo de servicios.

uno de los días de la semana, distinguiendo por grupo de servicios. En la misma imagen se puede apreciar como la duración media de las tareas que pertenecen al Grupo 1 es mayor que la del resto, no apreciándose una diferencia significativa en la duración del resto de grupos. Respecto al resto de grupos se aprecia una duración media similar. Además, se observa como existen numerosas tareas cuyas duraciones pueden ser consideradas como atípicas, por ejemplo, para el Grupo 1 y para el día de la semana miércoles, existe una tarea cuya duración es de 8 horas.

El diagrama de cajas anterior da una idea de la duración de las tareas, sus percentiles y los valores atípicos de estas. A pesar de ello, no nos aporta información sobre cómo se distribuyen dichas duraciones, en la figura 5.4 se muestra dicha información utilizando un gráfico de “violín”. En dicho gráfico el área de cada gráfico indica la densidad de tareas existentes con dicha duración, se puede observar claramente como las tareas que se desarrollan de lunes a viernes presentan una distribución muy similar, existiendo una mayor densidad de aquellas tareas cuya duración es de 120 y 60 minutos. Por el contrario, la distribución de duraciones de las tareas que se desarrollan durante el fin de semana, presentan una mayor densidad en duraciones entre 30 y 60 minutos.

A pesar de que el gráfico anterior muestra la distribución de probabilidad para la duración de cada tarea, no distingue entre los distintos grupos, en la gráfica 5.6 se muestran dichas densidades disgregadas por turno de trabajo, mostrándose claramente como las tareas que se desarrollan a partir de las 14:00, es decir aquellas pertenecientes a los grupos {Grupo 2, Grupo 4} presentan una menor duración. Además se puede observar cómo la distribución dentro de cada grupo es bastante homogénea.

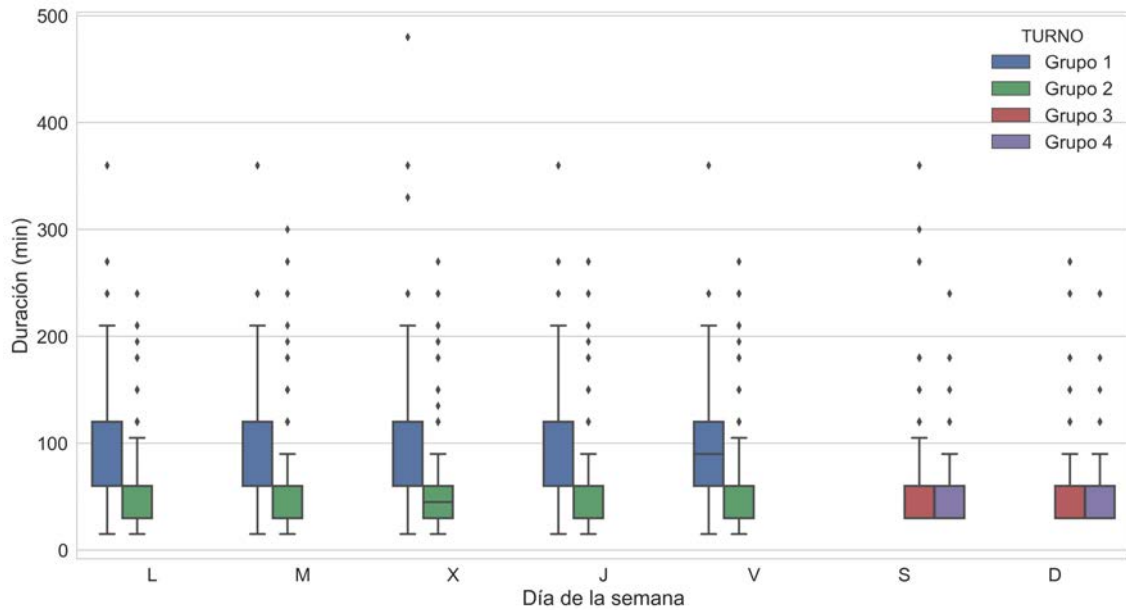


Figura 5.3: Diagrama de cajas en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo por cada turno de trabajo. Se observa una duración uniforme dentro de cada grupo de servicios, existiendo algunos valores atípicos representados con un rombo.

En la figura 5.7 se combinan la hora de inicio de la tarea, el día de la semana y el grupo al que pertenece dicha tarea detallándose la duración de cada tarea. Como se puede observar la única hora del día en la que comienzan tareas que pertenecen a distintas instancias es el periodo que abarca desde las 14:00 a las 15:00.

Como se expusiera en capítulos anteriores el problema abordado presenta dos grandes grupos de restricciones las espaciales y las temporales. Atendiendo a las primeras, la distribución geográfica de las mismas es un aspecto fundamental y no abordado hasta el momento. En las figuras 3.4 y 3.5 del capítulo 3, se mostraron diversas imágenes en las que se representaba el conjunto de servicios y el conjunto total de tareas geolocalizados. A pesar de que dichas imágenes son útiles para entender la distribución geográficas de las tareas, no aportan una información muy precisa, por ello en la imagen 5.8 se ofrece un mayor nivel de detalle. En dicha imagen cada servicio se ha representado por un punto y se ha coloreado de acuerdo a su latitud con el fin de facilitar la ubicación de la imagen ampliada. En la imagen también se muestra de un modo claro la alta densidad de los servicios, así como la imposibilidad de realizar algún tipo de agrupamiento geográfico con el fin de dividir el problema en instancias más pequeñas.

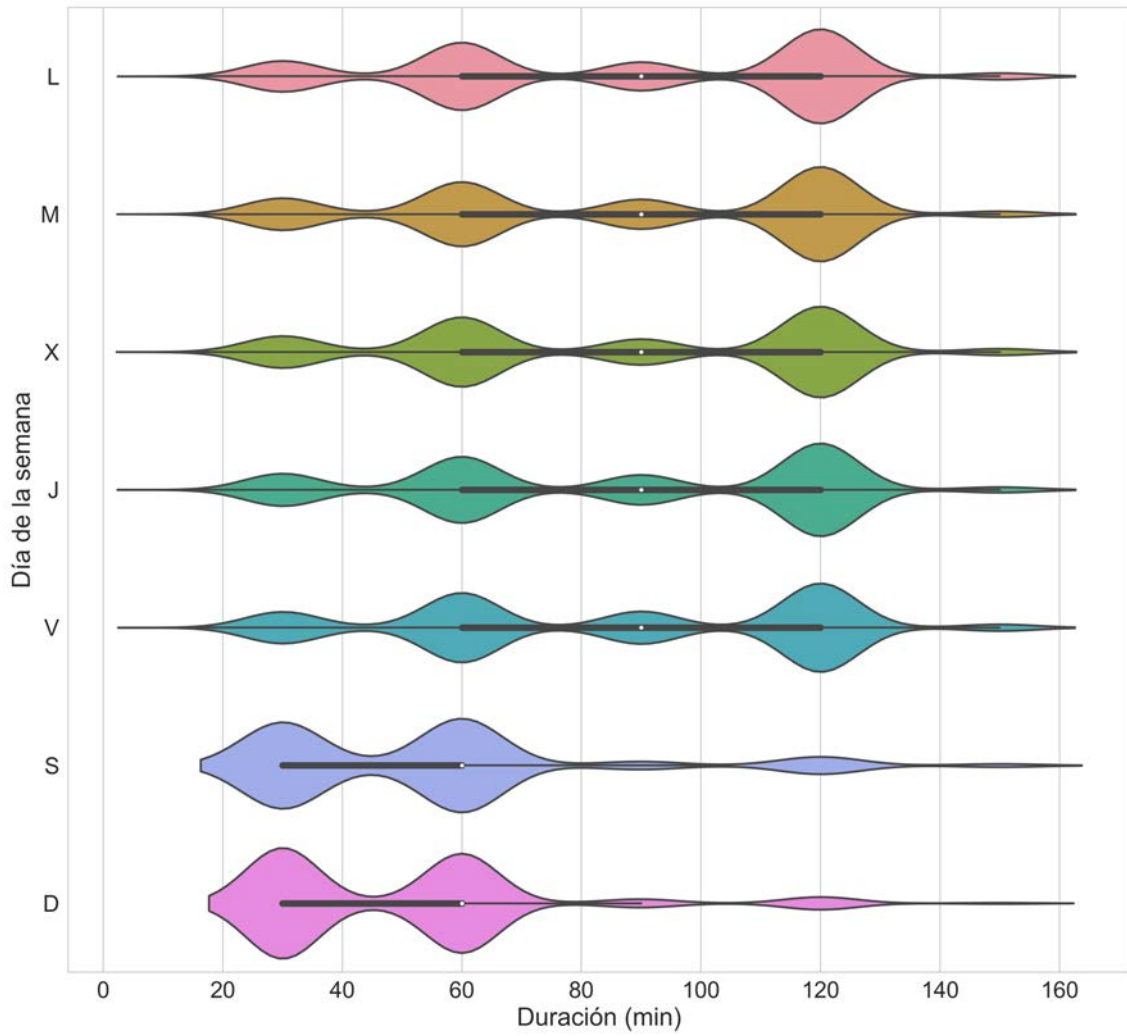


Figura 5.4

Figura 5.5: Gráfico de “violín” en el que se representa la duración de cada tarea, para cada día de la semana. Filtrándose todas aquellas tareas cuya duración sea superior de 150 minutos.

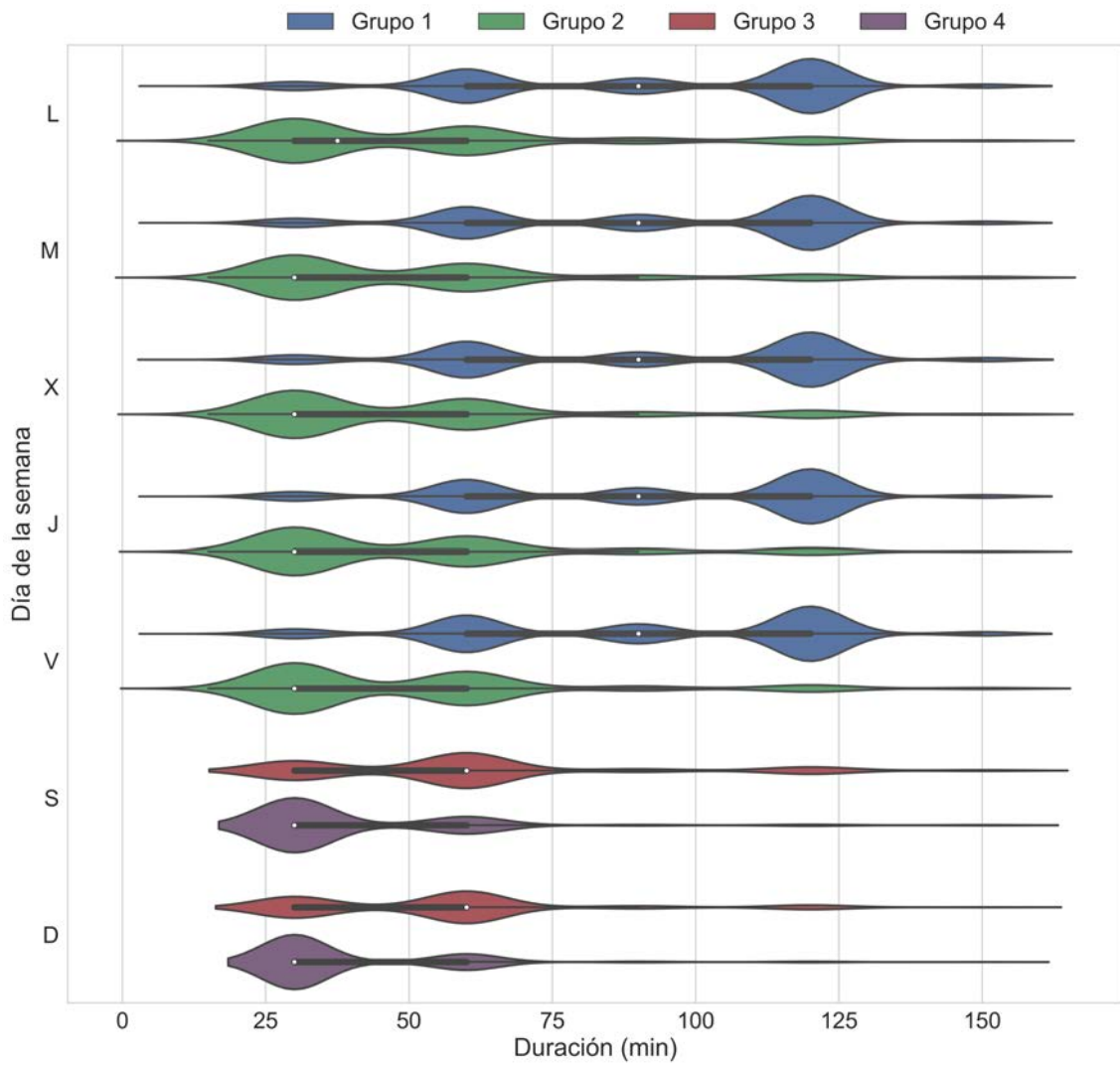


Figura 5.6: Gráfico de “violín” en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo para cada grupo de servicios. Filtrándose todas aquellas tareas cuya duración sea superior de 150 minutos.

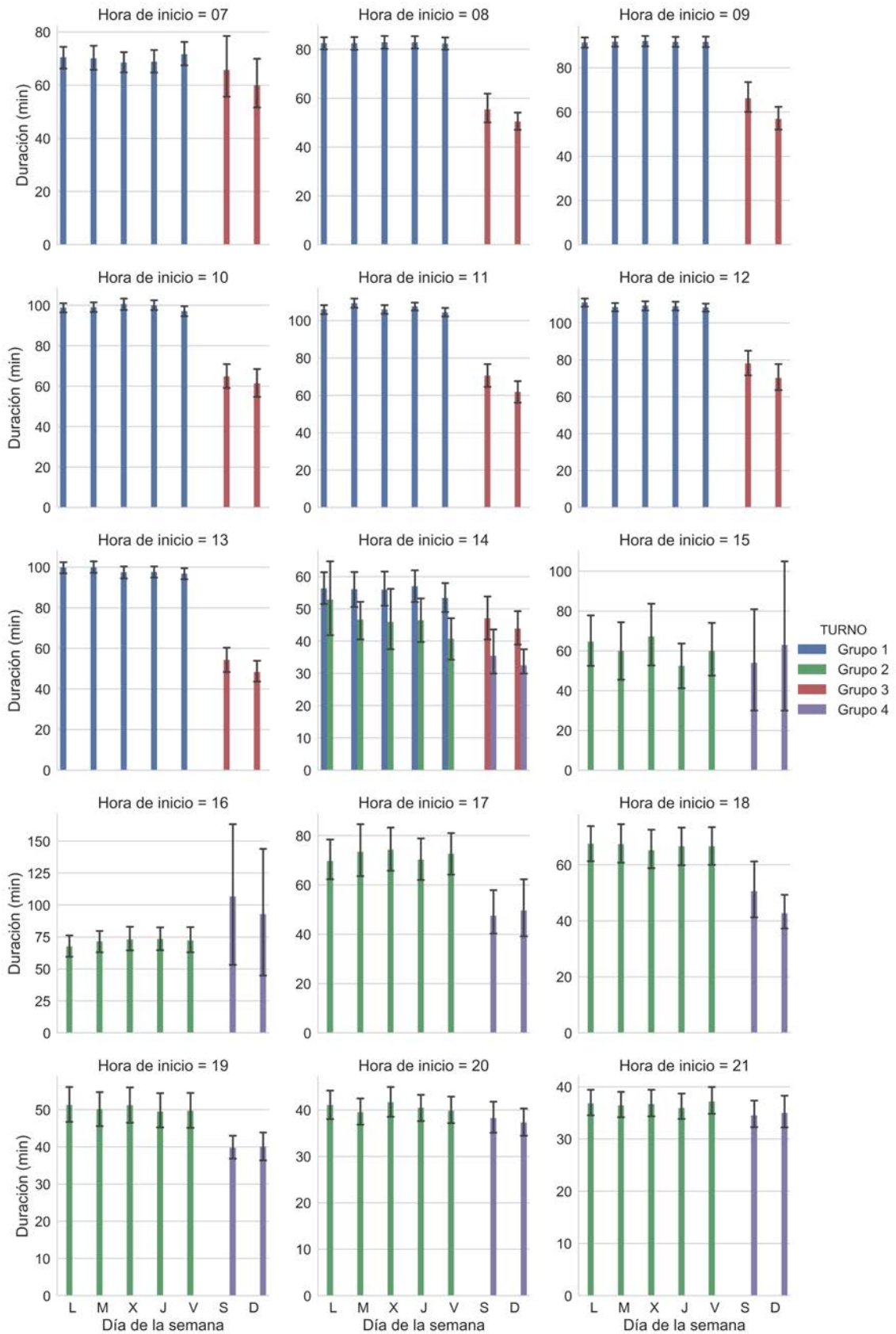


Figura 5.7: Gráfico en el que se representa la duración de cada tarea, para cada día de la semana, distinguiendo por hora de inicio de la tarea y por cada grupo de servicios.

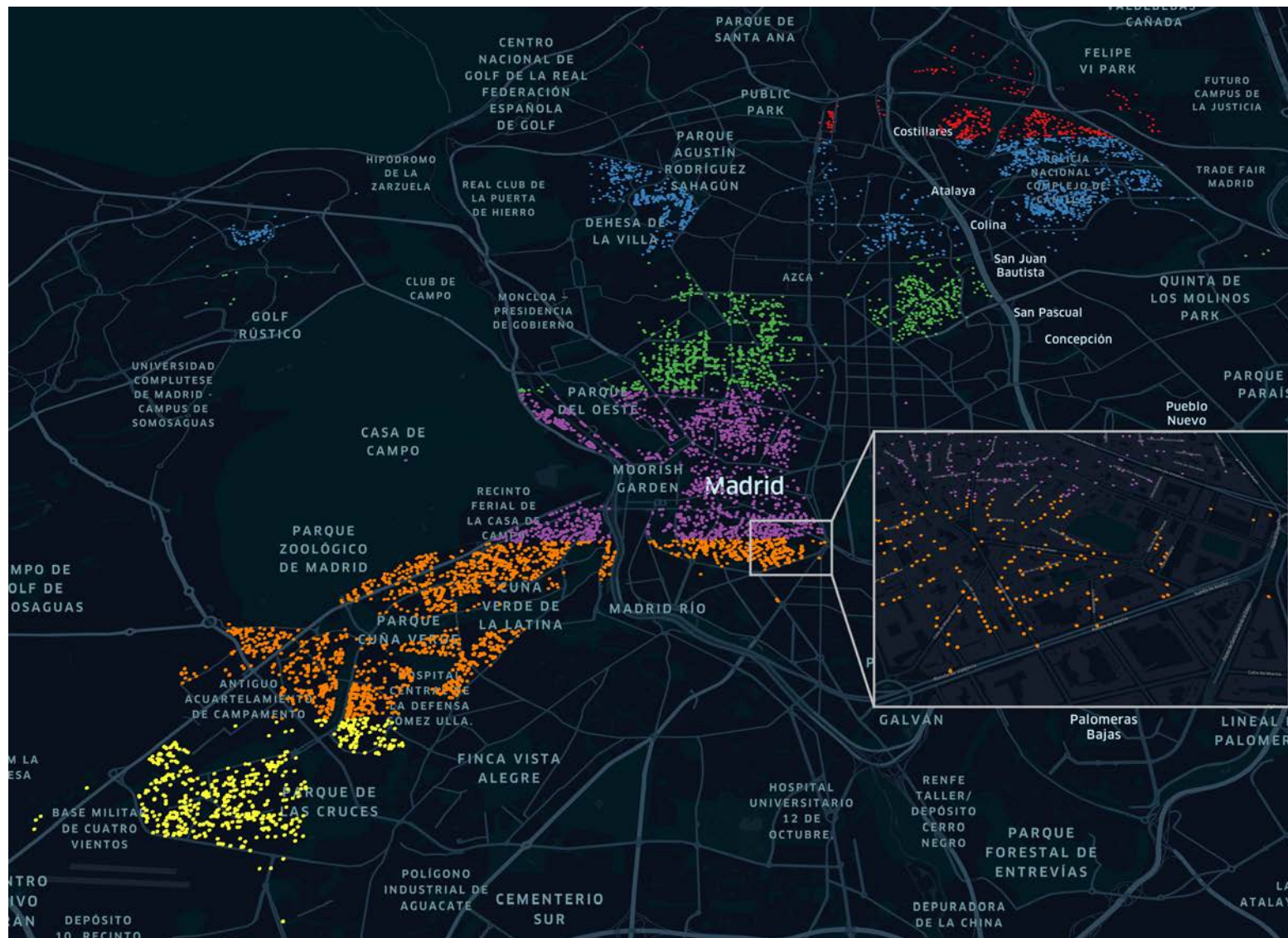


Figura 5.8: Imagen en la que se muestra el total de 13.344 servicios, mostrándose en detalle una zona seleccionada.

Otro aspecto importante a la hora de analizar las distintas instancias es el tiempo total requerido por cada servicio y la distribución geográfica del mismo. Una alternativa visual para representar la posición geográfica de cada tarea y la duración de las misma es utilizar un mapa de calor. Utilizando el API de Google Maps [Google Maps APIs \(2016c\)](#), en la imagen 5.9 se muestra un mapa de calor con pesos, en él, se muestran la posición de todas las tareas y se utiliza como indicador de intensidad la duración de cada una de las tareas. Finalmente, en la imagen 5.10 se muestran un mapa de calor con pesos para cada uno de los distintos grupos, como se puede apreciar en dicha imagen la localización geográfica de las tareas de un grupo no difiere de modo significativa de las de otro, sino que éstas parece distribuirse de modo uniforme sin importar el grupo al que pertenecen. La duración de cada tarea es utilizada como peso e indica el total de minutos de asistencia requeridos. Las zonas de color rojo y azul oscuro indican que en dichas zonas se requiere un mayor número de minutos de asistencia.

De igual modo y con el objetivo de mostrar el tiempo total requerido de un modo más claro el total de horas requeridas por cada localización en la imagen 5.11 se muestra un histograma en 3D sobre el mapa de la Comunidad de Madrid, en el cual tanto el color de las barras como la altura de las mismas representa el número total de horas requeridas por cada cliente. El cual ha sido construido del siguiente modo: en primer lugar, se ha calculado el número total de horas que requiere cada cliente en un único servicio y posteriormente cada servicio se ha agrupado espacialmente considerando un radio de 75 metros.

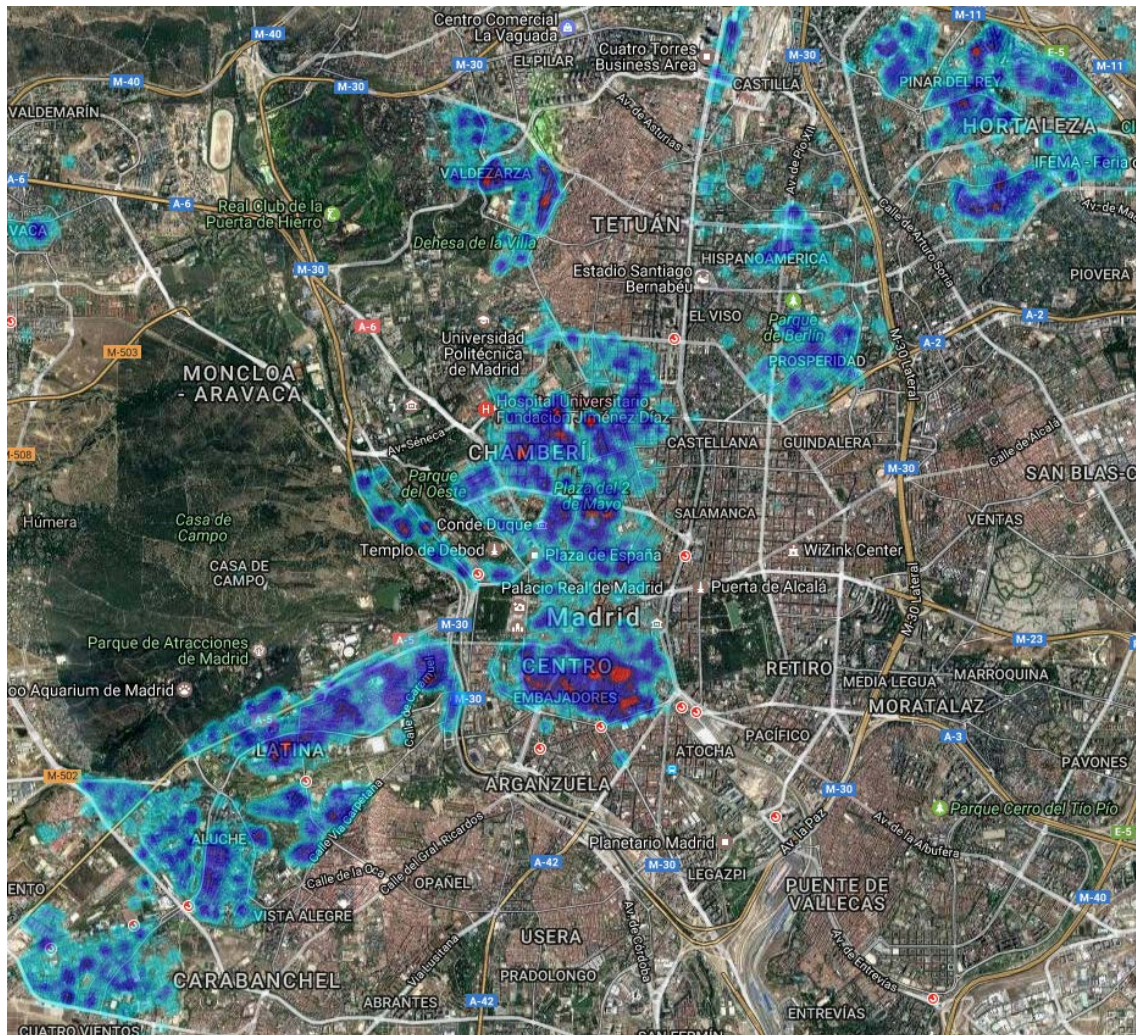
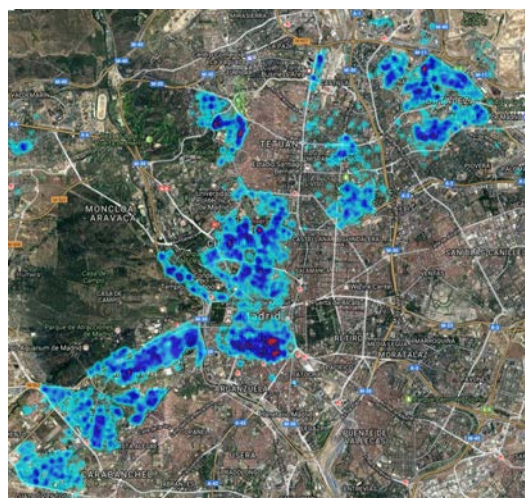
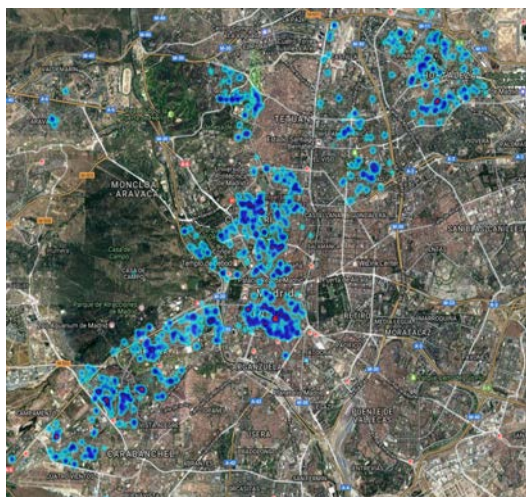


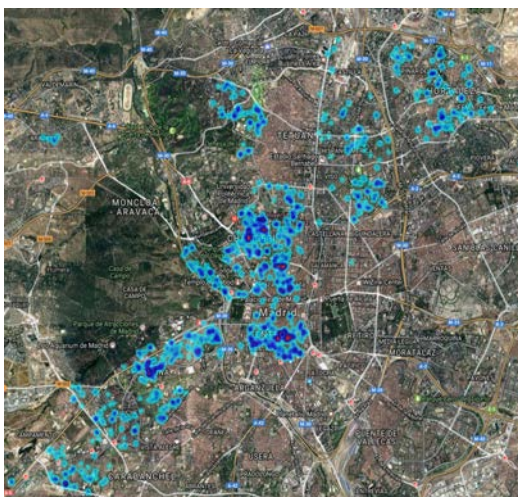
Figura 5.9: Mapa de calor con pesos en el que se representan todas las tareas de las distintas instancias, utilizando la librería [Google Maps APIs \(2016c\)](#)



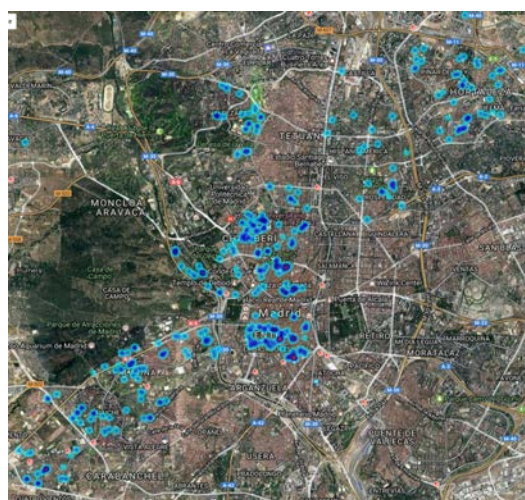
(a) Tareas del Grupo 1



(b) Tareas del Grupo 2



(c) Tareas del Grupo 3



(d) Tareas del Grupo 4

Figura 5.10: Mapa de calor con pesos para los distintos grupos de tareas.

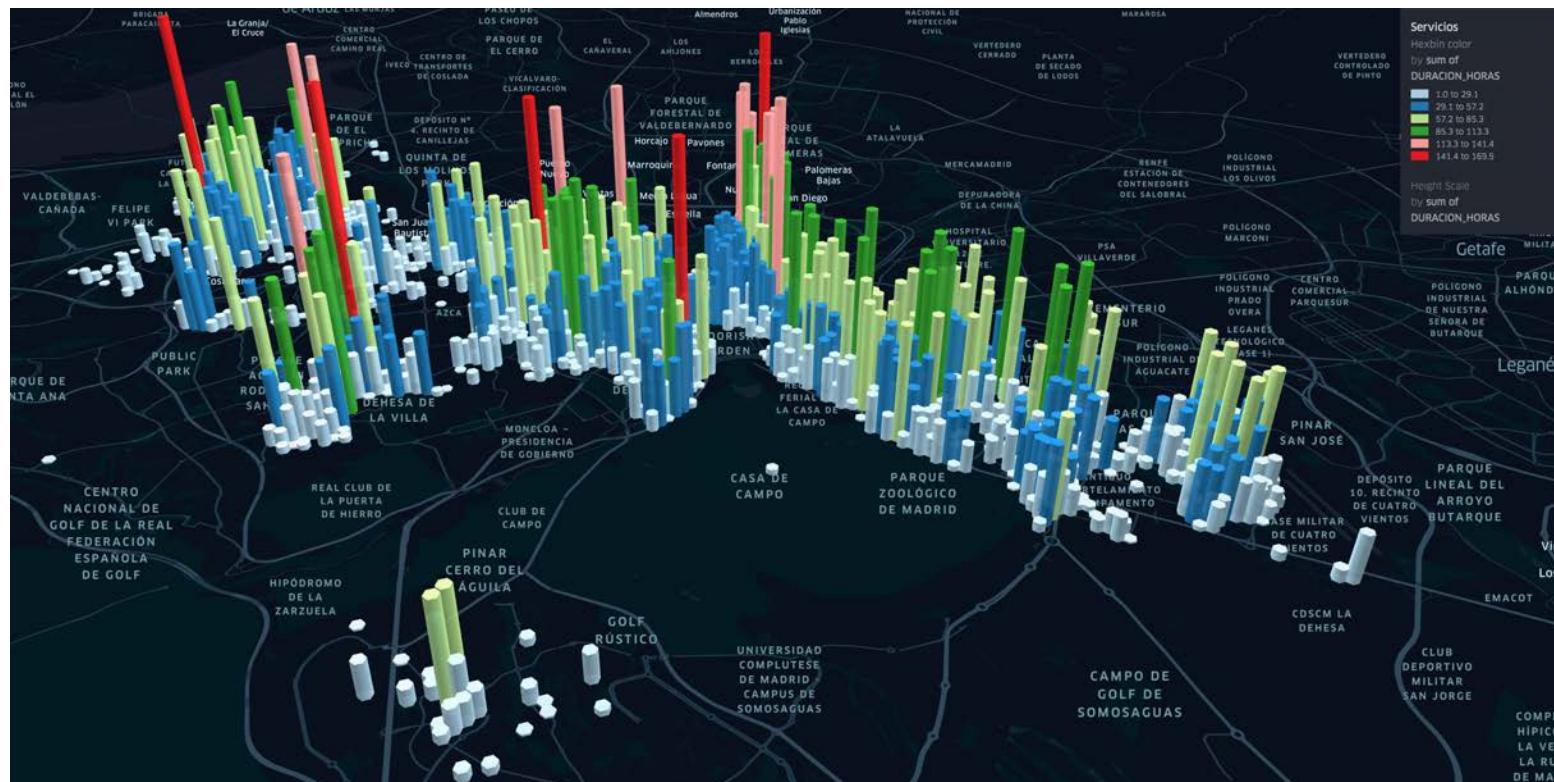


Figura 5.11: Histograma en 3D en el que los servicios se han agrupado espacialmente con un radio de 75m sumándose el total de horas requeridas.

5.1.1. Objetivos y restricciones

En la sección 3.4 se presentó la formalización del problema a resolver, además en esa misma sección se presentaba el problema a resolver como un problema objetivos enfrentados cuya importancia puede variar en función de las necesidades, en concreto se presentaban las ecuaciones (3.5) y (3.6), las cuales se describen a continuación con el fin de facilitar la lectura.

1. Minimizar el número de asistentes requeridos para atender todas las tareas. En lo sucesivo a dicho objetivo será representado como $Clust_{obj}$.

$$\text{Minimizar } Clust_{obj} = |C|$$

2. Minimizar el tiempo total requerido para visitar a todos los clientes a lo largo del horizonte de planificación. En lo sucesivo dicho objetivo será representado por Tot_{obj} el cual será medido en minutos.

$$\text{Minimizar } Tot_{obj} = \sum_{i=1}^k W_i + T_i + P_i$$

Donde W_i es el tiempo total de espera, T_i es el tiempo total de desplazamiento y P_i es el tiempo total productivo, donde todos los subíndices i hacen referencia al i -ésimo clúster. En este capítulo y subsiguientes para referirnos al valor de cada solución respecto al tiempo de espera W_i utilizaremos la abreviatura $Wait_{obj}$, de modo análogo para referirnos al valor obtenido por cada técnica respecto al tiempo de desplazamiento T_i utilizaremos la abreviatura $Trav_{obj}$, ambas expresadas en minutos.

Dichos objetivos fueron trasladados consecuentemente en dos tipos de funciones objetivo, en la que la primera de ellas solamente tiene en cuenta el número de asistentes necesarios para visitar a todos los clientes durante el periodo de planificación, mientras que la segunda realiza una combinación lineal entre el tiempo total requerido y el número de asistentes requeridos [Quintana et al. \(2017\)](#).

Otro de los aspectos fundamentales que determinan las soluciones encontradas son las restricciones del problema. En la sección 3.4 del capítulo 3 se presentaron las restricciones del problema, que son:

1. Todo servicio debe ser atendido durante la semana y debe ser realizado dentro de su ventana temporal. Cualquier solapamiento entre tareas está expresamente prohibido.
2. El número máximo de horas de trabajo que un asistente puede desempeñar a lo largo de la semana debe ser respetado. En sucesivo dicho parámetro será representado por $max_w = 40$ horas

$$W_i + T_i + P_i < max_w \quad \forall i \in [0, \dots, k] \quad (5.1)$$

3. Puesto que los asistentes se desplazan de un cliente a otro a pie, la distancia máxima que pueden recorrer entre clientes está limitada. Dicho parámetro será representado por $Dmax_k$.

4. El tiempo de espera en el que un asistente puede incurrir antes de iniciar la realización de una tarea no deberá exceder cierto límite. Dicho límite será representado por $Wmax_k$. Esta restricción fue añadida para asegurar que las soluciones obtenidas están alienadas con los objetivos de negocio de la compañía.

5.1.2. Parametrización

A lo largo de la fase de experimentación, algunas de estas restricciones se han parametrizado con el objetivo de estudiar su influencia en la construcción de soluciones. Especial atención merecen las restricciones relacionadas con la distancia máxima que pueden recorrer los asistentes y el tiempo máximo de espera que estos pueden incurrir. Como se verá a lo largo de este capítulo, dichas restricciones tienen una influencia significativa en el tiempo de cómputo requerido para la obtención de soluciones, así como en la calidad de las soluciones obtenidas. El conjunto de parámetros utilizados en esta fase experimental son los siguientes.

- **Tipo de agrupamiento:** Define el conjunto de horarios utilizados para agrupar las distintas instancias. Es decir, define el modo en el que han agrupado los distintos servicios y cómo estos han sido resueltos. Es importante destacar para cada tipo de agrupamiento se definen una serie de grupos, dichos grupos serán resueltos de modo independiente, con carácter general las soluciones reportadas serán la suma de la mejor solución encontrada por la técnica objeto de experimentación para cada uno de los grupos. En la fase de experimentación se han definido hasta tres tipos de agrupamientos distintos.
 1. **Tipo A:** Este es el agrupamiento preferido por la compañía, el cual será utilizado por todas las técnicas objeto de experimentación presentes en este capítulo, quedando compuesto por los siguientes grupos.
 - a) Grupo 1: Servicios y tareas realizados de lunes a viernes entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 22.982 tareas y agrupadas en 10.709 servicios.
 - b) Grupo 2: Servicios y tareas realizados de lunes a viernes entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 3.812 tareas y agrupadas 1.159 servicios.
 - c) Grupo 3: Servicios y tareas realizados de sábado a domingo entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 1.576 tareas distribuidas en 1.070 servicios.
 - d) Grupo 4: Servicios y tareas realizados de sábado a domingo entre las 14:31 y las 22:00 horas ambas incluidas, formado por un Total de 664 tareas, distribuidas en 406 servicios.
 2. **Tipo B:** Similar al agrupamiento anterior, contiene un grupo menos ya que los grupos 3 y 4 se fusionan en uno solo. Este tipo de agrupamiento es el utilizado en [Quintana et al. \(2017\)](#).
 - a) Grupo 1: Servicios y tareas realizados de lunes a viernes entre las 07:00 y las 14:30 horas ambas incluidas, formado por un total de 22.982 tareas y agrupadas en 10.709 servicios.

- b) Grupo 2: Servicios y tareas realizados de lunes a viernes entre las 14:31 y las 22:00 horas ambas incluidas, formado por un total de 3.812 tareas y agrupadas 1.159 servicios.
- c) Grupo 3: Servicios y tareas realizados de sábado a domingo entre las 07:00 y las 22:00 horas ambas incluidas, formado por un total de 2.240 tareas distribuidas en 1.476 servicios.

3. **Tipo C:** Todos los servicios están agrupados en un único grupo.

- a) Grupo 1: Servicios y tareas realizados de lunes a domingo entre las 07:00 y las 22:00 horas ambas incluidas, formado por un total de 13.344 servicios y 29.034 tareas.

- **Tiempo máximo de desplazamiento:** Como se detalló anteriormente existe una distancia máxima, D_{max_k} en la que puede incurrir los asistentes, ya que los desplazamientos entre cliente y cliente se realizarán a pie. Suponiendo una velocidad media, en la fase de experimentación se utilizará una media de 5 km/h, resulta sencillo transformar dicha distancia máxima en tiempo máximo de desplazamiento.
- **Tiempo máximo de espera:** Existe un tiempo máximo de espera W_{max_k} en el que pueden incurrir los asistentes a la hora realizar una tarea. El tiempo de espera se define como el número de minutos transcurridos desde que el asistente llega a la ubicación en donde debe realizar su siguiente tarea, hasta el comienzo de dicha tarea. En algunas configuraciones utilizadas en la fase de experimentación, la hora de comienzo de la tarea está fija, no existiendo una ventana temporal para su realización, lo cual implica que en la mayoría de las tareas se debe incurrir en un tiempo de espera.
- **Ventana temporal:** Para la realización de cada tarea se define una ventana temporal t_w la cual obliga a los asistentes a prestar su servicio dentro de dicha ventana. En algunas configuraciones utilizadas en la fase de experimentación la ventana temporal es igual a 0, por lo que la hora de comienzo de la tarea está fija, lo cual implica que en la mayoría de las tareas se debe incurrir en un tiempo de espera.
- **Función de evaluación:** Como se describió en el apartado 4.3.2.5 para las técnicas propuestas basadas en ACO se han definido dos funciones de evaluación. La primera de ellas se basa exclusivamente en considerar el número de asistentes requeridos para atender a todos los clientes, tratando de minimizarlos, a lo largo de este capítulo nos referiremos a dicha función como F_{clust} . La otra función de evaluación con la que se ha experimentado para dichas técnicas es la propuesta por [Quintana et al. \(2017\)](#) en la que el coste de una solución, expresado en €, y viene dado por la ecuación (3.7) la cual se vuelve a reproducir a fin de facilitar la lectura.

$$F_{clust} = Clust_{obj} = |C|$$

$$F_{cost} = 14,0 * \frac{Tot_{obj}}{60} + \frac{Clust_{obj} * 554,64}{52}$$

A lo largo del texto nos referiremos a una configuración concreta utilizando la siguiente nomenclatura $\{Técnica, Evaluación, Agrupamiento, D_{max_k}, W_{max_k}, t_w\}$, pudiendo obviar alguno de los parámetros cuando el resto de ellos sean evidentes, por ejemplo, si estamos dentro de la sección en la que se analizan los resultados de la técnica ACS-HCSP dicho parámetro podrá no ser especificado.

5.1.3. Hardware utilizado

La combinación anterior de parámetros da lugar a numerosos experimentos para cada una de las 3 técnicas propuestas, como son el método de Ward modificado, ACS-HCSP y IACS-HCSP. En el caso del método de Ward se ha experimentado con un total de 6 tipos de configuraciones distintas, repitiéndose cada uno de ellos un total de 10 veces. En el caso de los experimentos de las técnicas basadas en ACO, el total de experimentos realizados asciende a 96 por cada una de las técnicas, las cuales se han repetido un total de 5 veces dada la naturaleza estocástica de las técnicas utilizadas. Cada colonia de hormigas busca una solución completa para cada grupo de servicios, estando formada cada una de ellas por un total de 64 hormigas que son evolucionadas a lo largo de 100 rondas. Todos estos experimentos se han llevado a cabo utilizando un servidor de altas prestaciones (n1-highcpu-64) alojado en [Google Cloud Platform \(2017\)](#), el cual cuenta con un procesador Intel Xeon de 64 núcleos y con 60 GB de memoria RAM. Cabe destacar que para poder realizar la fase experimental en un periodo razonable de tiempo ha sido necesario adaptar las versiones de ACS-HCSP y IACS-HCSP a una versión que pueda aprovecharse de la ejecución paralela, debiéndose implementar mecanismos que sincronicen la actualización de feromona. En tiempo total de computo invertido en dicha plataforma ha sido de alrededor de 194 horas, tiempo que en un ordenador convencional hubiera supuesto aproximadamente algo más de 500 días de computación.

5.2. Análisis del método de Ward modificado

El método jerárquico de Ward modificado, presentado en la sección 4.2, ha sido utilizado en la fase experimental como una de las primeras aproximaciones con la que se ha abordado la resolución de las distintas instancias. La justificación de su utilización es doble, en primer lugar se ha adaptado una técnica de clustering jerárquico a un problema optimización combinatoria de alta dimensionalidad, el cual presenta restricciones temporales y espaciales. Por otro lado, dicha aproximación puede utilizarse como línea base para ser comparada con el resto de técnicas propuestas.

5.2.1. Parametrización

Antes de analizar los resultados obtenidos con el método de Ward modificado, es necesario describir la parametrización utilizada en los experimentos, se recomienda consultar la sección 5.1.2 para una mejor comprensión de los mismos.

- **Tipo de agrupamiento:** [Tipo A, Tipo B, Tipo C]. Se han utilizado todos los tipos de agrupamientos.
- **Tiempo máximo de desplazamiento:** $D_{max_k} \in [30]$. Se ha fijado el tiempo máximo de desplazamiento en 30 minutos.
- **Tiempo máximo de espera:** $W_{max_k} \in [30, \infty]$. Se han establecido dos tiempos máximos de espera, el primero de ellos de 30 minutos y el segundo sin ningún tipo de limitación.
- **Ventana temporal t_w** Sin ventana temporal.

Las combinaciones de los parámetros anteriores dan como resultado un total 6 experimentos. Uno de los aspectos que debe considerarse es la existencia de cierta componente aleatoria en el método desarrollado. Dicha componente se debe a como se resuelven los empates en el proceso de agrupamiento, en dicha situación se ha optado por elegir uno de ellos de forma aleatoria a fin de estudiar el impacto en la calidad de las soluciones obtenidas. Por este motivo, la experimentación con cada una de las configuraciones anteriores se ha repetido un total de 10 veces, dando como resultado un total de 60 experimentos.

5.2.2. Resultados

Una vez presentados los parámetros que han sido utilizado en la fase experimental para el método de Ward modificado se exponen los resultados obtenidos. En la tabla A.1 del apéndice B se encuentran todos los resultados de las distintas ejecuciones y configuraciones. Dichos resultados que se muestran de modo resumido en la tabla 5.1, mostrándose el valor medio valores medio (μ) y la desviación típica (σ) para cada una de las configuraciones.

Agrupamiento	D_{max_k}	W_{max_k}	$Trav_{obj}$		$Wait_{obj}$		Tot_{obj}		$Clust_{obj}$	
			μ	σ	μ	σ	μ	σ	μ	σ
A	30	30	94587	857	73722	740	2682924	1250	4963	8
A	30	∞	147830	1045	356357	3221	3018802	3716	3932	13
B	30	30	93996	607	73852	623	2682462	877	4959	9
B	30	∞	152046	877	422077	4752	3088738	4627	3800	15
C	30	30	79287	747	61571	837	2655474	1204	4442	9
C	30	∞	152125	1345	612686	8291	3279426	8577	3266	12

Tabla 5.1: Resultados resumidos obtenidos con el método de Ward para las distintas configuraciones. El valor medio viene definido por μ y la desviación estándar por σ , habiéndose obtenido dichos valores tras repetir 10 veces cada una de las configuraciones.

Centrándonos en el objetivo $Clust_{obj}$, minimizar el número de asistentes necesarios, se observa que la configuración que obtiene un mejor resultado es la configuración $[C, 30, \alpha]$ requiriendo un total de 3266 asistentes. De hecho, el tipo de agrupamiento C, obtiene mejores resultados en lo referido a dicho objetivo para los distintos valores de tiempo máximo de espera (W_{max_k}). Este resultado se explica por los siguientes motivos. Dicho tipo de agrupamiento C, formado por un único grupo de tareas, otorga una mayor flexibilidad

y permite soluciones que requieren un número significativamente menor de asistentes. Al existir un único grupo de servicios, las tareas de determinadas horas se pueden realizar junto con otras tareas que con otro tipo de agrupamiento no sería posible. Por ejemplo, las tareas que han de realizarse a las 14:00 de lunes a viernes, se pueden realizar conjuntamente con tareas que han de realizarse a las 15:00, lo cual con otro tipo de agrupamientos no sería posible (véase Tipo A o B). Finalmente, este tipo de agrupamiento es capaz de aprovecharse de la existencia de cierta continuidad entre las tareas, dicha continuidad puede apreciarse en la figura 5.7 para aquellas tareas cuya hora de inicio son las 14:00. En el caso del objetivo Tot_{obj} , minimizar el tiempo total requerido para visitar a todos los clientes, la configuración que obtiene los mejores resultados es $[C, 30, 30]$, requiriendo un total de 2.655.474 minutos y 4442 asistentes que de media trabajarían 10 horas semanales. En esta ocasión y a diferencia del objetivo anterior, la mejora obtenida respecto a los otros tipos agrupamientos, A y B es mucho menor estando en torno al 1 %. Para poder interpretar mejor los resultados obtenidos, es necesario remarcar el modo en que el método aglomerativo propuesto obtiene soluciones, en método propuesto un clúster de servicios sólo estará completo cuando no exista ningún otro clúster con el que se pueda unir. Este criterio de parada para la construcción de clústers resulta beneficioso para minimizar el objetivo $Clust_{obj}$, sin embargo, es perjudicial para minimizar el objetivo Tot_{obj} . Este hecho se puede apreciar en el incremento que se produce al variar el parámetro $Wmax_k$ para un mismo tipo de agrupamiento y valor de restricción de tiempo máximo de desplazamiento $Dmax_k$. Para cualquiera de las configuraciones, al no restringir el valor de tiempo máximo de espera se produce un incremento significativo en la distancia total recorrida, incremento que algunos casos como en el tipo de agrupamiento C llega a superar el 23 %. Dicho incremento se debe a dos factores, el incremento en el tiempo de desplazamiento y el incremento en el tiempo de espera. En la figura 5.12 se muestra el efecto que produce la combinación de cada uno de los tipos de agrupamientos con el tiempo máximo de espera en los objetivos Tot_{obj} y $Cluster_{obj}$, se puede apreciar de modo claro como el hecho de no limitar el tiempo máximo de espera produce un efecto doble, aumentando el tiempo total requerido Tot_{obj} y disminuyendo el número de asistentes.

De modo análogo en la figura 5.13 se muestra el efecto que tiene el hecho de no limitar el tiempo máximo de espera para cada uno de los tipos de agrupamientos. Se aprecia de modo claro que al no limitar este parámetro se produce aumento en los tiempos de espera $Wait_{obj}$ y desplazamiento $Travel_{obj}$.

En la figura 5.13 se muestra cual es el efecto que tiene en el método de Ward no limitar, apreciándose de modo claro que el mayor incremento al no limitar el tiempo de espera máximo ($Wmax_k$) se produce precisamente en el objetivo que mide dicho tiempo de espera $Wait_{obj}$.

Además del número de asistentes requeridos y los distintos tiempos considerados anteriormente, otro factor a analizar es la distribución del número de horas semanales requerida por cada agente en cada tipo de parametrización. En aquellas soluciones que requieren un mayor número de asistentes ($Cluster_{obj}$) el número de hora requeridas por cada agente será menor, en la figura 5.14 se muestran varios histogramas para las distintas parametrizaciones expuestas hasta el momento. La línea roja discontinua representa la media de la distribución, cuyo valor viene indicado por μ , siendo σ el valor de la desviación típica. Finalmente, el valor de $Cluster_{obj}$ representa el número de asistentes requeridos.

En la figura anterior se representan la distribución de horas semanales para un expe-

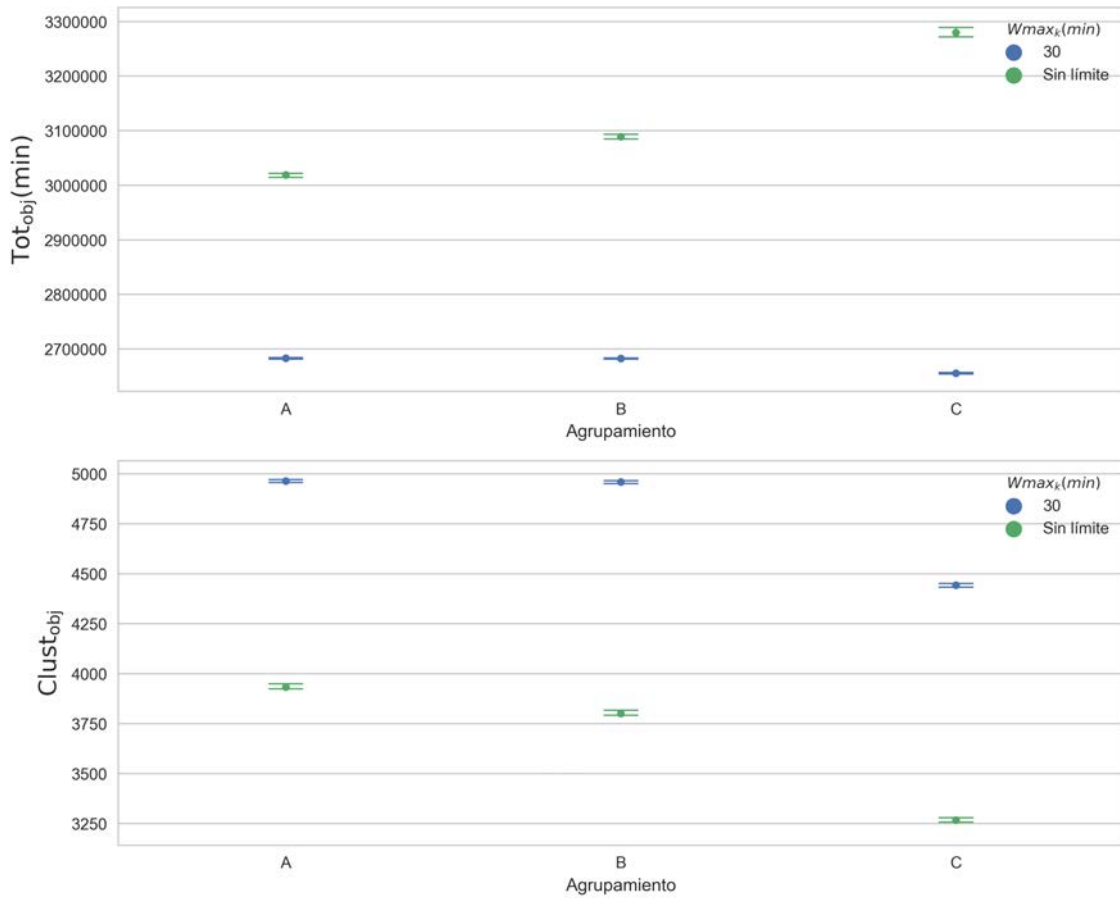


Figura 5.12: Valores medios obtenidos para cada uno de los objetivos Tot_{obj} y $Clust_{obj}$ distinguiendo por tipo de agrupamiento y valor de $Wmax_k$. Se puede apreciar como el hecho de no limitar el tiempo máximo de espera provoca que la distancia total aumente y el número de asistentes requeridos disminuya.

rimento en particular, obteniéndose resultados similares en el resto de experimentos, no apreciándose diferencias significativas que no sean compatibles con los valores de media y desviación estándar obtenidos. Para cualquiera de los tipos de agrupamientos se puede apreciar como el hecho de no limitar el tiempo máximo de espera hace que la distribución se desplace hacia la derecha, es decir, el número medio de horas semanales requeridos por cada agente es mayor. De modo similar la transición entre los agrupamientos $A \rightarrow B \rightarrow C$ para una misma configuración de tiempo máximo de espera provoca el mismo efecto, siendo dicho efecto más destacable cuando no se limita el tiempo máximo de espera.

En la figura 5.15 se muestra un gráfico de dispersión en el que se pueden apreciar los resultados obtenidos de los 60 experimentos realizados para el método de Ward modificado, mostrándose en el eje x el objetivo $Clust_{obj}$ y en el eje y el objetivo Tot_{obj} expresado en minutos. Para facilitar la legibilidad del mismo y dado que los resultados obtenidos para una misma configuración presentan poca dispersión se ha añadido cierto factor de

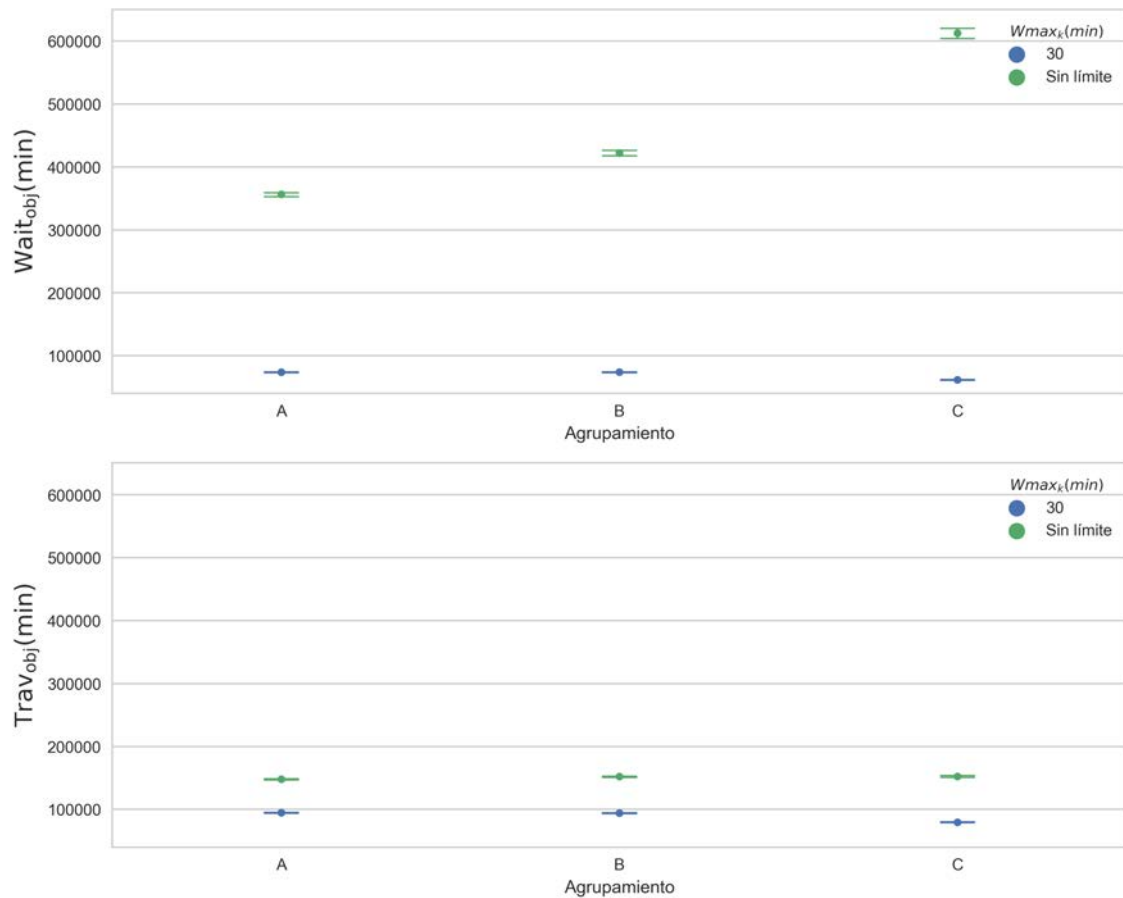


Figura 5.13: Valores medios obtenidos para cada uno de los objetivos $Wait_{obj}$ y $Trav_{obj}$ distinguiendo por tipo de agrupamiento y valor de $Wmax_k$. Nótese que los valores del eje Y no están en la misma escala.

ruido¹, lo cual permite visualizar de un mejor modo solución obtenida por cada tipo de configuración, los resultados exactos se recogen la tabla A.1 del anexo B.

En la parte izquierda del gráfico cuando el tiempo máximo de espera no está limitado, se observa claramente como ninguna de las soluciones obtenidas por cada tipo de agrupamiento domina sobre el resto, es decir las soluciones obtenidas para un tipo de agrupamiento siempre obtienen mejor resultado en alguno de los objetivos cuando se las compara con soluciones procedentes de otro tipo de agrupamientos. Por ejemplo, las soluciones con el tipo de agrupamiento A obtienen mejores soluciones para alguno de los objetivos cuando se las compara con soluciones con el agrupamiento B o C. Sin embargo, si nos centramos en el gráfico de dispersión que aparece a la derecha de la figura, en el cual se limita el tiempo de espera a 30 minutos, observamos como las soluciones obtenidas por el tipo de agrupamiento C domina al resto de soluciones obtenidas por los otros tipos de agrupamientos, obteniendo

¹Se ha añadido un ruido uniforme de 50 unidades en el objetivo $Cluster_{obj}$ y de 10000 en el objetivo Tot_{obj} .

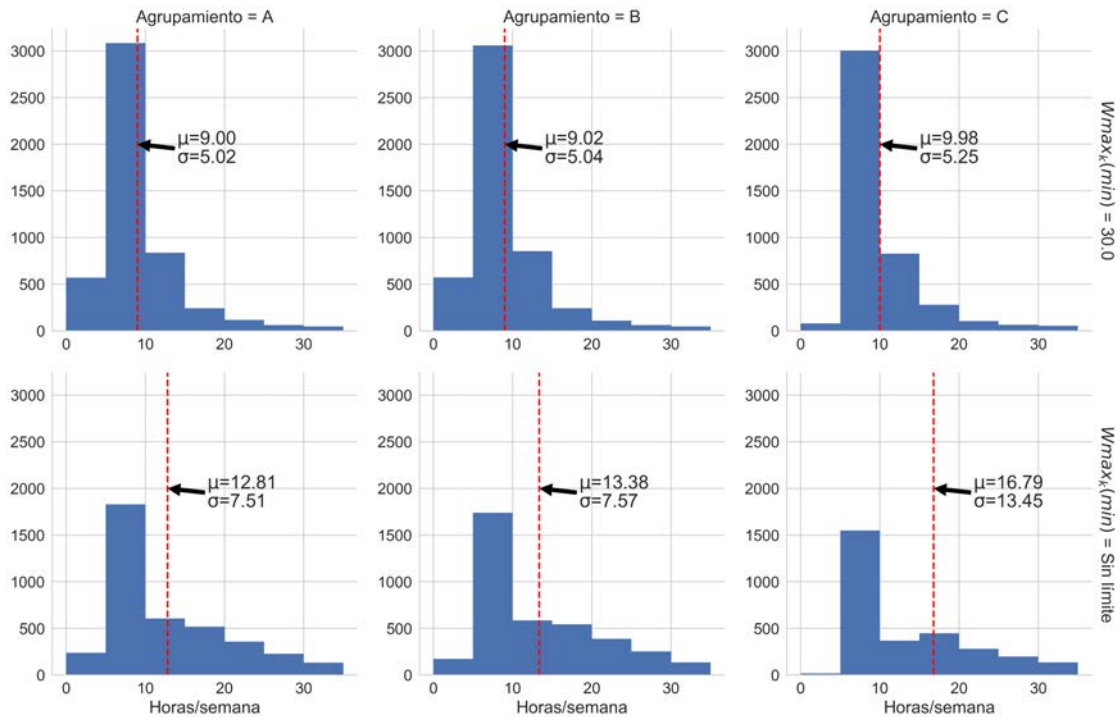


Figura 5.14: Histograma en el que se muestra el número de horas requeridas para cada tipo de agrupamiento y valor de tiempo máximo de espera W_{max_k} .

mejores resultados tanto en número de asistentes requeridos como en tiempo total requerido.

5.2.3. Áreas de mejora

Tal y como se presentó en la sección 4.2, la propuesta utilizada es una adaptación del método aglomerativo, en el que inicialmente cada servicio pertenece a un único clúster, en cada paso del algoritmo se integran dentro de un mismo clúster aquellos servicios/clústers cuya similitud sea menor ². Esta descripción coincide con la descripción de un método heurístico de mejor primero, utilizando como función heurística la medida de distancia descrita en la sección 4.2.1. Estos métodos de mejor primero o “avariciosos” son ineficientes en la obtención de soluciones de calidad en problemas de optimización combinatoria y como se mostrará en las secciones siguientes también lo son para la resolución de problemas de planificación de asistentes de atención domiciliaria. A pesar de que en cada ronda del método aglomerativo se unen aquellos clústers que se encuentren a menor distancia, es fácil observar, incluso con un número mínimo de servicios, cómo el resultado de aplicar dicho procedimiento puede dar como resultado soluciones de peor calidad que otras aproximaciones.

²En caso de que dos o más clúster se encuentre a la misma distancia se elegirá uno de ellos de modo aleatorio. Se adopta dicho enfoque ya que es la misma que la adoptada en [Quintana et al. \(2017\)](#), además la variabilidad resultante permitirá ilustrar la importancia de los empates.

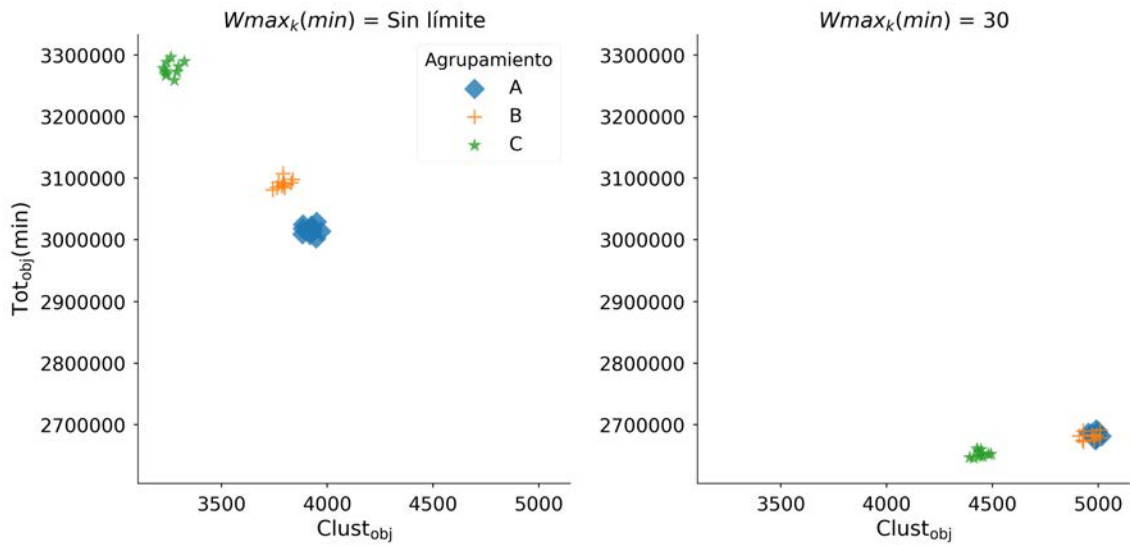


Figura 5.15: Gráfico de dispersión segmentado por el valor de tiempo máximo de espera $Wmax_k$, en el que se muestran los resultados obtenidos por los 60 experimentos realizados para el método de Ward modificado.

Tal y como se muestra en la figura 5.16, existen un total de 4 servicios, $\{S_1, S_2, S_3, S_4\}$ siendo las distancias iniciales entre ellos las que aparecen en la misma figura. Como se puede apreciar los pares servicios $\{S_1, S_2\}$ y $\{S_2, S_4\}$ no pueden estar dentro del mismo clúster, esto puede deberse a diversas causas, una de ellas podría ser que se deben prestar servicios a la misma. Con esta configuración la solución obtenida contendrá dos clústers, el $C_1 = \{S_1, S_4\}$ y $C_2 = \{S_2, S_3\}$. Inicialmente se elegirá unir los servicios S_2 y S_3 , el clúster resultante no será compatible con ninguno de los servicios S_1, S_4 restantes, por lo tanto la única unión posible será la de los servicios S_1, S_4 , que se unirán formando el clúster $C_2 = \{S_1, S_2\}$ obteniéndose en conjunto una solución con una distancia total de 4 horas y media ³. Como se puede observar, la decisión inicial de unir los servicios S_2 y S_3 por encontrarse a la menor distancia, que al comienzo del proceso parece ser una buena decisión, al final del proceso no resulta ser la mejor elección, dando como resultado una solución de peor calidad.

³Nótese que el tiempo de duración de cada servicio se ha omitido para simplificar el ejemplo. En todo caso la suma de duraciones totales de cada servicio es una constante para cualquier instancia, por lo que lo único que hace variar la distancia total recorrida son los tiempos de desplazamiento y espera.

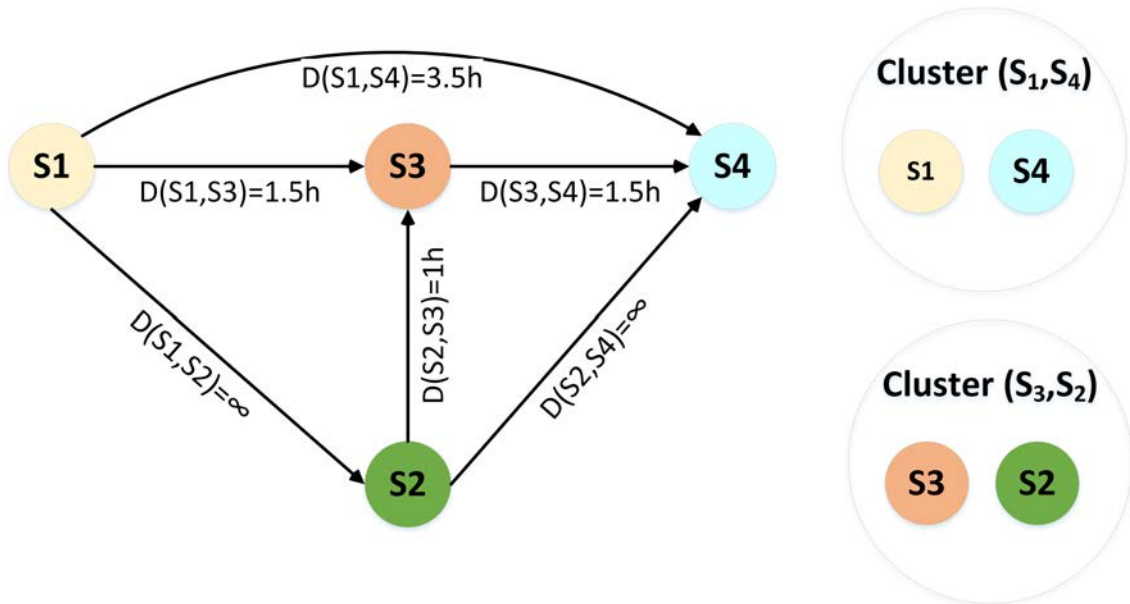


Figura 5.16: Ejemplo de la problemática que ocasiona utilizar el método de Ward basado en estrategias “avariciosas”. La solución resultante contendrá dos clústers, el $C_1 = \{S_2, S_3\}$ y $C_2 = \{S_1, S_4\}$ que no será óptima, ya que la solución formada por los clústers $C_1 = \{S_1, S_2, S_3\}$ y $C_2 = \{S_4\}$ obtendría una menor distancia y requeriría el mismo número de asistentes.

5.3. Análisis ACS-HCSP

Como se presentó en la sección 4.3.2 del capítulo 4 la propuesta denominada ACS-HCSP resulta una primera aproximación al problema de planificación de asistentes de atención domiciliaria utilizando la técnica de Optimización Basada en Colonias de Hormigas, en esta sección se lleva a cabo un análisis de los resultados obtenidos, en primer lugar se muestra la parametrización utilizada para posteriormente los resultados obtenidos.

5.3.1. Parametrización

Antes de analizar los resultados obtenidos con el método de ACS-HCSP, es necesario describir la parametrización utilizada en los experimentos, se recomienda consultar la sección 5.1.2 para una mejor comprensión de los mismos.

- **Tipo de agrupamiento:** [Tipo A, Tipo B]. Se han utilizado los tipos de agrupamientos A y B.
- **Tiempo máximo de desplazamiento:** $D_{max_k} \in [15, 30]$. Se ha fijado el tiempo máximo de desplazamiento entre 15 y 30 minutos.
- **Tiempo máximo de espera:** $W_{max_k} \in [15, 30, 45]$. Se han establecido tres tiempos máximos de espera, el primero de ellos de 15 minutos, el segundo de 30 y el último de 45 minutos.

- **Ventana temporal:** $t_w \in [0, 5, 10, 20]$. Se han establecido cuatro tipos de ventanas temporales, la primera de ellas sin ventana lo cual nos permitirá comparar los resultados con los resultados obtenidos con el método de Ward, las demás ventanas temporales se han establecido a intervalos de 5, 10 y 20 minutos respectivamente.
- **Función de evaluación:** $[F_{clust}, F_{cost}]$. Se ha utilizado los dos tipos de funciones de evaluación definidas en 5.1.2. En la primera de ellas F_{clust} se minimiza el número de clústers, en la segunda se minimiza la función F_{cost} que minimiza una estimación del coste de la misma.
- **Parámetros de ACS-HCSP:** Para el proceso de experimentación con la técnica ACS-HCSP se han utilizado numerosos parámetros, cuya representación, ecuación en la que se les referencia y valor se muestran en la tabla 5.2.

Parámetro	Ecuación	Valor
q_0^i	(4.7)	0.1
q_0	(4.8)	0.1
τ_0	(4.11)	0.1
ρ	(4.11)	0.001
α	(4.12)	0.1

Tabla 5.2: Valor de los parámetros utilizados por la técnica ACS-HCSP.

La combinación de los parámetros anteriores da como resultado un total 96 experimentos, los cuales se ha repetido un total de 5 veces, dando como resultado un total de 480 experimentos cuyo tiempo de cómputo ha ascendido a un total de 73 horas ⁴.

5.3.2. Resultados

Una vez presentados los parámetros que han sido utilizado en la fase experimental para el método de ACS-HCSP se exponen los resultados obtenidos. En la tabla A.2 del apéndice B pueden encontrarse los resultados detallados de cada uno de los 480 experimentos llevados a cabo. Cabe destacar que los resultados reportados siempre hacen referencia a la mejor solución encontrada por la colonia de hormigas tras las 100 rondas de evolución, el total de soluciones encontradas considerando todas las configuraciones y repeticiones de experimentos asciende a algo más de 3 millones. Aunque como se verá a continuación todos los parámetros tienen un impacto en la calidad de las soluciones obtenidas, quizás los dos más relevantes son el tipo de agrupamiento y la función de evaluación. Para el tipo de agrupamiento, los servicios que deben prestarse durante el fin de semana pueden resolverse como una sola instancia (Tipo B) o puedan considerarse dos instancias del problema distintas diferenciando entre horario de mañana y de tarde. La función objetivo utilizada también tiene un papel importante tal y como se mostró en la sección 4.3.2.5 del capítulo 4.

⁴Utilizando el servidor descrito en 5.1.3.

5.3.2.1. Función objetivo: F_{clust}

En la tabla 5.3 se muestran los resultados obtenidos para distintas configuraciones de parámetros cuando la función a objetivo minimizar es el número de clústers representado por F_{clust} . Los resultados reportados reflejan la mejor solución obtenida en cada una de las 5 repeticiones que se han realizado para cada configuración, mostrándose la media (μ) y desviación estándar (σ) de dicho conjunto de soluciones. En la figura 5.17 se representan parte de los resultados de un modo gráfico, en el eje Y se representa el número de clúster requeridos por cada configuración mientras que en el eje X se representan los distintos valores para la restricción de distancia máxima recorrida $D_{max_k} \in [15, 30]$. A su vez y por cada una de las combinaciones de los parámetros de tiempo máximo de espera $W_{max_k} \in [15, 30, 45]$ y ventana temporal $t_w \in [0, 5, 10, 20]$ se ha creado una subgráfica, utilizando para cada una de ellas el mismo rango de valores a fin de facilitar su comprensión. En cada una de estas subgráficas se representa el valor medio ⁵ del número de clústers para cada configuración distinguiendo a su vez por cada tipo de agrupamiento.

Centrándonos en la figura 5.17 se puede observar como no parece existir una diferencia significativa entre los tipos de agrupamiento A y B en las distintas configuraciones. En la misma figura, se puede observar como al aumentar la distancia máxima permitida D_{max_k} (eje X de cada subgráfica) se produce una reducción notable del número de clústers; por ejemplo, en el caso de la subgráfica $\{W_{max_k} = 15 | t_w = 0\}$ pasar de 15 minutos de tiempo máximo de desplazamiento a 30 supone pasar de requerir alrededor de 6.230 asistentes a requerir alrededor de 4.200, reducción que se produce en ambos tipos de agrupamientos. De modo análogo y a medida que aumenta el tiempo máximo de espera W_{max_k} , para una misma configuración de ventana temporal, se aprecia una clara disminución del número de asistentes requeridos. Finalmente, al aumentar la ventana temporal t_w y para una misma configuración, el espacio de búsqueda de posibles soluciones aumenta y consiguiéndose soluciones de mejor calidad.

Otro aspecto relevante no mencionado hasta el momento, es el coste computacional que tiene la técnica propuesta para cada una de las distintas configuraciones representado por $T_{ejecucion}$. En la figura 5.22 se muestra un ejemplo con los distintos tiempos de ejecución requeridos para las distintas configuraciones. Dicho tiempo es medido en segundos y representa la suma del tiempo total de ejecución de las distintas instancias de servicios, el cual incluye el tiempo requerido para ejecutar las 100 rondas de evolución y las 64 hormigas por ronda. De igual modo que los resultados obtenidos en la figura 5.17 se observa que al aumentar la calidad de las soluciones, también aumenta el tiempo requerido para obtenerlas. Por ejemplo pasar de la configuración $\{D_{max_k} = 30 | W_{max_k} = 15 | t_w = 0\}$ a una configuración de $\{D_{max_k} = 45 | W_{max_k} = 15 | t_w = 0\}$ supone incrementar el tiempo de resolución en más del doble, pasando de 357 a 746 segundos.

⁵Dicho valor y sus intervalos de confianza se han estimado realizando bootstrapping.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	$Trav_{obj}$		$Wait_{obj}$		Tot_{obj}		F_{clust}		$T_{ejecucion}$	
				μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
A	15	15	0	78458	709	59617	382	2652690	843	6237	12	265	2
A	15	15	5	90506	710	43949	527	2649070	838	4616	7	281	1
A	15	15	10	100666	71	35488	526	2650769	539	4203	9	316	1
A	15	15	20	112256	806	26258	485	2654842	4460	3865	11	377	2
A	15	30	0	105607	1102	164401	2130	2784623	3106	4418	16	320	1
A	15	30	5	107614	555	123526	1703	2745755	1767	3984	9	361	1
A	15	30	10	112575	778	100061	1211	2728964	3867	3807	20	409	2
A	15	30	20	119519	325	75098	838	2709232	819	3570	11	480	2
A	15	45	0	109598	542	216480	947	2840694	1043	4182	7	380	1
A	15	45	5	110135	415	165135	1848	2790030	1869	3839	6	428	2
A	15	45	10	115062	1108	138045	1218	2769435	5461	3674	15	476	1
A	15	45	20	122160	378	107196	1207	2743971	1498	3462	9	552	1
A	30	15	0	195956	1114	85978	658	2796693	1788	4220	7	357	1
A	30	15	5	186015	2296	59300	629	2759930	1884	3839	9	437	2
A	30	15	10	188402	435	44182	793	2747343	986	3661	12	531	2
A	30	15	20	199661	1017	31420	819	2747408	3902	3430	22	721	4
A	30	30	0	197231	1650	157261	1065	2869107	2450	3881	9	550	3
A	30	30	5	187497	2261	118439	1705	2820551	3481	3627	8	634	3
A	30	30	10	191477	2627	94332	1668	2800424	2482	3472	5	734	4
A	30	30	20	198510	1393	68049	170	2781174	1473	3283	9	932	3
A	30	45	0	207263	1596	219789	2056	2941667	2672	3691	5	753	5
A	30	45	5	196197	1308	166591	995	2877403	1510	3493	7	840	4
A	30	45	10	197056	1574	134128	1571	2845798	2526	3380	3	938	3
A	30	45	20	203790	813	101297	2209	2819702	2233	3187	6	1146	1
B	15	15	0	78279	816	59459	239	2652353	833	6230	14	262	6
B	15	15	5	90680	653	43941	900	2649235	1452	4608	7	281	1
B	15	15	10	100264	408	35526	523	2650404	818	4191	10	315	1
B	15	15	20	112606	277	26430	440	2653650	470	3856	3	375	2
B	15	30	0	105760	461	164347	849	2786435	3937	4421	29	319	2
B	15	30	5	107284	585	124477	1290	2746376	1506	3972	12	359	1
B	15	30	10	112401	736	100353	554	2727369	1064	3793	7	404	1
B	15	30	20	120886	457	74442	1265	2709943	1721	3553	8	475	2
B	15	45	0	110498	777	217945	1934	2843058	2627	4162	8	377	1
B	15	45	5	110037	803	166182	1023	2790834	1608	3825	9	423	1
B	15	45	10	114838	407	138734	2452	2768187	2532	3646	9	471	1
B	15	45	20	122154	582	106432	2327	2743202	2401	3454	8	544	3
B	30	15	0	195754	953	86373	411	2796742	1217	4213	11	355	1
B	30	15	5	187272	1519	59817	1223	2761705	2493	3826	5	430	2
B	30	15	10	190091	505	44762	735	2749467	1222	3646	10	526	3
B	30	15	20	199389	646	31229	645	2745233	1046	3422	9	714	4
B	30	30	0	196286	1954	158661	1544	2869563	3221	3866	14	542	3
B	30	30	5	187414	1095	119013	379	2822755	4473	3617	17	627	4
B	30	30	10	190291	1631	94405	1484	2799311	2354	3467	4	725	5
B	30	30	20	199358	506	67055	1772	2781029	1663	3272	7	921	3
B	30	45	0	207798	1166	221760	1949	2944173	875	3669	8	746	3
B	30	45	5	195576	431	166340	2605	2876530	2427	3481	5	832	2
B	30	45	10	197108	1381	135366	2537	2847088	3470	3357	15	935	3
B	30	45	20	205463	789	102122	1060	2822200	1545	3176	6	1133	3

Tabla 5.3: Resultados resumidos obtenidos para ACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{clust} .

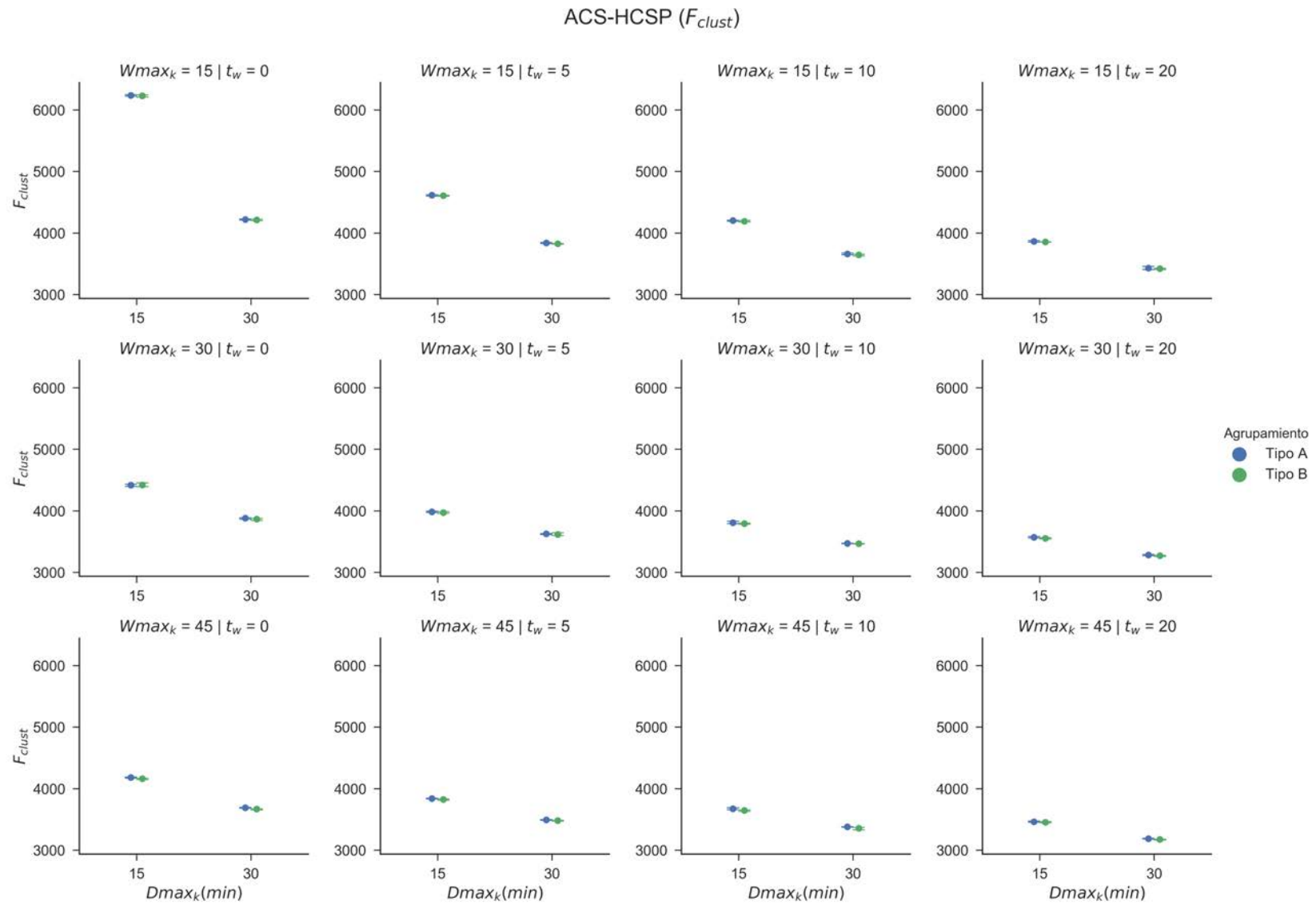


Figura 5.17: Resultados resumidos obtenidos para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust} .

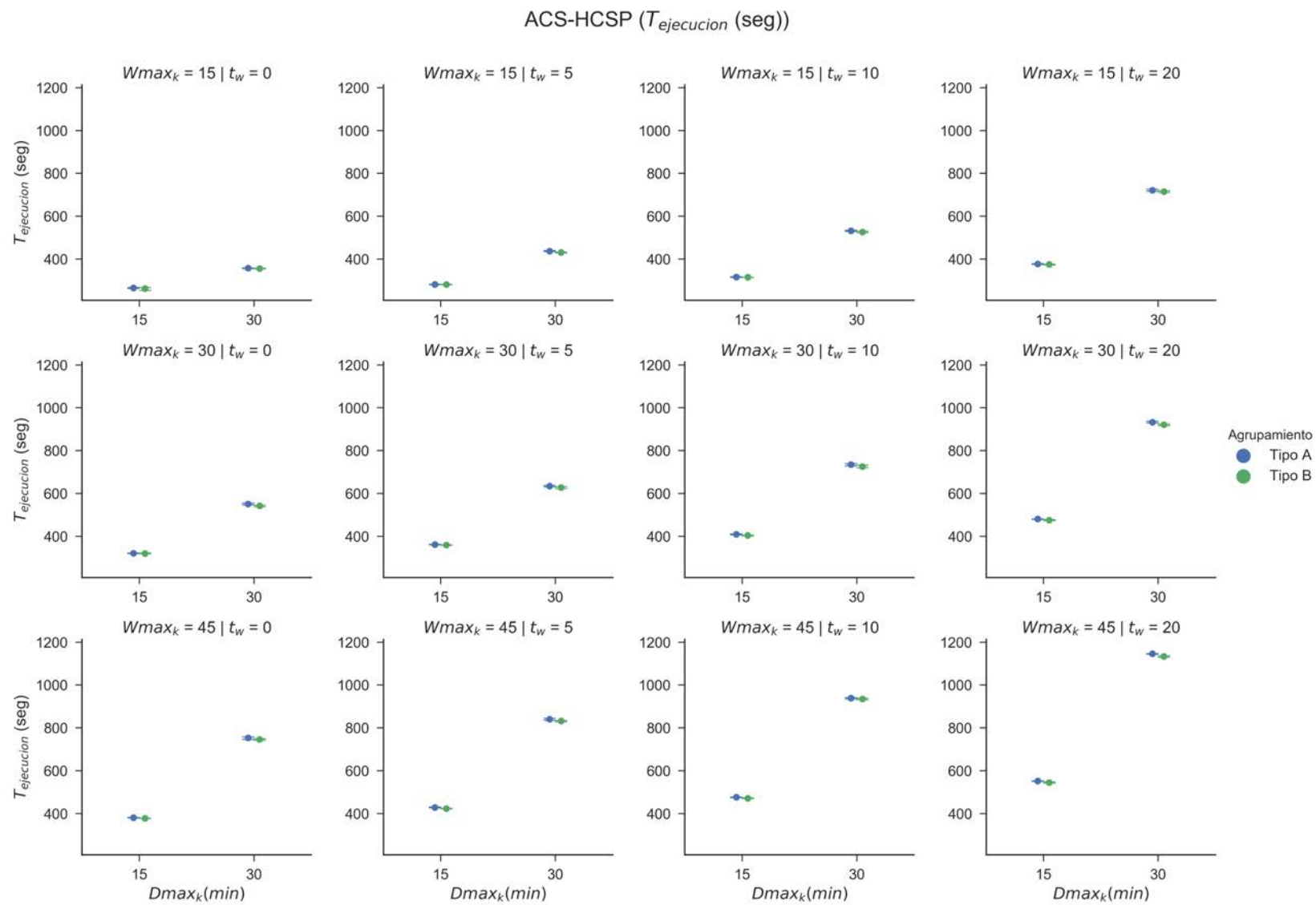


Figura 5.18: Coste computacional (medido en segundos) para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust} .

5.3.2.2. Función objetivo: F_{cost}

Una vez medido el objetivo de buscar soluciones que reduzcan el número de clústers, se analizan las soluciones obtenidas cuando el objetivo a minimizar es la función de coste definida en (3.7). En la tabla 5.4 se muestran los resultados obtenidos para distintas configuraciones de parámetros cuando la función a objetivo minimizar es el número de clústers representado por F_{cost} , mostrándose la media (μ) y desviación estándar (σ). En la figura 5.19 se representan parte de los resultados de un modo gráfico, en el eje Y se representa el coste de cada solución, mientras que en el eje X se representan los distintos valores para la restricción de distancia máxima recorrida $D_{max_k} \in [15, 30]$. A su vez y por cada una de las combinaciones de los parámetros de tiempo máximo de espera $W_{max_k} \in [15, 30, 45]$ y ventana temporal $t_w \in [0, 5, 10, 20]$ se ha creado una subgráfica, utilizando para cada una de ellas el mismo rango de valores a fin de facilitar se comprensión. En cada una de estas subgráficas se representa el valor medio ⁶ del número de clústers para cada configuración distinguiendo a su vez por cada tipo de agrupamiento.

Centrándonos en la figura 5.19 se puede observar como no parece existir una diferencia significativa entre los tipos de agrupamiento A y B en las distintas configuraciones. En la misma figura, se puede observar como al aumentar la distancia máxima permitida D_{max_k} (eje X de cada subgráfica) se produce un aumento del coste de la solución; por ejemplo en el caso de la subgráfica $\{W_{max_k} = 15 | t_w = 0\}$ pasar de 15 minutos de D_{max_k} a 30 supone un incremento de alrededor de 8.000 € semanales. En el caso de aumentar la ventana temporal t_w (conservando la misma configuración) se produce una disminución bastante significativa, ya que al aumentar la flexibilidad a la hora de atender los distintos servicios los tiempos de espera se reducen de tal modo que el coste global de la solución disminuye. Por ejemplo pasar de la configuración $\{D_{max_k} = 15 | W_{max_k} = 15 | t_w = 0\}$ a la configuración $\{D_{max_k} = 15 | W_{max_k} = 15 | t_w = 20\}$ supone un ahorro de costes de más de 25.000 € semanales.

Respecto al tiempo de ejecución, de igual modo que los resultados obtenidos en la figura 5.17 se observa que no existen diferencias muy significativas a la hora de resolver las instancias utilizando el tipo de agrupamiento A o el B. Se observa que al aumentar la calidad de las soluciones, también aumenta el tiempo requerido para obtenerlas. Por ejemplo pasar de la configuración $\{D_{max_k} = 15 | W_{max_k} = 45 | t_w = 20\}$ a una configuración de $\{D_{max_k} = 30 | W_{max_k} = 45 | t_w = 20\}$ supone incrementar el tiempo de resolución en más del doble, pasando de 357 a 746 segundos.

⁶Dicho valor y sus intervalos de confianza se han estimado realizando bootstrapping.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	$Trav_{obj}$		$Wait_{obj}$		Tot_{obj}		F_{cost}		$T_{ejecucion}$	
				μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
A	15	15	0	74069	884	56572	493	2645256	1128	684303	47	264	0
A	15	15	5	85458	616	39914	771	2641844	3573	666555	1128	281	0
A	15	15	10	95164	559	31347	307	2641126	419	661870	66	315	2
A	15	15	20	106424	596	23087	859	2644126	606	658749	122	378	1
A	15	30	0	84283	770	126575	738	2725473	573	691940	164	320	0
A	15	30	5	100908	977	106565	1003	2722088	1023	679097	249	362	3
A	15	30	10	106136	1173	85274	1126	2706025	1880	672937	295	408	2
A	15	30	20	114109	937	61520	574	2690244	1182	666734	154	480	2
A	15	45	0	88130	874	164321	1196	2767066	1440	699400	141	378	2
A	15	45	5	89823	1138	122933	585	2727371	1172	685919	101	428	2
A	15	45	10	93272	419	99584	1239	2707471	1033	679019	254	476	2
A	15	45	20	103218	7400	76501	5463	2694479	13157	671905	219	551	3
A	30	15	0	151719	970	67790	316	2734268	1274	692433	242	357	1
A	30	15	5	163047	10959	51324	3686	2728986	14582	680832	185	435	2
A	30	15	10	165825	12094	37236	2961	2717676	15050	676009	145	532	1
A	30	15	20	154345	1386	21135	548	2691808	5170	672717	1221	721	5
A	30	30	0	151081	778	119107	882	2784947	1304	701060	109	546	1
A	30	30	5	146591	1179	88182	921	2749388	1867	688959	183	632	3
A	30	30	10	149585	867	67218	1289	2731418	1614	682718	264	728	4
A	30	30	20	156441	1510	47064	959	2718121	1596	676865	241	926	2
A	30	45	0	163623	985	166281	1526	2844519	893	712424	179	746	4
A	30	45	5	159117	1183	123052	535	2796785	1004	697779	197	839	6
A	30	45	10	157200	1702	97617	1079	2769432	2533	690129	333	940	4
A	30	45	20	164678	1187	70524	857	2749816	1117	682720	186	1145	3
B	15	15	0	74022	737	55828	631	2644465	1247	684251	155	265	2
B	15	15	5	85754	532	39894	367	2641976	3559	666532	1259	281	1
B	15	15	10	95954	325	31566	168	2642135	252	661779	144	315	1
B	15	15	20	107308	814	23361	491	2645284	894	658721	143	375	1
B	15	30	0	84649	1076	126867	545	2726131	1556	692092	179	319	1
B	15	30	5	101301	735	107529	888	2723445	1327	679318	170	360	2
B	15	30	10	106553	471	86145	939	2707313	1354	673067	197	406	2
B	15	30	20	113603	725	62429	1219	2690647	1641	666736	203	475	2
B	15	45	0	89087	523	166734	1174	2772149	4797	700219	1341	377	2
B	15	45	5	90179	1013	125116	1272	2729910	1751	686091	243	423	1
B	15	45	10	94288	689	100988	79	2709890	662	679206	119	471	1
B	15	45	20	106361	8539	80544	7184	2701520	15699	671956	157	543	2
B	30	15	0	151841	591	67792	894	2734248	1171	692458	166	355	1
B	30	15	5	165550	10174	51910	3768	2732075	13896	680971	128	431	1
B	30	15	10	157356	13198	35269	3897	2707240	17049	676080	221	525	2
B	30	15	20	155213	1403	21765	594	2691593	1420	672286	156	712	2
B	30	30	0	152657	495	120078	663	2787350	886	701271	133	540	2
B	30	30	5	148570	1189	88804	1433	2751989	1428	689170	87	625	1
B	30	30	10	151764	1146	69191	1180	2737283	5336	683565	1268	723	3
B	30	30	20	157886	1568	46907	969	2721121	4629	677467	1264	920	2
B	30	45	0	164834	2249	168743	906	2848193	1632	712761	195	743	2
B	30	45	5	158515	1583	124136	475	2797266	1911	697951	229	830	3
B	30	45	10	159212	1621	97900	1251	2771726	2289	690219	321	931	2
B	30	45	20	165899	461	71699	593	2752213	443	682998	130	1136	5

Tabla 5.4: Resultados resumidos obtenidos para ACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{cost} .

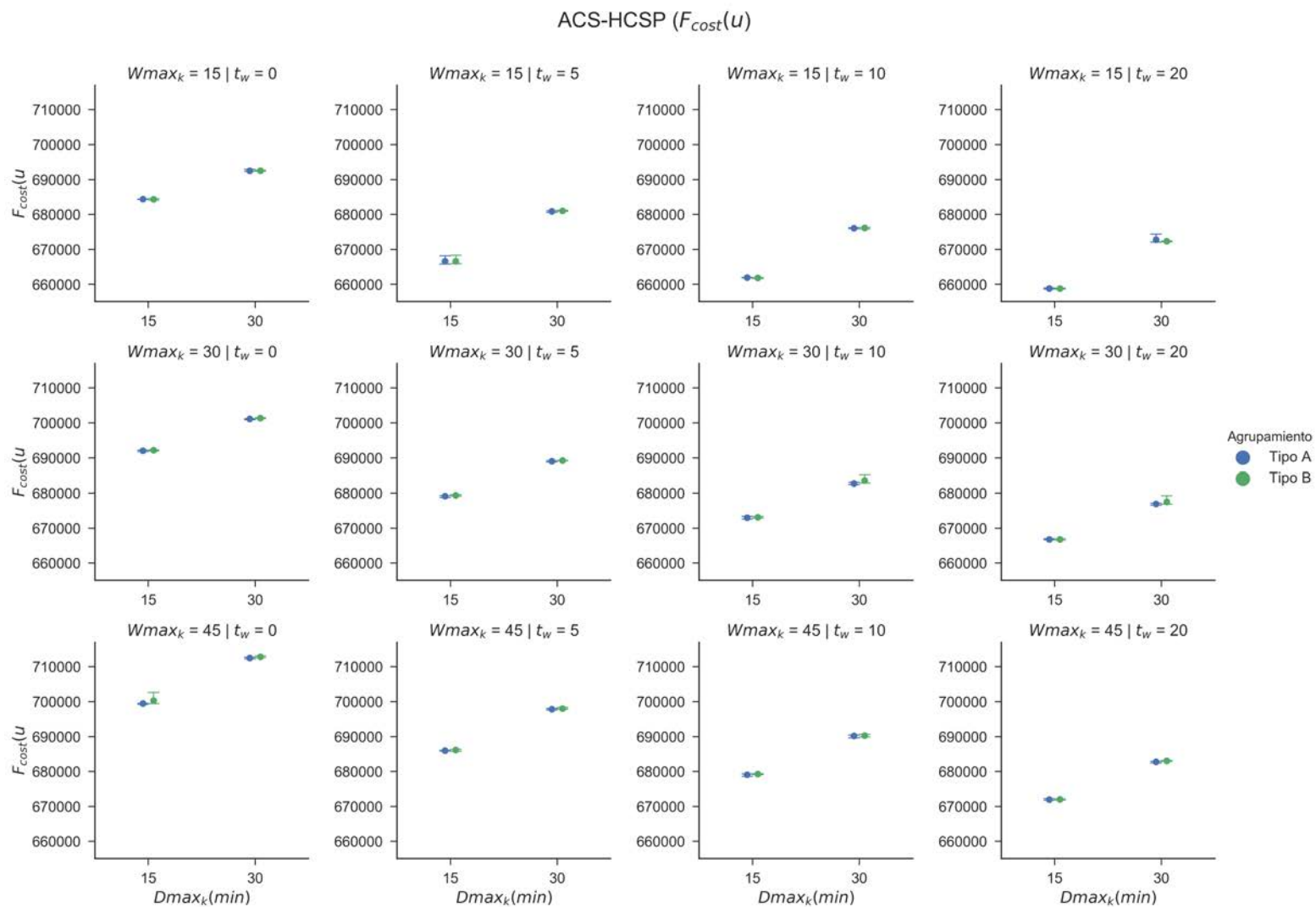


Figura 5.19: Resultados resumidos obtenidos para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost} .

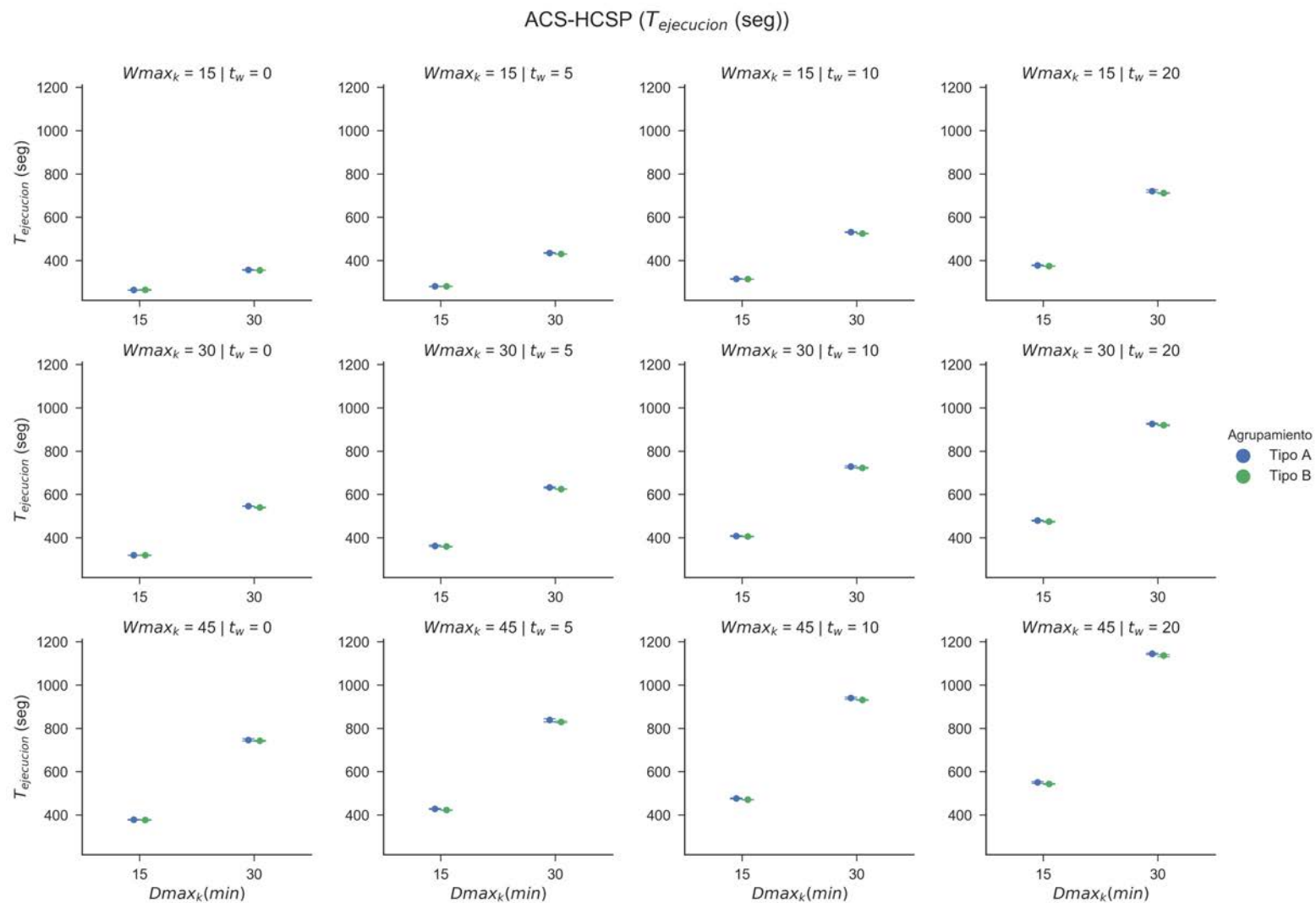


Figura 5.20: Coste computacional (medido en segundos) para ACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost} .

5.4. Análisis IACS-HCSP

5.4.1. Parametrización

Antes de analizar los resultados obtenidos con el método de IACS-HCSP, es necesario describir la parametrización utilizada en los experimentos, se recomienda consultar la sección 5.1.2 para una mejor comprensión de los mismos.

- **Tipo de agrupamiento:** [Tipo A, Tipo B]. Se han utilizado todos los tipos de agrupamientos A y B.
- **Tiempo máximo de desplazamiento:** $D_{\max_k} \in [15, 30]$. Se ha fijado el tiempo máximo de desplazamiento entre 15 y 30 minutos.
- **Tiempo máximo de espera:** $W_{\max_k} \in [15, 30, 45]$. Se han establecido tres tiempos máximos de espera, el primero de ellos de 15 minutos, el segundo de 30 y el último de 45 minutos.
- **Ventana temporal:** $t_w \in [0, 5, 10, 20]$. Se han establecido cuatro tipos de ventanas temporales, la primera de ellas sin ventana las demás ventanas temporales se han establecido a intervalos de 5, 10 y 20 minutos respectivamente.
- **Función de evaluación:** $[F_{clust}, F_{cost}]$. Se ha utilizado los dos tipos de funciones de evaluación definidas en 5.1.2. En la primera de ellas F_{clust} se minimiza el número de clústers, en la segunda se minimiza la función F_{cost} que minimiza una estimación del coste de la misma.
- **Parámetros de IACS-HCSP:** Para el proceso de experimentación con la técnica IACS-HCSP se han utilizado numerosos parámetros, cuya representación, ecuación en la que se les referencia y valor se muestran en la tabla 5.5.

Parámetro	Ecuación	Valor
q_0^i	(4.7)	0.1
q_0	(4.8)	0.1
τ_0	(4.11)	0.1
ρ	(4.11)	0.001
α	(4.12)	0.1
γ	(4.20)	0.1

Tabla 5.5: Valor de los parámetros utilizados por la técnica IACS-HCSP.

La combinación de los parámetros anteriores da como resultado un total 96 experimentos, los cuales se ha repetido un total de 5 veces, dando como resultado un total de 480 experimentos cuyo tiempo de cómputo ha ascendido a un total de 121 horas⁷.

⁷Utilizando el servidor descrito en 5.1.3.

5.4.2. Resultados

Al igual que con la técnica anterior en esta sección se exponen los resultados obtenidos para la técnica IACS-HCSP que fue diseñada originalmente para abordar las debilidades de la técnica ACS-HCSP. En la tabla A.3 del apéndice B pueden encontrarse los resultados detallados de cada uno de los 480 experimentos llevados a cabo. Al igual que en la sección anterior, para analizar los distintos resultados se utilizarán los parámetros de tipo de agrupamiento y función de evaluación. Para el tipo de agrupamiento, los servicios que deben prestarse durante el fin de semana pueden resolverse como una sola instancia (Tipo B) o puedan considerarse dos instancias del problema distintas diferenciando entre horario de mañana y de tarde. La función objetivo utilizada también tiene un papel importante tal y como se mostró en la sección 4.3.2.5 del capítulo 4 ya para ambas técnicas es la que determina la aptitud de cada hormiga.

5.4.2.1. Función objetivo: F_{clust}

En la tabla 5.6 se muestran los resultados obtenidos para distintas configuraciones de parámetros cuando la función a objetivo minimizar es el número de clústers representado por F_{clust} , mostrándose la media (μ) y desviación estándar (σ). En la figura 5.21 se representan parte de los resultados de un modo gráfico, en el eje Y se representa el número de clúster requeridos por cada configuración mientras que en el eje X se representan los distintos valores para la restricción de distancia máxima recorrida $D_{max_k} \in [15, 30]$. A su vez y por cada una de las combinaciones de los parámetros de tiempo máximo de espera $W_{max_k} \in [15, 30, 45]$ y ventana temporal $t_w \in [0, 5, 10, 20]$ se ha creado una subgráfica, utilizando para cada una de ellas el mismo rango de valores a fin de facilitar se comprensión. En cada una de estas subgráficas se representa el valor medio ⁸ del número de clústers para cada configuración distinguiendo a su vez por cada tipo de agrupamiento.

Centrándonos en la figura 5.21 se puede observar como no parece existir una diferencia significativa entre los tipos de agrupamiento A y B en las distintas configuraciones. En la misma figura, se puede observar como al aumentar la distancia máxima permitida D_{max_k} (eje X de cada subgráfica) se produce una reducción notable del número de clústers; por ejemplo en el caso de la subgráfica $\{W_{max_k} = 15 | t_w = 0\}$ pasar de 15 minutos de tiempo máximo de desplazamiento a 30 supone pasar de requerir alrededor de 3.982 asistentes a requerir alrededor de 2.968, reducción que se produce en ambos tipos de agrupamientos. Una reducción similar se observa al aumentar el tiempo máximo de espera W_{max_k} de 15 a 30 y para la configuración, $\{W_{max_k} = 15 | t_w = 0\}$ se experimenta una reducción de más de 1.000 asistentes. Finalmente al aumentar la ventana temporal t_w y para una misma configuración, la flexibilidad a la hora de atender a los clientes aumenta y para atender al mismo número de clientes se requiere un conjunto de asistentes significativamente menor.

Respecto al coste computacional este se muestra en figura 5.22 para las distintas configuraciones. De igual modo que los resultados obtenidos en la figura 5.21 se observa que al aumentar la calidad de las soluciones, también aumenta el tiempo requerido para obtenerlas. Por ejemplo pasar de la configuración $\{D_{max_k} = 30 | W_{max_k} = 15 | t_w = 0\}$ a una configuración de $\{D_{max_k} = 45 | W_{max_k} = 15 | t_w = 0\}$ supone incrementar el tiempo de

⁸Dicho valor y sus intervalos de confianza se han estimado realizando bootstrapping.

resolución en más del doble, pasando de 542 a 1.289 segundos.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	$Trav_{obj}$		$Wait_{obj}$		Tot_{obj}		F_{clust}		$T_{ejecucion}$	
				μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
A	15	15	0	98572	369	75161	600	2688348	930	3982	3	311	3
A	15	30	0	132352	622	208091	1634	2855058	1551	2968	4	461	4
A	15	45	0	139053	841	278707	2809	2932375	3310	2754	9	583	3
A	30	15	0	246004	1097	109937	598	2870556	721	2849	4	542	4
A	30	30	0	247571	2025	203185	1432	2965371	2966	2549	6	900	6
A	30	45	0	257266	1001	276964	1739	3048845	1744	2388	5	1289	15
A	15	15	5	113758	892	55683	434	2684056	901	3113	10	375	1
A	15	30	5	135451	877	157827	892	2807893	1702	2593	8	556	4
A	15	45	5	139384	478	215384	766	2869384	407	2466	9	680	3
A	30	15	5	236957	486	75110	1322	2826682	1324	2496	9	694	7
A	30	30	5	237755	1637	151148	1702	2903518	2352	2310	3	1067	11
A	30	45	5	241768	994	207312	1041	2963695	1555	2.240	1	1470	15
A	15	15	10	126196	465	43247	510	2684058	816	2789	7	499	98
A	15	30	10	141791	764	126938	965	2783344	1705	2428	7	647	5
A	15	45	10	144688	705	172519	1594	2831822	1843	2331	6	772	3
A	30	15	10	239901	998	54068	493	2808584	919	2339	7	870	10
A	30	30	10	238749	1.159	117588	1386	2870951	1243	2194	6	1264	8
A	30	45	10	242808	1163	162778	2389	2920201	2480	2146	3	1670	11
A	15	15	20	141668	355	30829	539	2687112	800	2500	9	591	3
A	15	30	20	151393	935	90243	2196	2756396	2881	2264	5	784	6
A	15	45	20	152729	672	127327	1326	2794671	1919	2181	4	921	5
A	30	15	20	251074	1040	36060	313	2801749	937	2165	7	1226	5
A	30	30	20	246563	1479	80741	1251	2841919	2639	2065	7	1654	18
A	30	45	20	248366	2025	114078	1690	2877060	3043	2030	3	2107	15
B	15	15	0	98299	374	75512	635	2688427	441	3965	11	318	8
B	15	30	0	133077	316	209602	2239	2857294	2431	2939	8	464	5
B	15	45	0	139552	612	281006	823	2935173	1198	2727	5	583	8
B	30	15	0	247217	794	109019	666	2872564	4293	2838	16	542	5
B	30	30	0	247941	756	201875	1602	2964431	1545	2531	4	894	7
B	30	45	0	258830	1282	278473	2660	3051919	3386	2371	6	1289	21
B	15	15	5	114163	675	55120	607	2683898	977	3101	9	379	5
B	15	30	5	136680	668	158277	1537	2809572	1915	2580	10	553	6
B	15	45	5	139751	397	213924	1968	2868290	1811	2449	8	677	6
B	30	15	5	237382	1016	75308	1148	2827306	1084	2486	5	693	8
B	30	30	5	237436	1177	150776	1315	2902828	2448	2302	5	1065	13
B	30	45	5	242979	2185	206646	3335	2964240	3866	2226	6	1470	12
B	15	15	10	126773	1137	43044	588	2684432	598	2771	7	493	84
B	15	30	10	141319	720	126347	2394	2782281	3043	2409	3	646	6
B	15	45	10	144648	595	174324	2169	2835300	6070	2318	9	771	8
B	30	15	10	240727	1534	54219	698	2809561	1692	2323	9	866	9
B	30	30	10	240225	2059	117156	1794	2871995	3112	2178	3	1263	13
B	30	45	10	242545	2163	164053	358	2921213	1986	2129	3	1662	12
B	15	15	20	141049	748	30838	305	2686502	935	2491	6	656	158
B	15	30	20	151205	537	91283	1026	2758816	4723	2249	13	783	6
B	15	45	20	152732	805	127101	2484	2794447	2072	2169	6	917	6
B	30	15	20	251637	1071	35810	770	2802062	1501	2157	8	1223	10
B	30	30	20	248490	1686	79752	956	2842857	1679	2045	4	1660	17
B	30	45	20	250020	1278	115699	1137	2880334	1406	2013	4	2094	17

Tabla 5.6: Resultados resumidos obtenidos para IACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{clust} .

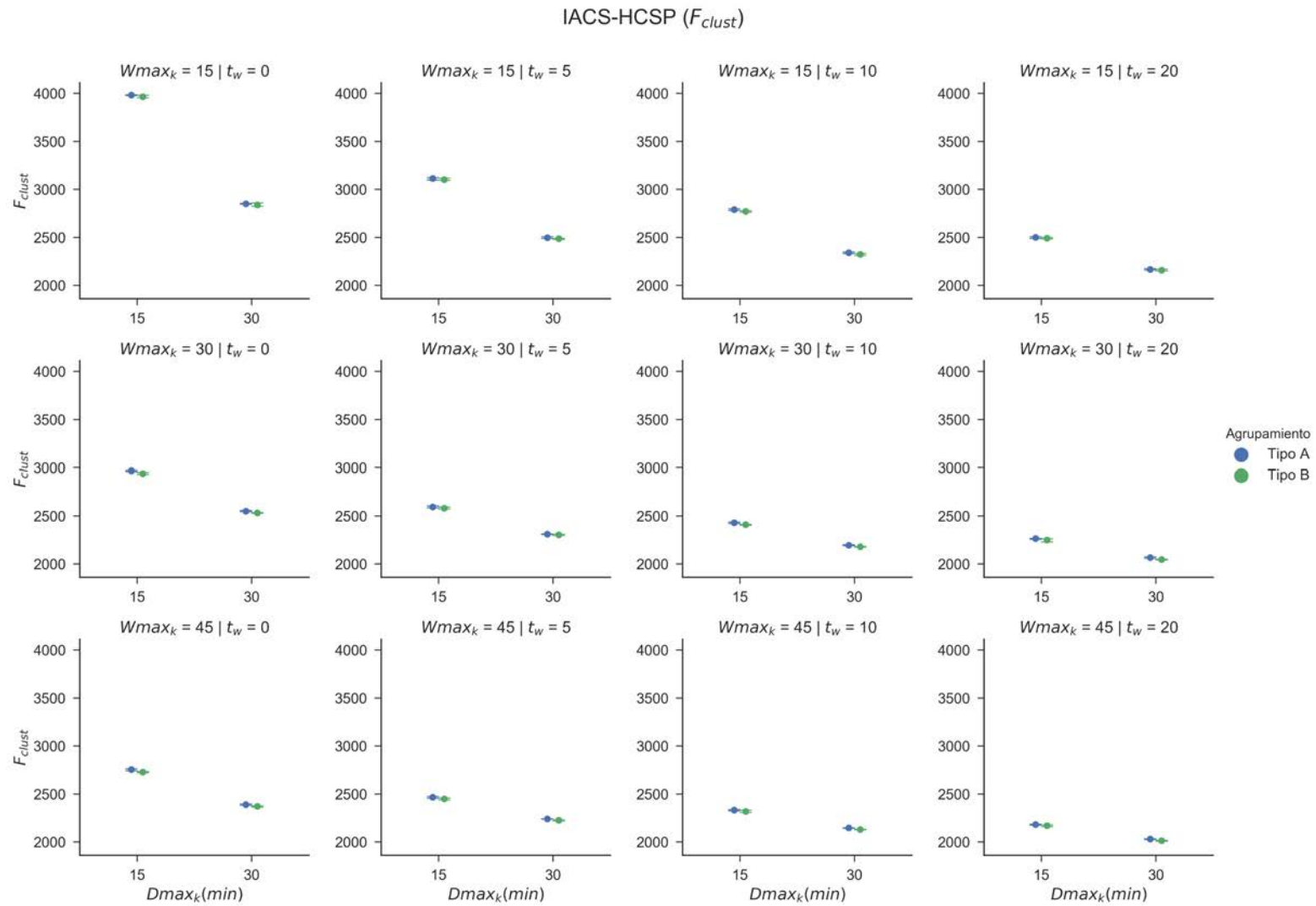


Figura 5.21: Resultados resumidos obtenidos para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust} .

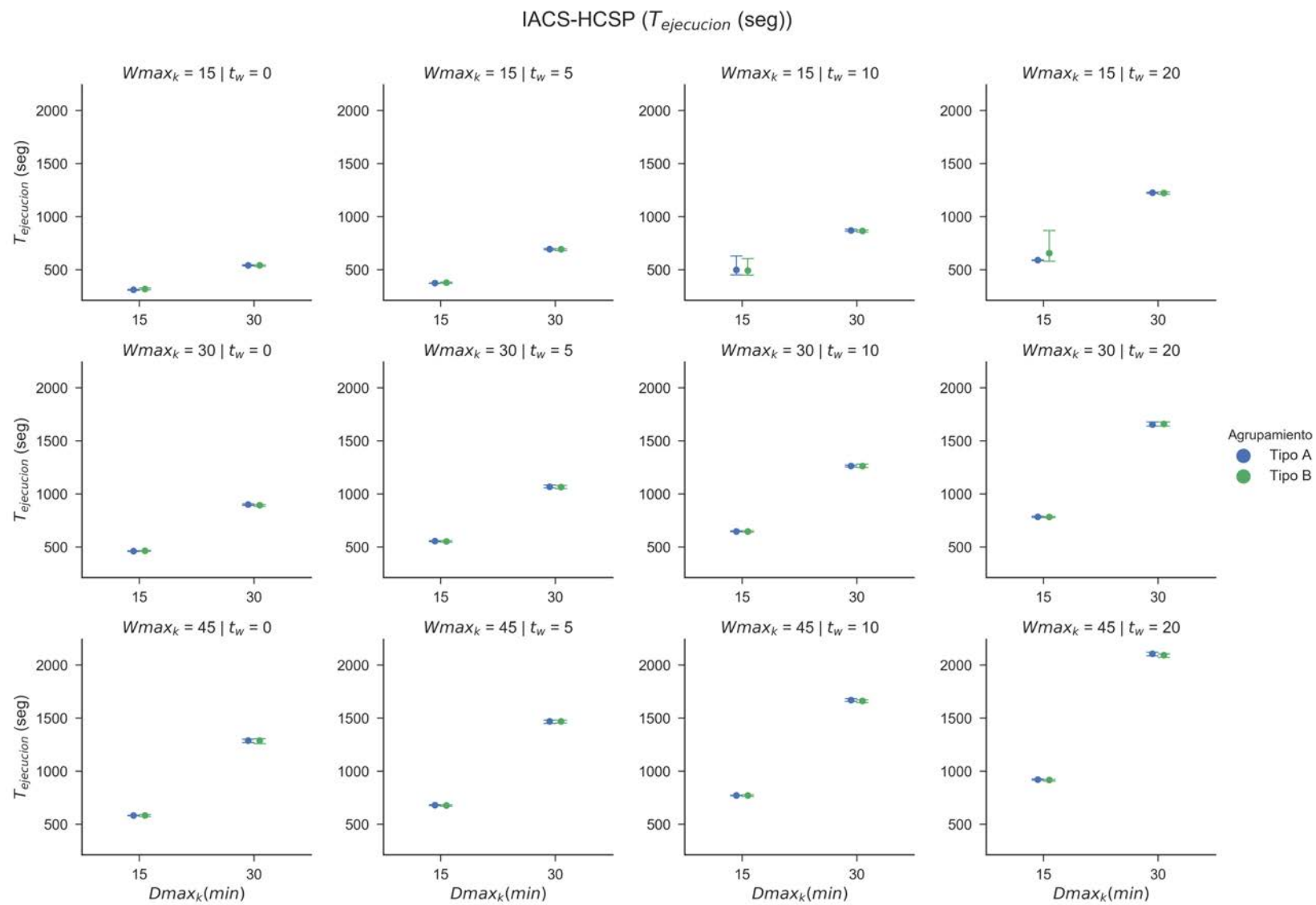


Figura 5.22: Coste computacional (medido en segundos) para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{clust} .

5.4.2.2. Función objetivo: F_{cost}

Una vez medido el objetivo de buscar soluciones que reduzcan el número de clústers, se analizan las soluciones obtenidas cuando el objetivo a minimizar es la función de coste definida en (3.7). En la tabla 5.4 se muestran los resultados obtenidos para distintas configuraciones de parámetros cuando la función a objetivo minimizar es el número de clústers representado por F_{cost} , mostrándose la media (μ) y desviación estándar (σ). En la figura 5.19 se representan parte de los resultados de un modo gráfico, en el eje Y se representa el coste de la solución, mientras que en el eje X se representan los distintos valores para la restricción de distancia máxima recorrida $D_{max_k} \in [15, 30]$. A su vez y por cada una de las combinaciones de los parámetros de tiempo máximo de espera $W_{max_k} \in [15, 30, 45]$ y ventana temporal $t_w \in [0, 5, 10, 20]$ se ha creado una subgráfica, utilizando para cada una de ellas el mismo rango de valores a fin de facilitar se comprensión. En cada una de estas subgráficas se representa el valor medio ⁹ del número de clústers para cada configuración distinguiendo a su vez por cada tipo de agrupamiento.

Centrándonos en la figura 5.19 se puede observar como no parece existir una diferencia significativa entre los tipos de agrupamiento A y B en las distintas configuraciones. En la misma figura, se puede observar como al aumentar la distancia máxima permitida D_{max_k} (eje X de cada subgráfica) se produce un aumento del coste de la solución; por ejemplo en el caso de la subgráfica $\{W_{max_k} = 15 | t_w = 0\}$ pasar de 15 minutos de D_{max_k} a 30 supone un incremento de alrededor de 7.000 € semanales. En el caso de aumentar la ventana temporal t_w (conservando la misma configuración) se produce una disminución bastante significativa, ya que al aumentar la flexibilidad a la hora de atender los distintos servicios los tiempos de espera se reducen de tal modo que el coste global de la solución disminuye. Por ejemplo pasar de la configuración $\{D_{max_k} = 15 | W_{max_k} = 15 | t_w = 0\}$ a la configuración $\{D_{max_k} = 15 | W_{max_k} = 15 | t_w = 20\}$ supone un ahorro de costes de más de 17.000 € semanales.

Respecto al tiempo de ejecución, de igual modo que los resultados obtenidos en la figura 5.17 se observa que no existen diferencias muy significativas a la hora de resolver las instancias utilizando el tipo de agrupamiento A o el B. Se observa que, al aumentar la calidad de las soluciones, también aumenta el tiempo requerido para obtenerlas. Por ejemplo pasar de la configuración $\{D_{max_k} = 15 | W_{max_k} = 45 | t_w = 20\}$ a una configuración de $\{D_{max_k} = 30 | W_{max_k} = 45 | t_w = 20\}$ supone incrementar el tiempo de resolución en más del doble, pasando de 910 a 2.020 segundos.

⁹Dicho valor y sus intervalos de confianza se han estimado realizando bootstrapping.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	$Trav_{obj}$		$Wait_{obj}$		Tot_{obj}		F_{cost}		$T_{ejecucion}$	
				μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
A	15	15	0	75951	1409	56831	767	2647396	2161	665214	112	327	4
A	15	30	0	83961	336	102479	983	2701054	877	673718	75	464	4
A	15	45	0	84827	763	114396	1547	2713838	2285	676377	253	579	4
A	30	15	0	131228	1950	65112	1204	2711100	2982	675146	281	536	5
A	30	30	0	121879	2695	90855	2863	2727349	5530	678858	531	880	5
A	30	45	0	118936	2579	98515	2157	2732066	4558	680620	368	1260	11
A	15	15	5	86752	1230	37090	1085	2640170	4938	654472	1039	380	9
A	15	30	5	87204	783	65555	1126	2667374	1628	660089	79	550	5
A	15	45	5	89682	1245	76118	1979	2680414	3165	662241	252	671	2
A	30	15	5	128075	1636	39417	649	2682107	2216	662297	181	683	5
A	30	30	5	121382	2373	59522	1298	2695519	3466	665510	238	1054	4
A	30	45	5	119450	836	65678	674	2699742	873	667143	69	1453	5
A	15	15	10	95132	905	27276	595	2638736	4782	651093	943	499	102
A	15	30	10	95630	438	50281	628	2660527	823	655623	104	633	5
A	15	45	10	96051	871	57933	1250	2670312	4179	657619	1088	763	5
A	30	15	10	134600	1141	27787	501	2677002	909	658249	161	854	5
A	30	30	10	128823	2512	44923	1307	2688361	3746	661114	239	1241	12
A	30	45	10	127269	831	50838	1242	2692722	1939	662437	183	1654	14
A	15	15	20	105299	1034	18230	447	2638144	1390	647746	122	581	6
A	15	30	20	105596	745	35868	726	2656079	975	651193	197	775	3
A	15	45	20	106299	884	42303	677	2663217	1482	652706	124	910	6
A	30	15	20	145533	763	18580	294	2678728	928	655229	137	1209	7
A	30	30	20	142829	2683	32598	1199	2690042	3177	657411	261	1643	24
A	30	45	20	138007	2218	36825	1374	2691160	4804	658876	996	2020	111
B	15	15	0	77320	1110	57851	1121	2649786	2204	665186	195	333	9
B	15	30	0	83966	1066	102191	1234	2700922	2333	673638	254	467	8
B	15	45	0	85030	1703	114076	2093	2713721	3758	676198	308	583	8
B	30	15	0	133206	1006	65942	370	2713763	1263	675290	111	539	6
B	30	30	0	123566	1242	92655	831	2730836	1709	678988	233	883	7
B	30	45	0	119559	3728	99768	3845	2733943	7487	680604	567	1260	18
B	15	15	5	86328	1266	36368	651	2637311	1862	654020	180	380	5
B	15	30	5	88215	528	67021	324	2669851	458	660126	77	549	7
B	15	45	5	88657	727	76145	482	2681004	3353	662573	1001	675	7
B	30	15	5	128422	1466	39316	242	2682353	1404	662295	87	682	5
B	30	30	5	121198	2747	58943	2370	2694756	5055	665613	413	1053	12
B	30	45	5	118702	3305	65128	2223	2698445	5466	667043	357	1449	15
B	15	15	10	96984	1487	27814	934	2639413	2365	650554	121	504	114
B	15	30	10	96131	1086	50869	712	2661615	1771	655545	132	635	8
B	15	45	10	96700	983	58072	1277	2669387	2117	657250	141	764	6
B	30	15	10	134912	505	28365	522	2677892	862	658139	131	852	8
B	30	30	10	131322	2758	46131	285	2692067	3010	661236	335	1236	13
B	30	45	10	129787	2135	52138	1182	2696539	3159	662597	204	1654	18
B	15	15	20	107517	810	18831	260	2640962	983	647745	81	582	8
B	15	30	20	106120	936	35985	395	2656720	1005	651232	99	771	6
B	15	45	20	106574	629	42285	515	2665187	3859	653129	1057	909	6
B	30	15	20	145997	1630	18789	813	2679401	1932	655348	149	1205	6
B	30	30	20	142804	1996	32460	804	2689878	2732	657437	128	1625	20
B	30	45	20	138325	4024	37070	1721	2690010	5626	658484	183	2040	39

Tabla 5.7: Resultados resumidos obtenidos para IACS-HCSP para el tipo de agrupamiento A y función de evaluación F_{cost} .

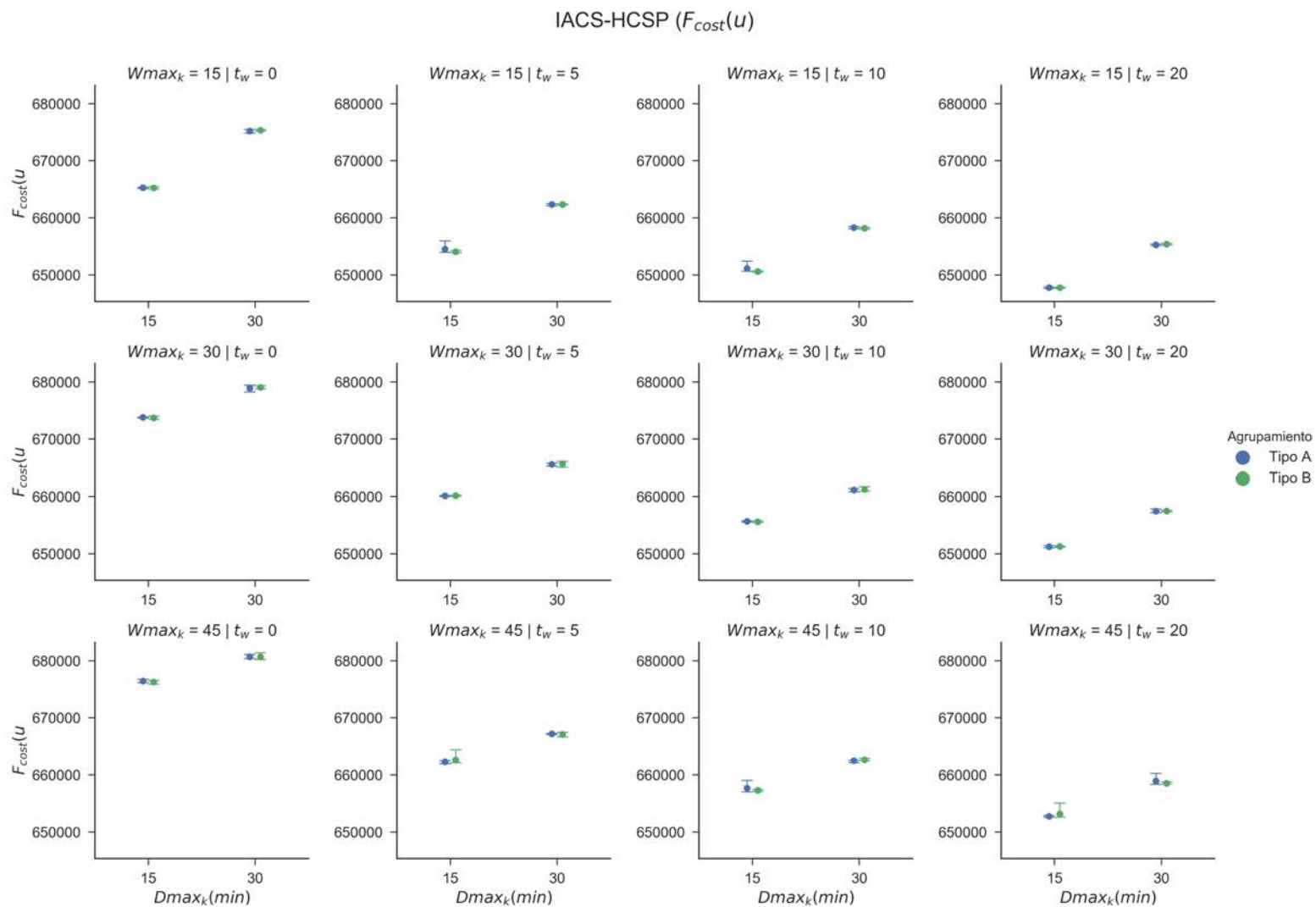


Figura 5.23: Resultados resumidos obtenidos para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost} .

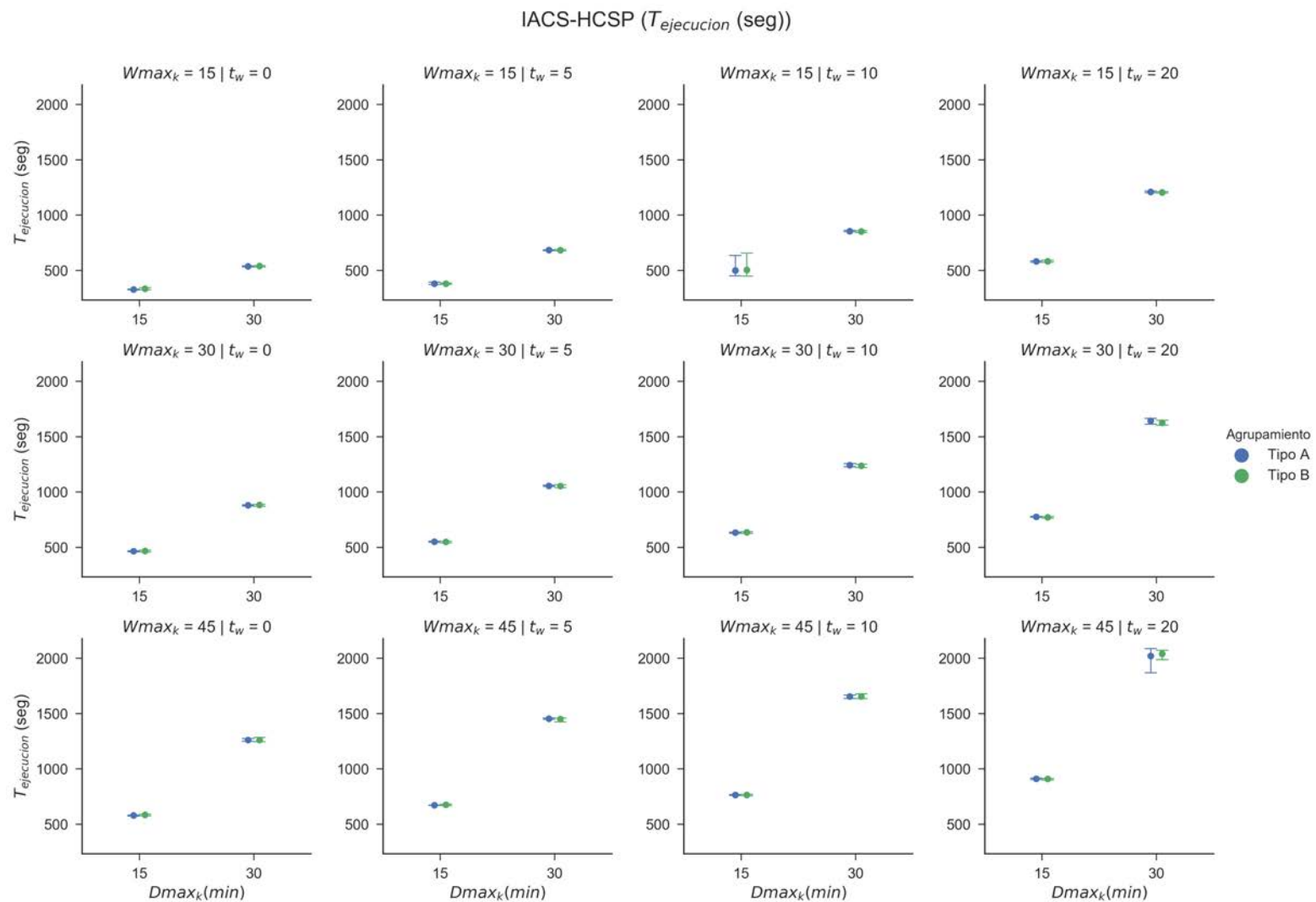


Figura 5.24: Coste computacional (medido en segundos) para IACS-HCSP y sus distintas parametrizaciones utilizando la función de evaluación F_{cost} .

5.5. Análisis comparativo

En esta sección se lleva a cabo un análisis comparativo entre las distintas técnicas propuestas. En primer lugar se comienza comparando las técnicas basadas en ACO, ACS-HCSP y IACS-HCSP, con el método de Ward modificado. Posteriormente se comparan las técnicas ACS-HCSP entre sí para las distintas configuraciones y finalmente se comparan las técnicas propuestas con la solución actual de la compañía y por la propuesta por [Quintana et al. \(2017\)](#).

5.5.1. Método de Ward modificado

Como se mostró en la sección 5.2.1 el método de Ward fue objeto de experimentación para un total de 6 tipos de configuraciones distintas, las cuales se repitieron un total de 10 veces. De entre las distintas configuraciones solamente dos ellas son directamente comparables con los experimentos realizados con ACS-HCSP y IACS-HCSP, en concreto con las siguientes configuraciones $\{TipoA|D_{max_k} = 30|W_{max_k} = 30|t_w = 0\}$ y $\{TipoB|D_{max_k} = 30|W_{max_k} = 30|t_w = 0\}$. Puesto que existe una diferencia tan grande entre los resultados obtenidos por el método de Ward y las técnicas basadas en ACO, no se requeriría de ningún test de contraste de hipótesis, ya que los resultados de cada método están separados unos de otros cientos de desviaciones típicas tal y como se puede ver en la figura 5.25. En todo caso, para asegurar que las técnicas propuestas tienen distintos rendimientos se procede a realizar dicha comprobación. En primer lugar, se debe comprobar si dichas poblaciones de resultados proceden de una distribución normal o no, para realizar esta comprobación se utiliza el test propuesto por ([Shapiro y Wilk, 1965](#)). El test establece como hipótesis nula que la población bajo prueba procede de una distribución normal, para rechazar dicha hipótesis será necesario obtener un p-valor que sea inferior a cierto nivel de significatividad, que en nuestro caso será igual 0.05. En la tabla 5.8 se muestran los resultados obtenidos en dicho test para cada de las técnicas y las configuraciones que serán objeto de prueba, puesto que todos los valores tienen un p-valor <0.05 no podemos rechazar la hipótesis de que cada una de dichas poblaciones procede de una distribución normal. Una vez comprobado que las muestras proceden de una población normal, se procede a realizar el test ANOVA, en dicho test la hipótesis nula es que las medias de las tres poblaciones son iguales, un p-valor menor de 0.05 indicaría que no lo son. En la tabla 5.9 se puede observar como los p-valores obtenidos para cada una de las dos configuraciones probadas están muy próximas a cero, hecho que es consistente con los resultados mostrados en la figura 5.25 en la cual se aprecia de un modo claro que la técnica que mejores resultados obtiene para dicha configuración y para la función objetivo F_{clust} es IACS-HCSP, requiriendo casi la mitad de clústers. Como se ha expuesto en la sección 5.1.1 el problema a resolver cuenta con objetivos enfrentados, este hecho se puede apreciar en la figura 5.26 en la cual se muestra el coste que hubieran tenido las soluciones obtenidas en la figura 5.25. Dicha comparación se muestra solamente para ilustrar como el reducir el número de clústers(asistentes) implica necesariamente aumentar el tiempo que se requiere a dichos asistentes, de hecho se puede apreciar como las figuras parecen estar invertidas. Dichas comparaciones no se deben tener en cuenta ya que tanto las técnica ACS-HCSP como IACS-HCSP no tenían como función objetivo reducir la función F_{cost} .

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	Técnica	est	p-valor
A	30	30	0	Ward	0.914	0.313
A	30	30	0	ACS-HCSP	0.911	0.4759
A	30	30	0	IACS-HCSP	0.89	0.23
B	30	30	0	Ward	0.898	0.210
B	30	30	0	ACS-HCSP	0.814	0.313
B	30	30	0	IACS-HCSP	0.952	0.758

Tabla 5.8: Resultados del test de normalidad Shapiro-Wilk para distintas configuraciones de las tres técnicas propuestas.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{clust}			est	p-valor
				Ward	ACS-HCSP	IACS-HCSP		
A	30	30	0	4963	3881	2549	158277	5.07E-37
B	30	30	0	4959	3866	2531	112680	9.10E-36

Tabla 5.9: Resultados obtenidos al realizar un test ANOVA para cada una de las tres poblaciones y configuraciones descritas.

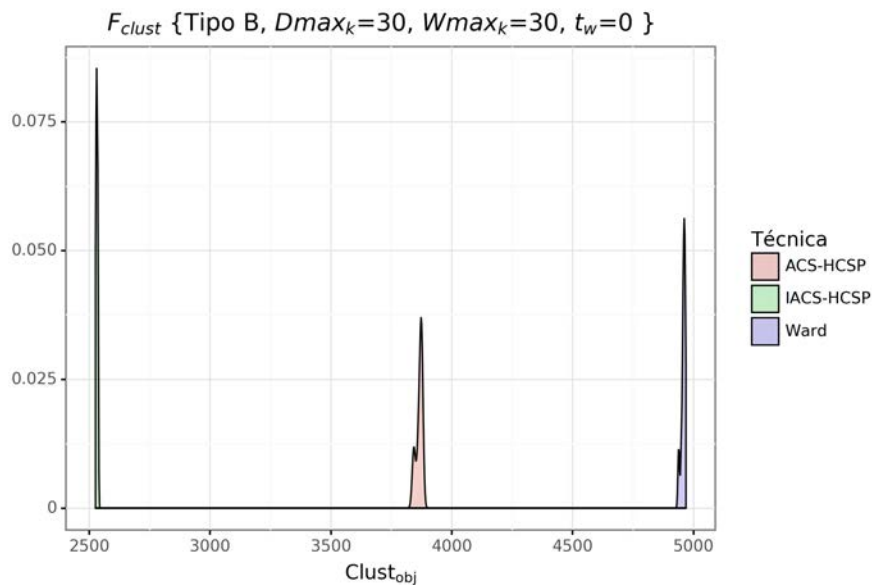


Figura 5.25: Función de densidad para los tres métodos propuestos, tipo de agrupamiento B y función de coste F_{clust} .

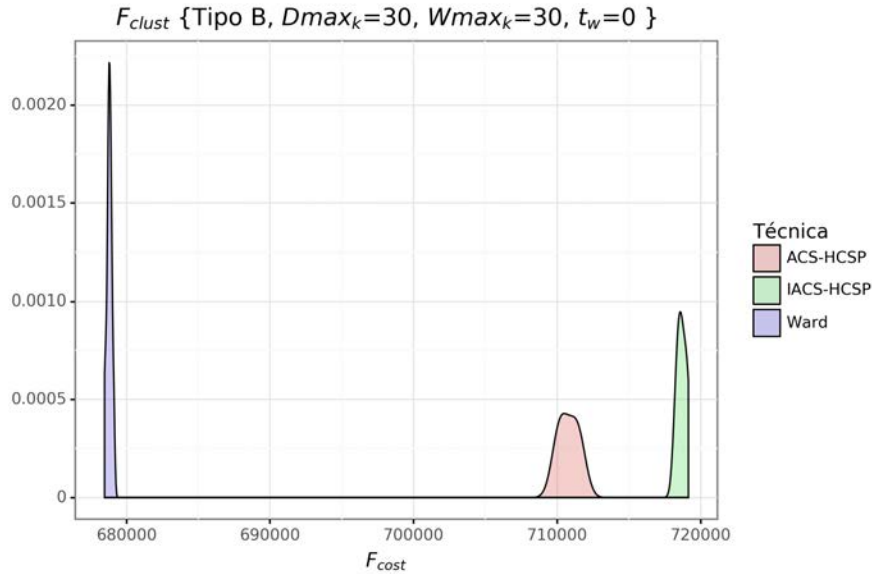


Figura 5.26: Función de densidad sobre el coste de las soluciones obtenidas en la figura 5.25.

5.5.2. ACS-HCSP y IACS-HCSP

En la sección anterior se mostró como el método de Ward obtenía resultados de menor calidad que las técnicas propuestas basadas en ACO. A su vez también se mostró como para las dos configuraciones analizadas en dicho apartado, la técnica IACS-HCSP obtenía mejores resultados que ACS-HCSP, en esta sección se expande dicho análisis a las 96 configuraciones distintas, distinguiéndose los resultados en los que se trata de minimizar el número de asistentes requeridos F_{clust} , de los que tratan de minimizar el coste total de la solución F_{cost} .

5.5.2.1. Función objetivo: F_{clust}

En la tabla 5.10 se muestran los resultados obtenidos para distintas configuraciones de parámetros, utilizando como función objetivo minimizar el número de clústers representado por F_{clust} , así mismo también se analiza el coste computacional requerido en cada uno de los experimentos, representado por $T_{ejecucion}$. En dicha tabla, puede apreciarse claramente como para todas las configuraciones posibles la técnica IACS-HCSP obtiene mejores resultados que ACS-HCSP para la función de evaluación F_{clust} , dicha mejora en los resultados se obtiene a cambio de obtener soluciones en un mayor tiempo, que en algunas ocasiones llega a suponer el doble. Estas diferencias en la calidad de las soluciones obtenidas y de tiempo de ejecución, se representan gráficamente en las figuras 5.27, 5.29, 5.28 y 5.30, en las cuales queda patente que la diferencia entre las soluciones obtenidas parece ser significativa desde el punto de vista estadístico.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{clust}		$T_{ejecucion}$	
				ACS-HCSP	IACS-HCSP	ACS-HCSP	IACS-HCSP
				$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$
A	15	15	0	6237±12	3982±3	265±2	311±3
A	15	15	5	4616±7	3113±10	281±1	375±1
A	15	15	10	4203±9	2789±7	316±1	499±98
A	15	15	20	3865±11	2500±9	377±2	591±3
A	15	30	0	4418±16	2968±4	320±1	461±4
A	15	30	5	3984±9	2593±8	361±1	556±4
A	15	30	10	3807±20	2428±7	409±2	647±5
A	15	30	20	3570±11	2264±5	480±2	784±6
A	15	45	0	4182±7	2754±9	380±1	583±3
A	15	45	5	3839±6	2466±9	428±2	680±3
A	15	45	10	3674±15	2331±6	476±1	772±3
A	15	45	20	3462±9	2181±4	552±1	921±5
A	30	15	0	4220±7	2849±4	357±1	542±4
A	30	15	5	3839±9	2496±9	437±2	694±7
A	30	15	10	3661±12	2339±7	531±2	870±10
A	30	15	20	3430±22	2165±7	721±4	1226±5
A	30	30	0	3881±9	2549±6	550±3	900±6
A	30	30	5	3627±8	2310±3	634±3	1067±11
A	30	30	10	3472±5	2194±6	734±4	1264±8
A	30	30	20	3283±9	2065±7	932±3	1654±18
A	30	45	0	3691±5	2388±5	753±5	1289±15
A	30	45	5	3493±7	2.240±1	840±4	1470±15
A	30	45	10	3380±3	2146±3	938±3	1670±11
A	30	45	20	3187±6	2030±3	1146±1	2107±15
B	15	15	0	6230±14	3965±11	262±6	318±8
B	15	15	5	4608±7	3101±9	281±1	379±5
B	15	15	10	4191±10	2771±7	315±1	493±84
B	15	15	20	3856±3	2491±6	375±2	656±158
B	15	30	0	4421±29	2939±8	319±2	464±5
B	15	30	5	3972±12	2580±10	359±1	553±6
B	15	30	10	3793±7	2409±3	404±1	646±6
B	15	30	20	3553±8	2249±13	475±2	783±6
B	15	45	0	4162±8	2727±5	377±1	583±8
B	15	45	5	3825±9	2449±8	423±1	677±6
B	15	45	10	3646±9	2318±9	471±1	771±8
B	15	45	20	3454±8	2169±6	544±3	917±6
B	30	15	0	4213±11	2838±16	355±1	542±5
B	30	15	5	3826±5	2486±5	430±2	693±8
B	30	15	10	3646±10	2323±9	526±3	866±9
B	30	15	20	3422±9	2157±8	714±4	1223±10
B	30	30	0	3866±14	2531±4	542±3	894±7
B	30	30	5	3617±17	2302±5	627±4	1065±13
B	30	30	10	3467±4	2178±3	725±5	1263±13
B	30	30	20	3272±7	2045±4	921±3	1660±17
B	30	45	0	3669±8	2371±6	746±3	1289±21
B	30	45	5	3481±5	2226±6	832±2	1470±12
B	30	45	10	3357±15	2129±3	935±3	1662±12
B	30	45	20	3176±6	2013±4	1133±3	2094±17

Continúa en la siguiente hoja

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{clust}		$T_{ejecucion}$	
				ACS-HCSP	IACS-HCSP	ACS-HCSP	IACS-HCSP
				$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$

Tabla 5.10: Comparación de resultados entre ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} .

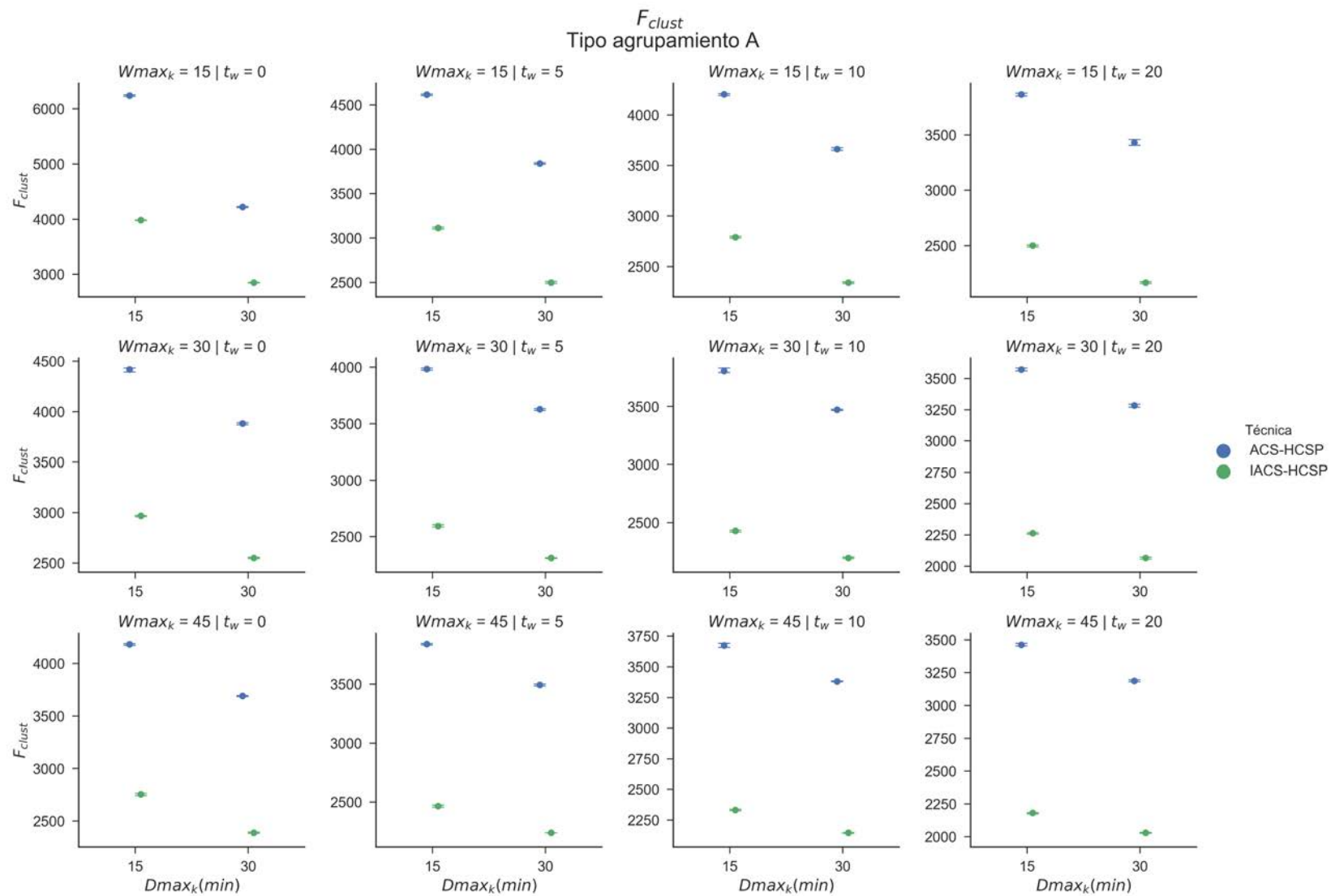


Figura 5.27: Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento A.

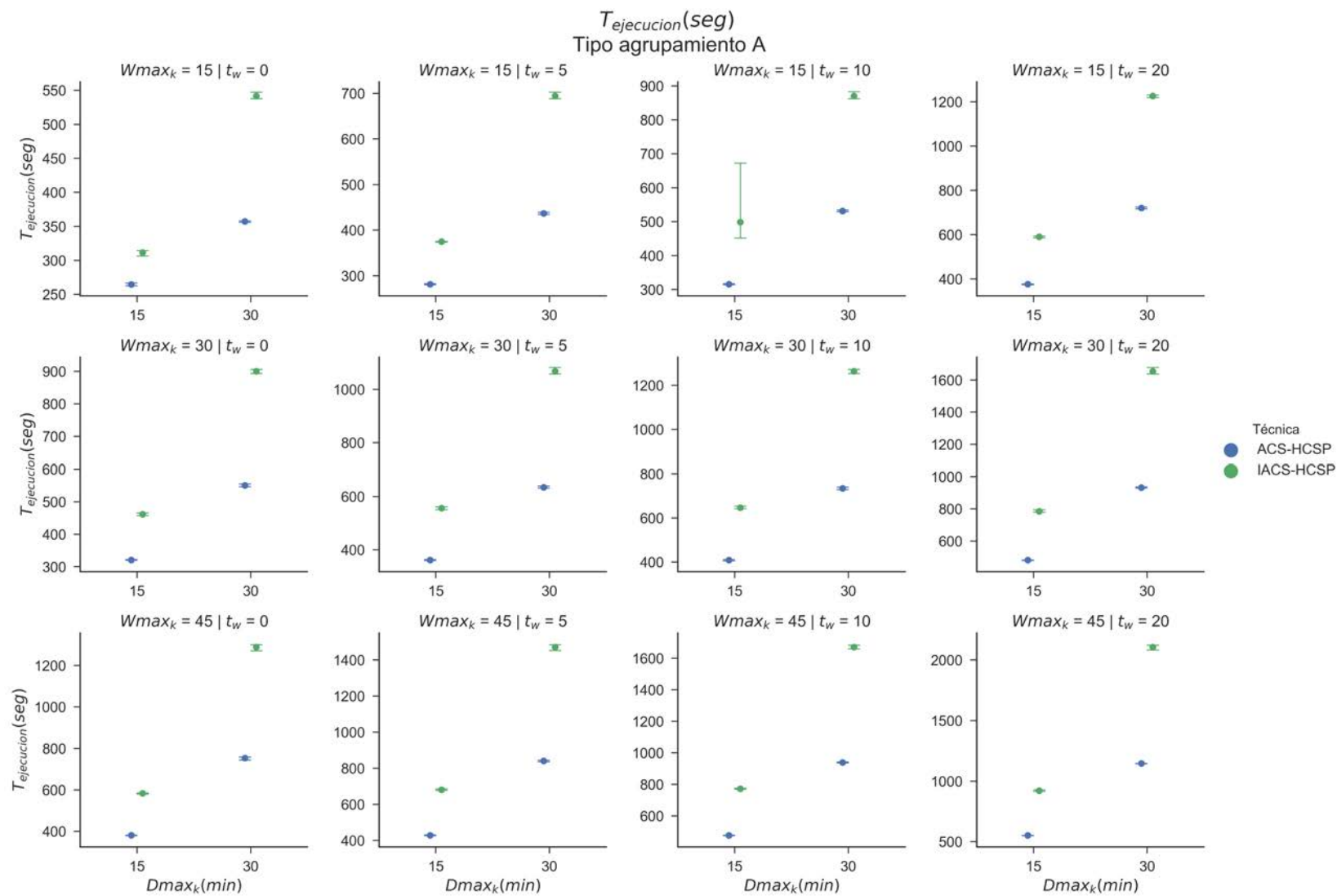


Figura 5.28: Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento A.

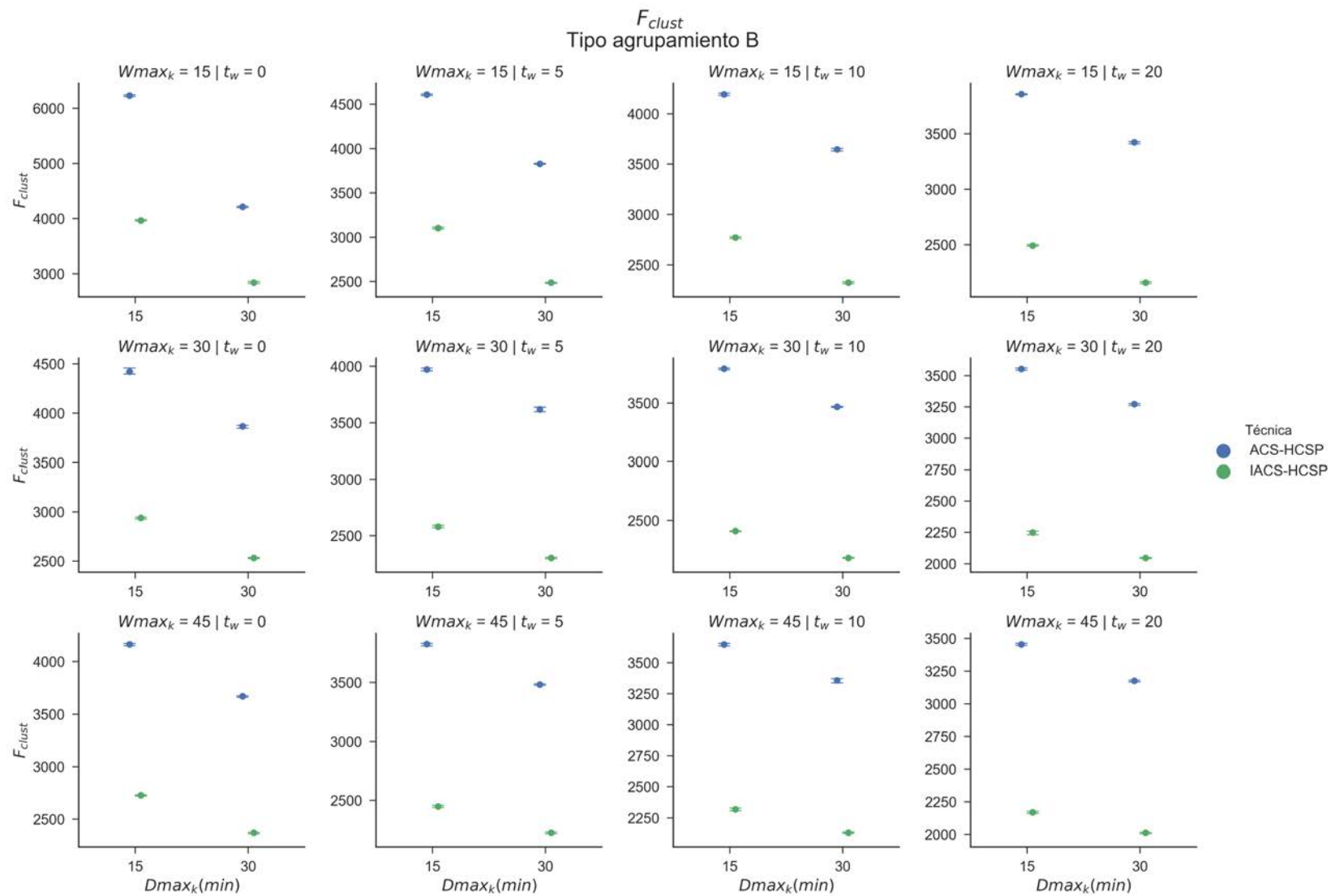


Figura 5.29: Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento B.

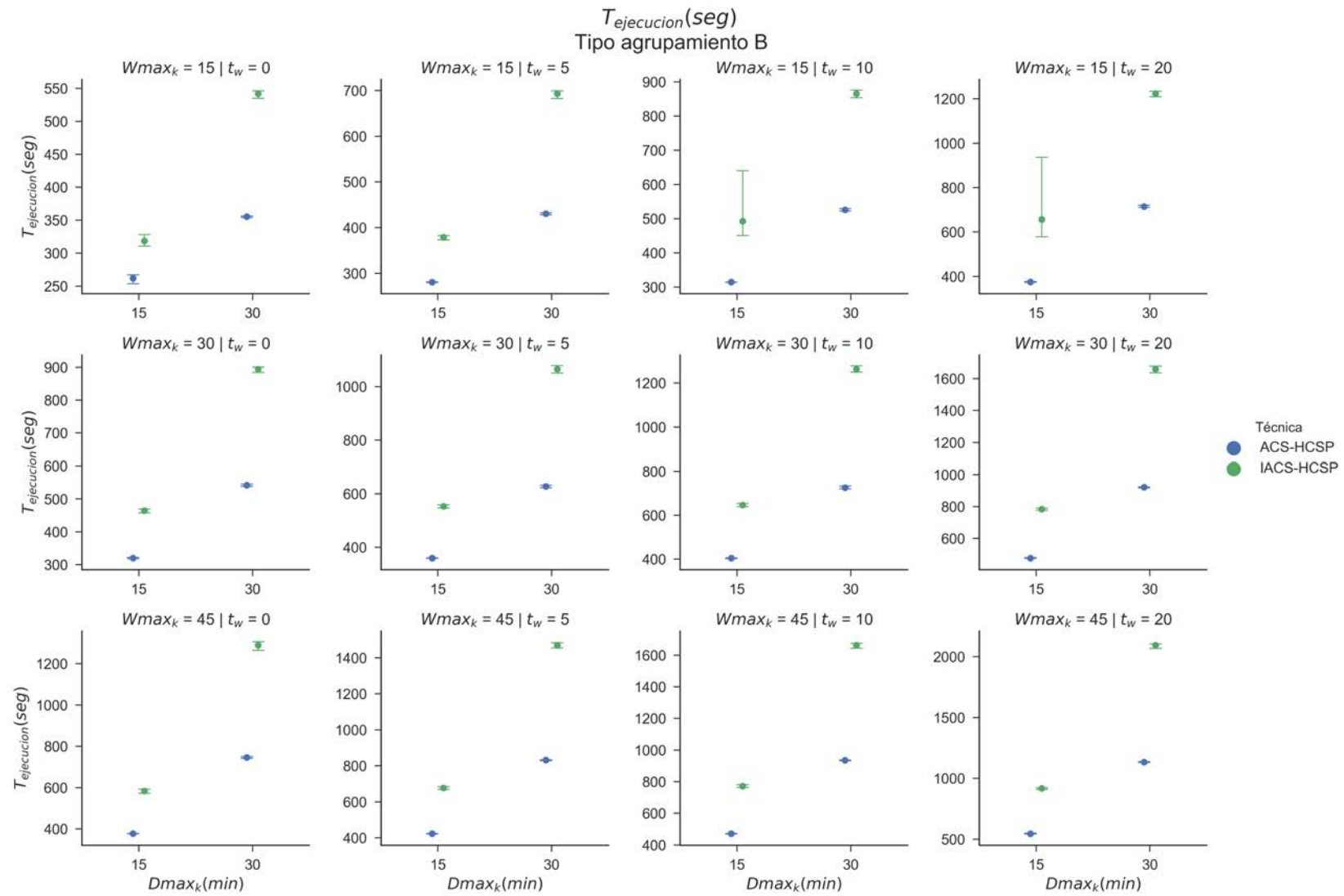


Figura 5.30: Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} y el tipo de agrupamiento B.

A pesar de que en las gráficas anteriores parece claro que los resultados de IACS-HCSP son significativamente superiores a los de ACS-HCSP, queda la incógnita si lo son para todas las soluciones que obtienen cada una de las técnicas, o solo lo son como resultado final. Puesto que solo estamos reportando las 5 mejores soluciones encontradas por cada técnica y cada parametrización, sería posible que las poblaciones de soluciones de las diferentes rondas de evolución no fueran diferentes. Por ejemplo, si en la técnica IACS-HCSP la mejora respecto a ACS-HCSP solo se lograra al final de la última ronda de evolución, puesto que solo estamos analizando las 5 mejores soluciones y no las 32.000 soluciones de cada configuración, sería posible que afirmásemos que la técnica IACS-HCSP es superior a ACS-HCSP cuando en realidad esa mejora solo se produce al final de la última ronda. En este punto es conveniente recordar que las soluciones reportadas hacen referencia a una solución completa, la cual implica varios grupos de servicios a lo largo de toda la semana, siendo dichos grupos resueltos de modo independiente por colonias distintas. Las soluciones reportadas hasta el momento se han obtenido al agregar los resultados obtenidos para cada grupo, con el objetivo de validar la hipótesis anterior se procede a examinar la evolución de las soluciones encontradas por cada técnica para el grupo de servicios Grupo 1 ¹⁰.

Con el objetivo de validar dicha hipótesis se llevan a cabo pruebas de contraste estadístico, en concreto se utilizará la prueba t de Student, ya que las muestras son independientes y el tamaño de la muestra para cada experimento es de 32.000. Con dicho tamaño y apoyados en el teorema central del límite, podemos obviar los test necesarios para asegurar que cada una de las muestras siguen una distribución normal. Al igual que hicieramos en apartados anteriores, nuestra hipótesis nula es que las poblaciones tienen igual media, fijando como p-valor el valor 0.05. Los resultados reflejados en la tabla 5.11 muestran tanto el valor del estadístico de la t-Student como sus p-valores asociados; se puede apreciar como todos los p-valores son iguales a cero, no se trata de ningún error simplemente el valor es tan cercano a cero que al realizar los cálculos el ordenador redondea a dicha cifra, por lo tanto debemos rechazar la hipótesis nula y concluir que las medias de las dos técnicas propuestas son diferentes.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{clust}		Est. t	p-valor
				ACS-HCSP	IACS-HCSP		
A	15	15	0	4898	3.812	345	0
A	15	15	5	3566	2467	347	0
A	15	15	10	3283	2169	400	0
A	15	15	20	3052	1957	359	0
A	15	30	0	3494	2368	334	0
A	15	30	5	3167	2021	427	0
A	15	30	10	3037	1895	497	0
A	15	30	20	2880	1771	468	0
A	15	45	0	3348	2179	433	0
A	15	45	5	3091	1926	536	0
A	15	45	10	2970	1825	502	0

Continúa en la siguiente hoja

¹⁰Que incluye a un total de 10.709 servicios que son prestados de lunes a viernes entre las 07:00 y las 14:30.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{clust}		Est. t	p-valor
				ACS-HCSP	IACS-HCSP		
A	15	45	20	2822	1719	449	0
A	30	15	0	3393	2279	387	0
A	30	15	5	3117	1990	396	0
A	30	15	10	2985	1872	442	0
A	30	15	20	2829	1739	489	0
A	30	30	0	3183	2030	478	0
A	30	30	5	2992	1826	553	0
A	30	30	10	2886	1742	553	0
A	30	30	20	2750	1644	562	0
A	30	45	0	3058	1882	534	0
A	30	45	5	2912	1758	553	0
A	30	45	10	2814	1686	604	0
A	30	45	20	2680	1603	606	0
B	15	15	0	4907	3822	278	0
B	15	15	5	3559	2468	330	0
B	15	15	10	3281	2172	329	0
B	15	15	20	3045	1944	412	0
B	15	30	0	3498	2352	439	0
B	15	30	5	3169	2030	414	0
B	15	30	10	3046	1903	420	0
B	15	30	20	2890	1770	467	0
B	15	45	0	3341	2164	486	0
B	15	45	5	3079	1927	511	0
B	15	45	10	2965	1826	476	0
B	15	45	20	2832	1718	511	0
B	30	15	0	3394	2281	387	0
B	30	15	5	3102	1989	402	0
B	30	15	10	2985	1867	408	0
B	30	15	20	2832	1729	560	0
B	30	30	0	3176	2028	486	0
B	30	30	5	2983	1834	512	0
B	30	30	10	2883	1741	550	0
B	30	30	20	2745	1636	617	0
B	30	45	0	3056	1877	526	0
B	30	45	5	2903	1758	591	0
B	30	45	10	2818	1684	628	0
B	30	45	20	2673	1604	567	0

Tabla 5.11: Prueba t-Student para las poblaciones de ACS-HCSP y IACS-HCSP para el Grupo 1 de servicios y función de evaluación F_{clust} .

Para ilustrar la diferencia clara de rendimiento entre las técnicas ACS-HCSP y IACS-HCSP, en la figura 5.31 se muestra la distribución de las soluciones encontradas por cada una de las técnicas con la configuración $\{F_{cost}, TipoA, Dmax_k = 30, Wmax_k = 45, t_w = 20\}$, se puede apreciar claramente como las dos distribuciones están perfectamente separadas. En este caso las soluciones pertenecen al Grupo 1, es decir representan al grupo más numeroso de servicios con un total de 10.709. Otro aspecto que podemos analizar para esta

misma configuración, es la evolución de las soluciones y ver la relación que tiene con la propia distribución que observamos. En la figura 5.32 se muestra la evolución de uno de los experimentos para la misma configuración anterior, en ella se puede observar como la técnica IACS-HCSP obtiene mejores soluciones desde las primeras rondas de evolución, de hecho más del 99 % de las soluciones encontradas por la técnica IACS-HCSP son mejores soluciones que la mejor solución encontrada por ACS-HCSP.

Con todas estas evidencias, podemos afirmar que las soluciones proporcionadas por la técnica IACS-HCSP para minimizar el número de asistentes, es superior al de la técnica ACS-HCSP. Esta mejora en calidad de las soluciones, va acompañada de un incremento en el coste computacional, y por lo tanto en tiempo, el cual está causado por la definición de vértices alcanzables utilizada por IACS-HCSP (ver ecuación (4.16)). La técnica IACS-HCSP considerar un mayor número de nodos a los que cada hormiga puede transitar por lo que resulta razonable que los tiempos de cada hormiga para encontrar una solución aumenten.

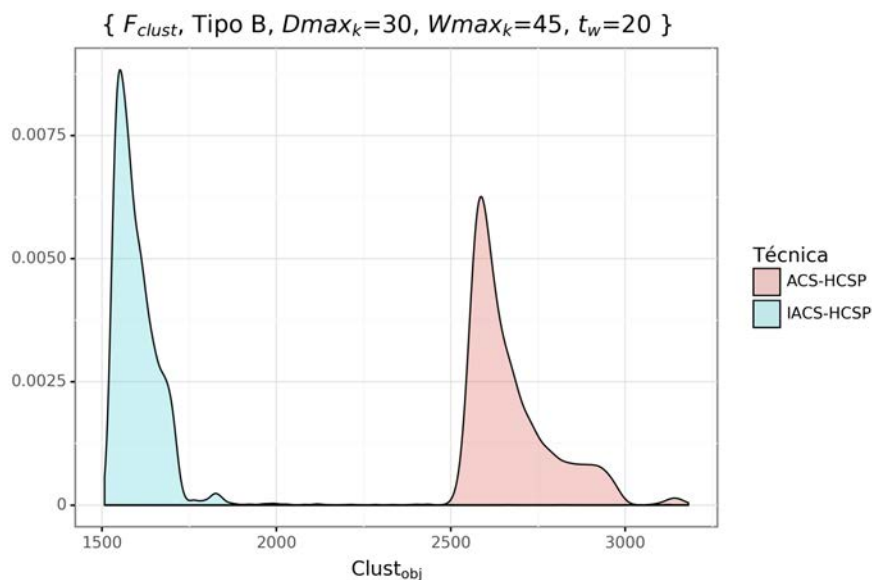


Figura 5.31: Distribuciones de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP durante las 100 rondas de evolución para la función de evaluación F_{clust} , el tipo de agrupamiento B y el grupo de servicios 1.

5.5.2.2. Función objetivo: F_{cost}

En la tabla 5.12 se muestran los resultados obtenidos para distintas configuraciones de parámetros, utilizando como función objetivo minimizar el coste de la solución representado por F_{cost} , así mismo también se analiza el coste computacional requerido en cada uno de los experimentos, representado por $T_{ejecucion}$. En dicha tabla, puede apreciarse claramente como para todas las configuraciones posibles la técnica IACS-HCSP obtiene mejores resultados que ACS-HCSP para la función de evaluación F_{cost} , dicha mejora en los resultados se obtiene a cambio de obtener soluciones en un mayor tiempo, que en algunas ocasiones llega a suponer el doble. Estas diferencias en la calidad de las soluciones obtenidas y de tiempo

de ejecución, se representan gráficamente en las figuras 5.33, 5.35, 5.34 y 5.36, en las cuales queda patente que la diferencia entre las soluciones obtenidas parece ser significativa desde el punto de vista estadístico.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{cost}		$T_{ejecucion}$	
				ACS-HCSP	IACS-HCSP	ACS-HCSP	IACS-HCSP
				$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$
A	15	15	0	684303±47	665214±112	264±0	327±4
A	15	15	5	666555±1128	654472±1039	281±0	380±9
A	15	15	10	661870±66	651093±943	315±2	499±102
A	15	15	20	658749±122	647746±122	378±1	581±6
A	15	30	0	691940±164	673718±75	320±0	464±4
A	15	30	5	679097±249	660089±79	362±3	550±5
A	15	30	10	672937±295	655623±104	408±2	633±5
A	15	30	20	666734±154	651193±197	480±2	775±3
A	15	45	0	699400±141	676377±253	378±2	579±4
A	15	45	5	685919±101	662241±252	428±2	671±2
A	15	45	10	679019±254	657619±1088	476±2	763±5
A	15	45	20	671905±219	652706±124	551±3	910±6
A	30	15	0	692433±242	675146±281	357±1	536±5
A	30	15	5	680832±185	662297±181	435±2	683±5
A	30	15	10	676009±145	658249±161	532±1	854±5
A	30	15	20	672717±1221	655229±137	721±5	1209±7
A	30	30	0	701060±109	678858±531	546±1	880±5
A	30	30	5	688959±183	665510±238	632±3	1054±4
A	30	30	10	682718±264	661114±239	728±4	1241±12
A	30	30	20	676865±241	657411±261	926±2	1643±24
A	30	45	0	712424±179	680620±368	746±4	1260±11
A	30	45	5	697779±197	667143±69	839±6	1453±5
A	30	45	10	690129±333	662437±183	940±4	1654±14
A	30	45	20	682720±186	658876±996	1145±3	2020±111
B	15	15	0	684251±155	665186±195	265±2	333±9
B	15	15	5	666532±1259	654020±180	281±1	380±5
B	15	15	10	661779±144	650554±121	315±1	504±114
B	15	15	20	658721±143	647745±81	375±1	582±8
B	15	30	0	692092±179	673638±254	319±1	467±8
B	15	30	5	679318±170	660126±77	360±2	549±7
B	15	30	10	673067±197	655545±132	406±2	635±8
B	15	30	20	666736±203	651232±99	475±2	771±6
B	15	45	0	700219±1341	676198±308	377±2	583±8
B	15	45	5	686091±243	662573±1001	423±1	675±7
B	15	45	10	679206±119	657250±141	471±1	764±6
B	15	45	20	671956±157	653129±1057	543±2	909±6
B	30	15	0	692458±166	675290±111	355±1	539±6
B	30	15	5	680971±128	662295±87	431±1	682±5
B	30	15	10	676080±221	658139±131	525±2	852±8
B	30	15	20	672286±156	655348±149	712±2	1205±6
B	30	30	0	701271±133	678988±233	540±2	883±7
B	30	30	5	689170±87	665613±413	625±1	1053±12
B	30	30	10	683565±1268	661236±335	723±3	1236±13
B	30	30	20	677467±1264	657437±128	920±2	1625±20
B	30	45	0	712761±195	680604±567	743±2	1260±18

Continúa en la siguiente hoja

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{cost}		$T_{ejecucion}$	
				ACS-HCSP	IACS-HCSP	ACS-HCSP	IACS-HCSP
				$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$	$\mu \pm \sigma$
B	30	45	5	697951±229	667043±357	830±3	1449±15
B	30	45	10	690219±321	662597±204	931±2	1654±18
B	30	45	20	682998±130	658484±183	1136±5	2040±39

Tabla 5.12: Comparación de resultados entre ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} .

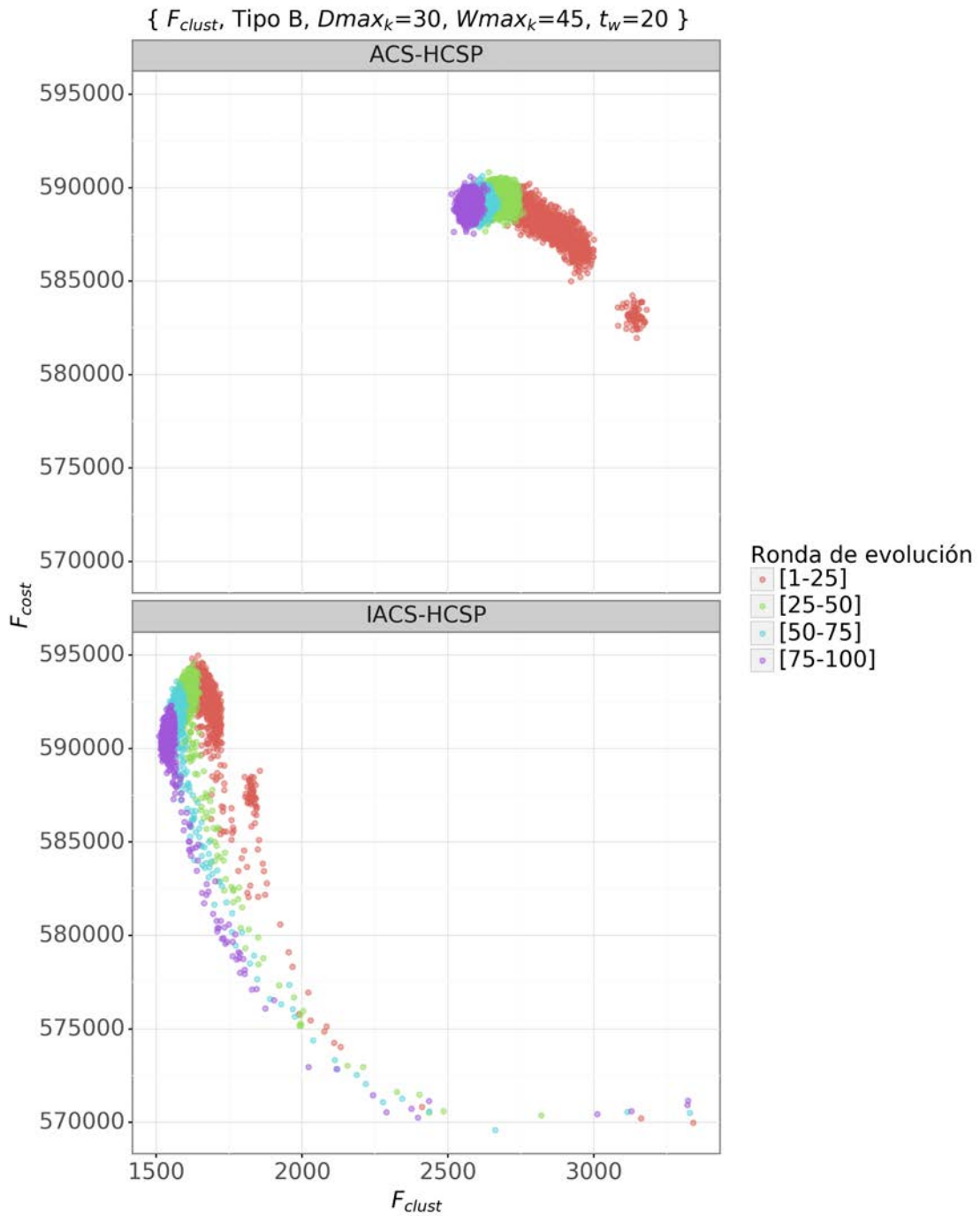


Figura 5.32: Evolución de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{clust} , el tipo de agrupamiento B y el grupo de servicios 1.

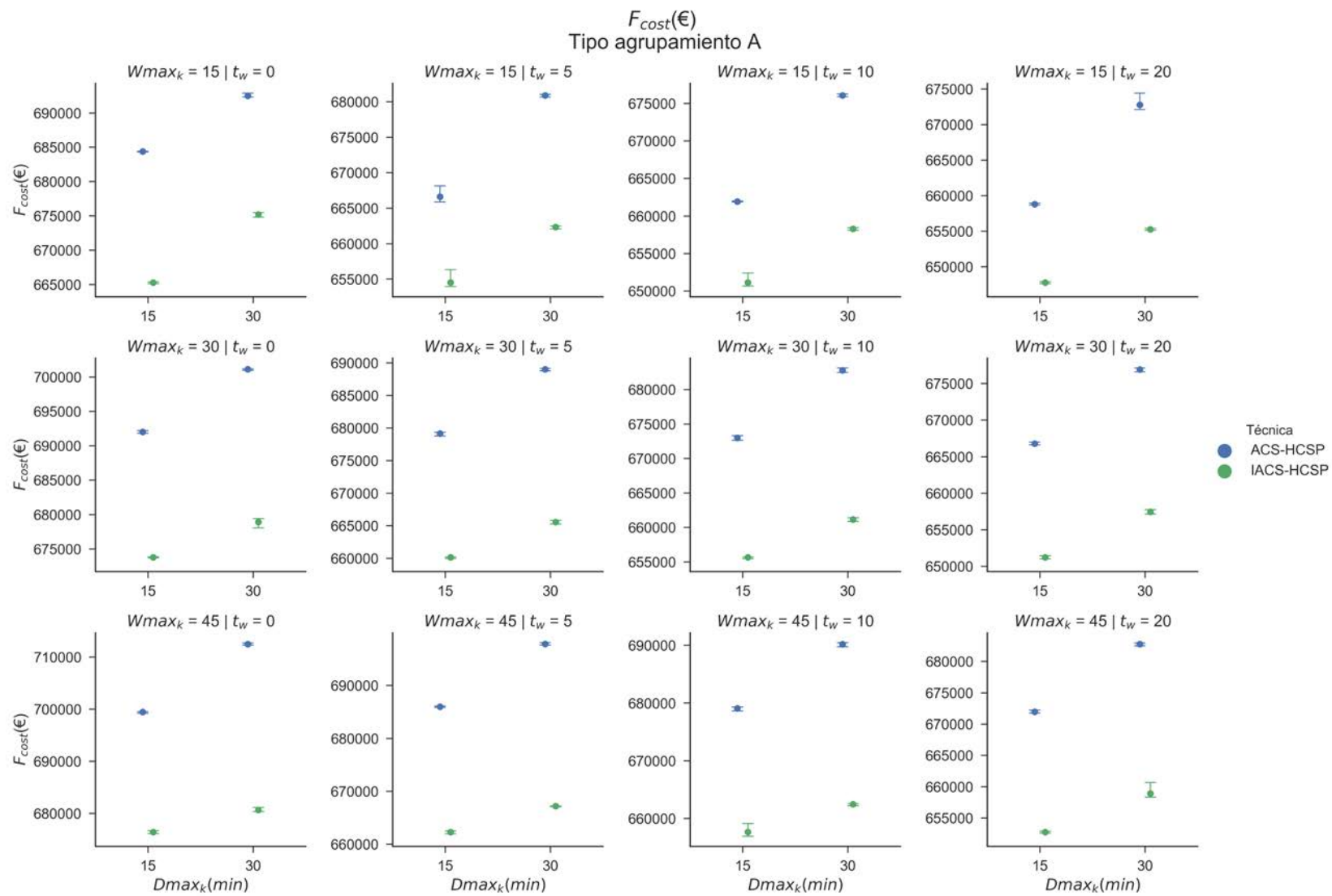


Figura 5.33: Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento A.

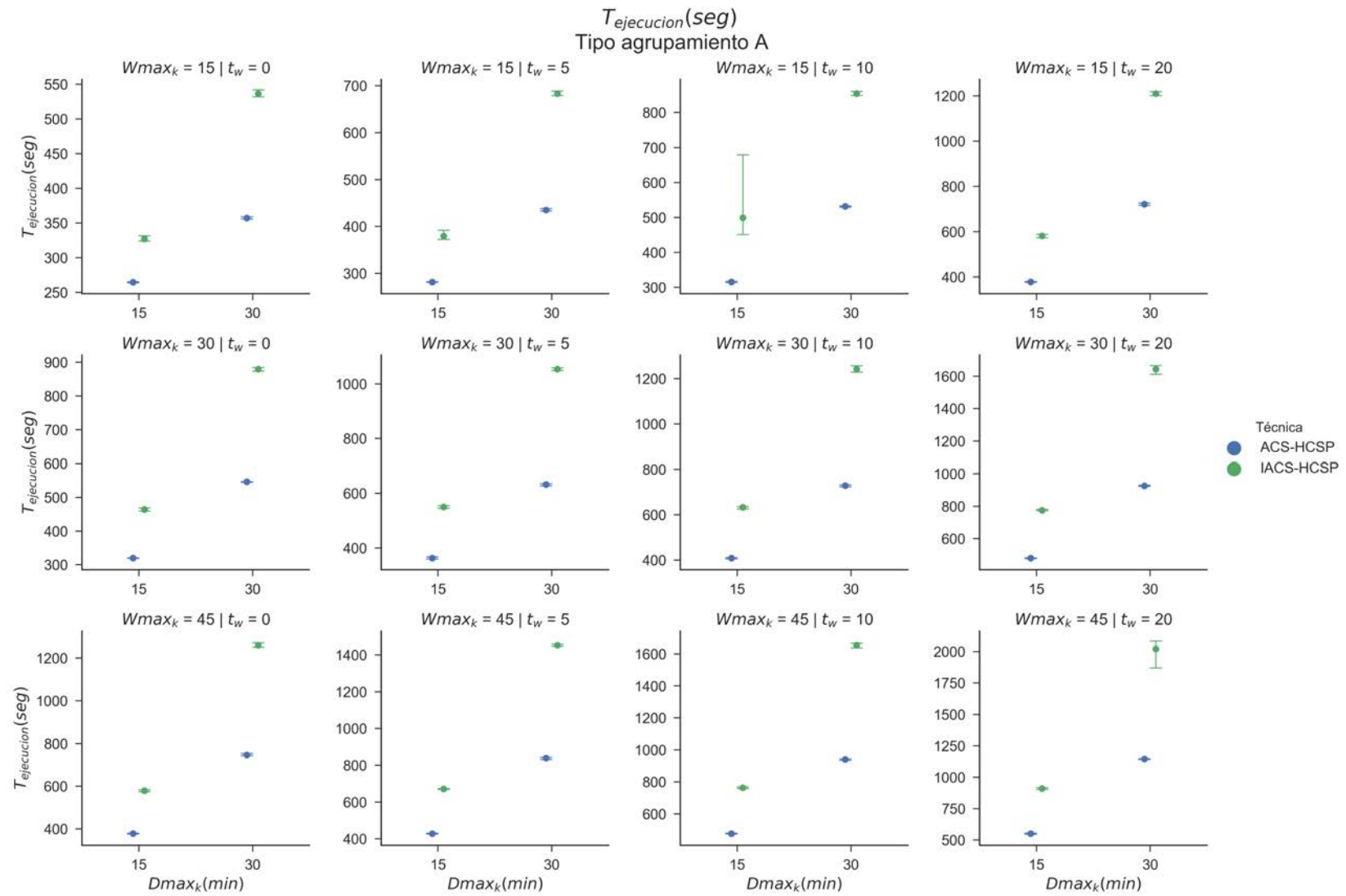


Figura 5.34: Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento A.

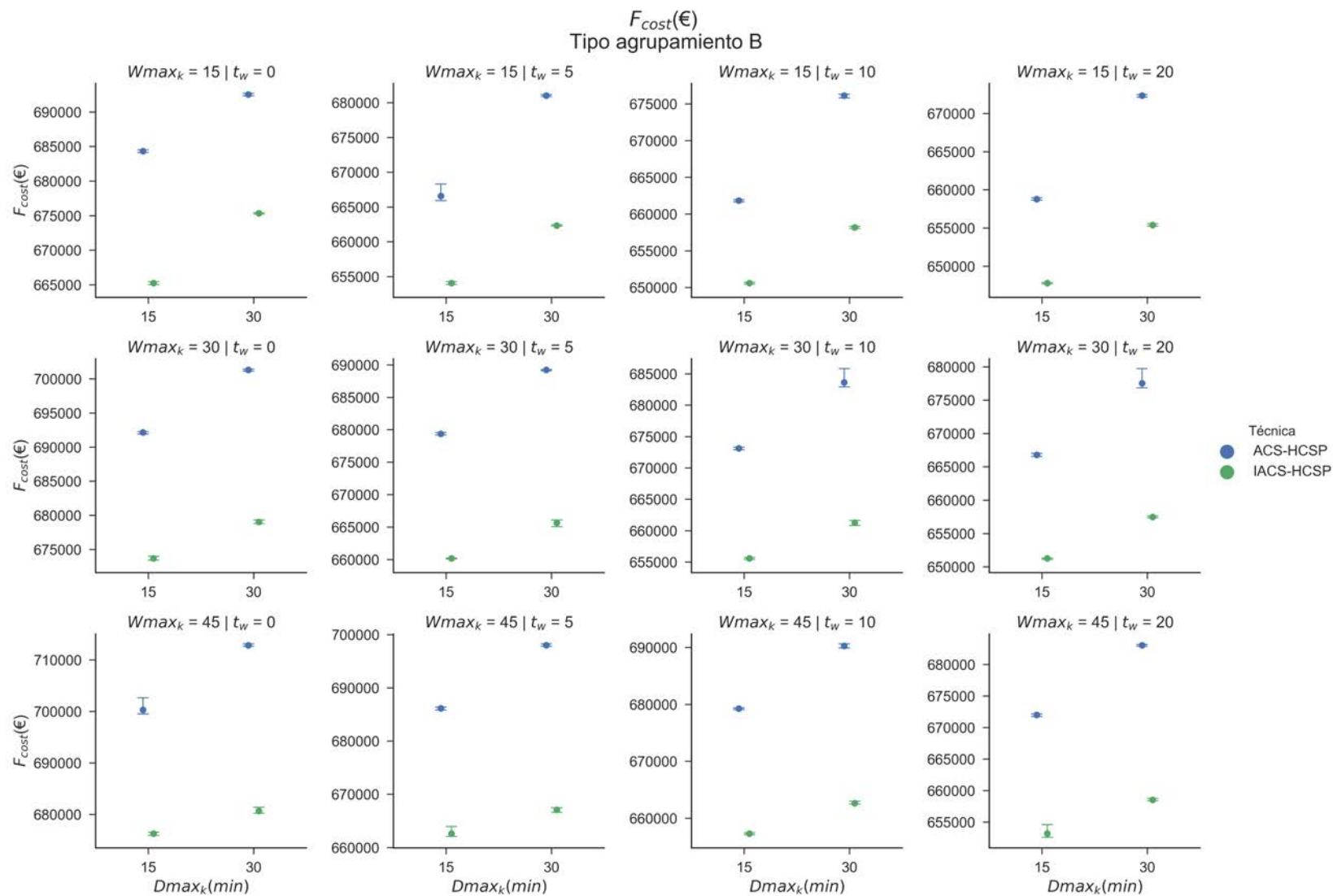


Figura 5.35: Comparativa entre los resultados obtenidos por ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento B.

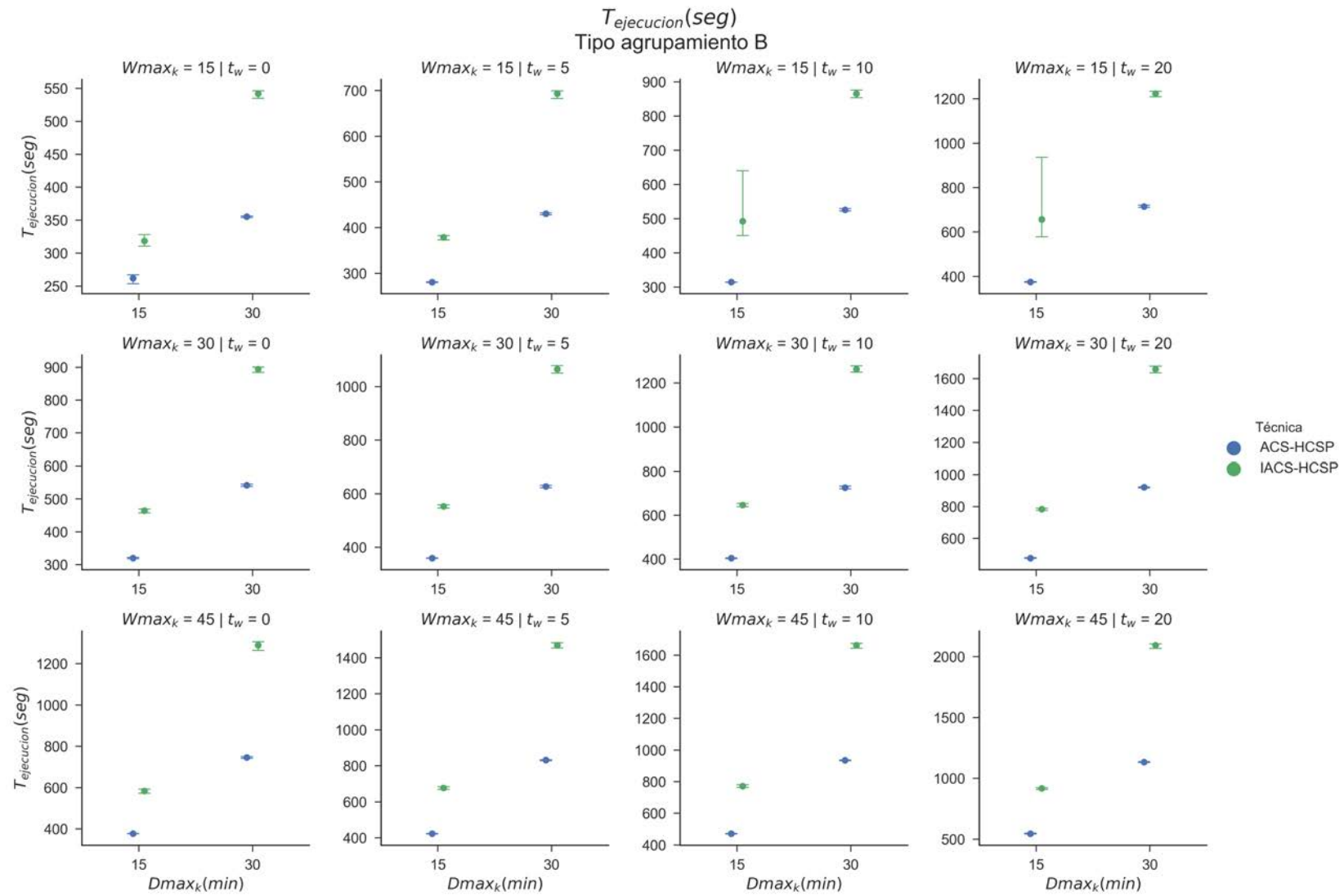


Figura 5.36: Comparativa de los tiempos requeridos entre las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} y el tipo de agrupamiento B.

A pesar de que en las gráficas anteriores parece claro que los resultados de IACS-HCSP son significativamente superiores a los de ACS-HCSP, queda la incógnita si lo son para todas las soluciones que obtienen cada una de las técnicas, o solo lo son como resultado final. Puesto que solo estamos reportando las 5 mejores soluciones encontradas por cada técnica y cada parametrización, sería posible que las poblaciones de soluciones de las diferentes rondas de evolución no fueran diferentes. Por ejemplo, si en la técnica IACS-HCSP la mejora respecto a ACS-HCSP solo se lograra al final de la última ronda de evolución, puesto que solo estamos analizando las 5 mejores soluciones y no las 32.000 soluciones de cada configuración, sería posible que afirmásemos que la técnica IACS-HCSP es superior a ACS-HCSP cuando en realidad esa mejora solo se produce al final de la última ronda. En este punto es conveniente recordar que las soluciones reportadas hacen referencia a una solución completa, la cual implica varios grupos de servicios a lo largo de toda la semana, siendo dichos grupos resueltos de modo independiente por colonias distintas. Las soluciones reportadas hasta el momento se han obtenido al agregar los resultados obtenidos para cada grupo, con el objetivo de validar la hipótesis anterior se procede a examinar la evolución de las soluciones encontradas por cada técnica para el grupo de servicios Grupo 1 ¹¹.

Con el objetivo de validar dicha hipótesis se llevan a cabo pruebas de contraste estadístico, en concreto se utilizará la prueba t de Student, ya que las muestras son independientes y el tamaño de la muestra para cada experimento es de 32.000. Con dicho tamaño y apoyados en el teorema central del límite, podemos obviar cálculos necesarios para asegurar que cada una de las muestras siguen una distribución normal. Al igual que hicieramos en apartados anteriores, nuestra hipótesis nula es que las poblaciones tienen igual media, fijando como p-valor el valor 0.05. Los resultados reflejados en la tabla 5.13 muestran tanto el valor del estadístico de la t-Student como sus p-valores asociados; se puede apreciar como todos los p-valores son iguales a cero, no se trata de ningún error simplemente el valor es tan cercano a cero que al realizar los cálculos el ordenador redondea a dicha cifra, por lo tanto debemos rechazar la hipótesis nula y concluir que las medias de las dos técnicas propuestas son diferentes.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{cost}		Est. t	p-valor
				ACS-HCSP	IACS-HCSP		
A	15	15	0	583888	578725	157	0
A	15	15	5	568437	562127	217	0
A	15	15	10	565809	558533	305	0
A	15	15	20	563767	556295	367	0
A	15	30	0	590437	579712	304	0
A	15	30	5	579865	567564	416	0
A	15	30	10	575073	563301	462	0
A	15	30	20	570897	559845	508	0
A	15	45	0	598099	582313	430	0
A	15	45	5	586465	569435	487	0
A	15	45	10	581063	565048	511	0

Continúa en la siguiente hoja

¹¹Que incluye a un total de 10.709 servicios que son prestados de lunes a viernes entre las 07:00 y las 14:30.

Agrupamiento	$Dmax_k$	$Wmax_k$	t_w	F_{cost}		Est. t	p-valor
				ACS-HCSP	IACS-HCSP		
A	15	45	20	575762	561382	542	0
A	30	15	0	590822	580089	318	0
A	30	15	5	581273	568843	458	0
A	30	15	10	577533	565689	488	0
A	30	15	20	575532	563339	501	0
A	30	30	0	599917	583695	428	0
A	30	30	5	589317	572143	495	0
A	30	30	10	585140	568604	532	0
A	30	30	20	580843	565885	529	0
A	30	45	0	611367	586152	469	0
A	30	45	5	598254	574503	504	0
A	30	45	10	592362	570539	513	0
A	30	45	20	587234	567287	543	0
B	15	15	0	583819	578728	158	0
B	15	15	5	568594	562137	225	0
B	15	15	10	565712	558625	291	0
B	15	15	20	563632	556213	367	0
B	15	30	0	590452	579583	332	0
B	15	30	5	580126	567520	436	0
B	15	30	10	575295	563281	485	0
B	15	30	20	570804	559708	514	0
B	15	45	0	598135	582157	411	0
B	15	45	5	586424	569293	484	0
B	15	45	10	581055	565054	533	0
B	15	45	20	575917	561489	556	0
B	30	15	0	590676	580130	336	0
B	30	15	5	581038	568929	433	0
B	30	15	10	577640	565525	507	0
B	30	15	20	575566	563441	530	0
B	30	30	0	599870	583716	401	0
B	30	30	5	589567	572337	478	0
B	30	30	10	585019	568689	498	0
B	30	30	20	580834	565695	499	0
B	30	45	0	611209	586138	467	0
B	30	45	5	598371	574462	500	0
B	30	45	10	592476	570554	525	0
B	30	45	20	587048	567462	527	0

Tabla 5.13: Prueba t-Student para las poblaciones de ACS-HCSP y IACS-HCSP para el Grupo 1 de servicios y función de evaluación F_{clust} .

Para ilustrar la diferencia clara de rendimiento entre las técnicas ACS-HCSP y IACS-HCSP, en la figura 5.31 se muestra la distribución de las soluciones encontradas por cada una de las técnicas con la configuración $\{F_{cost}, TipoA, Dmax_k = 30, Wmax_k = 45, t_w = 20\}$, se puede apreciar claramente como las dos distribuciones están perfectamente separadas. En este caso las soluciones pertenecen al Grupo 1, es decir representan al grupo más numeroso de servicios con un total de 10.709. Otro aspecto que podemos analizar para esta

misma configuración, es la evolución de las soluciones y ver la relación que tiene con la propia distribución que observamos. En la figura 5.32 se muestra la evolución de uno de los experimentos para la misma configuración anterior, en ella se puede observar como la técnica IACS-HCSP obtiene mejores soluciones desde las primeras rondas de evolución, de hecho más del 99 % de las soluciones encontradas por la técnica IACS-HCSP son mejores soluciones que la mejor solución encontrada por ACS-HCSP. En dicha figura también se aprecia como la técnica ACS-HCSP no varía respecto a las soluciones descritas en la figura 5.38, es decir, las soluciones encontradas por la técnica ACS-HCSP son exactamente iguales si la función a minimizar es F_{clust} o si es F_{cost} . Por otro lado, en dicha figura se puede apreciar como las soluciones obtenidas por ACS-HCSP durante las primeras rondas de evolución (1-25) obtienen un mejor coste que las soluciones de las últimas rondas de evolución, lo cual indica que la técnica no es capaz de evolucionar y de encontrar mejores a medida que se suceden el número de iteraciones. Por el contrario, en el caso de IACS-HCSP, se puede apreciar como las soluciones de las últimas rondas de evolución (75-100) son mejores que la de las primeras (1-25), obteniéndose soluciones de distinta calidad gracias a la utilización del criterio de parada explicado en el apartado 4.3.3.3 del capítulo 4. Al igual que sucediera con en el apartado anterior, donde el criterio a optimizar eran el número de clúster, esta mejora en calidad de las soluciones también va acompañada de un incremento en el coste computacional, el cual se traduce en un mayor tiempo de ejecución. Este incremento en tiempo está causado por la definición de vértices alcanzables utilizada por IACS-HCSP (ver ecuación (4.16)), gracias a la cual la técnica IACS-HCSP es capaz de encontrar soluciones de mejor calidad que ACS-HCSP desde las primeras iteraciones, dicha definición junto con el criterio de parada hacen que las soluciones no se concentren en un solo punto del espacio de búsqueda obteniendo poblaciones como la mostrada en la figura 5.38. Con todas estas evidencias, podemos afirmar que las soluciones proporcionadas por la técnica IACS-HCSP para minimizar el coste la función de coste propuesta por [Quintana et al. \(2017\)](#), es superior a de la técnica ACS-HCSP.

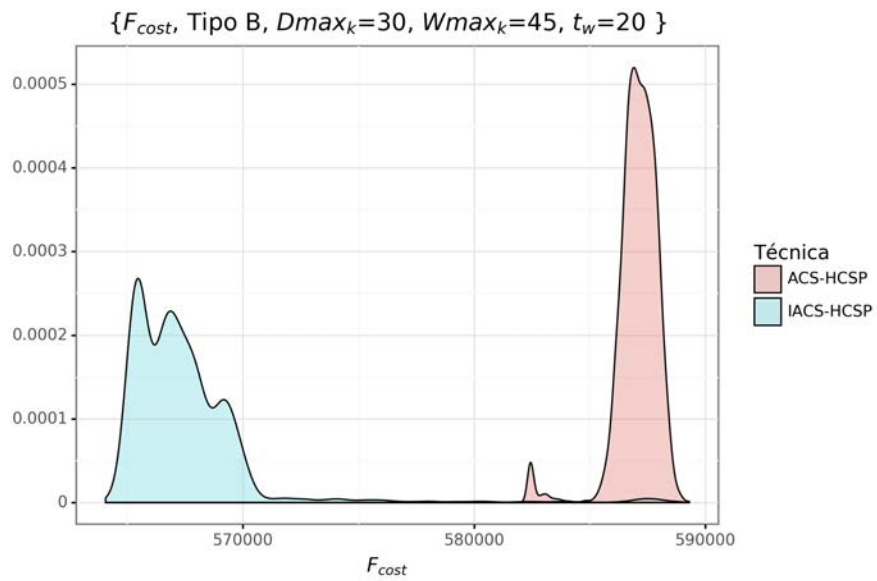


Figura 5.37: Distribuciones de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP durante las 100 rondas de evolución para la función de evaluación F_{cost} , el tipo de agrupamiento B y el grupo de servicios 1.

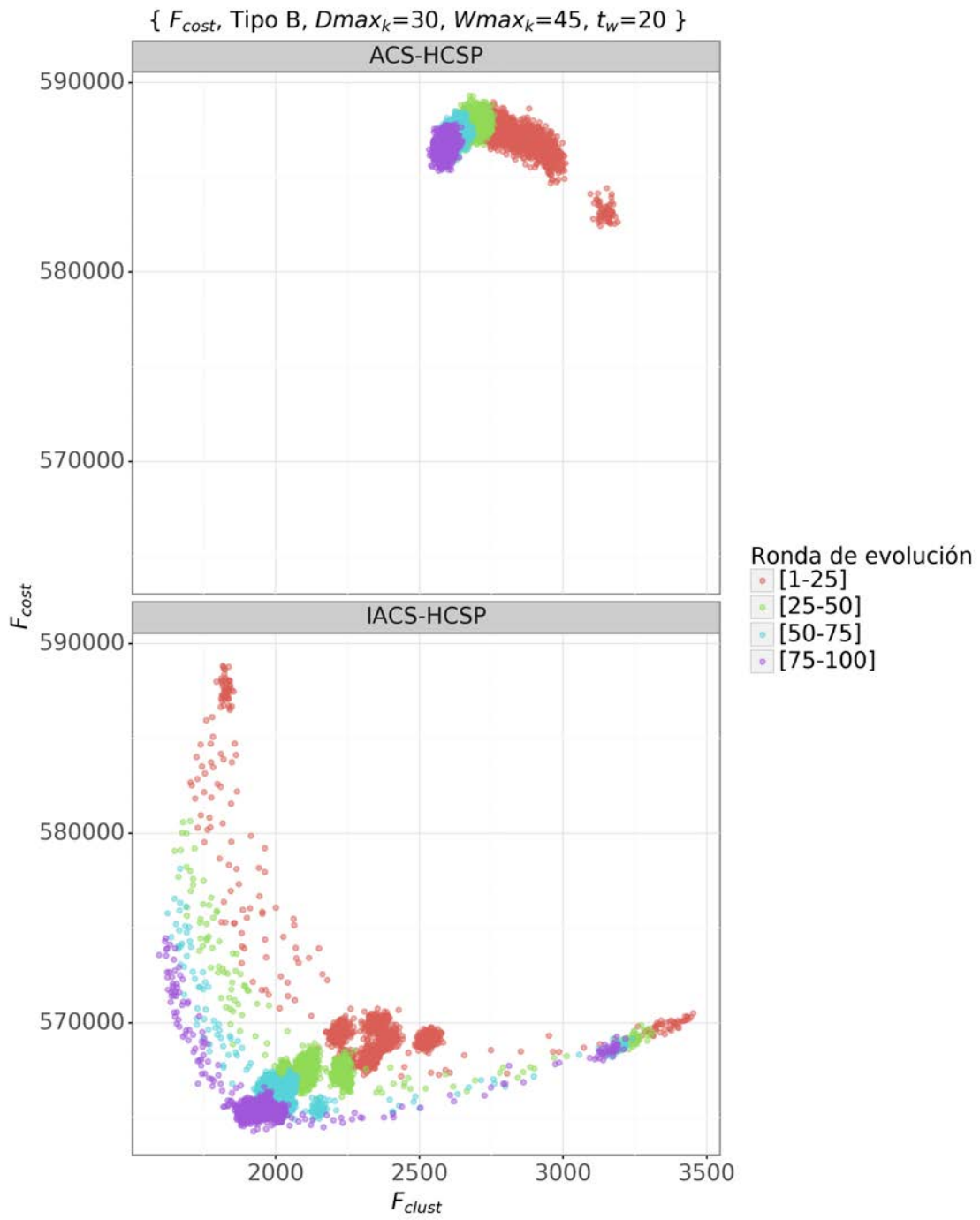


Figura 5.38: Evolución de las soluciones obtenidas por las técnicas ACS-HCSP y IACS-HCSP para la función de evaluación F_{cost} , el tipo de agrupamiento B y el grupo de servicios 1.

5.5.3. Comparativa con (Quintana et al., 2017)

Hasta el momento todas las comparaciones realizadas han sido entre las distintas técnicas propuestas por esta investigación, parece recomendable validar dichas técnicas con otras existentes en el estado del arte. En primer lugar cabe destacar que a pesar de que la mayoría de investigadores ponen a disposición de otros investigadores sus instancias, raramente estas son reutilizadas; ya que las regulaciones, particularidades y restricciones suelen cambiar de una región a otra (incluso dentro del mismo país) haciendo poco común su reutilización Fikar y Hirsch (2017). En el caso de esta investigación contamos con la importante ventaja de poder reutilizar las instancias usadas por (Quintana et al., 2017) las cuales, siguiendo la nomenclatura de la sección 5.1.2 quedan parametrizadas del siguiente modo:

- **Tipo de agrupamiento:** [Tipo B].
- **Tiempo máximo de desplazamiento:** $D_{\max_k} \in [30]$.
- **Tiempo máximo de espera:** $W_{\max_k} = \infty$.
- **Ventana temporal:** $t_w \in [5]$.
- **Función de evaluación:** $[F_{cost}]$

Los autores proponen un total de dos técnicas heurísticas, denominadas “heurística 1” y “heurística 2”. Ambas son bastante similares al método de Ward al tratarse de dos algoritmos aglomerativos “avariciosos” ya que en cada iteración aquellos clústers cuya unión represente un menor incremento del tiempo no productivo serán unidos en uno solo. Dado el tamaño de las instancias resulta muy probable, especialmente en las primeras iteraciones, que se produzcan empates la política sobre cómo se resuelven dichos empates da lugar a las citadas heurísticas. En caso de la “heurística 1” dicha elección se realiza de un modo probabilístico, mientras que en el caso de la heurística “heurística 2” los empates se resuelven seleccionando aquel clúster que presentan un mayor coste respecto a los demás clústers. De acuerdo con los autores la idea subyacente es prevenir la unión de clústers de baja calidad.

Una vez expuestas las técnicas propuestas por (Quintana et al., 2017) se procede a realizar la comparación con las descritas en esta investigación, Ward, ACS-HCSP y IACS-HCSP. La configuración de las dos heurísticas propuestas no coincide exactamente con las configuraciones de ACS-HCSP y IACS-HCSP, ya que los autores no limitan el tiempo máximo de espera $W_{\max_k} = \infty$. Por ello la comparación en el caso de ACS-HCSP y IACS-HCSP se realizará con el valor máximo de dicho parámetro $W_{\max_k} = 45$. En la tabla 5.14 se presentan los resultados obtenidos por cada una de las técnicas, en primer lugar, se presenta el resultado del que partía la compañía ¹², posteriormente se muestra el valor medio de los resultados obtenidos por el método de Ward para la configuración $\{F_{cost}, TipoB, D_{\max_k} = 30, W_{\max_k} = \infty, t_w = 0\}$.

¹²Solución para la cual fue necesario hacer un ajuste de 161.106 minutos con el objetivo de hacer la solución proporcionada factible.

	$F_{cost}(\text{€})$	F_{clust}
Eulen	739.009	1.527
Ward (μ)	761.184	3.800
Heurística 1	717.441	3.882
Heurística 2	680.938	4.051
ACS-HCSP (μ)	697.951	4.248
IACS-HCSP (μ)	667.043	3.511

Tabla 5.14: Comparativa entre los resultados obtenidos por las técnicas propuestas Ward, ACS-HCSP y IACS-HCSP, y las heurísticas propuestas por [Quintana et al. \(2017\)](#)

En la figura 5.39 se muestra una representación visual de los resultados reportados en la tabla 5.14. En dicha representación se puede observar como la técnica IACS-HCSP obtienen mejores resultados que el resto de técnicas tanto en lo referido al coste de la solución F_{cost} , como en lo referido al número de clústers F_{clust} . Por otro lado, también se aprecia como la técnica ACS-HCSP a pesar de tener fuertes puntos de mejora como los expuestos en el capítulo 4, obtiene mejores resultados que la Heurística 1 y que el método de Ward. En el aspecto económico la solución obtenida por la técnica IACS-HCSP supone un ahorro semanal de unos 14.000^{13} € lo que anualmente supondría un ahorro de 728.000 €. Si comparamos dicha solución con la obtenida por la compañía el ahorro anual supondría alrededor de $3.742.232$ €, cifra que se incrementaría hasta $4.441.000$ € si la comparamos con la mejor solución obtenida por la técnica IACS-HCSP con la configuración $\{F_{cost}, TipoB, Dmax_k = 15, Wmax_k = 15, tw = 5\}$.

¹³Cabe destacar que en la tabla 5.7 con parámetros de tiempo de desplazamiento máximo y tiempo de espera máximo en 15 minutos se han reportado soluciones que supondrían un ahorro semanal de aproximadamente 27.000 €.

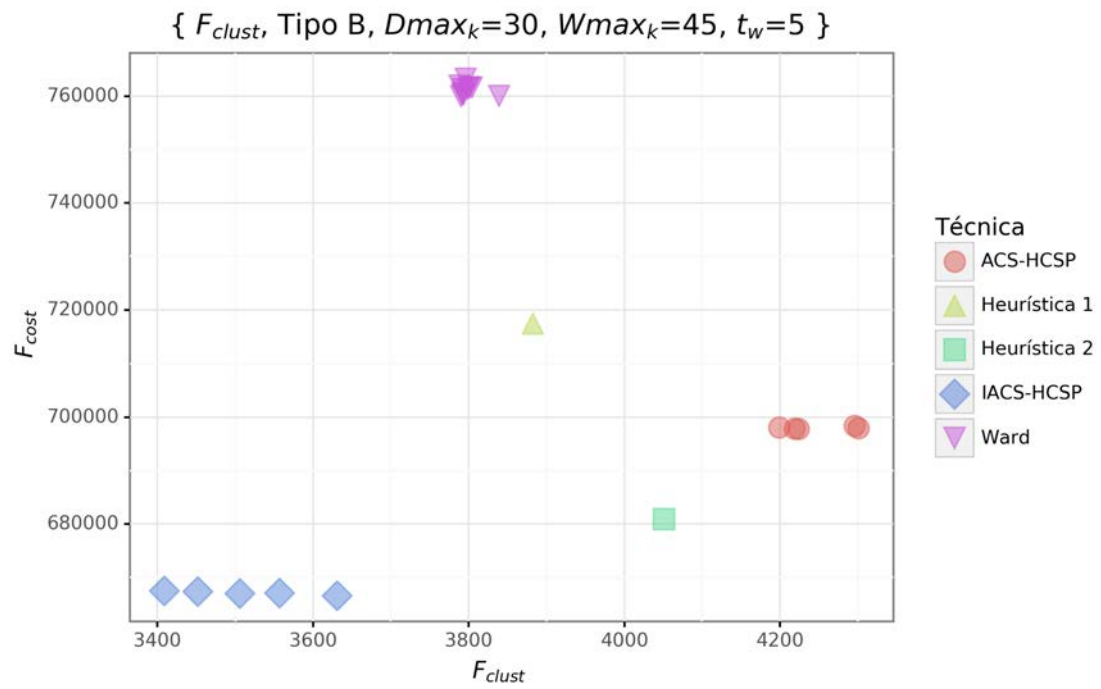


Figura 5.39: Comparativa entre las técnicas propuestas Ward, ACS-HCSP y IACS-HCSP respecto a las heurísticas propuestas por Quintana et al. (2017)

Capítulo 6

Conclusiones

La conclusión es que sabemos muy poco y sin embargo es asombroso lo mucho que conocemos.

Bertrand Russell

Resumen:

En este último capítulo se muestran las conclusiones obtenidas como resultado de la presente investigación, analizando en primer lugar el cumplimiento de los objetivos marcados en el capítulo 1, destacándose posteriormente las aportaciones realizadas por esta investigación. Por último, se detallan las posibles extensiones de la presente investigación proporcionando dominios de aplicación concretos en los que las técnicas propuestas podrían ser utilizados, destacando de modo sucinto los aspectos que han quedado fuera del alcance de la misma, los cuales conforman las posibles líneas de investigación que podrían ser abordadas en el futuro.

6.1. Objetivos

Tal y como se presentó en el capítulo 1 el objetivo principal de esta investigación era la resolución de problemas de optimización combinatoria, los cuales debían presentar las siguientes características.

- Las instancias resueltas debían pertenecer a problemas reales de la industria, de modo que los resultados de esta investigación sean aplicables y fácilmente transferibles a dicho mercado.
- Las instancias debían ser instancias de alta dimensionalidad, entendiéndose por tal, instancias cuya dimensión fuera de al menos un orden de magnitud superior a las instancias resueltas por las técnicas exactas en el estado del arte actual.
- Los problemas seleccionados debían presentar restricciones temporales y espaciales. Por restricciones temporales se entienden aquellas restricciones que tienen en cuenta el tiempo, incluyéndose al menos los tiempos en los que se puede realizar una tarea o servicio, conocido como ventana temporal, el tiempo de desplazamiento y el tiempo de espera. Respecto a restricciones espaciales, se entienden aquellas que teniendo en cuenta la localización geográfica de cada tarea o servicio, fijan restricciones sobre la realización de estas. Por ejemplo, la distancia máxima que puede recorrer un asistente de atención domiciliaria entre dos tareas realizadas de modo consecutivo.

Dentro de los diversos problemas de optimización combinatoria, la familia de problemas estudiada en esta investigación es la familia de problemas de planificación de asistentes de atención domiciliaria. Conocida por sus siglas en inglés HCSP (Home Care Scheduling Problem), dicha familia de problemas ha suscitado el interés de numerosos investigadores en los últimos años, interés motivado por varios factores como, las previsiones demográficas, el envejecimiento de nuestra población y la creciente demanda de este tipo de servicios, factores recogidos en recientes revisiones del estado del arte [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#). Tal y como se presentó a lo largo del capítulo 1 todos los estudios y previsiones apuntan a que nuestra población envejecerá a lo largo de las próximas décadas, en la Europa de los 28 para 2050 la tasa de dependencia superará el 50 % de la población [European Commission \(2015\)](#). Esta población de personas mayores prefiere envejecer en su hogar o al menos en la comunidad en que han vivido [Costa-Font et al. \(2009\)](#), lo cual se conoce como “Envejecimiento en el lugar” (Ageing in place). De acuerdo con [World Health Organization \(2015\)](#) “es necesario reorientar los modelos de atención para priorizar la atención primaria y la atención en la comunidad. Esto abarca un cambio de la asistencia hospitalaria a la asistencia ambulatoria y las consultas externas, más intervenciones en el hogar, la participación de la comunidad y un sistema de derivación completamente integrado”. Es precisamente en dicho contexto donde las compañías prestadoras de servicios de atención domiciliaria desarrollan su actividad, debiéndose enfrentar a distintas variantes del HCSP, lo cual supone numerosos desafíos, tanto logísticos, como organizativos a la hora de planificar su operativa diaria. El problema estudiado a pesar de tener una enunciación aparentemente sencilla, ha requerido y requiere el esfuerzo de numerosos investigadores. Dicha enunciación se puede definir del siguiente modo: “Asignar a cada cliente aquel

asistente de atención domiciliaria que mejor se adapte a sus necesidades, respetando las numerosas restricciones y minimizando ciertas métricas como el coste o el número de asistentes requeridos.” El objetivo de esta investigación es por lo tanto, proporcionar a las compañías prestadoras de servicios de atención domiciliaria herramientas y técnicas que les permitan la automatización de dichas planificaciones, permitiendo la resolución de problemas de alta dimensionalidad, de un modo sencillo y con unos tiempos suficientes para que su operativa diaria no se vea afectada. En los escenarios actuales y futuros es y será fundamental, dotar a dichas compañías de dichas herramientas, de tal modo que aumenten su eficiencia, reduzcan sus costes y posibiliten en última instancia una atención domiciliaria de calidad y sostenible para nuestros mayores.

El objetivo anterior se ha tratado desde dos perspectivas distintas, complementarias y totalmente novedosas en el dominio del HCSP, como son las técnicas de clústering y la técnica metaheurística ACO. La concepción, desarrollo y evaluación de dichas técnicas, surgen como solución a las necesidades del caso de estudio tratado en esta investigación, el cual fue expuesto en el capítulo 3. Dicho caso de estudio cumple con las características que se establecieron inicialmente en los objetivos; se trata de un caso de estudio real, de fácil transferencia al mercado, cuyo tamaño supera en hasta dos órdenes de magnitud el tamaño de los problemas de la misma familia que son resueltos mediante técnicas exactas y en un orden de magnitud aquellos que son resueltos con técnicas no exactas, presentando restricciones espaciales y temporales. El caso de estudio describe el problema al que se enfrenta la compañía EULEN en la Comunidad de Madrid, la cual debe planificar semanalmente más de 29.000 visitas al domicilio de sus clientes. Estas más de 29.000 visitas, son agrupadas en unidades semanales de planificación denominadas servicios, conformando un total de 13.344. El problema presenta varias características que imposibilitan la resolución del mismo en instancias más pequeñas; en primer lugar la división en subconjuntos de servicios no resulta trivial, al tener que considerarse las componentes temporales y espaciales del problema. Desde el punto de vista espacial, los servicios presentan tanta densidad que cualquier tipo de particionamiento resultaría del todo arbitrario (ver imagen 5.8). Además dada la alta dimensionalidad de las instancias resueltas se deberían encontrar más de 300 particiones para que las instancias resultantes tuvieran un tamaño próximo al tamaño de las instancias resueltas por técnicas exactas. Por otra parte y desde el punto de vista temporal, más del 45% de las tareas deben comenzar entre las 08:00 y las 10:00 por lo que cualquier tipo de particionamiento resultaría infructuoso. Otro factor limitante en la utilización de otro tipo de técnicas, como técnicas exactas, es el tiempo de resolución y el tamaño de instancias que dichas técnicas son capaces de manejar. Desde el punto de vista de complejidad computacional el HCSP pertenece a la clase de complejidad \mathcal{NP} -difícil, ya que se puede tratar como una generalización del TSP (ver sección 2.1) y el VRP (ver sección 2.2). La pertenencia a dicha clase de complejidad implica que hasta la fecha no se conoce un algoritmo que sea capaz de resolver cualquier instancia del mismo en tiempo polinomial. Es decir, a medida que crecen los tamaños de las instancias a resolver, el esfuerzo computacional requerido para obtener soluciones óptimas a dichas instancias crece de modo exponencial, imposibilitando en la mayoría de los casos la utilización de técnicas exactas exactas. En el caso del HCSP las técnicas exactas no son capaces de abordar problemas con más de 50 asistentes, mientras que algunas técnicas metaheurísticas, como la presentada en (Hiermann et al., 2015), son

capaces de resolver problemas con hasta 518 asistentes y 717 tareas. Entre las motivaciones que han dado lugar al presente trabajo destacan las siguientes: La alta dimensionalidad del caso de estudio presentado, la imposibilidad de dividirlo en instancias más pequeñas, la clase complejidad del problema tratado y finalmente la falta de trabajos que aborden el problema del HCSP desde la perspectiva del clústering. Todas estas razones constituyen las principales motivaciones que dieron lugar a la presente investigación, las cuales justifican el diseño, desarrollo y evaluación de las distintas técnicas propuestas cuyas aportaciones principales son resumidas a continuación.

6.2. Aportaciones

Desde la perspectiva de las técnicas de clústering existen pocos trabajos, como el propuesto por [Qi et al. \(2012\)](#), en los que la resolución de problemas de optimización combinatoria sea abordada únicamente mediante técnicas de clústering. Quizás uno de los motivos para dicha falta de trabajos, es la dificultad inherente a las técnicas de clústering para considerar las distintas restricciones existentes en los problemas de optimización combinatoria. En el caso concreto del HCSP, tal como se expuso en el capítulo 2 y de acuerdo a recientes revisiones sobre estado del arte [Fikar y Hirsch \(2017\)](#); [Cisse et al. \(2017\)](#), no existen investigaciones relevantes donde se aborde la resolución de esta familia de problemas utilizando técnicas de clústering. Muy probablemente, entre los motivos para la falta de trabajos en dicha dirección, se encuentre la dificultad existente en la definición de una función de similitud o distancia apropiada para problemas que presenten restricciones espaciales y temporales, como el tratado en esta investigación. A dicha dificultad, se debe sumar la dificultad de evaluar el cumplimiento de dichas restricciones de un modo dinámico, dependiente de la propia composición del grupo. Este dinamismo en el cumplimiento o incumplimiento de restricciones hace que dos elementos (servicios) no puedan pertenecer por si solos al mismo clúster, pero si lo puedan hacer si en dicho clúster se encuentra un tercer elemento (servicio). La primera aproximación para resolver el problema objeto de estudio está basada en el método clústering jerárquico propuesto por [Ward \(1963\)](#). Para adaptar dicha técnica al dominio estudiado han sido necesarias diversas modificaciones, las cuales se resumen a continuación. En primer lugar y partiendo de simples tareas en la sección 4.2.1.1 se definió un concepto de similitud para tareas que deben realizarse en el mismo día. Dicho concepto de similitud, fue posteriormente extendido en la sección 4.2.1.2 a varios días de tal forma que se pudiera aplicar a los servicios, siendo capaz de capturar las restricciones espaciales y temporales del problema de un modo eficaz. Posteriormente el algoritmo original propuesto por [Ward \(1963\)](#) ha sido modificado ya que en el caso del HCSP no se quiere obtener una jerarquía, sino un conjunto de clústers. Finalmente, dado que la complejidad de la técnica propuesta es $O(n^3)$, se han implementado una serie de mecanismos de caché con el objetivo de evitar calcular aquellas similitudes entre pares de clústers que ya han sido calculadas en iteraciones anteriores reduciéndose los tiempos de cómputo. La técnica propuesta obtiene unos resultados discretos cuando es evaluada utilizando como función objetivo F_{clust} , la cual representa el número total de asistentes requeridos para llevar a cabo la planificación de todos los servicios. Este resultado se debe al modo en que el método propuesto construye sus soluciones. Siempre que un par de clústers que minimizan la métrica de similitud, este método “avaricioso” da como resultado

soluciones que requieren un número elevado de asistentes.

Posteriormente y con el objetivo de mejorar los resultados obtenidos por la técnica basada en el método de Ward, se han desarrollado dos técnicas de clústering adaptadas al HCSP objeto de estudio, utilizando como algoritmo base la metaheurística ACO. Respecto a la técnica metaheurística ACO (Dorigo y Stützle, 2010), a pesar de haber sido utilizada para multitud de problemas de optimización combinatoria, siendo muchos de ellos de optimización de rutas con características similares al problema abordado en esta investigación Rizzoli et al. (2007); Liu y He (2012), así como en diversos problemas de planificación Neto y Filho (2013) no existen publicaciones en el estado del arte actual en el que dichas técnicas se utilicen para la resolución del HCSP Fikar y Hirsch (2017); Cisse et al. (2017). La adaptación de dicha metaheurística al problema abordado, ha supuesto varios desafíos que se enumeran a continuación. La metaheurística ACO necesita la creación de un grafo en el cual la colonia de hormigas pueda llevar a cabo la búsqueda de soluciones. Por ello uno de los retos y de las aportaciones de esta técnica, está relacionada con la creación de dicho grafo de tal modo que se represente el problema objeto de estudio, así como sus restricciones. Como se expuso en la sección 4.3.1, para la representación del HCSP se utiliza un grafo no dirigido. En dicho grafo cada uno de los servicios a agrupar, está representado por un vértice. Tradicionalmente en otros problemas en los que se ha aplicado ACO, como el TSP Dorigo y Gambardella (1997) o el VRP Gambardella et al. (1999) las aristas simplemente representan distancias, siendo el grafo un grafo completo, es decir, para todo par de nodos existe una arista que representa la distancia entre dichos nodos. En el caso de las técnicas propuestas, las aristas del grafo no solo representan las componentes temporales y espaciales del problema (función de similitud), sino que además representan la compatibilidad entre servicios dos a dos. Esta propuesta de crear solo algunas aristas del grafo responde a la necesidad de evitar evaluar la similitud entre servicios que a priori se saben que son incompatibles, por ejemplo dos servicios cuyas tareas se solapen. El objetivo último al reducir el número de aristas, es el de mejorar los tiempos de cómputo y reducir el consumo de memoria. Este tipo de mejoras han sido necesarias debido a la alta dimensionalidad del problema abordado. Cabe destacar, que algunas de las instancias a resolver cuentan con más de 10.700 servicios, lo cual para un grafo completo se traduciría en la creación de casi 60 millones de aristas, en las que además de la propia arista, se debería almacenar información sobre los niveles de feromona.

El siguiente reto y aportación de las técnicas basadas en ACO para tratar el HCSP, ha sido realizar una primera adaptación del algoritmo propuesto por (Dorigo y Gambardella, 1997) desde una perspectiva de clústering. Como resultado de dicha adaptación en la sección 4.3.2 se presentó la técnica denominada ACS-HCSP, la cual fue diseñada con la directriz de realizar el menor número de cambios posibles respecto de la técnica original propuesta por (Dorigo y Gambardella, 1997). Entre los cambios realizados destacan los siguientes: la creación de una función de selección del nodo inicial para dar comienzo a la construcción de un nuevo clúster, la adaptación de la regla de transición probabilística y la creación de una función que determina la finalización del clúster en construcción. El resto de funciones como las reglas de evaporación y actualización de feromona no han sufrido cambios respecto a las propuestas por (Dorigo y Gambardella, 1997).

Una vez analizados los resultados de la técnica ACS-HCSP, se presentó la técnica IACS-HCSP cuyo principal objetivo fue solventar las vías de mejora detectadas en ACS-HCSP,

objetivo que se consigue a través de dos propuestas novedosas. La primera vía de mejora detectada está relacionada con el criterio de parada. La función de dicho criterio de parada consiste en determinar cuando el clúster que está en construcción se debe dar por finalizado y se debe comenzar con la construcción uno nuevo. La aproximación sencilla sugerida por ACS-HCSP, consistente en parar cuando no existe ningún servicio compatible (ver sección 4.3.2.3), no obtiene buenos resultados en determinados escenarios. En concreto cuando el objetivo no es minimizar el número de asistentes requeridos, como es el caso de la función de coste propuesta por Quintana et al. (2017). Para solventar dicho problema se definió un nuevo criterio de parada probabilístico, el cual puede consultarse en la sección 4.3.3.3. En dicho criterio de parada, cada hormiga en cada ronda de ejecución define una eficiencia objetivo, la cuál determina la probabilidad de finalizar la construcción del clúster actual. Dicha eficiencia objetivo viene determinada y condicionada por la eficiencia de la mejor solución encontrada hasta el momento por la colonia. La segunda vía de mejora está relacionada con las aristas y con la definición de vértices alcanzables desde un vértice dado. Tal y como se presentó en la sección 4.3.3.2, la definición estática de aristas entre vértices puede ser demasiado restrictiva o demasiado permisiva. A medida que la hormiga visita nuevos vértices y conforma el clúster en construcción, la definición original de compatibilidad entre servicios puede dejar de ser válida. Este problema solo se presenta en problemas como el abordado; en el cual y dada la alta dimensionalidad de las instancias se necesita limitar el número de aristas existentes en el grafo. Para solucionarlo se propuso una nueva definición de vértices alcanzables, en la cual a medida que se construye un nuevo clúster el conjunto de vértices alcanzables se modifica de modo dinámico. Dicha definición presenta múltiples ventajas, en primer lugar, no requiere de la construcción de un grafo completo, lo cual supone un ahorro en tiempo de cómputo y de memoria. Puesto que el grafo original no se modifica, sino que es cada hormiga la que recalcula el conjunto de vértices alcanzables a medida que se avanza en la construcción del clúster actual, se permite una implementación paralelizable del algoritmo. Dicha implementación paralela ha sido aplicada a ambas propuestas ACS-HCSP y IACS-HCSP, permitiendo la obtención de soluciones en servidores de alto rendimiento tal y como se mostró en la sección 5.1.3.

Todas las propuestas han sido evaluadas y comparadas de modo exhaustivo a lo largo del capítulo 6, en primer lugar se compararon las tres técnicas propuestas entre sí a fin de determinar su rendimiento, tanto en calidad como en tiempo de ejecución, realizándose los pertinentes análisis de significación estadística. Una vez comprobado que las técnicas basadas en ACO obtienen un mejor rendimiento, se pasó a comparar de modo exhaustivo las técnicas ACS-HCSP y IACS-HCSP a fin de determinar si las modificaciones propuestas producen mejoras significativas. En total se han llevado a cabo 960 experimentos, los cuales han requerido un total de 194 horas de computación utilizando un servidor de alto rendimiento, cuyas características se pueden consultar en la sección 5.1.3. Para cada una de las 96 configuraciones posibles se han realizado pruebas estadísticas, en las cuales queda patente que la técnica IACS-HCSP obtiene mejores resultados que la técnica originalmente propuesta. Dichas pruebas de significancia estadística no solo se realizaron con los resultados finales obtenidos por cada técnica, sino por la población de soluciones obteniendo en ambos casos la técnica IACS-HCSP mejores resultados y siendo dichas mejoras significativas desde el punto de vista estadístico. En particular, se mostró como la definición de vértices alcanzables proporcionada en 4.3.3.2 proporciona a la técnica IACS-HCSP

mejores soluciones desde la primera ronda de evolución, no solo cuando el objetivo a minimizar es el número de asistentes requeridos, sino también cuando se quiere minimizar el coste de la solución la cual está basada en la propuesta de [Quintana et al. \(2017\)](#). Respecto al coste de la solución, se ha mostrado como la técnica original ACS-HCSP no es capaz de encontrar soluciones de buena calidad ni se aprecia una clara evolución en las sucesivas rondas. Por el contrario, la técnica IACS-HCSP gracias al criterio de parada definido en 4.3.3.3 es capaz de encontrar soluciones de buena calidad, observándose una clara mejoría en los resultados a medida que transcurren las rondas. Finalmente y con el objetivo de comparar las técnicas propuestas con técnicas existentes en el estado del arte, las técnicas propuestas son comparadas con las heurísticas propuestas por [Quintana et al. \(2017\)](#) y con la solución actual de la compañía. Ambas técnicas se muestran competitivas con los resultados existentes en el estado del arte, destacando los resultados obtenidos por la técnica IACS-HCSP la cual permite un ahorro económico estimado de 3.7 millones de euros anuales.

Por todo lo anteriormente expuesto se considera que esta investigación ha cumplido con éxito los objetivos establecidos al comienzo de la misma.

6.3. Limitaciones y líneas futuras

Dentro de las líneas futuras que no han sido abordadas dentro de esta investigación se pueden distinguir dos tipos, aquellas centradas en la mejora y modificación de los métodos propuestos y aquellas que pretendan extender las técnicas propuestas a otras instancias del HCSP o incluso a otros dominios de aplicación.

La extensión de las técnicas propuestas a otras instancias del HCSP utilizadas por otros investigadores es sin duda un trabajo interesante que podrá ser objeto de investigación y extensión en el futuro. Como se expuso con anterioridad las particularidades y normativa de cada región dificultan la reutilización de instancias [Fikar y Hirsch \(2017\)](#). En este sentido sería realmente útil la creación de una librería de pruebas, similar a la existente para algunos problemas de optimización combinatoria como el TSP [Reinelt \(1991\)](#), el VRP [Uchoa et al. \(2016\)](#) o el VRPTW [Solomon \(1987\)](#); [Gehring y Homberger \(2001\)](#). En la investigación actual siempre se ha considerado que los asistentes se desplazan entre los distintos clientes a pie, una posible ampliación sería considerar que dicho desplazamiento se puede realizar de distintos modos, a pie, en transporte público o en coche particular. Como se mostró en la sección 3.3.2 del capítulo 3 la obtención de la matriz de distancias para instancias de una alta dimensionalidad, como la tratada en esta investigación, puede suponer un coste elevado, el cual se vería multiplicado si los asistentes pueden utilizar varios medios de transporte. Atendiendo a la eficiencia del medio de transporte, el uso del vehículo privado para visitar a los clientes puede resultar excesivamente costoso, siendo más eficiente y respetuoso con el medio ambiente compartir vehículo entre varios asistentes, alternativa que podría ser explorada en el futuro y de la cual existen algunos trabajos previos [Fikar y Hirsch \(2018\)](#). Integrar las distintas alternativas de transporte, modelar los tiempos de desplazamiento y las distintas interacciones entre los asistentes con el fin de evaluar una solución puede resultar bastante complejo. En este sentido, la utilización de herramientas de simulación pueden ser de gran ayuda, en concreto el simulador multiagente ([Horni et al., 2016](#)), puede resultar una alternativa útil con el fin de solventar dicha complejidad,

ya que permite distintos tipos de transporte, como ir a pie, en transporte público o en automóvil, así como las distintas interacciones entre los agentes implicados en la simulación.

Otra línea de investigación a explorar es la aplicación de las técnicas propuestas, especialmente aquellas basadas en ACO, a otros dominios en los que se requieran la resolución de problemas de optimización combinatoria de una alta dimensionalidad. Uno de los primeros dominios de aplicación a explorar, sería la resolución de subproblemas de la familia VRP o m-TSP. En especial aquellos, en los existan restricciones entre los vértices que pueden ser visitados dentro de la misma ruta y en los que se deban realizar planificaciones dentro de un horizonte temporal de varios días. Ejemplos de este tipo de problemas los encontramos entre las innumerables variantes del VRP, como es el caso del PVRP (Periodic Vehicle Routing Problem) [Francis et al. \(2008\)](#) en el que se debe elegir entre distintos patrones de visita semanales o el caso del conVRP (Consistent VRP) [Groër et al. \(2009\)](#) en el cual se debe garantizar que cada cliente es atendido siempre por el mismo conductor a lo largo de horizonte de planificación. Otro dominio de aplicación distinto, sería la resolución de problemas de planificación con recursos limitados [Klein \(2000\)](#), en este tipo de problemas se deben llevar a cabo una serie de tareas entre las cuales existen dependencias temporales y para las que se requiere una serie de recursos limitados.

Finalmente la tercera línea de investigación la componen aquellas mejoras que podrían ser implementadas en las técnicas propuestas ACS-HCSP y IACS-HCSP. La primera vía de mejora sería la inclusión de técnicas de búsqueda local similares a las propuestas en el capítulo 2. De modo alternativo, se podría estudiar la viabilidad de utilizar técnicas exactas para determinar el orden de visita de los clientes en aquellos escenarios en los que las ventanas temporales fueran muy extensas. La idea subyacente consistiría en resolver un TSPTW utilizando una aproximación exacta para cada uno de los días de la semana y los servicios seleccionados.

Apéndices

Apéndice A

Resultados detallados

En este apéndice se han incluido todos los resultados obtenidos por cada una de las tres técnicas propuestas y que han sido objeto de experimentación a lo largo del capítulo 5. Todos los resultados fueron obtenidos utilizando un servidor (n1-highcpu-64) de la plataforma [Google Cloud Platform \(2017\)](#) con un procesador Intel Xeon de 64 núcleos y un total de 60Gb de memoria RAM.

A.1. Ward

A continuación se exponen los resultados de todos los experimentos realizados para el método de Ward modificado.

Agrupamiento	Dmax _k (min)	Wmax _k (min)	Trav _{obj} (min)	Wait _{obj} (min)	Tot _{obj} (min)	Clust _{obj}
A	30	30	94765	72950	2682330	4967
A	30	30	94109	74844	2683568	4957
A	30	30	94712	72986	2682313	4963
A	30	30	95613	73759	2683987	4953
A	30	30	93905	74545	2683065	4962
A	30	30	94724	74324	2683663	4962
A	30	30	95328	73785	2683728	4962
A	30	30	93512	72968	2681095	4977
A	30	30	95843	74180	2684638	4954
A	30	30	93358	72878	2680851	4975
A	30	∞	148643	355013	3018271	3926
A	30	∞	146572	349704	3010891	3937
A	30	∞	148950	359288	3022853	3928
A	30	∞	147696	358078	3020389	3937
A	30	∞	147390	358768	3020773	3966
A	30	∞	147976	356228	3018819	3930
A	30	∞	146988	352331	3013934	3931
A	30	∞	148910	358829	3022354	3920
A	30	∞	149020	356214	3019849	3927
A	30	∞	146154	359115	3019884	3920
B	30	30	94186	74014	2682815	4959
B	30	30	94552	72863	2682030	4969

Continúa en la siguiente hoja

Agrupamiento	Dmax _k (min)	Wmax _k (min)	Trav _{obj} (min)	Wait _{obj} (min)	Tot _{obj} (min)	Clust _{obj}
B	30	30	94224	73874	2682713	4963
B	30	30	92978	73855	2681448	4950
B	30	30	93494	74047	2682156	4964
B	30	30	94501	73159	2682275	4957
B	30	30	93523	73134	2681272	4966
B	30	30	94365	74623	2683603	4962
B	30	30	93358	74259	2682232	4957
B	30	30	94777	74688	2684080	4939
B	30	∞	150774	416538	3081928	3839
B	30	∞	151510	424203	3090328	3804
B	30	∞	152409	422566	3089590	3793
B	30	∞	152037	419437	3086089	3792
B	30	∞	152373	425635	3092623	3788
B	30	∞	153272	416113	3084000	3790
B	30	∞	150612	425434	3090661	3801
B	30	∞	153000	421798	3089413	3799
B	30	∞	151896	431205	3097716	3796
B	30	∞	152574	417842	3085031	3793
C	30	30	79775	61365	2655755	4433
C	30	30	78148	62640	2655403	4435
C	30	30	80313	62921	2657849	4426
C	30	30	78826	59877	2653318	4449
C	30	30	78317	61191	2654123	4439
C	30	30	78706	61701	2655022	4451
C	30	30	79945	61455	2656015	4448
C	30	30	79887	61118	2655620	4453
C	30	30	79280	61693	2655588	4443
C	30	30	79678	61749	2656042	4442
C	30	∞	153503	611284	3279402	3255
C	30	∞	154541	610767	3279923	3254
C	30	∞	150475	611684	3276774	3272
C	30	∞	151850	628358	3294823	3265
C	30	∞	152247	606178	3273040	3261
C	30	∞	153047	619076	3286738	3251
C	30	∞	151291	597839	3263745	3272
C	30	∞	151995	619961	3286572	3277
C	30	∞	152212	612120	3278947	3268
C	30	∞	150086	609593	3274294	3290

Tabla A.1: Resultados detallados obtenidos con el método de Ward para las distintas configuraciones.

A.2. ACS-HCSP

A continuación se exponen los resultados de todos los experimentos realizados para el método ACS-HCSP cuyos detalles de implementación fueron detallados en la sección 4.3.2. Tal y como se expuso en el capítulo 5 para la técnica ACS se han llevado a cabo un total 96 experimentos, repetidos cada uno de ellos 5 veces dada la naturaleza estocástica de la técnica utilizada. Dicha experimentación ha requerido un total de 100 horas de cómputo en la plataforma Google Cloud en un servidor con 64 núcleos y 60 Gb de Ram.

La parametrización utilizada para dichos experimentos es la siguiente.

- **Tipo de agrupamiento:**[Tipo A, Tipo B] Se han utilizado todos los tipos de agrupamientos A y B.
- **Tiempo máximo de desplazamiento:**[15, 30] Se establecido el tiempo máximo de desplazamiento entre 15 y 30 minutos.
- **Tiempo máximo de espera:**[15, 30, 45] Se han establecido tres tiempos máximos de espera, el primero de ellos de 15 minutos, el segundo de 30 y el último de 45 minutos.
- **Ventana temporal** $|\alpha_{el} - \alpha_{ei}|$ [0,5,10,20] Se han establecido cuatro tipos de ventanas temporales, la primera de ellas sin ventana lo cual nos permitirá comparar los resultados con los resultados obtenidos con el método de Ward, las demás ventanas temporales se han establecido a intervalos de 5, 10 y 20 minutos respectivamente.
- **Función de evaluación** $[F_{clust}, F_{cost}]$ Se ha utilizado los dos tipos de funciones de evaluación definidas en 5.1.2. En la primera de ellas F_{clust} se minimiza el número de clusters, en la segunda se minimiza la función F_{cost} definida en (3.7).

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	15	0	<i>F_{clust}</i>	79238	59224	2653077	6225	6225	267
A	15	15	0	<i>F_{clust}</i>	77681	59222	2651518	6239	6239	263
A	15	15	0	<i>F_{clust}</i>	77737	59796	2652148	6242	6242	265
A	15	15	0	<i>F_{clust}</i>	78918	60091	2653624	6252	6252	265
A	15	15	0	<i>F_{clust}</i>	78718	59750	2653083	6225	6225	263
A	15	15	5	<i>F_{clust}</i>	89792	43322	2647728	4613	4613	281
A	15	15	5	<i>F_{clust}</i>	90233	44652	2649500	4624	4624	283
A	15	15	5	<i>F_{clust}</i>	91513	43538	2649666	4619	4619	280
A	15	15	5	<i>F_{clust}</i>	90953	44116	2649684	4617	4617	281
A	15	15	5	<i>F_{clust}</i>	90037	44119	2648771	4605	4605	281
A	15	15	10	<i>F_{clust}</i>	100687	36065	2651366	4189	4189	318
A	15	15	10	<i>F_{clust}</i>	100780	35375	2650769	4202	4202	314
A	15	15	10	<i>F_{clust}</i>	100634	34740	2649989	4211	4211	315
A	15	15	10	<i>F_{clust}</i>	100599	35347	2650561	4209	4209	316
A	15	15	10	<i>F_{clust}</i>	100633	35914	2651161	4206	4206	317
A	15	15	20	<i>F_{clust}</i>	112154	25991	2652760	3860	3860	378
A	15	15	20	<i>F_{clust}</i>	111374	26471	2652460	3869	3869	377
A	15	15	20	<i>F_{clust}</i>	111865	25808	2652289	3866	3866	374
A	15	15	20	<i>F_{clust}</i>	112346	27007	2653968	3849	3849	377
A	15	15	20	<i>F_{clust}</i>	113540	26013	2662733	3880	3880	378
A	15	30	0	<i>F_{clust}</i>	103925	162392	2780932	4435	4435	321
A	15	30	0	<i>F_{clust}</i>	106765	167824	2789204	4420	4420	320
A	15	30	0	<i>F_{clust}</i>	105836	164766	2785217	4419	4419	319
A	15	30	0	<i>F_{clust}</i>	105213	162931	2782758	4426	4426	320
A	15	30	0	<i>F_{clust}</i>	106296	164091	2785002	4392	4392	322
A	15	30	5	<i>F_{clust}</i>	107290	122739	2744643	3973	3973	363
A	15	30	5	<i>F_{clust}</i>	108024	126186	2748825	3985	3985	360
A	15	30	5	<i>F_{clust}</i>	107599	123188	2745402	3997	3997	361
A	15	30	5	<i>F_{clust}</i>	106888	123899	2745402	3978	3978	360
A	15	30	5	<i>F_{clust}</i>	108270	121615	2744500	3988	3988	361
A	15	30	10	<i>F_{clust}</i>	112458	99429	2726502	3794	3794	412
A	15	30	10	<i>F_{clust}</i>	112675	99738	2727028	3804	3804	410
A	15	30	10	<i>F_{clust}</i>	111454	102136	2728204	3798	3798	408
A	15	30	10	<i>F_{clust}</i>	113641	99035	2727292	3797	3797	405

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	30	10	<i>F_{clust}</i>	112645	99969	2735794	3842	3842	409
A	15	30	20	<i>F_{clust}</i>	119518	76030	2710163	3560	3560	482
A	15	30	20	<i>F_{clust}</i>	119734	73869	2708218	3571	3571	481
A	15	30	20	<i>F_{clust}</i>	119090	74830	2708535	3588	3588	481
A	15	30	20	<i>F_{clust}</i>	119918	75062	2709596	3572	3572	478
A	15	30	20	<i>F_{clust}</i>	119334	75698	2709647	3561	3561	481
A	15	45	0	<i>F_{clust}</i>	109156	216059	2839829	4187	4187	381
A	15	45	0	<i>F_{clust}</i>	110208	216934	2841757	4177	4177	380
A	15	45	0	<i>F_{clust}</i>	110103	215255	2839973	4171	4171	380
A	15	45	0	<i>F_{clust}</i>	109513	217780	2841908	4188	4188	379
A	15	45	0	<i>F_{clust}</i>	109012	216373	2840000	4185	4185	381
A	15	45	5	<i>F_{clust}</i>	110056	162820	2787491	3845	3845	431
A	15	45	5	<i>F_{clust}</i>	110254	166213	2791082	3837	3837	427
A	15	45	5	<i>F_{clust}</i>	110659	163447	2788721	3831	3831	428
A	15	45	5	<i>F_{clust}</i>	109511	166666	2790793	3840	3840	427
A	15	45	5	<i>F_{clust}</i>	110195	166530	2792060	3844	3844	428
A	15	45	10	<i>F_{clust}</i>	114313	139195	2768123	3659	3659	476
A	15	45	10	<i>F_{clust}</i>	115175	137749	2767539	3672	3672	476
A	15	45	10	<i>F_{clust}</i>	116496	139386	2779061	3697	3697	475
A	15	45	10	<i>F_{clust}</i>	115655	136549	2766819	3677	3677	476
A	15	45	10	<i>F_{clust}</i>	113673	137347	2765634	3664	3664	477
A	15	45	20	<i>F_{clust}</i>	122745	107801	2745161	3461	3461	553
A	15	45	20	<i>F_{clust}</i>	121758	105769	2742142	3465	3465	552
A	15	45	20	<i>F_{clust}</i>	122259	108422	2745296	3453	3453	550
A	15	45	20	<i>F_{clust}</i>	122112	107954	2744681	3455	3455	551
A	15	45	20	<i>F_{clust}</i>	121925	106035	2742575	3475	3475	553
A	30	15	0	<i>F_{clust}</i>	196772	85867	2797254	4213	4213	358
A	30	15	0	<i>F_{clust}</i>	196026	85355	2795996	4212	4212	358
A	30	15	0	<i>F_{clust}</i>	196745	87058	2799138	4225	4225	358
A	30	15	0	<i>F_{clust}</i>	194054	85578	2794247	4227	4227	355
A	30	15	0	<i>F_{clust}</i>	196185	86032	2796832	4223	4223	357
A	30	15	5	<i>F_{clust}</i>	183977	60271	2758862	3849	3849	439
A	30	15	5	<i>F_{clust}</i>	189656	58662	2762933	3833	3833	435
A	30	15	5	<i>F_{clust}</i>	186836	59187	2760638	3835	3835	437
A	30	15	5	<i>F_{clust}</i>	184634	59501	2758750	3848	3848	434

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	15	5	<i>F_{clust}</i>	184973	58876	2758464	3830	3830	437
A	30	15	10	<i>F_{clust}</i>	188708	44958	2749001	3679	3679	534
A	30	15	10	<i>F_{clust}</i>	188371	43486	2746471	3653	3653	529
A	30	15	10	<i>F_{clust}</i>	187951	44572	2747138	3665	3665	529
A	30	15	10	<i>F_{clust}</i>	188960	43184	2746759	3649	3649	533
A	30	15	10	<i>F_{clust}</i>	188020	44712	2747347	3658	3658	532
A	30	15	20	<i>F_{clust}</i>	198692	30629	2743936	3435	3435	726
A	30	15	20	<i>F_{clust}</i>	198425	32436	2754041	3463	3463	721
A	30	15	20	<i>F_{clust}</i>	200300	30812	2745727	3406	3406	716
A	30	15	20	<i>F_{clust}</i>	200307	31071	2745993	3416	3416	723
A	30	15	20	<i>F_{clust}</i>	200579	32150	2747344	3430	3430	717
A	30	30	0	<i>F_{clust}</i>	196794	156746	2868155	3890	3890	552
A	30	30	0	<i>F_{clust}</i>	198582	158543	2871740	3867	3867	547
A	30	30	0	<i>F_{clust}</i>	197228	155957	2867799	3889	3889	550
A	30	30	0	<i>F_{clust}</i>	198831	158155	2871601	3878	3878	548
A	30	30	0	<i>F_{clust}</i>	194719	156903	2866237	3883	3883	555
A	30	30	5	<i>F_{clust}</i>	186193	119693	2820501	3627	3627	638
A	30	30	5	<i>F_{clust}</i>	190514	119191	2824321	3627	3627	633
A	30	30	5	<i>F_{clust}</i>	184546	115795	2814956	3631	3631	635
A	30	30	5	<i>F_{clust}</i>	187818	119829	2822262	3614	3614	630
A	30	30	5	<i>F_{clust}</i>	188413	117685	2820713	3635	3635	633
A	30	30	10	<i>F_{clust}</i>	191080	96248	2801943	3472	3472	737
A	30	30	10	<i>F_{clust}</i>	193194	92945	2800754	3473	3473	734
A	30	30	10	<i>F_{clust}</i>	188689	93126	2796430	3478	3478	728
A	30	30	10	<i>F_{clust}</i>	189414	96058	2800086	3469	3469	733
A	30	30	10	<i>F_{clust}</i>	195008	93284	2802907	3466	3466	740
A	30	30	20	<i>F_{clust}</i>	197388	68017	2780020	3271	3271	938
A	30	30	20	<i>F_{clust}</i>	199041	68244	2781900	3280	3280	933
A	30	30	20	<i>F_{clust}</i>	199013	67824	2781452	3287	3287	932
A	30	30	20	<i>F_{clust}</i>	200281	68188	2783083	3295	3295	930
A	30	30	20	<i>F_{clust}</i>	196829	67970	2779414	3280	3280	929
A	30	45	0	<i>F_{clust}</i>	206202	218284	2939101	3693	3693	756
A	30	45	0	<i>F_{clust}</i>	205275	221847	2941737	3695	3695	752
A	30	45	0	<i>F_{clust}</i>	209425	221578	2945618	3693	3693	757
A	30	45	0	<i>F_{clust}</i>	207571	217124	2939310	3683	3683	744

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	45	0	F_{clust}	207843	220110	2942568	3689	3689	755
A	30	45	5	F_{clust}	198023	167174	2879812	3496	3496	844
A	30	45	5	F_{clust}	196569	164856	2876040	3486	3486	839
A	30	45	5	F_{clust}	196305	166915	2877835	3485	3485	836
A	30	45	5	F_{clust}	195636	166717	2876968	3500	3500	844
A	30	45	5	F_{clust}	194453	167291	2876360	3500	3500	839
A	30	45	10	F_{clust}	195174	133662	2843451	3378	3378	941
A	30	45	10	F_{clust}	197753	136587	2848955	3379	3379	938
A	30	45	10	F_{clust}	199183	133320	2847118	3378	3378	935
A	30	45	10	F_{clust}	197260	134596	2846471	3383	3383	937
A	30	45	10	F_{clust}	195908	132473	2842996	3384	3384	942
A	30	45	20	F_{clust}	204508	103991	2823114	3191	3191	1144
A	30	45	20	F_{clust}	203589	99536	2817740	3184	3184	1146
A	30	45	20	F_{clust}	204761	99135	2818512	3182	3182	1147
A	30	45	20	F_{clust}	202905	103276	2820796	3196	3196	1146
A	30	45	20	F_{clust}	203186	100546	2818347	3183	3183	1147
A	15	15	0	F_{cost}	72664	56616	2643895	6332	684369	265
A	15	15	0	F_{cost}	74325	56053	2644993	6303	684316	264
A	15	15	0	F_{cost}	75038	57367	2647019	6257	684299	265
A	15	15	0	F_{cost}	74416	56316	2645347	6293	684292	264
A	15	15	0	F_{cost}	73903	56506	2645024	6295	684238	264
A	15	15	5	F_{cost}	85472	39347	2647998	4757	668547	282
A	15	15	5	F_{cost}	85555	39173	2639343	4686	665771	281
A	15	15	5	F_{cost}	84423	39591	2639349	4732	666262	281
A	15	15	5	F_{cost}	85860	40918	2641393	4674	666121	282
A	15	15	5	F_{cost}	85981	40542	2641138	4675	666072	281
A	15	15	10	F_{cost}	95481	31172	2641268	4285	661948	318
A	15	15	10	F_{cost}	95696	30895	2641205	4272	661794	315
A	15	15	10	F_{cost}	95424	31612	2641651	4275	661931	314
A	15	15	10	F_{cost}	94306	31585	2640506	4291	661834	315
A	15	15	10	F_{cost}	94916	31470	2641001	4281	661843	316
A	15	15	20	F_{cost}	106217	23705	2644537	3912	658736	380
A	15	15	20	F_{cost}	106827	22874	2644315	3933	658908	377
A	15	15	20	F_{cost}	107208	22341	2644164	3929	658830	378
A	15	15	20	F_{cost}	105687	24234	2644536	3900	658608	377

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	15	20	F _{cost}	106183	22281	2643079	3937	658663	377
A	15	30	0	F _{cost}	84051	126749	2725415	5239	691746	320
A	15	30	0	F _{cost}	83918	127642	2726175	5261	692157	320
A	15	30	0	F _{cost}	84225	126072	2724912	5271	691969	319
A	15	30	0	F _{cost}	85602	125713	2725930	5234	691813	320
A	15	30	0	F _{cost}	83619	126697	2724931	5275	692016	320
A	15	30	5	F _{cost}	100178	106934	2721727	4134	679113	367
A	15	30	5	F _{cost}	102196	106593	2723405	4102	679163	361
A	15	30	5	F _{cost}	99861	106778	2721254	4153	679205	363
A	15	30	5	F _{cost}	100701	107610	2722926	4128	679329	362
A	15	30	5	F _{cost}	101605	104908	2721128	4106	678675	359
A	15	30	10	F _{cost}	106221	83523	2704359	3898	672546	411
A	15	30	10	F _{cost}	106849	86448	2707912	3894	673332	405
A	15	30	10	F _{cost}	107427	86045	2708087	3868	673096	408
A	15	30	10	F _{cost}	105850	85102	2705567	3899	672838	407
A	15	30	10	F _{cost}	104335	85249	2704199	3932	672871	410
A	15	30	20	F _{cost}	114493	60751	2689859	3694	666989	480
A	15	30	20	F _{cost}	115050	62110	2691775	3613	666573	481
A	15	30	20	F _{cost}	114486	61336	2690437	3656	666719	483
A	15	30	20	F _{cost}	113928	62073	2690616	3649	666686	478
A	15	30	20	F _{cost}	112590	61328	2688534	3696	666701	478
A	15	45	0	F _{cost}	88909	165474	2768998	5008	699454	381
A	15	45	0	F _{cost}	88990	164168	2767773	5036	699466	377
A	15	45	0	F _{cost}	87353	165231	2767199	5047	699450	377
A	15	45	0	F _{cost}	88304	162439	2765358	5059	699148	378
A	15	45	0	F _{cost}	87093	164294	2766002	5076	699479	378
A	15	45	5	F _{cost}	87987	123441	2726044	4670	685830	428
A	15	45	5	F _{cost}	89657	122577	2726848	4656	685869	431
A	15	45	5	F _{cost}	91034	123428	2729077	4628	686091	427
A	15	45	5	F _{cost}	90213	122097	2726925	4657	685897	426
A	15	45	5	F _{cost}	90224	123124	2727963	4635	685905	429
A	15	45	10	F _{cost}	92829	101288	2708732	4433	679266	479
A	15	45	10	F _{cost}	93765	98042	2706422	4421	678599	476
A	15	45	10	F _{cost}	93585	100219	2708420	4428	679140	475
A	15	45	10	F _{cost}	93313	98921	2706849	4458	679093	474

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	45	10	F _{cost}	92869	99448	2706932	4447	678995	478
A	15	45	20	F _{cost}	99262	73798	2687675	4200	671870	554
A	15	45	20	F _{cost}	100173	73426	2688214	4179	671772	547
A	15	45	20	F _{cost}	116436	86167	2717938	3571	672230	551
A	15	45	20	F _{cost}	100344	75461	2690420	4151	671989	548
A	15	45	20	F _{cost}	99878	73654	2688146	4170	671661	554
A	30	15	0	F _{cost}	150402	67952	2732969	5126	692304	359
A	30	15	0	F _{cost}	151163	67452	2733230	5133	692440	356
A	30	15	0	F _{cost}	152803	68037	2736175	5107	692850	356
A	30	15	0	F _{cost}	152455	67442	2734512	5089	692270	356
A	30	15	0	F _{cost}	151770	68068	2734453	5093	692299	358
A	30	15	5	F _{cost}	143644	44893	2703152	4716	680979	438
A	30	15	5	F _{cost}	165473	52817	2732905	4023	680538	434
A	30	15	5	F _{cost}	169784	53443	2737842	3954	680955	433
A	30	15	5	F _{cost}	167797	51659	2734071	4034	680927	434
A	30	15	5	F _{cost}	168537	53807	2736959	3955	680760	435
A	30	15	10	F _{cost}	171152	38519	2724286	3780	675938	533
A	30	15	10	F _{cost}	172391	38987	2725994	3766	676188	532
A	30	15	10	F _{cost}	169544	37864	2722023	3817	675804	529
A	30	15	10	F _{cost}	171763	38815	2725192	3769	676033	532
A	30	15	10	F _{cost}	144274	31996	2690884	4525	676082	531
A	30	15	20	F _{cost}	153604	20609	2688827	4203	672171	726
A	30	15	20	F _{cost}	156326	21335	2700841	4195	674889	716
A	30	15	20	F _{cost}	153074	21698	2689387	4211	672387	716
A	30	15	20	F _{cost}	153466	20498	2688580	4195	672028	719
A	30	15	20	F _{cost}	155253	21536	2691405	4141	672112	727
A	30	30	0	F _{cost}	152326	119244	2786185	4788	701120	547
A	30	30	0	F _{cost}	150562	118357	2784254	4828	701096	546
A	30	30	0	F _{cost}	151290	120562	2786467	4780	701101	545
A	30	30	0	F _{cost}	150857	118860	2784332	4828	701114	546
A	30	30	0	F _{cost}	150370	118511	2783496	4823	700866	547
A	30	30	5	F _{cost}	147026	87264	2748905	4439	688704	636
A	30	30	5	F _{cost}	146234	87775	2748624	4483	689107	632
A	30	30	5	F _{cost}	145011	88054	2747680	4478	688833	628
A	30	30	5	F _{cost}	146444	88098	2749157	4464	689029	635

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	30	5	F _{cost}	148241	89719	2752574	4398	689123	632
A	30	30	10	F _{cost}	150546	67768	2732929	4263	683101	725
A	30	30	10	F _{cost}	150239	67230	2732084	4250	682765	731
A	30	30	10	F _{cost}	148465	66178	2729258	4293	682564	723
A	30	30	10	F _{cost}	148965	69061	2732641	4238	682767	732
A	30	30	10	F _{cost}	149712	65852	2730179	4257	682395	731
A	30	30	20	F _{cost}	154412	48116	2717144	4012	676743	923
A	30	30	20	F _{cost}	156832	47320	2718767	4009	677090	924
A	30	30	20	F _{cost}	155982	45501	2716098	4014	676521	926
A	30	30	20	F _{cost}	156388	47309	2718312	4000	676888	930
A	30	30	20	F _{cost}	158593	47077	2720285	3975	677082	925
A	30	45	0	F _{cost}	164717	163705	2843037	4583	712202	754
A	30	45	0	F _{cost}	163519	166557	2844691	4592	712684	743
A	30	45	0	F _{cost}	163177	167040	2844832	4557	712344	747
A	30	45	0	F _{cost}	162272	167699	2844586	4576	712489	744
A	30	45	0	F _{cost}	164428	166404	2845447	4549	712402	743
A	30	45	5	F _{cost}	160546	122632	2797793	4202	697586	842
A	30	45	5	F _{cost}	159562	123583	2797760	4249	698079	829
A	30	45	5	F _{cost}	159624	122586	2796825	4242	697786	838
A	30	45	5	F _{cost}	157561	123681	2795857	4267	697827	844
A	30	45	5	F _{cost}	158293	122779	2795688	4251	697617	841
A	30	45	10	F _{cost}	156934	97902	2769451	4122	690120	942
A	30	45	10	F _{cost}	157001	96110	2767726	4110	689590	941
A	30	45	10	F _{cost}	159056	98947	2772618	4075	690359	934
A	30	45	10	F _{cost}	158384	98078	2771078	4117	690447	944
A	30	45	10	F _{cost}	154624	97047	2766286	4192	690128	941
A	30	45	20	F _{cost}	165237	69555	2749408	3839	682429	1147
A	30	45	20	F _{cost}	164258	69778	2748651	3891	682806	1147
A	30	45	20	F _{cost}	163145	71617	2749378	3861	682656	1145
A	30	45	20	F _{cost}	164418	70999	2750032	3870	682905	1143
A	30	45	20	F _{cost}	166330	70668	2751613	3826	682805	1141
B	15	15	0	F _{clust}	78939	59127	2652681	6227	6227	251
B	15	15	0	F _{clust}	77033	59342	2650990	6245	6245	265
B	15	15	0	F _{clust}	78918	59644	2653177	6241	6241	263
B	15	15	0	F _{clust}	77887	59723	2652225	6230	6230	267

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	15	0	<i>F_{clust}</i>	78617	59458	2652690	6209	6209	263
B	15	15	5	<i>F_{clust}</i>	90856	43228	2648699	4612	4612	280
B	15	15	5	<i>F_{clust}</i>	90706	44696	2650017	4611	4611	281
B	15	15	5	<i>F_{clust}</i>	90780	44389	2649783	4616	4616	282
B	15	15	5	<i>F_{clust}</i>	89628	42736	2646979	4606	4606	280
B	15	15	5	<i>F_{clust}</i>	91429	44655	2650698	4597	4597	281
B	15	15	10	<i>F_{clust}</i>	99927	35981	2650523	4191	4191	315
B	15	15	10	<i>F_{clust}</i>	100369	35570	2650554	4187	4187	315
B	15	15	10	<i>F_{clust}</i>	99752	34632	2648999	4177	4177	314
B	15	15	10	<i>F_{clust}</i>	100591	35669	2650875	4205	4205	315
B	15	15	10	<i>F_{clust}</i>	100678	35777	2651071	4196	4196	314
B	15	15	20	<i>F_{clust}</i>	112697	26064	2653376	3851	3851	377
B	15	15	20	<i>F_{clust}</i>	112981	26307	2653904	3855	3855	373
B	15	15	20	<i>F_{clust}</i>	112261	26830	2653707	3858	3858	374
B	15	15	20	<i>F_{clust}</i>	112673	26954	2654241	3858	3858	374
B	15	15	20	<i>F_{clust}</i>	112417	25993	2653025	3859	3859	375
B	15	30	0	<i>F_{clust}</i>	105596	164650	2784861	4413	4413	322
B	15	30	0	<i>F_{clust}</i>	105261	162923	2782799	4413	4413	319
B	15	30	0	<i>F_{clust}</i>	106512	165068	2786195	4410	4410	320
B	15	30	0	<i>F_{clust}</i>	105751	164833	2785199	4396	4396	318
B	15	30	0	<i>F_{clust}</i>	105679	164263	2793122	4472	4472	319
B	15	30	5	<i>F_{clust}</i>	106773	122605	2743994	3967	3967	360
B	15	30	5	<i>F_{clust}</i>	107165	125654	2747434	3958	3958	358
B	15	30	5	<i>F_{clust}</i>	107824	123676	2746115	3990	3990	358
B	15	30	5	<i>F_{clust}</i>	106697	125167	2746479	3974	3974	358
B	15	30	5	<i>F_{clust}</i>	107960	125285	2747860	3970	3970	359
B	15	30	10	<i>F_{clust}</i>	112675	100039	2727329	3789	3789	406
B	15	30	10	<i>F_{clust}</i>	111941	99538	2726093	3802	3802	404
B	15	30	10	<i>F_{clust}</i>	111855	100740	2727210	3797	3797	402
B	15	30	10	<i>F_{clust}</i>	113577	100861	2729053	3783	3783	404
B	15	30	10	<i>F_{clust}</i>	111958	100588	2727162	3795	3795	403
B	15	30	20	<i>F_{clust}</i>	120286	72700	2707601	3556	3556	478
B	15	30	20	<i>F_{clust}</i>	121571	76264	2712450	3549	3549	475
B	15	30	20	<i>F_{clust}</i>	120894	74365	2709874	3544	3544	474
B	15	30	20	<i>F_{clust}</i>	120871	74594	2710080	3554	3554	473

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	30	20	<i>F_{clust}</i>	120807	74290	2709712	3564	3564	474
B	15	45	0	<i>F_{clust}</i>	111420	219691	2845726	4161	4161	378
B	15	45	0	<i>F_{clust}</i>	110496	218749	2843859	4168	4168	377
B	15	45	0	<i>F_{clust}</i>	109459	214632	2838706	4171	4171	377
B	15	45	0	<i>F_{clust}</i>	110068	218306	2842989	4163	4163	377
B	15	45	0	<i>F_{clust}</i>	111045	218349	2844009	4149	4149	378
B	15	45	5	<i>F_{clust}</i>	109062	164647	2788324	3810	3810	425
B	15	45	5	<i>F_{clust}</i>	110950	166144	2791709	3824	3824	423
B	15	45	5	<i>F_{clust}</i>	110297	167526	2792438	3832	3832	422
B	15	45	5	<i>F_{clust}</i>	109342	166295	2790252	3830	3830	422
B	15	45	5	<i>F_{clust}</i>	110535	166296	2791446	3827	3827	423
B	15	45	10	<i>F_{clust}</i>	115024	140371	2770010	3644	3644	473
B	15	45	10	<i>F_{clust}</i>	114600	142068	2771283	3643	3643	470
B	15	45	10	<i>F_{clust}</i>	114728	137049	2766391	3652	3652	471
B	15	45	10	<i>F_{clust}</i>	114399	136089	2765102	3657	3657	469
B	15	45	10	<i>F_{clust}</i>	115442	138093	2768150	3634	3634	470
B	15	45	20	<i>F_{clust}</i>	121486	105848	2741949	3446	3446	549
B	15	45	20	<i>F_{clust}</i>	122088	103482	2740186	3466	3466	543
B	15	45	20	<i>F_{clust}</i>	121710	108716	2745041	3455	3455	542
B	15	45	20	<i>F_{clust}</i>	122821	105240	2742675	3456	3456	543
B	15	45	20	<i>F_{clust}</i>	122666	108876	2746156	3447	3447	546
B	30	15	0	<i>F_{clust}</i>	196379	87045	2798039	4225	4225	355
B	30	15	0	<i>F_{clust}</i>	194070	86081	2794766	4221	4221	354
B	30	15	0	<i>F_{clust}</i>	196230	86477	2797322	4218	4218	355
B	30	15	0	<i>F_{clust}</i>	195959	86212	2796786	4199	4199	356
B	30	15	0	<i>F_{clust}</i>	196130	86053	2796798	4204	4204	357
B	30	15	5	<i>F_{clust}</i>	189268	60775	2764658	3821	3821	431
B	30	15	5	<i>F_{clust}</i>	187318	59079	2761012	3833	3833	428
B	30	15	5	<i>F_{clust}</i>	188172	60686	2763473	3824	3824	429
B	30	15	5	<i>F_{clust}</i>	185576	58014	2758205	3826	3826	432
B	30	15	5	<i>F_{clust}</i>	186027	60532	2761174	3828	3828	432
B	30	15	10	<i>F_{clust}</i>	189598	44111	2748323	3630	3630	530
B	30	15	10	<i>F_{clust}</i>	189644	44407	2748666	3645	3645	523
B	30	15	10	<i>F_{clust}</i>	190178	44650	2749443	3655	3655	525
B	30	15	10	<i>F_{clust}</i>	190195	44623	2749432	3655	3655	523

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	15	10	<i>F_{clust}</i>	190839	46018	2751473	3643	3643	529
B	30	15	20	<i>F_{clust}</i>	199999	32341	2746955	3423	3423	721
B	30	15	20	<i>F_{clust}</i>	199336	31061	2745012	3406	3406	713
B	30	15	20	<i>F_{clust}</i>	199537	30758	2744911	3429	3429	711
B	30	15	20	<i>F_{clust}</i>	198321	31176	2744111	3428	3428	713
B	30	15	20	<i>F_{clust}</i>	199753	30809	2745177	3425	3425	714
B	30	30	0	<i>F_{clust}</i>	199109	160170	2873894	3843	3843	540
B	30	30	0	<i>F_{clust}</i>	193719	157913	2866247	3874	3874	537
B	30	30	0	<i>F_{clust}</i>	195664	156698	2866977	3862	3862	543
B	30	30	0	<i>F_{clust}</i>	196826	160291	2871732	3873	3873	544
B	30	30	0	<i>F_{clust}</i>	196114	158233	2868962	3876	3876	545
B	30	30	5	<i>F_{clust}</i>	185781	119183	2819579	3619	3619	632
B	30	30	5	<i>F_{clust}</i>	188249	119184	2822049	3611	3611	625
B	30	30	5	<i>F_{clust}</i>	186779	119075	2820469	3598	3598	628
B	30	30	5	<i>F_{clust}</i>	188137	119277	2830593	3644	3644	622
B	30	30	5	<i>F_{clust}</i>	188125	118347	2821087	3614	3614	629
B	30	30	10	<i>F_{clust}</i>	191142	93268	2799025	3468	3468	733
B	30	30	10	<i>F_{clust}</i>	187414	93678	2795707	3468	3468	722
B	30	30	10	<i>F_{clust}</i>	191217	93163	2798995	3473	3473	720
B	30	30	10	<i>F_{clust}</i>	191133	95399	2801146	3462	3462	723
B	30	30	10	<i>F_{clust}</i>	190549	96517	2801681	3465	3465	728
B	30	30	20	<i>F_{clust}</i>	200025	66250	2780890	3278	3278	923
B	30	30	20	<i>F_{clust}</i>	199399	64549	2778564	3274	3274	920
B	30	30	20	<i>F_{clust}</i>	198715	68708	2782038	3274	3274	916
B	30	30	20	<i>F_{clust}</i>	199042	67011	2780669	3275	3275	922
B	30	30	20	<i>F_{clust}</i>	199611	68759	2782984	3259	3259	924
B	30	45	0	<i>F_{clust}</i>	208553	220165	2943333	3673	3673	750
B	30	45	0	<i>F_{clust}</i>	208345	220934	2943894	3673	3673	742
B	30	45	0	<i>F_{clust}</i>	208115	222068	2944798	3674	3674	745
B	30	45	0	<i>F_{clust}</i>	205732	225013	2945360	3670	3670	744
B	30	45	0	<i>F_{clust}</i>	208246	220620	2943481	3655	3655	747
B	30	45	5	<i>F_{clust}</i>	195160	170616	2880391	3475	3475	834
B	30	45	5	<i>F_{clust}</i>	196045	163981	2874641	3481	3481	831
B	30	45	5	<i>F_{clust}</i>	195376	165971	2875962	3483	3483	828
B	30	45	5	<i>F_{clust}</i>	195262	164578	2874455	3488	3488	834

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	45	5	F_{clust}	196036	166552	2877204	3480	3480	832
B	30	45	10	F_{clust}	197587	133034	2845235	3363	3363	938
B	30	45	10	F_{clust}	195836	133156	2843608	3371	3371	934
B	30	45	10	F_{clust}	196312	134510	2845437	3363	3363	932
B	30	45	10	F_{clust}	199297	138325	2852237	3356	3356	933
B	30	45	10	F_{clust}	196506	137803	2848925	3331	3331	937
B	30	45	20	F_{clust}	205969	102495	2823079	3165	3165	1135
B	30	45	20	F_{clust}	204662	100749	2820026	3178	3178	1131
B	30	45	20	F_{clust}	206509	102837	2823961	3181	3181	1130
B	30	45	20	F_{clust}	204758	103244	2822616	3178	3178	1137
B	30	45	20	F_{clust}	205416	101286	2821317	3176	3176	1133
B	15	15	0	F_{cost}	74360	55547	2644522	6286	684025	265
B	15	15	0	F_{cost}	75031	56687	2646333	6285	684437	265
B	15	15	0	F_{cost}	74008	55462	2644085	6311	684190	263
B	15	15	0	F_{cost}	73078	55165	2642858	6351	684329	267
B	15	15	0	F_{cost}	73634	56277	2644526	6309	684271	264
B	15	15	5	F_{cost}	85513	39602	2639730	4697	665978	280
B	15	15	5	F_{cost}	86045	40407	2641067	4655	665843	281
B	15	15	5	F_{cost}	85507	39517	2648204	4774	668776	282
B	15	15	5	F_{cost}	85178	39842	2639635	4714	666137	280
B	15	15	5	F_{cost}	86525	40105	2641245	4659	665927	282
B	15	15	10	F_{cost}	95929	31553	2642097	4241	661672	316
B	15	15	10	F_{cost}	96373	31497	2642485	4245	661805	315
B	15	15	10	F_{cost}	95507	31667	2641789	4268	661888	314
B	15	15	10	F_{cost}	96130	31335	2642080	4234	661594	315
B	15	15	10	F_{cost}	95830	31776	2642221	4263	661936	314
B	15	15	20	F_{cost}	106565	23629	2644809	3895	658619	374
B	15	15	20	F_{cost}	107306	22720	2644642	3901	658644	375
B	15	15	20	F_{cost}	108684	23407	2646706	3881	658912	375
B	15	15	20	F_{cost}	107025	23987	2645627	3875	658597	375
B	15	15	20	F_{cost}	106959	23063	2644637	3919	658834	373
B	15	30	0	F_{cost}	85828	126992	2727435	5241	692238	321
B	15	30	0	F_{cost}	83070	126018	2723703	5283	691815	319
B	15	30	0	F_{cost}	84106	126798	2725519	5281	692217	319
B	15	30	0	F_{cost}	85182	127522	2727319	5238	692179	319

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	30	0	F_{cost}	85060	127003	2726678	5236	692008	320
B	15	30	5	F_{cost}	100528	107775	2722918	4153	679593	362
B	15	30	5	F_{cost}	101176	107426	2723217	4103	679130	361
B	15	30	5	F_{cost}	100755	107252	2722621	4131	679289	357
B	15	30	5	F_{cost}	102345	108822	2725781	4065	679324	360
B	15	30	5	F_{cost}	101701	106373	2722688	4126	679252	361
B	15	30	10	F_{cost}	106529	86157	2707301	3883	673072	408
B	15	30	10	F_{cost}	106336	85448	2706399	3882	672851	406
B	15	30	10	F_{cost}	106080	84994	2705689	3901	672888	404
B	15	30	10	F_{cost}	107335	87191	2709141	3863	673289	405
B	15	30	10	F_{cost}	106484	86935	2708034	3882	673233	408
B	15	30	20	F_{cost}	114185	62585	2691385	3659	666972	474
B	15	30	20	F_{cost}	113094	60406	2688116	3687	666508	474
B	15	30	20	F_{cost}	114561	63366	2692542	3620	666827	476
B	15	30	20	F_{cost}	112916	63405	2690936	3656	666836	472
B	15	30	20	F_{cost}	113260	62382	2690257	3643	666539	477
B	15	45	0	F_{cost}	89051	167347	2771013	5007	699914	376
B	15	45	0	F_{cost}	88279	165201	2768095	5037	699552	376
B	15	45	0	F_{cost}	89422	166778	2770815	4959	699356	376
B	15	45	0	F_{cost}	89656	166076	2770347	5000	699684	376
B	15	45	0	F_{cost}	89026	168268	2780474	5051	702590	380
B	15	45	5	F_{cost}	89922	123841	2728378	4607	685704	422
B	15	45	5	F_{cost}	89766	123649	2728030	4644	686017	423
B	15	45	5	F_{cost}	88837	126349	2729800	4622	686195	424
B	15	45	5	F_{cost}	91330	126018	2731962	4587	686327	421
B	15	45	5	F_{cost}	91040	125723	2731377	4589	686212	424
B	15	45	10	F_{cost}	95063	100957	2710635	4369	679028	471
B	15	45	10	F_{cost}	93834	101050	2709498	4425	679360	471
B	15	45	10	F_{cost}	93350	100990	2708956	4425	679233	471
B	15	45	10	F_{cost}	94707	100872	2710193	4394	679191	469
B	15	45	10	F_{cost}	94485	101070	2710169	4397	679218	472
B	15	45	20	F_{cost}	100168	75819	2690602	4142	671935	546
B	15	45	20	F_{cost}	115005	88060	2717681	3569	672149	542
B	15	45	20	F_{cost}	99890	75869	2690374	4152	671989	541
B	15	45	20	F_{cost}	116390	88695	2719700	3510	671992	544

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	45	20	F _{cost}	100353	74276	2689244	4151	671714	544
B	30	15	0	F _{cost}	151400	66978	2732993	5123	692278	355
B	30	15	0	F _{cost}	152120	68206	2734941	5113	692626	355
B	30	15	0	F _{cost}	152450	68851	2735916	5088	692587	356
B	30	15	0	F _{cost}	152189	66747	2733551	5133	692515	355
B	30	15	0	F _{cost}	151046	68180	2733841	5105	692284	356
B	30	15	5	F _{cost}	169308	53319	2737243	3955	680826	431
B	30	15	5	F _{cost}	169918	54227	2738761	3932	680935	430
B	30	15	5	F _{cost}	171103	52827	2738545	3959	681172	430
B	30	15	5	F _{cost}	147387	45239	2707241	4628	680996	431
B	30	15	5	F _{cost}	170034	53937	2738586	3935	680926	432
B	30	15	10	F _{cost}	147608	31826	2694049	4464	676170	526
B	30	15	10	F _{cost}	171253	39790	2725659	3736	675790	522
B	30	15	10	F _{cost}	147929	32150	2694694	4462	676299	524
B	30	15	10	F _{cost}	147628	33405	2695648	4435	676234	524
B	30	15	10	F _{cost}	172361	39175	2726151	3736	675905	527
B	30	15	20	F _{cost}	157453	21291	2693359	4123	672376	715
B	30	15	20	F _{cost}	155343	22488	2692445	4134	672280	711
B	30	15	20	F _{cost}	153693	21831	2690139	4188	672317	709
B	30	15	20	F _{cost}	155080	22161	2691856	4161	672430	711
B	30	15	20	F _{cost}	154498	21054	2690167	4160	672026	712
B	30	30	0	F _{cost}	152947	120896	2788458	4764	701395	542
B	30	30	0	F _{cost}	152365	120548	2787528	4759	701125	537
B	30	30	0	F _{cost}	152020	119345	2785980	4805	701254	539
B	30	30	0	F _{cost}	152659	120098	2787372	4790	701419	540
B	30	30	0	F _{cost}	153294	119501	2787410	4765	701161	541
B	30	30	5	F _{cost}	147222	88575	2750411	4454	689215	626
B	30	30	5	F _{cost}	150091	87399	2752106	4399	689024	624
B	30	30	5	F _{cost}	147939	91162	2753716	4376	689155	623
B	30	30	5	F _{cost}	148082	88011	2750708	4447	689210	624
B	30	30	5	F _{cost}	149514	88875	2753004	4400	689244	625
B	30	30	10	F _{cost}	151652	68385	2734653	4203	682864	722
B	30	30	10	F _{cost}	150508	70912	2736034	4196	683112	724
B	30	30	10	F _{cost}	151357	68300	2734271	4233	683094	719
B	30	30	10	F _{cost}	151672	68417	2734704	4208	682929	722

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	30	10	F_{cost}	153631	69943	2746754	4216	685826	727
B	30	30	20	F_{cost}	158139	46134	2718888	3986	676873	923
B	30	30	20	F_{cost}	155410	47237	2717262	4026	676920	920
B	30	30	20	F_{cost}	157493	48421	2729094	4030	679724	917
B	30	30	20	F_{cost}	159007	46064	2719687	3962	676804	922
B	30	30	20	F_{cost}	159380	46680	2720675	3960	677013	919
B	30	45	0	F_{cost}	168401	167422	2850438	4457	712586	744
B	30	45	0	F_{cost}	164096	168334	2847045	4545	712732	739
B	30	45	0	F_{cost}	162581	169140	2846336	4546	712577	742
B	30	45	0	F_{cost}	165476	169009	2849100	4528	713031	743
B	30	45	0	F_{cost}	163618	169812	2848045	4537	712880	746
B	30	45	5	F_{cost}	157788	124237	2796640	4296	698318	829
B	30	45	5	F_{cost}	159199	123647	2797460	4224	697743	828
B	30	45	5	F_{cost}	156277	123639	2794531	4301	697879	826
B	30	45	5	F_{cost}	160498	124652	2799765	4199	698014	834
B	30	45	5	F_{cost}	158812	124508	2797935	4219	697800	831
B	30	45	10	F_{cost}	161755	98108	2774478	4036	690377	932
B	30	45	10	F_{cost}	157437	96915	2768967	4116	689944	932
B	30	45	10	F_{cost}	159598	98772	2772985	4091	690615	929
B	30	45	10	F_{cost}	158444	99348	2772408	4076	690320	930
B	30	45	10	F_{cost}	158825	96354	2769795	4088	689838	934
B	30	45	20	F_{cost}	166060	71248	2751923	3854	683175	1142
B	30	45	20	F_{cost}	166559	71045	2752219	3820	682882	1140
B	30	45	20	F_{cost}	165601	71580	2751796	3827	682858	1130
B	30	45	20	F_{cost}	165351	72223	2752189	3836	683046	1135
B	30	45	20	F_{cost}	165926	72397	2752937	3818	683028	1135

Tabla A.2: Resultados detallados obtenidos con ACS-HCSP para las distintas configuraciones.

A.3. IACS-HCSP

A continuación se exponen los resultados de todos los experimentos realizados para el método IACS-HCSP cuyos detalles de implementación fueron detallados en la sección 4.3.3. Tal y como se expuso en el capítulo 5 para la técnica IACS se han llevado a cabo un total 96 experimentos, repetidos cada uno de ellos 5 veces dada la naturaleza estocástica de la técnica utilizada. Dicha experimentación ha requerido un total de 100 horas de cómputo en la plataforma Google Cloud en un servidor con 64 núcleos y 60 Gb de Ram. La parametrización utilizada para dichos experimentos es la misma que la expuesta en la sección anterior A.2.

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	15	0	<i>F_{clust}</i>	98337	75314	2688266	3983	3983	306
A	15	15	0	<i>F_{clust}</i>	98263	74299	2687177	3981	3981	314
A	15	15	0	<i>F_{clust}</i>	98634	75128	2688377	3986	3986	313
A	15	15	0	<i>F_{clust}</i>	99183	75978	2689776	3984	3984	309
A	15	15	0	<i>F_{clust}</i>	98444	75087	2688145	3977	3977	315
A	15	15	5	<i>F_{clust}</i>	113391	55267	2683274	3120	3120	375
A	15	15	5	<i>F_{clust}</i>	114242	55208	2684064	3123	3123	374
A	15	15	5	<i>F_{clust}</i>	114647	56063	2685325	3115	3115	375
A	15	15	5	<i>F_{clust}</i>	112383	56138	2683137	3109	3109	375
A	15	15	5	<i>F_{clust}</i>	114127	55737	2684479	3098	3098	375
A	15	15	10	<i>F_{clust}</i>	125454	43260	2683329	2800	2800	453
A	15	15	10	<i>F_{clust}</i>	126695	43798	2685108	2788	2788	452
A	15	15	10	<i>F_{clust}</i>	126267	42783	2683665	2790	2790	457
A	15	15	10	<i>F_{clust}</i>	126129	42691	2683435	2785	2785	675
A	15	15	10	<i>F_{clust}</i>	126435	43705	2684754	2781	2781	457
A	15	15	20	<i>F_{clust}</i>	141319	30730	2686663	2504	2504	586
A	15	15	20	<i>F_{clust}</i>	141427	30949	2686991	2502	2502	588
A	15	15	20	<i>F_{clust}</i>	141510	29957	2686082	2507	2507	594
A	15	15	20	<i>F_{clust}</i>	142136	31167	2687918	2502	2502	592
A	15	15	20	<i>F_{clust}</i>	141950	31342	2687907	2485	2485	593
A	15	30	0	<i>F_{clust}</i>	132940	206810	2854365	2969	2969	459
A	15	30	0	<i>F_{clust}</i>	132466	209328	2856409	2961	2961	457
A	15	30	0	<i>F_{clust}</i>	132607	209530	2856752	2968	2968	467
A	15	30	0	<i>F_{clust}</i>	132449	205900	2852964	2972	2972	462
A	15	30	0	<i>F_{clust}</i>	131296	208887	2854798	2969	2969	461
A	15	30	5	<i>F_{clust}</i>	136508	158623	2809746	2589	2589	551
A	15	30	5	<i>F_{clust}</i>	134281	156687	2805583	2592	2592	554
A	15	30	5	<i>F_{clust}</i>	135238	158421	2808274	2606	2606	560
A	15	30	5	<i>F_{clust}</i>	136113	158360	2809088	2585	2585	560
A	15	30	5	<i>F_{clust}</i>	135116	157044	2806776	2593	2593	553
A	15	30	10	<i>F_{clust}</i>	140796	126004	2781414	2435	2435	644
A	15	30	10	<i>F_{clust}</i>	142745	127954	2785314	2425	2425	644
A	15	30	10	<i>F_{clust}</i>	142163	127566	2784344	2435	2435	654
A	15	30	10	<i>F_{clust}</i>	141974	127350	2783939	2418	2418	649

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	30	10	<i>F_{clust}</i>	141277	125819	2781711	2429	2429	643
A	15	30	20	<i>F_{clust}</i>	151093	90118	2755826	2267	2267	785
A	15	30	20	<i>F_{clust}</i>	153008	93646	2761268	2255	2255	783
A	15	30	20	<i>F_{clust}</i>	151343	89925	2755882	2266	2266	794
A	15	30	20	<i>F_{clust}</i>	150752	87493	2753580	2266	2266	781
A	15	30	20	<i>F_{clust}</i>	150771	90036	2755422	2265	2265	779
A	15	45	0	<i>F_{clust}</i>	138130	275618	2928363	2763	2763	579
A	15	45	0	<i>F_{clust}</i>	139228	281431	2935274	2746	2746	585
A	15	45	0	<i>F_{clust}</i>	139824	281851	2936290	2743	2743	584
A	15	45	0	<i>F_{clust}</i>	138220	277968	2930803	2757	2757	583
A	15	45	0	<i>F_{clust}</i>	139863	276667	2931145	2761	2761	584
A	15	45	5	<i>F_{clust}</i>	139701	214570	2868885	2475	2475	677
A	15	45	5	<i>F_{clust}</i>	139014	216264	2869893	2474	2474	678
A	15	45	5	<i>F_{clust}</i>	139411	215673	2869699	2460	2460	680
A	15	45	5	<i>F_{clust}</i>	138816	215826	2869257	2455	2455	686
A	15	45	5	<i>F_{clust}</i>	139980	214589	2869184	2464	2464	680
A	15	45	10	<i>F_{clust}</i>	144544	171492	2830651	2337	2337	769
A	15	45	10	<i>F_{clust}</i>	145351	173806	2833771	2337	2337	772
A	15	45	10	<i>F_{clust}</i>	145347	171082	2831044	2325	2325	773
A	15	45	10	<i>F_{clust}</i>	144544	174639	2833798	2331	2331	776
A	15	45	10	<i>F_{clust}</i>	143652	171579	2829846	2326	2326	769
A	15	45	20	<i>F_{clust}</i>	152986	129012	2796613	2184	2184	916
A	15	45	20	<i>F_{clust}</i>	152807	127398	2794820	2183	2183	929
A	15	45	20	<i>F_{clust}</i>	152416	126535	2793566	2184	2184	921
A	15	45	20	<i>F_{clust}</i>	151812	125602	2792030	2179	2179	921
A	15	45	20	<i>F_{clust}</i>	153622	128090	2796327	2175	2175	919
A	30	15	0	<i>F_{clust}</i>	244947	110678	2870240	2849	2849	537
A	30	15	0	<i>F_{clust}</i>	246665	109933	2871213	2855	2855	547
A	30	15	0	<i>F_{clust}</i>	246675	109109	2870399	2850	2850	543
A	30	15	0	<i>F_{clust}</i>	247044	109671	2871330	2846	2846	540
A	30	15	0	<i>F_{clust}</i>	244688	110293	2869596	2846	2846	540
A	30	15	5	<i>F_{clust}</i>	237183	74984	2826782	2490	2490	688
A	30	15	5	<i>F_{clust}</i>	236474	77033	2828122	2488	2488	703
A	30	15	5	<i>F_{clust}</i>	237483	74212	2826310	2500	2500	699
A	30	15	5	<i>F_{clust}</i>	236406	73643	2824664	2510	2510	691

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	15	5	<i>F_{clust}</i>	237239	75677	2827531	2493	2493	689
A	30	15	10	<i>F_{clust}</i>	239528	54191	2808334	2339	2339	863
A	30	15	10	<i>F_{clust}</i>	238870	54107	2807592	2349	2349	865
A	30	15	10	<i>F_{clust}</i>	241055	53229	2808900	2341	2341	885
A	30	15	10	<i>F_{clust}</i>	240868	54514	2809997	2331	2331	874
A	30	15	10	<i>F_{clust}</i>	239185	54299	2808099	2337	2337	864
A	30	15	20	<i>F_{clust}</i>	251833	35662	2802110	2163	2163	1227
A	30	15	20	<i>F_{clust}</i>	249249	36240	2800104	2175	2175	1217
A	30	15	20	<i>F_{clust}</i>	251279	36213	2802107	2170	2170	1231
A	30	15	20	<i>F_{clust}</i>	251563	35796	2801974	2161	2161	1228
A	30	15	20	<i>F_{clust}</i>	251446	36390	2802452	2158	2158	1229
A	30	30	0	<i>F_{clust}</i>	248886	202079	2965580	2559	2559	892
A	30	30	0	<i>F_{clust}</i>	246520	202860	2963995	2548	2548	906
A	30	30	0	<i>F_{clust}</i>	246284	201780	2962679	2545	2545	907
A	30	30	0	<i>F_{clust}</i>	245695	203943	2964252	2547	2547	899
A	30	30	0	<i>F_{clust}</i>	250467	205264	2970346	2546	2546	898
A	30	30	5	<i>F_{clust}</i>	239232	150820	2904667	2312	2312	1057
A	30	30	5	<i>F_{clust}</i>	238223	153301	2906139	2305	2305	1058
A	30	30	5	<i>F_{clust}</i>	235101	151765	2901481	2308	2308	1085
A	30	30	5	<i>F_{clust}</i>	238825	151247	2904687	2311	2311	1069
A	30	30	5	<i>F_{clust}</i>	237396	148607	2900618	2313	2313	1068
A	30	30	10	<i>F_{clust}</i>	239333	119051	2872999	2195	2195	1253
A	30	30	10	<i>F_{clust}</i>	239637	116503	2870755	2203	2203	1274
A	30	30	10	<i>F_{clust}</i>	239418	116778	2870812	2189	2189	1259
A	30	30	10	<i>F_{clust}</i>	236811	119149	2870575	2190	2190	1264
A	30	30	10	<i>F_{clust}</i>	238543	116457	2869616	2193	2193	1270
A	30	30	20	<i>F_{clust}</i>	246525	81091	2842231	2074	2074	1638
A	30	30	20	<i>F_{clust}</i>	245937	79247	2839799	2060	2060	1685
A	30	30	20	<i>F_{clust}</i>	248260	82275	2845150	2071	2071	1651
A	30	30	20	<i>F_{clust}</i>	244473	79688	2838776	2065	2065	1649
A	30	30	20	<i>F_{clust}</i>	247618	81404	2843637	2057	2057	1649
A	30	45	0	<i>F_{clust}</i>	255892	278612	3049119	2390	2390	1269
A	30	45	0	<i>F_{clust}</i>	258385	275370	3048370	2387	2387	1296
A	30	45	0	<i>F_{clust}</i>	257344	278014	3049973	2396	2396	1302
A	30	45	0	<i>F_{clust}</i>	258005	278016	3050636	2386	2386	1299

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	45	0	F_{clust}	256703	274810	3046128	2383	2383	1277
A	30	45	5	F_{clust}	243060	207143	2964818	2239	2239	1462
A	30	45	5	F_{clust}	241828	208859	2965302	2241	2241	1485
A	30	45	5	F_{clust}	240283	206553	2961451	2239	2239	1478
A	30	45	5	F_{clust}	242013	206244	2962872	2239	2239	1477
A	30	45	5	F_{clust}	241659	207758	2964032	2240	2240	1449
A	30	45	10	F_{clust}	244383	162394	2921392	2148	2148	1660
A	30	45	10	F_{clust}	241565	163126	2919305	2140	2140	1684
A	30	45	10	F_{clust}	242215	166373	2923203	2148	2148	1673
A	30	45	10	F_{clust}	242231	159713	2916559	2148	2148	1659
A	30	45	10	F_{clust}	243647	162283	2920545	2145	2145	1675
A	30	45	20	F_{clust}	246144	113919	2874678	2031	2031	2083
A	30	45	20	F_{clust}	249246	115719	2879579	2026	2026	2116
A	30	45	20	F_{clust}	251214	115217	2881047	2033	2033	2123
A	30	45	20	F_{clust}	246765	114179	2875559	2028	2028	2100
A	30	45	20	F_{clust}	248463	111358	2874436	2031	2031	2110
A	15	15	0	F_{cost}	77803	57885	2650303	4398	665260	324
A	15	15	0	F_{cost}	76433	56919	2647966	4456	665332	324
A	15	15	0	F_{cost}	75577	56422	2646614	4472	665187	329
A	15	15	0	F_{cost}	73918	55832	2644365	4507	665035	325
A	15	15	0	F_{cost}	76022	57097	2647734	4454	665257	332
A	15	15	5	F_{cost}	87471	37398	2648048	3608	656317	375
A	15	15	5	F_{cost}	85956	36429	2637000	3628	653952	372
A	15	15	5	F_{cost}	85117	35655	2635388	3659	653906	376
A	15	15	5	F_{cost}	86988	37477	2639080	3583	653958	395
A	15	15	5	F_{cost}	88225	38493	2641334	3559	654228	379
A	15	15	10	F_{cost}	96032	27772	2646984	3299	652777	452
A	15	15	10	F_{cost}	95925	27985	2638525	3295	650760	451
A	15	15	10	F_{cost}	94879	27023	2636517	3330	650665	455
A	15	15	10	F_{cost}	93805	26527	2634947	3358	650597	682
A	15	15	10	F_{cost}	95021	27072	2636708	3326	650667	456
A	15	15	20	F_{cost}	104798	17887	2637300	3044	647800	576
A	15	15	20	F_{cost}	106167	18778	2639560	2973	647571	573
A	15	15	20	F_{cost}	103792	17750	2636157	3059	647693	588
A	15	15	20	F_{cost}	105448	18604	2638667	3023	647896	587

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	15	20	F _{cost}	106289	18130	2639034	3003	647768	581
A	15	30	0	F _{cost}	83605	103505	2701724	4063	673689	461
A	15	30	0	F _{cost}	83775	103300	2701690	4078	673841	458
A	15	30	0	F _{cost}	83972	101046	2699633	4112	673723	469
A	15	30	0	F _{cost}	84500	102311	2701426	4065	673641	467
A	15	30	0	F _{cost}	83951	102232	2700798	4084	673697	466
A	15	30	5	F _{cost}	87460	64132	2666207	3565	660096	548
A	15	30	5	F _{cost}	87436	65832	2667883	3534	660157	546
A	15	30	5	F _{cost}	85946	64644	2665205	3574	659958	556
A	15	30	5	F _{cost}	88066	66483	2669164	3505	660147	554
A	15	30	5	F _{cost}	87112	66685	2668412	3516	660088	546
A	15	30	10	F _{cost}	95353	50434	2660403	3268	655577	631
A	15	30	10	F _{cost}	95482	51094	2661191	3240	655463	625
A	15	30	10	F _{cost}	95165	49368	2659148	3307	655700	637
A	15	30	10	F _{cost}	96240	50085	2660940	3264	655660	635
A	15	30	10	F _{cost}	95910	50426	2660951	3269	655716	635
A	15	30	20	F _{cost}	106537	35857	2657009	2927	651153	772
A	15	30	20	F _{cost}	104832	34979	2654426	2974	651051	774
A	15	30	20	F _{cost}	104896	36913	2656424	2966	651432	776
A	15	30	20	F _{cost}	106107	35467	2656189	2928	650972	773
A	15	30	20	F _{cost}	105609	36123	2656347	2961	651360	780
A	15	45	0	F _{cost}	85924	116802	2717341	4004	676704	577
A	15	45	0	F _{cost}	85128	114402	2714145	4055	676502	574
A	15	45	0	F _{cost}	83912	112485	2711012	4080	676037	578
A	15	45	0	F _{cost}	84782	114110	2713507	4059	676396	583
A	15	45	0	F _{cost}	84388	114180	2713183	4052	676245	582
A	15	45	5	F _{cost}	87978	73995	2676588	3501	661836	671
A	15	45	5	F _{cost}	90502	78031	2683149	3414	662440	672
A	15	45	5	F _{cost}	91053	78405	2684073	3396	662464	670
A	15	45	5	F _{cost}	88881	74839	2678336	3502	662255	673
A	15	45	5	F _{cost}	89993	75317	2679925	3463	662210	669
A	15	45	10	F _{cost}	95750	58230	2677160	3273	659541	760
A	15	45	10	F _{cost}	95300	57805	2667720	3249	657082	770
A	15	45	10	F _{cost}	95395	58318	2668328	3244	657171	761
A	15	45	10	F _{cost}	97401	59364	2671380	3198	657393	768

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	15	45	10	F _{cost}	96411	55946	2666972	3249	656908	758
A	15	45	20	F _{cost}	106404	41792	2662811	2943	652677	904
A	15	45	20	F _{cost}	104927	41456	2660998	2992	652776	918
A	15	45	20	F _{cost}	106340	42356	2663311	2920	652549	912
A	15	45	20	F _{cost}	107403	42956	2664974	2914	652873	911
A	15	45	20	F _{cost}	106418	42958	2663990	2915	652654	904
A	30	15	0	F _{cost}	127772	63512	2705899	4065	674684	532
A	30	15	0	F _{cost}	132384	66540	2713539	3969	675444	538
A	30	15	0	F _{cost}	132284	64282	2711901	3983	675211	543
A	30	15	0	F _{cost}	131721	65669	2712005	3984	675246	532
A	30	15	0	F _{cost}	131981	65559	2712155	3971	675143	537
A	30	15	5	F _{cost}	128428	39419	2682462	3427	662419	679
A	30	15	5	F _{cost}	128902	39358	2682875	3421	662451	686
A	30	15	5	F _{cost}	125639	38859	2679113	3464	662031	690
A	30	15	5	F _{cost}	129962	40493	2685069	3368	662398	681
A	30	15	5	F _{cost}	127443	38958	2681015	3437	662188	681
A	30	15	10	F _{cost}	135806	27361	2677782	3140	658269	855
A	30	15	10	F _{cost}	134039	27430	2676084	3176	658256	849
A	30	15	10	F _{cost}	132906	28457	2675978	3155	658008	862
A	30	15	10	F _{cost}	135223	27497	2677335	3148	658250	852
A	30	15	10	F _{cost}	135029	28190	2677834	3157	658462	852
A	30	15	20	F _{cost}	144624	18535	2677774	2843	655103	1200
A	30	15	20	F _{cost}	145389	18202	2678206	2835	655118	1220
A	30	15	20	F _{cost}	146366	19020	2680001	2826	655441	1210
A	30	15	20	F _{cost}	145026	18630	2678272	2848	655272	1210
A	30	15	20	F _{cost}	146260	18511	2679386	2818	655213	1205
A	30	30	0	F _{cost}	122902	90982	2728499	3959	678828	880
A	30	30	0	F _{cost}	123725	93359	2731699	3923	679191	885
A	30	30	0	F _{cost}	121803	90724	2727142	3996	678906	884
A	30	30	0	F _{cost}	117267	86186	2718068	4109	677993	877
A	30	30	0	F _{cost}	123698	93022	2731335	3948	679373	873
A	30	30	5	F _{cost}	125422	60942	2700980	3339	665802	1050
A	30	30	5	F _{cost}	120716	60088	2695419	3438	665559	1061
A	30	30	5	F _{cost}	119289	58495	2692399	3470	665195	1056
A	30	30	5	F _{cost}	121269	60244	2696127	3430	665639	1051

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
A	30	30	5	F_{cost}	120214	57842	2692671	3479	665355	1054
A	30	30	10	F_{cost}	128483	45052	2688151	3187	661189	1229
A	30	30	10	F_{cost}	131081	46448	2692144	3104	661236	1256
A	30	30	10	F_{cost}	125689	43785	2684089	3240	660806	1230
A	30	30	10	F_{cost}	131606	45901	2692122	3120	661402	1242
A	30	30	10	F_{cost}	127256	43427	2685299	3226	660939	1251
A	30	30	20	F_{cost}	143651	32988	2691254	2765	657417	1608
A	30	30	20	F_{cost}	143724	33827	2692166	2754	657513	1663
A	30	30	20	F_{cost}	141837	30595	2687047	2838	657213	1666
A	30	30	20	F_{cost}	138857	32838	2686311	2846	657127	1646
A	30	30	20	F_{cost}	146077	32740	2693432	2752	657787	1631
A	30	45	0	F_{cost}	118240	97637	2730492	4094	680732	1255
A	30	45	0	F_{cost}	117125	96857	2728596	4097	680321	1250
A	30	45	0	F_{cost}	118540	96933	2730088	4070	680382	1268
A	30	45	0	F_{cost}	123427	102008	2740050	3930	681215	1274
A	30	45	0	F_{cost}	117349	99140	2731104	4054	680448	1252
A	30	45	5	F_{cost}	120574	64716	2699906	3493	667192	1447
A	30	45	5	F_{cost}	119942	66042	2700599	3477	667183	1460
A	30	45	5	F_{cost}	118412	65238	2698266	3513	667022	1454
A	30	45	5	F_{cost}	119012	66316	2699944	3490	667169	1454
A	30	45	5	F_{cost}	119308	66075	2699998	3487	667149	1450
A	30	45	10	F_{cost}	126922	49272	2690809	3256	662544	1644
A	30	45	10	F_{cost}	127372	51959	2693946	3181	662477	1663
A	30	45	10	F_{cost}	126040	49855	2690510	3222	662112	1659
A	30	45	10	F_{cost}	127873	51062	2693550	3193	662513	1635
A	30	45	10	F_{cost}	128140	52040	2694795	3168	662537	1668
A	30	45	20	F_{cost}	137488	36394	2697062	2941	660647	2056
A	30	45	20	F_{cost}	138397	37243	2690255	2872	658324	2089
A	30	45	20	F_{cost}	141578	38701	2694894	2791	658543	2069
A	30	45	20	F_{cost}	135751	34910	2685277	2982	658334	2062
A	30	45	20	F_{cost}	136818	36877	2688310	2934	658531	1823
B	15	15	0	F_{clust}	98254	75126	2687995	3982	3982	315
B	15	15	0	F_{clust}	98910	74906	2688431	3956	3956	310
B	15	15	0	F_{clust}	98302	75765	2688682	3971	3971	315
B	15	15	0	F_{clust}	98128	75264	2688007	3957	3957	328

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	15	0	<i>F_{clust}</i>	97903	76499	2689017	3960	3960	324
B	15	15	5	<i>F_{clust}</i>	113548	54443	2682606	3116	3116	372
B	15	15	5	<i>F_{clust}</i>	113320	55405	2683340	3097	3097	377
B	15	15	5	<i>F_{clust}</i>	114705	54538	2683858	3094	3094	379
B	15	15	5	<i>F_{clust}</i>	114526	55854	2684994	3096	3096	383
B	15	15	5	<i>F_{clust}</i>	114719	55360	2684694	3104	3104	383
B	15	15	10	<i>F_{clust}</i>	126490	43154	2684259	2779	2779	451
B	15	15	10	<i>F_{clust}</i>	127897	42599	2685110	2773	2773	451
B	15	15	10	<i>F_{clust}</i>	124952	44012	2683579	2769	2769	458
B	15	15	10	<i>F_{clust}</i>	127401	42864	2684881	2761	2761	642
B	15	15	10	<i>F_{clust}</i>	127123	42593	2684331	2771	2771	461
B	15	15	20	<i>F_{clust}</i>	140207	30684	2685506	2486	2486	579
B	15	15	20	<i>F_{clust}</i>	141457	31181	2687254	2501	2501	582
B	15	15	20	<i>F_{clust}</i>	140488	30896	2685999	2490	2490	593
B	15	15	20	<i>F_{clust}</i>	142071	31028	2687714	2485	2485	940
B	15	15	20	<i>F_{clust}</i>	141020	30402	2686037	2493	2493	589
B	15	30	0	<i>F_{clust}</i>	132930	210588	2858132	2939	2939	459
B	15	30	0	<i>F_{clust}</i>	133355	212813	2860783	2926	2926	458
B	15	30	0	<i>F_{clust}</i>	133114	207442	2855171	2949	2949	470
B	15	30	0	<i>F_{clust}</i>	133372	209588	2857575	2941	2941	469
B	15	30	0	<i>F_{clust}</i>	132616	207578	2854808	2938	2938	466
B	15	30	5	<i>F_{clust}</i>	135765	158497	2808877	2597	2597	547
B	15	30	5	<i>F_{clust}</i>	137173	160568	2812356	2571	2571	547
B	15	30	5	<i>F_{clust}</i>	136914	157521	2809050	2577	2577	559
B	15	30	5	<i>F_{clust}</i>	136212	156391	2807218	2576	2576	557
B	15	30	5	<i>F_{clust}</i>	137338	158410	2810363	2578	2578	556
B	15	30	10	<i>F_{clust}</i>	140426	125204	2780245	2412	2412	639
B	15	30	10	<i>F_{clust}</i>	140790	123275	2778680	2404	2404	641
B	15	30	10	<i>F_{clust}</i>	142157	128938	2785710	2408	2408	651
B	15	30	10	<i>F_{clust}</i>	141362	125717	2781694	2408	2408	654
B	15	30	10	<i>F_{clust}</i>	141862	128600	2785077	2411	2411	645
B	15	30	20	<i>F_{clust}</i>	151206	90745	2756566	2228	2228	777
B	15	30	20	<i>F_{clust}</i>	151418	90927	2756960	2253	2253	776
B	15	30	20	<i>F_{clust}</i>	151868	91945	2766994	2262	2262	789
B	15	30	20	<i>F_{clust}</i>	151143	92690	2758448	2250	2250	789

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	30	20	<i>F_{clust}</i>	150390	90110	2755114	2251	2251	784
B	15	45	0	<i>F_{clust}</i>	140101	282225	2936941	2729	2729	573
B	15	45	0	<i>F_{clust}</i>	139445	280260	2934320	2723	2723	581
B	15	45	0	<i>F_{clust}</i>	138683	280536	2933834	2730	2730	580
B	15	45	0	<i>F_{clust}</i>	139355	281477	2935446	2733	2733	589
B	15	45	0	<i>F_{clust}</i>	140178	280532	2935324	2721	2721	593
B	15	45	5	<i>F_{clust}</i>	140174	215081	2869870	2454	2454	669
B	15	45	5	<i>F_{clust}</i>	139648	213426	2867689	2443	2443	673
B	15	45	5	<i>F_{clust}</i>	139341	214176	2868132	2448	2448	681
B	15	45	5	<i>F_{clust}</i>	140163	210876	2865654	2459	2459	685
B	15	45	5	<i>F_{clust}</i>	139428	216060	2870103	2439	2439	678
B	15	45	10	<i>F_{clust}</i>	144141	175032	2833788	2307	2307	764
B	15	45	10	<i>F_{clust}</i>	144006	174325	2832947	2319	2319	764
B	15	45	10	<i>F_{clust}</i>	145475	177332	2845987	2332	2332	781
B	15	45	10	<i>F_{clust}</i>	144876	171377	2830868	2319	2319	776
B	15	45	10	<i>F_{clust}</i>	144742	173554	2832910	2312	2312	772
B	15	45	20	<i>F_{clust}</i>	151580	129525	2795720	2160	2160	907
B	15	45	20	<i>F_{clust}</i>	152553	130025	2797193	2170	2170	917
B	15	45	20	<i>F_{clust}</i>	152804	125151	2792570	2177	2177	921
B	15	45	20	<i>F_{clust}</i>	152895	124817	2792327	2167	2167	923
B	15	45	20	<i>F_{clust}</i>	153827	125985	2794427	2173	2173	918
B	30	15	0	<i>F_{clust}</i>	247108	108902	2870625	2822	2822	533
B	30	15	0	<i>F_{clust}</i>	246979	108319	2869913	2839	2839	543
B	30	15	0	<i>F_{clust}</i>	248398	108532	2880110	2864	2864	547
B	30	15	0	<i>F_{clust}</i>	247394	109972	2871981	2831	2831	544
B	30	15	0	<i>F_{clust}</i>	246203	109371	2870189	2832	2832	542
B	30	15	5	<i>F_{clust}</i>	236410	75040	2826065	2490	2490	680
B	30	15	5	<i>F_{clust}</i>	237158	77003	2828775	2489	2489	699
B	30	15	5	<i>F_{clust}</i>	238467	74952	2828034	2478	2478	699
B	30	15	5	<i>F_{clust}</i>	236452	75677	2826744	2483	2483	695
B	30	15	5	<i>F_{clust}</i>	238426	73871	2826912	2490	2490	691
B	30	15	10	<i>F_{clust}</i>	238828	53311	2806754	2312	2312	853
B	30	15	10	<i>F_{clust}</i>	241978	54161	2810754	2327	2327	860
B	30	15	10	<i>F_{clust}</i>	239677	55125	2809417	2327	2327	878
B	30	15	10	<i>F_{clust}</i>	242498	53863	2810976	2333	2333	872

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	15	10	<i>F_{clust}</i>	240654	54636	2809905	2315	2315	866
B	30	15	20	<i>F_{clust}</i>	251874	34786	2801275	2167	2167	1216
B	30	15	20	<i>F_{clust}</i>	253373	36592	2804580	2154	2154	1209
B	30	15	20	<i>F_{clust}</i>	250796	36416	2801827	2152	2152	1234
B	30	15	20	<i>F_{clust}</i>	251358	36001	2801974	2165	2165	1230
B	30	15	20	<i>F_{clust}</i>	250783	35254	2800652	2148	2148	1226
B	30	30	0	<i>F_{clust}</i>	247384	201701	2963700	2530	2530	884
B	30	30	0	<i>F_{clust}</i>	248686	199197	2962498	2534	2534	895
B	30	30	0	<i>F_{clust}</i>	248673	203309	2966596	2526	2526	899
B	30	30	0	<i>F_{clust}</i>	247961	202589	2965164	2536	2536	901
B	30	30	0	<i>F_{clust}</i>	247000	202579	2964194	2530	2530	892
B	30	30	5	<i>F_{clust}</i>	239401	152798	2906814	2309	2309	1049
B	30	30	5	<i>F_{clust}</i>	237220	149837	2901672	2306	2306	1057
B	30	30	5	<i>F_{clust}</i>	237333	151311	2903259	2299	2299	1080
B	30	30	5	<i>F_{clust}</i>	236977	150387	2901979	2296	2296	1063
B	30	30	5	<i>F_{clust}</i>	236249	149550	2900414	2301	2301	1075
B	30	30	10	<i>F_{clust}</i>	238817	119061	2872492	2179	2179	1249
B	30	30	10	<i>F_{clust}</i>	240577	114945	2870137	2176	2176	1263
B	30	30	10	<i>F_{clust}</i>	243009	118394	2876018	2176	2176	1281
B	30	30	10	<i>F_{clust}</i>	241036	117780	2873431	2183	2183	1253
B	30	30	10	<i>F_{clust}</i>	237685	115598	2867898	2178	2178	1269
B	30	30	20	<i>F_{clust}</i>	250680	79285	2844580	2049	2049	1637
B	30	30	20	<i>F_{clust}</i>	248647	80071	2843334	2043	2043	1681
B	30	30	20	<i>F_{clust}</i>	245966	79821	2840403	2049	2049	1655
B	30	30	20	<i>F_{clust}</i>	248292	81082	2843989	2040	2040	1656
B	30	30	20	<i>F_{clust}</i>	248863	78501	2841979	2045	2045	1669
B	30	45	0	<i>F_{clust}</i>	258034	277076	3049725	2376	2376	1259
B	30	45	0	<i>F_{clust}</i>	259047	282057	3055719	2374	2374	1304
B	30	45	0	<i>F_{clust}</i>	260482	277109	3052206	2362	2362	1300
B	30	45	0	<i>F_{clust}</i>	257156	275684	3047455	2368	2368	1307
B	30	45	0	<i>F_{clust}</i>	259434	280439	3054488	2374	2374	1272
B	30	45	5	<i>F_{clust}</i>	242384	208571	2965571	2218	2218	1453
B	30	45	5	<i>F_{clust}</i>	245645	208528	2968788	2231	2231	1473
B	30	45	5	<i>F_{clust}</i>	240151	209498	2964264	2223	2223	1476
B	30	45	5	<i>F_{clust}</i>	242064	201455	2958134	2225	2225	1484

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	45	5	F_{clust}	244651	205176	2964442	2231	2231	1462
B	30	45	10	F_{clust}	243348	163462	2921425	2127	2127	1644
B	30	45	10	F_{clust}	239337	164377	2918329	2130	2130	1662
B	30	45	10	F_{clust}	243973	164146	2922734	2128	2128	1674
B	30	45	10	F_{clust}	241411	164272	2920298	2134	2134	1659
B	30	45	10	F_{clust}	244656	164008	2923279	2126	2126	1672
B	30	45	20	F_{clust}	248726	116471	2879812	2012	2012	2067
B	30	45	20	F_{clust}	248572	116174	2879361	2016	2016	2103
B	30	45	20	F_{clust}	251194	115769	2881578	2014	2014	2106
B	30	45	20	F_{clust}	250523	113722	2878860	2006	2006	2089
B	30	45	20	F_{clust}	251085	116358	2882058	2017	2017	2106
B	15	15	0	F_{cost}	76468	56917	2648000	4420	664957	324
B	15	15	0	F_{cost}	77734	58335	2650684	4379	665146	325
B	15	15	0	F_{cost}	78360	58372	2651347	4384	665354	334
B	15	15	0	F_{cost}	78195	59167	2651977	4376	665416	339
B	15	15	0	F_{cost}	75842	56465	2646922	4453	665057	345
B	15	15	5	F_{cost}	87265	36546	2638426	3614	654136	377
B	15	15	5	F_{cost}	85740	36034	2636389	3641	653948	375
B	15	15	5	F_{cost}	84605	35846	2635066	3657	653810	377
B	15	15	5	F_{cost}	87812	37432	2639858	3595	654268	384
B	15	15	5	F_{cost}	86216	35982	2636814	3631	653941	385
B	15	15	10	F_{cost}	96145	27236	2637996	3285	650530	451
B	15	15	10	F_{cost}	98631	29133	2642379	3178	650413	448
B	15	15	10	F_{cost}	95216	27226	2637057	3300	650471	452
B	15	15	10	F_{cost}	98423	28466	2641504	3221	650667	706
B	15	15	10	F_{cost}	96505	27009	2638129	3297	650689	460
B	15	15	20	F_{cost}	106666	18672	2639953	2976	647695	574
B	15	15	20	F_{cost}	108168	19184	2641967	2947	647856	572
B	15	15	20	F_{cost}	108466	18736	2641816	2945	647799	583
B	15	15	20	F_{cost}	107524	19011	2641150	2952	647719	591
B	15	15	20	F_{cost}	106760	18551	2639925	2973	647656	588
B	15	30	0	F_{cost}	85527	102886	2703028	4034	673684	460
B	15	30	0	F_{cost}	84230	103809	2703404	4058	674028	457
B	15	30	0	F_{cost}	82835	101045	2698495	4102	673351	470
B	15	30	0	F_{cost}	84130	102312	2701057	4073	673640	474

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	30	0	F _{cost}	83110	100902	2698627	4112	673488	472
B	15	30	5	F _{cost}	88159	66936	2669710	3499	660210	541
B	15	30	5	F _{cost}	88662	67051	2670329	3469	660035	542
B	15	30	5	F _{cost}	87515	67002	2669131	3497	660054	555
B	15	30	5	F _{cost}	87938	67510	2670063	3487	660165	554
B	15	30	5	F _{cost}	88803	66606	2670024	3488	660166	551
B	15	30	10	F _{cost}	94388	49675	2658678	3288	655388	625
B	15	30	10	F _{cost}	97224	51577	2663416	3216	655727	629
B	15	30	10	F _{cost}	96827	50946	2662388	3221	655540	641
B	15	30	10	F _{cost}	96110	51147	2661872	3239	655611	642
B	15	30	10	F _{cost}	96108	50999	2661722	3228	655459	636
B	15	30	20	F _{cost}	106407	35693	2656716	2936	651180	765
B	15	30	20	F _{cost}	105103	36076	2655793	2968	651306	765
B	15	30	20	F _{cost}	105678	35543	2655836	2962	651252	771
B	15	30	20	F _{cost}	105839	36560	2657014	2921	651090	776
B	15	30	20	F _{cost}	107573	36053	2658241	2917	651333	777
B	15	45	0	F _{cost}	82532	111602	2708748	4106	675786	574
B	15	45	0	F _{cost}	86311	116011	2716937	3975	676301	577
B	15	45	0	F _{cost}	86408	115310	2716333	4005	676480	584
B	15	45	0	F _{cost}	83998	112013	2710626	4082	675968	590
B	15	45	0	F _{cost}	85900	115446	2715961	4011	676457	592
B	15	45	5	F _{cost}	88732	75487	2686769	3515	664361	668
B	15	45	5	F _{cost}	88004	76569	2679188	3474	662155	667
B	15	45	5	F _{cost}	88138	75904	2678657	3478	662074	675
B	15	45	5	F _{cost}	89842	76651	2681109	3433	662167	681
B	15	45	5	F _{cost}	88568	76112	2679295	3467	662106	682
B	15	45	10	F _{cost}	97903	59392	2671910	3177	657293	756
B	15	45	10	F _{cost}	95412	57392	2667419	3256	657087	768
B	15	45	10	F _{cost}	97126	57631	2669372	3229	657255	762
B	15	45	10	F _{cost}	96021	56541	2667177	3268	657158	771
B	15	45	10	F _{cost}	97038	59405	2671059	3211	657457	762
B	15	45	20	F _{cost}	106543	41469	2662627	2951	652719	898
B	15	45	20	F _{cost}	107649	42856	2665120	2890	652651	909
B	15	45	20	F _{cost}	106236	42461	2671877	2964	655016	910
B	15	45	20	F _{cost}	106395	42190	2663200	2922	652544	914

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	15	45	20	F _{cost}	106048	42450	2663113	2940	652715	912
B	30	15	0	F _{cost}	133104	65570	2713289	3963	675322	529
B	30	15	0	F _{cost}	133994	66197	2714806	3922	675239	541
B	30	15	0	F _{cost}	133983	65926	2714523	3939	675354	542
B	30	15	0	F _{cost}	131541	65600	2711756	3978	675124	543
B	30	15	0	F _{cost}	133409	66418	2714442	3946	675410	541
B	30	15	5	F _{cost}	129479	39367	2683461	3407	662439	675
B	30	15	5	F _{cost}	125983	39505	2680103	3463	662252	684
B	30	15	5	F _{cost}	129625	39366	2683606	3389	662281	689
B	30	15	5	F _{cost}	128696	38896	2682207	3421	662295	684
B	30	15	5	F _{cost}	128328	39446	2682389	3409	662210	679
B	30	15	10	F _{cost}	134959	28807	2678381	3128	658281	842
B	30	15	10	F _{cost}	135078	28982	2678675	3108	658136	845
B	30	15	10	F _{cost}	134047	27866	2676528	3139	657966	862
B	30	15	10	F _{cost}	135113	27852	2677580	3125	658062	855
B	30	15	10	F _{cost}	135360	28319	2678294	3127	658250	856
B	30	15	20	F _{cost}	144916	18853	2678384	2841	655224	1197
B	30	15	20	F _{cost}	145688	18370	2678673	2842	655302	1203
B	30	15	20	F _{cost}	144245	19250	2678109	2867	655437	1203
B	30	15	20	F _{cost}	146743	17677	2679034	2826	655216	1213
B	30	15	20	F _{cost}	148393	19795	2682802	2776	655562	1206
B	30	30	0	F _{cost}	123560	91639	2729814	3935	678880	872
B	30	30	0	F _{cost}	123706	92229	2730550	3904	678721	888
B	30	30	0	F _{cost}	125049	93883	2733546	3868	679037	887
B	30	30	0	F _{cost}	121607	92858	2729080	3958	678953	885
B	30	30	0	F _{cost}	123906	92667	2731188	3949	679349	880
B	30	30	5	F _{cost}	119922	57550	2692088	3503	665474	1039
B	30	30	5	F _{cost}	122281	61107	2698003	3397	665725	1046
B	30	30	5	F _{cost}	123928	60730	2699274	3410	666160	1065
B	30	30	5	F _{cost}	117039	55497	2687151	3569	665025	1050
B	30	30	5	F _{cost}	122820	59828	2697264	3409	665681	1067
B	30	30	10	F _{cost}	131623	46308	2692546	3106	661352	1220
B	30	30	10	F _{cost}	135140	46388	2696143	3060	661701	1230
B	30	30	10	F _{cost}	132090	46317	2693022	3088	661271	1242
B	30	30	10	F _{cost}	127608	45800	2688023	3154	660808	1234

Continúa en la siguiente hoja

Agrupamiento	Dmax _k	Wmax _k	t _w	Fitness func	Trav _{obj}	Wait _{obj}	Tot _{obj}	Clust _{obj}	Valor fitness	Tiempo (seg)
B	30	30	10	F_{cost}	130147	45840	2690602	3120	661047	1255
B	30	30	20	F_{cost}	143276	33026	2690916	2765	657338	1600
B	30	30	20	F_{cost}	139747	31576	2685938	2878	657381	1655
B	30	30	20	F_{cost}	144117	32691	2691423	2766	657467	1625
B	30	30	20	F_{cost}	144824	33347	2692787	2753	657647	1622
B	30	30	20	F_{cost}	142054	31658	2688327	2823	657352	1622
B	30	45	0	F_{cost}	116657	95861	2727133	4112	680140	1243
B	30	45	0	F_{cost}	123937	104674	2743226	3872	681338	1252
B	30	45	0	F_{cost}	115534	97399	2727548	4113	680247	1272
B	30	45	0	F_{cost}	122902	103027	2740544	3908	681096	1286
B	30	45	0	F_{cost}	118768	97879	2731262	4027	680198	1249
B	30	45	5	F_{cost}	114027	62040	2690683	3631	666510	1425
B	30	45	5	F_{cost}	120312	67308	2702236	3452	667299	1458
B	30	45	5	F_{cost}	118698	64958	2698271	3506	666949	1460
B	30	45	5	F_{cost}	117541	64106	2696262	3557	667023	1460
B	30	45	5	F_{cost}	122933	67227	2704775	3409	667433	1444
B	30	45	10	F_{cost}	132117	53446	2700178	3089	662951	1630
B	30	45	10	F_{cost}	130982	52050	2697646	3102	662499	1650
B	30	45	10	F_{cost}	130412	52075	2697102	3112	662479	1664
B	30	45	10	F_{cost}	126624	50296	2691535	3233	662469	1649
B	30	45	10	F_{cost}	128799	52822	2696236	3141	662586	1678
B	30	45	20	F_{cost}	143579	39573	2697767	2751	658788	2032
B	30	45	20	F_{cost}	138167	36202	2688984	2910	658432	2073
B	30	45	20	F_{cost}	140911	38026	2693553	2817	658508	2064
B	30	45	20	F_{cost}	133793	36297	2684704	2997	658361	2055
B	30	45	20	F_{cost}	135176	35249	2685040	2987	658332	1975

Tabla A.3: Resultados detallados obtenidos con el método de IACS para las distintas configuraciones.

Id Cliente	Fecha	Hora	Duración	Localización
1345	10-10-2016	10:30	80	(40.421501, -3.7002308)

Tabla B.1: Ejemplo de tarea

Apéndice B

Formulación alternativa

B.1. Formulación matemática

El problema de planificación de asistentes de atención domiciliaria abordado en este trabajo, también puede plantearse desde una perspectiva exacta. Sin embargo, tal y como quedó expuesto en los capítulos 2 y 3 la alta dimensionalidad del problema planteado no hace posible la utilización de dicho tipo de técnicas. En todo caso en este anexo se proporciona una formulación exacta para el mismo. Existen un conjunto de tareas que se deben planificar semanalmente, para cada tarea se conoce su localización, su fecha y hora, la duración de la misma y el cliente que requiere de dicha tarea (ver tabla B.1).

B.1.1. Objetivos

El problema consiste en determinar qué tarea debe realizar cada asistente minimizando dos métricas de calidad que están enfrentadas, como son:

- Minimizar el número de asistentes requeridos para atender todas las tareas.
- Minimizar la distancia total recorrida por los asistentes.

B.1.2. Restricciones

Además se deben respetar las siguientes restricciones:

- Todas las tareas han de ser asignadas a uno y solo un asistente.
- Todas las tareas han de iniciarse dentro de su ventana temporal.

Id Cliente	Día	Hora	Duración	Localización
1345	10-10-2016	10:30	80	(40.421501, -3.7002308)
1345	12-10-2016	11:30	60	(40.421501, -3.7002308)
1345	14-10-2016	09:30	120	(40.421501, -3.7002308)

Tabla B.2: Ejemplo de servicio

- Todas las tareas que pertenezcan al mismo cliente y que estén dentro del mismo horario de trabajo han de ser realizadas por el mismo asistente.
- Ningún asistente puede realizar tareas que estén fuera de su horario.
- La duración total de la jornada de trabajo diaria de los asistentes no superará cierto límite (8 horas).
- La duración total de la jornada de trabajo semanal de los asistentes no superará cierto límite (40 horas).
- La distancia recorrida por los asistentes entre cada una de sus tareas no superará cierto límite.
- El tiempo máximo de espera incurrido por los asistentes para iniciar una tarea no superará cierto límite.

El problema de determinar la asignación óptima de asistentes a tareas, y puesto que todas las tareas de un cliente para un turno de trabajo deben ser realizadas por el mismo asistente, es equivalente a determinar qué servicios realiza cada asistente de modo óptimo para un turno dado.

De un modo más formal, sea $G = \{V, A\}$ un grafo finito, donde $V = \{v_1, \dots, v_n\}$ representa el conjunto de vértices, $A = \{(i, j) | i, j \in V\}$ el conjunto de aristas y $d_{ij} = d_{ji} \in \mathbb{R}^+$ la distancia de transitar desde el vértice v_i al v_j o viceversa. Cada vértice representa un cliente que debe ser visitado dentro de un horizonte de planificación D , compuesto por d días con $d \in D$. Además para cada cliente se define una periodicidad P_i que determina el número de tareas a realizar dentro de dicho horizonte. Puesto que existen clientes a los que se les debe visitar en distintos días, y en distintos horarios, se definen una serie de parámetros auxiliares, como son la hora de visita del cliente i en el día d α_i^d , la duración de dicha visita por d_i^d y un parámetro auxiliar X_{id} que determina si el cliente i tiene que ser visitado en el día d . Para cada asistente $k \in K$ y día d se define una ventana de disponibilidad $[e_k^d, l_k^d]$ que determina el horario en el que el asistente k puede trabajar el día d . Además para cada asistente k y día d se define un número máximo max^{kd} y mínimo min^{kd} de horas de trabajo, así como un número máximo max^k y mínimo min^k de horas de trabajo dentro del horizonte de planificación D . Finalmente para cada asistente k se define una constante $Dmax_k$, que determina la distancia máxima que puede recorrer entre cada tarea. Los parámetros del problema vienen especificados en la tabla B.3.

Una vez definidos los parámetros del problema se definen las variables de decisión, así como sus variables auxiliares. Para cada cliente i , asistente k se define la variable binaria x_{ik} , que toma valor 1 si el cliente i es visitado por el asistente k y valor 0 en otro caso. La

Parámetros	Descripción
d_{ij}	Distancia existente entre las tareas i y j .
X_{id}	Parámetro que indica si el cliente i debe ser visitado el día d .
P_i	Número de visitas a realizar al cliente i dentro del horizonte de planificación
d_i^d	Duración de la visita al cliente i en el día d .
$[\alpha_{ei}^d, \alpha_{li}^d]$	Horario en el que se debe visitar al cliente i el día d . Siendo α_{ei}^d la hora de comienzo más temprana y α_{li}^d la hora más tarde a la que debe comenzar la visita al cliente.
$[e_k^d, l_k^d]$	Horario en el que el asistente k puede trabajar el día d . Siendo e_k^d, l_k^d el comienzo y el final de su disponibilidad para el día d .
max_k^d, min_k^d	Horas máximas y mínimas de trabajo para el asistente k en el día d .
max_k, min_k	Horas máximas y mínimas de trabajo para el asistente k dentro del horizonte de planificación.
$Dmax_k$	Distancia máxima que puede recorrer el asistente k entre dos clientes.
$Wmax_k$	Tiempo máximo de espera en el que puede incurrir el asistente k antes de iniciar una tarea

Tabla B.3: Parámetros.

variable binaria x_{ik}^d que tomará valor 1 si el cliente i es visitado por el asistente k el día d . La variable x_{ijk}^d tomará valor 1 si el cliente j es visitado inmediatamente después del i por el asistente k el día d . Y finalmente la variable c_k que tomará valor 1 si el asistente k visita a algún cliente dentro del horizonte de planificación.

Una vez presentadas las variables y los parámetros del problema, se muestra una formulación para un problema de programación lineal entera mixta (MILP). La naturaleza del problema a resolver presenta varios objetivos a resolver, los cuales se detallan a continuación:

1. Minimizar el número de asistentes utilizados para atender a todos los clientes durante el periodo de planificación. En lo sucesivo nos referiremos a dicho objetivo como $Clust_{obj}$.

$$\text{Minimizar } Clust_{obj} = \sum_{k \in K} c_k \quad (\text{B.1})$$

2. Minimizar el tiempo total requerido para visitar a todos los clientes durante el periodo de planificación. En lo sucesivo nos referiremos a dicho objetivo como Tot_{obj} .

$$\text{Minimizar } Tot_{obj} = Trav_{obj} + Wait_{obj} \quad (\text{B.2})$$

3. Minimizar el tiempo total de desplazamiento de todos los asistentes. En lo sucesivo nos referiremos a dicho objetivo como $Trav_{obj}$.

$$\text{Minimizar } Trav_{obj} = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} \sum_{d \in D} x_{ijk}^d d_{ij} \quad (\text{B.3})$$

Variable	Descripción
i, j	Índices para los clientes.
k	Índice para los asistentes.
d	Índice para el día.
x_{ik}	Variable de decisión; su valor es 1 si el cliente i es visitado por el asistente k y 0 en otro caso.
x_{ik}^d	Variable auxiliar; su valor es 1 si el cliente i es visitado por el asistente k en el día d , 0 en otro caso.
x_{ijk}^d	Variable auxiliar; su valor es 1 si el cliente j es visitado justo después de i por el asistente k en el día d , 0 en otro caso.
h_{ik}^d	Variable que determina la hora a la que comenzará el asistente k la visita al cliente i en el día d .
s_{ij}^d	Diferencia temporal entre el comienzo de la visita al cliente i y el j para el día d . Dicha variable toma valores siempre que $X_{id} = X_{jd} = 1$ y es igual a $h_{jk}^d - h_{ik}^d$.
w_{ij}^d	Tiempo de espera incurrido antes de realizar la tarea j en el día d , el cual es igual a $s_{ij}^d - d_{ij} - d_i^d$.
c_k	Variable auxiliar; su valor es 1 si el asistente k realiza alguna visita a algún cliente dentro del horizonte de planificación.

4. Minimizar el tiempo total de espera de todos los asistentes. En lo sucesivo nos referiremos a dicho objetivo como Wait_{obj} .

$$\text{Minimizar } \text{Wait}_{obj} = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} \sum_{d \in D} x_{ijk}^d w_{ij} \quad (\text{B.4})$$

Estando sujeto a las siguientes restricciones

1. Cada cliente solamente puede ser visitado por un asistente.

$$\sum_{k \in K} x_{ik} = 1 \quad \forall i \in V \quad (\text{B.5})$$

2. Cada cliente solamente puede ser visitado por un asistente para un día d , siempre y cuando el cliente deba ser visitado dicho día.

$$\sum_{k \in K} x_{ik}^d = X_{id} \quad \forall i \in V, d \in D \quad (\text{B.6})$$

3. Cada cliente debe ser visitado un número P_i de veces dentro del horizonte de planificación.

$$\sum_{k \in K} \sum_{d \in D} x_{ik}^d = P_i \quad \forall i \in V \quad (\text{B.7})$$

4. Una vez realizada una visita a un cliente i en un día d por un asistente k , se deberá asignar al mismo asistente para el resto de visitas que el cliente requiera dentro del horizonte de planificación.

$$\sum_{d \in D} x_{ik}^d x_{ik} X_{id} = P_i x_{ik} \quad \forall i \in V, k \in K \quad (\text{B.8})$$

5. Desde el un cliente i , para un día d y un asistente k solamente se puede transitar a un solo cliente j .

$$\sum_{j \in V} x_{ijk}^d = X_{ik}^d \quad \forall i \in V, d \in D, k \in K \quad (\text{B.9})$$

6. A un cliente cliente j , para un día d y un asistente k solamente se puede transitar desde un solo cliente i .

$$\sum_{i \in V} x_{ijk}^d = X_{jk}^d \quad \forall j \in V, d \in D, k \in K \quad (\text{B.10})$$

7. El orden de visitas de un asistente k para un día d debe ser factible, es decir, debe existir suficiente tiempo para ir del cliente i al cliente j .

$$h_{jk}^d \leq h_{ik}^d + d_i^d + d_{ij} - M(1 - x_{ijk}^d) \quad \forall k \in K, d \in D, (i, j) \in A \quad (\text{B.11})$$

8. La hora de visita a un cliente i por parte de un asistente k para un día d debe estar dentro de los límites de ventana temporal.

$$\alpha_{ei}^d \leq h_{ik}^d \leq \alpha_{li}^d \quad \forall i \in V, d \in D, k \in K \quad (\text{B.12})$$

9. La visita a cualquier cliente i por parte de un asistente k para el día d , debe respetar la disponibilidad del asistente.

$$x_{ik}^d e_k^d \leq x_{ik}^d \alpha_i^d \leq x_{ik}^d l_k^d \quad \forall i \in V, k \in K, d \in D \quad (\text{B.13})$$

10. Se deben respetar las horas mínimas y máximas diarias de trabajo para los asistentes.

$$\min_k^d \leq \sum_{i \in V} x_{ik}^d d_i^d \leq \max_k^d \quad \forall k \in K, d \in D \quad (\text{B.14})$$

11. Se deben respetar las horas mínimas y máximas de trabajo para los asistentes dentro del horizonte de planificación.

$$\min_k \leq \sum_{d \in D} \sum_{i \in V} x_{ik}^d d_i^d \leq \max_k \quad \forall k \in K \quad (\text{B.15})$$

12. Se debe respetar la distancia máxima que un asistente puede recorrer entre dos clientes.

$$x_{ijk}^d d_{ij} \leq D \max_k \quad \forall k \in K, d \in D, (i, j) \in A \quad (\text{B.16})$$

13. Se debe respetar el tiempo máximo de espera en el que puede incurrir un asistente en cada tarea.

$$x_{ijk}^d w_{ij} \leq W \max_k \quad \forall k \in K, d \in D, (i, j) \in A \quad (\text{B.17})$$

14. Las variables del problema toman valores binarios

$$x_{ik} \in \{0, 1\} \quad \forall i \in V, k \in K \quad (\text{B.18})$$

$$x_{ik}^d \in \{0, 1\} \quad \forall i \in V, k \in K, d \in D \quad (\text{B.19})$$

$$x_{ijk}^d \in \{0, 1\} \quad \forall k \in K, d \in D, (i, j) \in A \quad (\text{B.20})$$

Bibliografía

- A., T. y P, H. Optimization of daily scheduling for home health care services. *Journal of Applied Operational Research*, vol. 3, páginas 124 – 136, 2011.
- ACHUTHAN, N. y CACCETTA, L. Integer linear programming formulation for a vehicle routing problem. *European Journal of Operational Research*, vol. 52(1), páginas 86–89, 1991.
- ADANSON, M. *Familles des plantes.Par M. Adanson..* Smithsonian Institution, 1763.
- AGARWAL, Y., MATHUR, K. y SALKIN, H. M. A set-partitioning-based exact algorithm for the vehicle routing problem. *Networks*, vol. 19(7), páginas 731–749, 1989.
- AKJIRATIKARL, C., YENRADEE, P. y DRAKE, P. R. PSO-based algorithm for home care worker scheduling in the UK. *Computers & Industrial Engineering*, vol. 53(4), páginas 559–583, 2007a. ISSN 03608352.
- AKJIRATIKARL, C., YENRADEE, P. y DRAKE, P. R. Pso-based algorithm for home care worker scheduling in the {UK}. *Computers Industrial Engineering*, vol. 53(4), páginas 559 – 583, 2007b. ISSN 0360-8352.
- ALTINEL, I. y ONCAN, T. A new enhancement of the clarke and wright savings heuristic for the capacitated vehicle routing problem. *J Oper Res Soc*, vol. 56(8), páginas 954–961, 2004.
- ALTINKEMER, K. y GAVISH, B. Parallel savings based heuristics for the delivery problem. *Operations Research*, vol. 39(3), páginas 456–469, 1991.
- ANDERBERG, M. R. *Cluster Analysis for Applications.* Elsevier BV, 1973.
- ANDREA, T. y PATRICK, H. A branch-price-and-cut approach for solving the medium-term home health care planning problem. *Networks*, vol. 64(3), páginas 143–159, 2014.
- APPLEGATE, D. L., BIXBY, R. E., CHVÁTAL, V. y COOK, W. J. *The Traveling Salesman Problem: A Computational Study.* Princeton University Press, Princeton, 2007. ISBN 0691129932.
- ARABIE, P., HUBERT, L. J. y SOETE, G. D. *Clustering and Classification.* World Scientific Pub Co Pte Lt, 1996.

- BAKER, E. K. Technical note—an exact algorithm for the time-constrained traveling salesman problem. *Operations Research*, vol. 31(5), páginas 938–945, 1983.
- BALAS, E. y SIMONETTI, N. Linear time dynamic-programming algorithms for new classes of restricted TSPs: A computational study. *INFORMS Journal on Computing*, vol. 13(1), páginas 56–75, 2001.
- BALDACCI, R., CHRISTOFIDES, N. y MINGOZZI, A. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Math. Program.*, vol. 115(2), páginas 351–385, 2008.
- BALDACCI, R., HADJICONSTANTINO, E. y MINGOZZI, A. An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations Research*, vol. 52(5), páginas 723–738, 2004.
- BALDACCI, R., MINGOZZI, A. y ROBERTI, R. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, vol. 59(5), páginas 1269–1283, 2011.
- BALDACCI, R., MINGOZZI, A. y ROBERTI, R. Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research*, vol. 218(1), páginas 1–6, 2012.
- BALINSKI, M. L. y QUANDT, R. E. On an integer program for a delivery problem. *Operations Research*, vol. 12(2), páginas 300–304, 1964.
- BARD, J. F., SHAO, Y. y JARRAH, A. I. A sequential grasp for the therapist routing and scheduling problem. *Journal of Scheduling*, vol. 17(2), páginas 109–133, 2014. ISSN 1099-1425.
- BARD, J. F., SHAO, Y. y WANG, H. Weekly scheduling models for traveling therapists. *Socio-Economic Planning Sciences*, 2012. ISSN 00380121.
- BARD, J. F., SHAO, Y. y WANG, H. Weekly scheduling models for traveling therapists. *Socio-Economic Planning Sciences*, vol. 47(3), páginas 191 – 204, 2013. ISSN 0038-0121. Health Care Modelling.
- BATRA, S. W. T. Nests and social behavior of halictine bees of india (hymenoptera: Halictidae. *Indian J. Entomol*, vol. 1(28), páginas 375–393, 1966.
- BATTARRA, M., GOLDEN, B. y VIGO, D. Tuning a parametric clarke–wright heuristic via a genetic algorithm. *J Oper Res Soc*, vol. 59(11), páginas 1568–1572, 2007.
- BAXTER, J. Depot location: A technique for the avoidance of local optima. *European Journal of Operational Research*, vol. 18(2), páginas 208–214, 1984.
- BEGUR, S. V. An integrated spatial dss for scheduling and routing home-health-care nurses. *Interfaces*, vol. 27(4), páginas 35–48, 1997. ISSN 0092-2102.
- BEKTAS, T. The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, vol. 34(3), páginas 209–219, 2006.

- BENNETT, A. R. y ERERA, A. L. Dynamic periodic fixed appointment scheduling for home health. *IIE Transactions on Healthcare Systems Engineering*, vol. 1(1), páginas 6–19, 2011.
- BERTELS, S. y FAHLE, T. A hybrid setup for a hybrid scenario: Combining heuristics for the home health care problem. *Comput. Oper. Res.*, vol. 33(10), páginas 2866–2890, 2006. ISSN 0305-0548.
- BIGGS, N., LLOYD, E. K. y WILSON, R. J. *Graph Theory, 1736-1936*. Clarendon Press, New York, NY, USA, 1986. ISBN 0-198-53916-9.
- BLAIS, M., LAPIERRE, S. D. y LAPORTE, G. Solving a home-care districting problem in an urban setting. *Journal of the Operational Research Society*, vol. 54(11), 2003. ISSN 1476-9360.
- BLOOM, D. E. 7 billion and counting. *Science*, vol. 333(6042), páginas 562–569, 2011.
- BOWERS, J., CHEYNE, H., MOULD, G. y PAGE, M. Continuity of care in community midwifery. *Health Care Management Science*, vol. 18(2), páginas 195–204, 2015. ISSN 1572-9389.
- BRÄAYSY, O., NAKARI, P., DULLAERT, W. y NEITTAANMÄKI, P. An optimization approach for communal home meal delivery service: A case study. *Journal of Computational and Applied Mathematics*, vol. 232(1), páginas 46–53, 2009.
- BRAEKERS, K., HARTL, R. F., PARRAGH, S. N. y TRICOIRE, F. A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *European Journal of Operational Research*, vol. 248(2), páginas 428 – 443, 2016. ISSN 0377-2217.
- BRAYSY, O., DULLAERT, W. y NAKARI, P. The potential of optimization in communal routing problems: case studies from finland. *Journal of Transport Geography*, vol. 17(6), páginas 484–490, 2009. ISSN 0966-6923.
- BREDSTRÖM, D. y RÖNNQVIST, M. A branch and price algorithm for the combined vehicle routing and scheduling problem with synchronization constraints. Discussion Papers 2007/7, Norwegian School of Economics, Department of Business and Management Science, 2007.
- BREDSTRÖM, D. y RÖNNQVIST, M. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, vol. 191(1), páginas 19–31, 2008. ISSN 03772217.
- BULLNHEIMER, B., HARTL, R. F. y STRAUSS, C. Applying the ANT system to the vehicle routing problem. En *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, páginas 285–296. Springer Science + Business Media, 1999.
- CALVO, R. W. A new heuristic for the traveling salesman problem with time windows. *Transportation Science*, vol. 34(1), páginas 113–124, 2000.

- CAPPANERA, P. y SCUTELLÁ, M. G. Joint assignment, scheduling, and routing models to home care optimization: A pattern-based approach. *Transportation Science*, vol. 49(4), páginas 830–852, 2015.
- CARLTON, W. B. y BARNES, J. W. Solving the travelling salesman problem with time windows using tabu search. *IIE Transactions*, vol. 28, páginas 617–629, 1996.
- CARO, G. D., DUCATELLE, F. y GAMBARDELLA, L. M. Anthocnet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *EUROPEAN TRANSACTIONS ON TELECOMMUNICATIONS*, vol. 16, páginas 443–455, 2005.
- CASTILLO-SALAZAR, J. A., LANDA-SILVA, D. y QU, R. Workforce scheduling and routing problems: literature survey and computational study. *Annals of Operations Research*, vol. 239(1), páginas 39–67, 2016. ISSN 1572-9338.
- CENSOR, Y. Pareto optimality in multiobjective problems. *Applied Mathematics & Optimization*, vol. 4(1), páginas 41–59, 1977.
- CHENG, C.-B. y MAO, C.-P. A modified ant colony system for solving the travelling salesman problem with time windows. *Mathematical and Computer Modelling*, vol. 46(9-10), páginas 1225–1235, 2007.
- CHENG, E. y RICH, J. L. A home health care routing and scheduling problem. *Technical report CAAM TR98-04.*, 1998.
- CHERIF-KHETTAF, W. R., RACHID, M. H., BLOCH, C. y CHATONNAY, P. New notation and classification scheme for vehicle routing problems. *RAIRO - Operations Research*, vol. 49(1), páginas 161–194, 2014.
- CHRISTOFIDES, N. y EILON, S. An algorithm for the vehicle-dispatching problem. *OR*, vol. 20(3), página 309, 1969.
- CHRISTOFIDES, N., MINGOZZI, A. y TOTH, P. An algorithm for the time constrained travelling salesman problem. *Technical Report IC OR 8125*, 1981a.
- CHRISTOFIDES, N., MINGOZZI, A. y TOTH, P. State-space relaxation procedures for the computation of bounds to routing problems. *Networks*, vol. 11(2), páginas 145–164, 1981c.
- CHUNYU, R. y XIAOBO, W. Research on VRP optimizing based on hierarchy clustering and IGA under common distribution. En *2006 International Conference on Computational Intelligence and Security*. Institute of Electrical & Electronics Engineers (IEEE), 2006.
- CIRE, A. A. y HOOKER, J. N. A heuristic logic-based benders method for the home health care problem. 2012.
- CISSE, M., YALCINDAG, S., KERGOSIEN, Y., SAHIN, E., LENTE, C. y MATTA, A. Or problems related to home health care: A review of relevant routing and scheduling problems. *Operations Research for Health Care*, vol. 13-14(Supplement C), páginas 1 – 22, 2017. ISSN 2211-6923.

- CLARKE, G. y WRIGHT, J. W. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, vol. 12(4), páginas 568–581, 1964.
- CLAY INSTITUTE OF MATHEMATICS. P vs np problem. 2000.
- COOK, W. J. *In Pursuit of the Traveling Salesman*. Walter de Gruyter GmbH, 2015.
- CORDEAU, J.-F., DESAULNIERS, G., DESROSIERS, J., SOLOMON, M. M. y SOUMIS, F. 7. VRP with time windows. En *The Vehicle Routing Problem*, páginas 157–193. Society for Industrial & Applied Mathematics (SIAM), 2002.
- CORDEAU, J.-F. y LAPORTE, G. *Tabu Search Heuristics for the Vehicle Routing Problem*, páginas 145–163. Springer US, Boston, MA, 2005. ISBN 978-0-387-23667-4.
- CORDON, O., DE VIANA, I. F., HERRERA, F. y MORENO, L. A new aco model integrating evolutionary computation concepts: The best-worst ant system. 2000.
- CORNILLIER, F., LAPORTE, G., BOCTOR, F. F. y RENAUD, J. The petrol station replenishment problem with time windows. *Computers & Operations Research*, vol. 36(3), páginas 919–935, 2009.
- COSTA-FONT, J., ELVIRA, D. y MASCARILLA-MIRO, O. ‘ageing in place’? exploring elderly people’s housing preferences in spain. *Urban Studies*, vol. 46(2), páginas 295–316, 2009.
- D. APPLGATE, R. E. BIXBY, V. CHVÁTAL, AND W. COOK. Concorde – a code for solving traveling salesman problems. release, 12 2003. 2003. Disponible en <http://www.math.uwaterloo.ca/tsp/concorde/index.html> (último acceso, Agosto, 2015).
- DANTZIG, G., FULKERSON, R. y JOHNSON, S. Solution of a large-scale traveling-salesman problem. *Operations Research*, vol. 2, páginas 393–410, 1954.
- DANTZIG, G. B. y RAMSER, J. H. The truck dispatching problem. *Management Science*, vol. 6, páginas 80–91, 1959.
- DANTZIG, G. B. y WOLFE, P. Decomposition principle for linear programs. *Oper. Res.*, vol. 8(1), páginas 101–111, 1960. ISSN 0030-364X.
- DARWIN, C. *Origin of species / Charles Darwin..* Smithsonian Institution, 1900.
- DENEUBOURG, ARON, S., GOSS, S. y PASTEELS, J. M. The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, vol. 3(2), páginas 159–168, 1990. ISSN 0892-7553.
- DESAULNIERS, G., LESSARD, F. y HADJAR, A. Tabu search, partial elementarity, and generalizedk-path inequalities for the vehicle routing problem with time windows. *Transportation Science*, vol. 42(3), páginas 387–404, 2008.
- DESAULNIERS, G., MADSEN, O. B. y ROPKE, S. Chapter 5: The vehicle routing problem with time windows. En *Problems, Methods, and Applications, Second Edition*, páginas 119–159. Society for Industrial & Applied Mathematics (SIAM), 2014.

- DESROCHERS, M., LENSTRA, J. y SAVELSBERGH, M. A classification scheme for vehicle routing and scheduling problems. *European Journal of Operational Research*, vol. 46(3), páginas 322–332, 1990.
- DI GASPERO, L. y URLI, T. A cp/lms approach for multi-day homecare scheduling problems. En *Hybrid Metaheuristics* (editado por M. J. Blesa, C. Blum y S. Voß), páginas 1–15. Springer International Publishing, Cham, 2014. ISBN 978-3-319-07644-7.
- DOERNER, K. F. y SCHMID, V. Survey: Matheuristics for rich vehicle routing problems. En *Hybrid Metaheuristics*, páginas 206–221. Springer Science Business Media, 2010.
- DONDO, R. y CERDÁ, J. A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. *European Journal of Operational Research*, vol. 176(3), páginas 1478–1507, 2007.
- DORIGO, M. *Optimization, Learning and Natural Algorithms*. Tesis Doctoral, Dipartimento di Elettronica e Informazione, Politecnico di Milano, IT, 1992.
- DORIGO, M., CARO, G. D. y GAMBARDELLA, L. M. Ant Algorithms for Discrete Optimization. *Artificial Life*, vol. 5(2), páginas 137–172, 1999. ISSN 1064-5462.
- DORIGO, M. y GAMBARDELLA, L. M. A study of some properties of ant-q. En *Parallel Problem Solving from Nature — PPSN IV*, páginas 656–665. Springer Nature, 1996.
- DORIGO, M. y GAMBARDELLA, L. M. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, 1997.
- DORIGO, M., MANIEZZO, V. y COLORNI, A. The ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, vol. 26(1), páginas 29–41, 1996.
- DORIGO, M. y STÜTZLE, T. *Ant Colony Optimization (Bradford Books)*. The MIT Press, 2004. ISBN 0262042193.
- DORIGO, M. y STÜTZLE, T. Ant colony optimization: Overview and recent advances. 2009.
- DORIGO, M. y STÜTZLE, T. Ant colony optimization: Overview and recent advances. En *Handbook of Metaheuristics*, páginas 227–263. Springer Science Business Media, 2010.
- DRÉO, J., SIARRY, P., PÉTROWSKI, A. y TAILLARD, E. *Metaheuristics for Hard Optimization*. Springer-Verlag, Berlin, 2006. ISBN 3-540-23022-X.
- DU, D.-Z. y PARDALOS, P. M., editores. *Handbook of Combinatorial Optimization*. Springer Science Business Media, 2005.
- DUMAS, Y., DESROSIERS, J., GELINAS, E. y SOLOMON, M. M. An optimal algorithm for the traveling salesman problem with time windows. *Operations Research*, vol. 43(2), páginas 367–371, 1995.

- DUQUE, P. M., CASTRO, M., SÅPRENSEN, K. y GOOS, P. Home care service planning. the case of landelijke thuiszorg. *European Journal of Operational Research*, vol. 243(1), páginas 292 – 301, 2015. ISSN 0377-2217.
- EKSIÖGLU, B., VURAL, A. V. y REISMAN, A. The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, vol. 57(4), páginas 1472–1483, 2009.
- ERWIN, T. L. Tropical Forests: Their Richness in Coleoptera and Other Arthropod Species. *The Coleopterists Bulletin*, vol. 36(1), páginas 74–75, 1982.
- EULEN GROUP. Servicios sociosanitarios. 2016. Disponible en <http://www.eulen.com/cl/servicios-sociosanitarios/ayuda-domicilio/> (último acceso, Diciembre, 2016).
- EUROPEAN COMMISSION. Demography report. 2015. Disponible en <http://ec.europa.eu/eurostat/documents/3217494/6917833/KE-BM-15-003-EN-N.pdf> (último acceso, Online; accessed 24 de septiembre de 2018).
- EVEBORN, P., FLISBERG, P. y RÖNNQVIST, M. Laps Care—An operational system for staff planning of home care. *European Journal of Operational Research*, vol. 171(3), páginas 962–976, 2006. ISSN 03772217.
- EVEBORN, P., RANNQVIST, M., EINARSDATTIR, H., EKLUND, M., LIDAN, K. y ALMROTH, M. Operations research improves quality and efficiency in home care. *Interfaces*, vol. 39(1), páginas 18–34,97–98,100,102, 2009.
- EVERITT, B. S., LANDAU, S., LEESE, M. y STAHL, D. *Cluster Analysis*. John Wiley & Sons, Ltd, 2011.
- FERNANDEZ, A., GREGORY, G., HINDLE, A. y LEE, A. C. A model for community nursing in a rural county. *Journal of the Operational Research Society*, vol. 25(2), páginas 231–239, 1974. ISSN 1476-9360.
- FIKAR, C. y HIRSCH, P. Home health care routing and scheduling: A review. *Computers & Operations Research*, vol. 77, páginas 86 – 95, 2017. ISSN 0305-0548.
- FIKAR, C. y HIRSCH, P. Evaluation of trip and car sharing concepts for home health care services. *Flexible Services and Manufacturing Journal*, vol. 30(1), páginas 78–97, 2018. ISSN 1936-6590.
- FOCACCI, F., LODI, A. y MILANO, M. A hybrid exact algorithm for the TSPTW. *INFORMS Journal on Computing*, vol. 14(4), páginas 403–417, 2002.
- FRANÇA, P. M., GENDREAU, M., LAPORTE, G. y MÃLLER, F. M. The m -traveling salesman problem with minmax objective. *Transportation Science*, vol. 29(3), páginas 267–275, 1995.
- FRANCIS, P. M., SMILOWITZ, K. R. y TZUR, M. *The Period Vehicle Routing Problem and its Extensions*, páginas 73–102. Springer US, Boston, MA, 2008. ISBN 978-0-387-77778-8.

- FREEMAN, G. y HUGHES, J. Continuity of care and the patient experience. 2010.
- FUELLERER, G., DOERNER, K. F., HARTL, R. F. y IORI, M. Ant colony optimization for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, vol. 36(3), páginas 655–673, 2009.
- FUELLERER, G., DOERNER, K. F., HARTL, R. F. y IORI, M. Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research*, vol. 201(3), páginas 751–759, 2010. ISSN 03772217.
- FÜGENSCHUH, A. Solving a school bus scheduling problem with integer programming. *European Journal of Operational Research*, vol. 193(3), páginas 867–884, 2009.
- GAMBARDELLA, L. M. y DORIGO, M. Ant-q: A reinforcement learning approach to the traveling salesman problem. En *Machine Learning Proceedings 1995*, páginas 252–260. Elsevier BV, 1995.
- GAMBARDELLA, L. M., TAILLARD, E. y AGAZZI, G. Macs-vrptw: A multiple ant colony system for vehicle routing problems with time windows. En *New Ideas in Optimization* (editado por D. Corne, M. Dorigo, F. Glover, D. Dasgupta, P. Moscato, R. Poli y K. V. Price), páginas 63–76. McGraw-Hill Ltd., UK, Maidenhead, UK, England, 1999. ISBN 0-07-709506-5.
- GAMST, M. y JENSEN, T. S. *A branch-and-price algorithm for the long-term home care scheduling problem*, páginas 483–488. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-29210-1.
- GASKELL, T. J. Bases for vehicle fleet scheduling. *OR*, vol. 18(3), página 281, 1967.
- GEHRING, H. y HOMBERGER, J. A parallel two-phase metaheuristic for routing problems with time windows. *Asia-Pacific Journal of Operational Research*, vol. 18, páginas 35–47, 2001.
- GEHRING, H. y HOMBERGER, J. Parallelization of a two-phase metaheuristic for routing problems with time windows. *Journal of Heuristics*, vol. 8(3), páginas 251–276, 2002.
- GENDREAU, M., HERTZ, A. y LAPORTE, G. New insertion and postoptimization procedures for the traveling salesman problem. *Operations Research*, vol. 40(6), páginas 1086–1094, 1992.
- GENDREAU, M., HERTZ, A., LAPORTE, G. y STAN, M. A generalized insertion heuristic for the traveling salesman problem with time windows. *Operations Research*, vol. 46(3), páginas 330–335, 1998.
- GENDREAU, M. y POTVIN, J.-Y., editores. *Handbook of Metaheuristics*. Springer US, 2010.
- GOLDEN, B. L., MAGNANTI, T. L. y NGUYEN, H. Q. Implementing vehicle routing algorithms. *Networks*, vol. 7(2), páginas 113–148, 1977.

- GOLDEN, B. L., WASIL, E. A., KELLY, J. P. y CHAO, I.-M. The impact of metaheuristics on solving the vehicle routing problem: Algorithms, problem sets, and computational results. En *Fleet Management and Logistics*, páginas 33–56. Springer Science Business Media, 1998.
- GOOGLE CLOUD PLATFORM. Geocoding cloud platform:compute engine. 2017. Disponible en <https://cloud.google.com/compute/pricing> (último acceso, Junio, 2017).
- GOOGLE MAPS APIS. Directions api. 2016a. Disponible en <https://developers.google.com/maps/documentation/directions/start> (último acceso, Diciembre, 2016).
- GOOGLE MAPS APIS. Geocoding api. 2016b. Disponible en <https://developers.google.com/maps/documentation/geocoding/start> (último acceso, Diciembre, 2016).
- GOOGLE MAPS APIS. Heatmap layer. 2016c. Disponible en <https://developers.google.com/maps/documentation/javascript/heatmaplayer?hl=en> (último acceso, Diciembre, 2016).
- GOOGLE MAPS APIS. Directions api usage limits. 2018. Disponible en <https://developers.google.com/maps/documentation/directions/usage-limits> (último acceso, Mayo, 2018).
- GOSS, S., ARON, S., DENEUBOURG, J. y PASTEELS, J. Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, vol. 76(12), páginas 579–581, 1989.
- GRENE, D., MARJORIE; DEPEW. The philosophy of biology: an episodic history. *Cambridge University Press.*, vol. 1(1), 2004.
- GROËR, C., GOLDEN, B. y WASIL, E. The consistent vehicle routing problem. *Manufacturing & Service Operations Management*, vol. 11(4), páginas 630–643, 2009.
- GUROBI OPTIMIZATION, I. Gurobi optimization tutorial, traveling salesman problem. 2015. Disponible en <http://examples.gurobi.com/traveling-salesman-problem/> (último acceso, Agosto, 2015).
- GUTIERREZ, E., GUTIERREZ, V. y VIDAL, C. Home health care logistics management: Framework and research perspectives. *International Journal of Industrial Engineering and Management*, vol. 4(3), páginas 173–182, 2013.
- GUTJAHR, W. On the finite-time dynamics of ant colony optimization. *Methodology and Computing in Applied Probability*, vol. 8, páginas 105–133, 2006. ISSN 1387-5841. 10.1007/s11009-006-7291-4.
- GUTJAHR, W. J. A graph based ant system and its convergence. *Future Generation Computer Systems*, vol. 16, páginas 873–888, 2000.
- GUTJAHR, W. J. Mathematical runtime analysis of aco algorithms: survey on an emerging issue. *Swarm Intelligence*, vol. 1, páginas 59–79, 2007.

- HADJICONSTANTINOY, E., CHRISTOFIDES, N. y MINGOZZI, A. A new exact algorithm for the vehicle routing problem based on q -paths and k -shortest paths relaxations. *Annals of Operations Research*, vol. 61(1), páginas 21–43, 1995.
- HANSEN, P., MLADENOVIC, N., BRIMBERG, J. y PÉREZ, J. A. M. Variable neighborhood search. En *Handbook of Metaheuristics*, páginas 61–86. Springer Science + Business Media, 2010.
- HARTMANIS, J. Computers and intractability: A guide to the theory of NP completeness (michael r. Garey and david s. Johnson). *SIAM Rev.*, vol. 24(1), páginas 90–91, 1982.
- HEWITT, M., NOWAK, M. y NATARAJ, N. Planning strategies for home health care delivery. *Asia-Pacific Journal of Operational Research*, vol. 33(05), página 1650041, 2016.
- HIERMANN, G., PRANDTSTETTER, M., RENDL, A., PUCHINGER, J. y RAIDL, G. R. Metaheuristics for solving a multimodal home-healthcare scheduling problem. *Central European Journal of Operations Research*, vol. 23(1), páginas 89–113, 2015. ISSN 1613-9178.
- HILLIER, F. S. y LIEBERMAN, G. J. *Introduction to Operations Research 10th edition*. McGraw-Hill Science/Engineering/Math, 2015. ISBN 0073523453.
- HOLDOBLER, B. y WILSON, E. O. *The Ants*. Belknap Press of Harvard University Press, 1st edición, 1990. ISBN 0674040759.
- HONDA, K., NAGATA, Y. y ONO, I. A parallel genetic algorithm with edge assembly crossover for 100,000-city scale TSPs. En *2013 IEEE Congress on Evolutionary Computation*. Institute of Electrical & Electronics Engineers (IEEE), 2013.
- HORNER, P. Edelman Award: ORION delivers success for UPS. 2016. Disponible en <https://www.informs.org/ORMS-Today/Public-Articles/June-Volume-43-Number-3/Edelman-Award-ORION-delivers-success-for-UPS> (último acceso, Junio, 2016).
- HORNI, A., NAGEL, K. y AXHAUSEN, K. W. *The Multi-Agent Transport Simulation MATSim*. Ubiquity Press, United Kingdom, 2016. ISBN 1909188751, 9781909188754.
- IKEGAMI, A. y UNO, A. Bounds for staff size in home help staff scheduling. *Journal of the Operations Research Society of Japan*, vol. 50(4), páginas 563–575, 2007.
- ILOG CPLEX OPTIMIZATION STUDIO, I. Ilog cplex optimization studio, user manual. 2015. Disponible en http://www.ibm.com/support/knowledgecenter/SSSA5P_12.6.3/ilog.odms.cplex.help/CPLEX/UsrMan/topics/progr_adv/usr_cut_lazy_constr/02_defn.html (último acceso, Agosto, 2015).
- INSTITUTO NACIONAL DE ESTADÍSTICA . Encuesta Anual de Coste Laboral. 2017. Disponible en <http://www.ine.es/prensa/np983.pdf>. (último acceso, Online; accessed 24 de septiembre de 2018).

- JEPSEN, M., PETERSEN, B., SPOORENDONK, S. y PISINGER, D. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research*, vol. 56(2), páginas 497–511, 2008.
- JOHNSON, D. S. A brief history of np-completeness, 1954-2012. 2012.
- JONKER, R. y VOLGENANT, T. Transforming asymmetric into symmetric traveling salesman problems. *Operations Research Letters*, vol. 2(4), páginas 161–163, 1983.
- JUAN, A. A., GRASMAN, S. E., CACERES-CRUZ, J. y BEKTAŞ, T. A simheuristic algorithm for the single-period stochastic inventory-routing problem with stock-outs. *Simulation Modelling Practice and Theory*, vol. 46, páginas 40–52, 2014.
- KALLEHAUGE, B. Formulations and exact algorithms for the vehicle routing problem with time windows. *Computers & Operations Research*, vol. 35(7), páginas 2307–2330, 2008.
- KAPLAN, C. S. y BOSCH, R. Tsp art. En *Renaissance Banff: Mathematics, Music, Art, Culture* (editado por R. Sarhangi y R. V. Moody), páginas 301–308. Bridges Conference, 2005. ISBN 0-9665201-6-5. ISSN 1099-6702.
- KARP, R. M. Reducibility among combinatorial problems. En *Complexity of Computer Computations*, páginas 85–103. Springer Science and Business Media, 1972.
- KATRAGJINI, K., RUIZ, R. y RODRÍGUEZ, R. Using real world distances in logistics management. En *International Conference on Industrial Engineering and Industrial Management*. XIV Congreso de Ingeniería de Organización, 2010.
- KERGOSIEN, Y., LENTE, C. y BILLAUT, J.-C. Home health care problem: An extended multiple traveling salesman problem. En *Proceedings of the 4th Multidisciplinary International Scheduling Conference: Theory and Applications (MISTA 2009), 10-12 Aug 2009, Dublin, Ireland* (editado por J. Blazewicz, M. Drozdowski, G. Kendall y B. McCollum), páginas 85–92. 2009a.
- KERGOSIEN, Y., LENTÉ, C. y BILLAUT, J.-C. Home health care problem: An extended multiple Traveling Salesman Problem. En *4th Multidisciplinary International Conference on Scheduling: Theory and Applications*, páginas –. Dublin, Ireland, 2009b.
- KIM, B.-I., KIM, S. y SAHOO, S. Waste collection vehicle routing problem with time windows. *Computers & Operations Research*, vol. 33(12), páginas 3624–3642, 2006.
- KIRKPATRICK, S., GELATT, C. D. y VECCHI, M. P. Optimization by simulated annealing. *Science (New York, N.Y.)*, vol. 220(4598), páginas 671–680, 1983. ISSN 0036-8075.
- KLEIN, R. *Resource-Constrained Scheduling Problems*, páginas 73–109. Springer US, Boston, MA, 2000. ISBN 978-1-4615-4629-0.
- KOLEN, A. W. J., KAN, A. H. G. R. y TRIENEKENS, H. W. J. M. Vehicle routing with time windows. *Operations Research*, vol. 35(2), páginas 266–273, 1987.

- KOVACS, A. A., GOLDEN, B. L., HARTL, R. F. y PARRAGH, S. N. The generalized consistent vehicle routing problem. *Transportation Science*, vol. 49(4), páginas 796–816, 2015.
- KYTOJOKI, J., NUORTIO, T., BRAYSY, O. y GENDREAU, M. An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. *Computers & Operations Research*, vol. 34(9), páginas 2743–2757, 2007.
- L, H.-K., M, S. y E, P.-L. Long-term care providers and services users in the united states: Data from the national study of long-term care providers, 2013-2014. *Vital Health Stat. National Center for Health Statistics.*, vol. 3(38), 2016.
- LABADIE, N., PRINS, C. y PRODHON, C. *Metaheuristics for Vehicle Routing Problems*. Wiley-Blackwell, 2016.
- LANGEVIN, A., DESROCHERS, M., DESROSIERS, J., GÉLINAS, S. y SOUMIS, F. A two-commodity flow formulation for the traveling salesman and the makespan problems with time windows. *Networks*, vol. 23(7), páginas 631–640, 1993.
- LANZARONE, E., MATTA, A. y SAHIN, E. Operations management applied to home care services: The problem of assigning human resources to patients. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 42(6), páginas 1346–1363, 2012. ISSN 1083-4427.
- LAPORTE, G. Fifty years of vehicle routing. *Transportation Science*, vol. 43(4), páginas 408–416, 2009.
- LAPORTE, G., NOBERT, Y. y DESROCHERS, M. Optimal routing under capacity and distance restrictions. *Operations Research*, vol. 33(5), páginas 1050–1073, 1985.
- LAPORTE, G., ROPKE, S. y VIDAL, T. Chapter 4: Heuristics for the vehicle routing problem. En *Vehicle Routing: Problems, Methods, and Applications, Second Edition*, páginas 87–116. Society for Industrial & Applied Mathematics (SIAM), 2014.
- LAPORTE, G. y SEMET, F. 5. classical heuristics for the capacitated VRP. En *The Vehicle Routing Problem*, páginas 109–128. Society for Industrial & Applied Mathematics (SIAM), 2002.
- LENSTRA, J. K. y KAN, A. H. G. R. Complexity of vehicle routing and scheduling problems. *Networks*, vol. 11(2), páginas 221–227, 1981.
- LI, F., GOLDEN, B. y WASIL, E. Very large-scale vehicle routing: new test problems, algorithms, and results. *Computers & Operations Research*, vol. 32(5), páginas 1165–1179, 2005.
- LIN, S. Computer solutions of the traveling salesman problem. *Bell System Technical Journal*, vol. 44(10), páginas 2245–2269, 1965.
- LIU, J. y HE, Y. A Clustering-Based Multiple Ant Colony System for the Waste Collection Vehicle Routing Problems. En *2012 Fifth International Symposium on Computational Intelligence and Design*. IEEE, 2012.

- LIU, R., XIE, X., AUGUSTO, V. y RODRIGUEZ, C. Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *European Journal of Operational Research*, vol. 230(3), páginas 475 – 486, 2013. ISSN 0377-2217.
- LIU, R., XIE, X. y GARAIX, T. Hybridization of tabu search with feasible and infeasible local searches for periodic home health care logistics. *Omega*, vol. 47, páginas 17 – 32, 2014. ISSN 0305-0483.
- LIU, R., YUAN, B. y JIANG, Z. Mathematical model and exact algorithm for the home care worker scheduling and routing problem with lunch break requirements. *International Journal of Production Research*, vol. 55(2), páginas 558–575, 2017.
- LÓPEZ-IBÁÑEZ, M. y BLUM, C. Beam-ACO for the travelling salesman problem with time windows. *Computers & Operations Research*, vol. 37(9), páginas 1570–1583, 2010.
- LÓPEZ-IBÁÑEZ, M., BLUM, C., OHLMANN, J. W. y THOMAS, B. W. The travelling salesman problem with time windows: Adapting algorithms from travel-time to makespan optimization. *Applied Soft Computing*, vol. 13(9), páginas 3806–3815, 2013.
- LUMER, E. D. y FAIETA, B. Diversity and adaptation in populations of clustering ants. En *Proceedings of the Third International Conference on Simulation of Adaptive Behavior : From Animals to Animats 3: From Animals to Animats 3*, SAB94, páginas 501–508. MIT Press, Cambridge, MA, USA, 1994. ISBN 0-262-53122-4.
- LYSGAARD, J., LETCHFORD, A. N. y EGGLESE, R. W. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, vol. 100(2), páginas 423–445, 2004.
- MACHNIK, L. Ants in text documents clustering. En *Advances in Systems, Computing Sciences and Software Engineering*, páginas 209–212. Springer Science Business Media, 2006.
- MANKOWSKA, D. S., MEISEL, F. y BIERWIRTH, C. The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, vol. 17(1), páginas 15–30, 2014. ISSN 1572-9389.
- MARCON, E. y CHAABANE, S. *Caregivers Routing Problem in Home Health Care: Literature Review*, páginas 319–326. Springer International Publishing, Cham, 2017. ISBN 978-3-319-51100-9.
- MASc, C. R. B., PHD y PHD, M. W. C. A decision tool for negotiating home care funding levels in ontario. *Home Health Care Services Quarterly*, vol. 25(3-4), páginas 91–106, 2006. PMID: 17062513.
- MENDELEYEV, D. I., KAMENSKY, G. y POPE, T. H. *The principles of chemistry by D. Mendeléeff*. Smithsonian Institution, 1905.

- MINGOZZI, A., BIANCO, L. y RICCIARDELLI, S. Dynamic programming strategies for the traveling salesman problem with time window and precedence constraints. *Operations Research*, vol. 45(3), páginas 365–377, 1997.
- MLADENOVIĆ, N. y HANSEN, P. Variable neighborhood search. *Computers & Operations Research*, vol. 24(11), páginas 1097–1100, 1997.
- MLADENOVIC, N., TODOSIJEVIC, R. y UROSEVIC, D. An efficient general variable neighborhood search for large travelling salesman problem with time windows. *Yugoslav Journal of Operations Research*, vol. 23(1), páginas 19–30, 2013.
- MONMARCHÉ, N., GUINAND, F. y SIARRY, P., editores. *Artificial Ants: from collective intelligence to real life optimization and beyond*. ISTE – Wiley, 2010.
- MOSCATO, P. y COTTA, C. A modern introduction to memetic algorithms. En *Handbook of Metaheuristics*, páginas 141–183. Springer Science + Business Media, 2010.
- N., C., A., M. y TOTH, P. The vehicle routing problem. En *Combinatorial Optimization*, páginas 315–338. Wiley, Chichester, 1979.
- NAGATA, Y. Edge assembly crossover for the capacitated vehicle routing problem. En *Evolutionary Computation in Combinatorial Optimization*, páginas 142–153. Springer Science + Business Media, 2007a.
- NAGATA, Y. Efficient evolutionary algorithm for the vehicle routing problem with time windows: edge assembly crossover for the VRPTW. En *2007 IEEE Congress on Evolutionary Computation*. Institute of Electrical and Electronics Engineers (IEEE), 2007b.
- NAGATA, Y. y BRÄYSY, O. Edge assembly based memetic algorithm for the capacitated vehicle routing problem. *Networks*, vol. 54(4), páginas 205–215, 2009. ISSN 00283045.
- NAGATA, Y. y BRAYSY, O. Edge assembly-based memetic algorithm for the capacitated vehicle routing problem. *Networks*, vol. 54(4), páginas 205–215, 2009.
- NAGATA, Y., BRAYSY, O. y DULLAERT, W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. *Computers & Operations Research*, vol. 37(4), páginas 724–737, 2010.
- NAGATA, Y. y KOBAYASHI, S. A powerful genetic algorithm using edge assembly crossover for the traveling salesman problem. *INFORMS J. on Computing*, vol. 25(2), páginas 346–363, 2013. ISSN 1526-5528.
- NECULA, R., BREABAN, M. y RASCHIP, M. Performance evaluation of ant colony systems for the single-depot multiple traveling salesman problem. En *Lecture Notes in Computer Science*, páginas 257–268. Springer Science Business Media, 2015a.
- NECULA, R., BREABAN, M. y RASCHIP, M. Tackling the bi-criteria facet of multiple traveling salesman problem with ant colony systems. En *2015 IEEE 27th International Conference on Tools with Artificial Intelligence*. Institute of Electrical Electronics Engineers, 2015b.

- NELSON, M. D., NYGARD, K. E., GRIFFIN, J. H. y SHREVE, W. E. Implementation techniques for the vehicle routing problem. *Computers & Operations Research*, vol. 12(3), páginas 273–283, 1985.
- NETO, R. T. y FILHO, M. G. Literature review regarding ant colony optimization applied to scheduling problems: Guidelines for implementation and directions for future research. *Engineering Applications of Artificial Intelligence*, vol. 26(1), páginas 150 – 161, 2013. ISSN 0952-1976.
- NICKEL, S., SCHRÖDER, M. y STEEG, J. Mid-term and short-term planning support for home health care services. *European Journal of Operational Research*, vol. 219(3), páginas 574–587, 2012. ISSN 03772217.
- NOVOTNY, V., BASSET, Y., MILLER, S. E., WEIBLEN, G. D., BREMER, B., CIZEK, L. y DROZD, P. Low host specificity of herbivorous insects in a tropical forest. *Nature*, vol. 416(6883), páginas 841–844, 2002. ISSN 0028-0836.
- OBERLIN, P., RATHINAM, S. y DARBHA, S. A transformation for a heterogeneous, multiple depot, multiple traveling salesman problem. En *2009 American Control Conference*. Institute of Electrical & Electronics Engineers (IEEE), 2009.
- OHIO, U. Hymenoptera Name Server. 2011. Disponible en http://osuc.biosci.ohio-state.edu/hymenoptera/tsa.sppcount?the_taxon=Formicidae (último acceso, June 15, 2011).
- OHLMANN, J. W. y THOMAS, B. W. A compressed-annealing heuristic for the traveling salesman problem with time windows. *INFORMS Journal on Computing*, vol. 19(1), páginas 80–90, 2007.
- PAESSENS, H. The savings algorithm for the vehicle routing problem. *European Journal of Operational Research*, vol. 34(3), páginas 336–344, 1988.
- PARDALOS, P. y RESENDE, M. *Handbook of Applied Optimization*. Oxford University Press, 2002. ISBN 9780195125948.
- PARTYKA, J. y HALL, R. Vehicle routing software survey. 2016. Disponible en <https://www.informs.org/ORMS-Today/Public-Articles/February-Volume-39-Number-1/Software-Survey-Vehicle-Routing> (último acceso, Junio, 2016).
- PASCHOS, V. T., editor. *Applications of Combinatorial Optimization*. Wiley-Blackwell, 2014.
- PECIN, D., PESSOA, A., POGGI, M. y UCHOA, E. Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation*, 2016.
- PESANT, G., GENDREAU, M., POTVIN, J.-Y. y ROUSSEAU, J.-M. An exact constraint logic programming algorithm for the traveling salesman problem with time windows. *Transportation Science*, vol. 32(1), páginas 12–29, 1998.

- POGGI, M. y UCHOA, E. Chapter 3: New exact algorithms for the capacitated vehicle routing problem. En *Vehicle Routing: Problems, Methods, and Applications, Second Edition*, páginas 59–86. Society for Industrial & Applied Mathematics (SIAM), 2014.
- POTVIN, J.-Y. A review of bio-inspired algorithms for vehicle routing. En *Bio-inspired Algorithms for the Vehicle Routing Problem* (editado por F. B. Pereira y J. Tavares), páginas 1–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009a. ISBN 978-3-540-85152-3.
- POTVIN, J.-Y. *A Review of Bio-inspired Algorithms for Vehicle Routing*, páginas 1–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009b. ISBN 978-3-540-85152-3.
- POTVIN, J.-Y., KERVAHUT, T., GARCIA, B.-L. y ROUSSEAU, J.-M. The vehicle routing problem with time windows part i: Tabu search. *INFORMS Journal on Computing*, vol. 8(2), páginas 158–164, 1996.
- PRINS, C. A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, vol. 31(12), páginas 1985–2002, 2004.
- QI, M., LIN, W.-H., LI, N. y MIAO, L. A spatiotemporal partitioning approach for large-scale vehicle routing problems with time windows. *Transportation Research Part E: Logistics and Transportation Review*, vol. 48(1), páginas 248 – 257, 2012. ISSN 1366-5545. Select Papers from the 19th International Symposium on Transportation and Traffic Theory.
- QUINTANA, D., CERVANTES, A., SAEZ, Y. y ISASI, P. Clustering technique for large-scale home care crew scheduling problems. *Applied Intelligence*, 2017. ISSN 1573-7497.
- QUINTIQ. Qiquintiq vrptw world record. 2017. Disponible en <http://www.quintiq.com/optimization/vrptw-world-records.html> (último acceso, Agosto, 2016).
- RASMUSSEN, M. S., JUSTESEN, T., DOHN, A. y LARSEN, J. The Home Care Crew Scheduling Problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, vol. 219(3), páginas 598–610, 2012. ISSN 03772217.
- REAKA-KUDLA M.L., W. E., WILSON D.E. *Biodiversity II: Understanding and Protecting Our Biological Resources*. Joseph Henry Press, 1997.
- REDJEM, R. y MARCON, E. Operations management in the home care services: a heuristic for the caregivers’ routing problem. *Flexible Services and Manufacturing Journal*, vol. 28(1), páginas 280–303, 2016. ISSN 1936-6590.
- REIMANN, M., STUMMER, M. y DOERNER, K. A savings based ant system for the vehicle routing problem. En *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '02*, páginas 1317–1326. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002. ISBN 1-55860-878-8.
- REINELT, G. TSPLIB- a traveling salesman problem library. *ORSA Journal of Computing*, vol. 3(4), páginas 376–384, 1991.

- RENDL, A., PRANDTSTETTER, M., HIERMANN, G., PUCHINGER, J. y RAIDL, G. Hybrid heuristics for multimodal homecare scheduling. En *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems* (editado por N. Beldiceanu, N. Jussien y É. Pinson), páginas 339–355. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. ISBN 978-3-642-29828-8.
- REST, K., TRAUTSAMWIESER, A. y HIRSCH, P. Trends and risks in home health care. *Journal of Humanitarian Logistics and Supply Chain Management*, vol. 2(1), páginas 34–53, 2012a.
- REST, K.-D. y HIRSCH, P. Daily scheduling of home health care services using time-dependent public transport. *Flexible Services and Manufacturing Journal*, vol. 28(3), páginas 495–525, 2016. ISSN 1936-6590.
- REST, K. D., TRAUTSAMWIESER, A. y HIRSCH, P. Trends and risks in home health care. *Journal of Humanitarian Logistics and Supply Chain Management*, vol. 2(1), páginas 34–53, 2012b.
- RIZZOLI, A. E., MONTEMANNI, R., LUCIBELLO, E. y GAMBARDELLA, L. M. Ant colony optimization for real-world vehicle routing problems. *Swarm Intelligence*, vol. 1(2), páginas 135–151, 2007.
- RODRÍGUEZ, A. y RUIZ, R. The effect of the asymmetry of road transportation networks on the traveling salesman problem. *Computers & Operations Research*, vol. 39(7), páginas 1566–1576, 2012. ISSN 03050548.
- ROPKE, S. Branching decisions in branch-and-cut-and-price algorithms for vehicle routing problems, 2012. International Workshop on Column Generation.
- ROSS, W. D. *Aristotle*. Methuen & Co., 1923.
- S., C. G. G., CRUZ-REYES, L., B., J. J. G., H., H. J. F., R., R. A. P. y P., J. J. M. Ant colony system with characterization-based heuristics for a bottled-products distribution logistics system. *Journal of Computational and Applied Mathematics*, vol. 259, páginas 965–977, 2014.
- SAVELSBERGH, M. W. P. Local search in routing problems with time windows. *Annals of Operations Research*, vol. 4(1), páginas 285–305, 1985.
- SCHRIJVER, A. On the history of combinatorial optimization (till 1960). En *Discrete Optimization*, páginas 1–68. Elsevier BV, 2005.
- SEMET, F. y TAILLARD, E. Solving real-life vehicle routing problems efficiently using tabu search. *Annals of Operations Research*, vol. 41(4), páginas 469–488, 1993.
- SEMET, F., TOTH, P. y VIGO, D. Chapter 2: Classical exact algorithms for the capacitated vehicle routing problem. En *Vehicle Routing: Problems, Methods, and Applications, Second Edition*, páginas 37–57. Society for Industrial & Applied Mathematics (SIAM), 2014.

- SEMIH, CAPPANERA, P., SCUTELLA, M. G., SAHIN, E. y MATTA, A. Pattern-based decompositions for human resource planning in home health care services. *Computers & Operations Research*, vol. 73(Supplement C), páginas 12 – 26, 2016a. ISSN 0305-0548.
- SEMIH, MATTA, A., SAHIN, E. y SHANTHIKUMAR, J. G. The patient assignment problem in home health care: using a data-driven method to estimate the travel times of care givers. *Flexible Services and Manufacturing Journal*, vol. 28(1), páginas 304–335, 2016b.
- SHAPIRO, S. S. y WILK, M. B. An analysis of variance test for normality (complete samples). *Biometrika*, vol. 52(3-4), páginas 591–611, 1965.
- SHI, W. y WEISE, T. An initialized aco for the vrptw. En *Proceedings of the 14th International Conference on Intelligent Data Engineering and Automated Learning — IDEAL 2013 - Volume 8206*, IDEAL 2013, páginas 93–100. Springer-Verlag New York, Inc., New York, NY, USA, 2013. ISBN 978-3-642-41277-6.
- SHIN, K. y HAN, S. A centroid-based heuristic algorithm for the capacitated vehicle routing problem. *Computing and Informatics*, vol. 30(4), páginas 721–732, 2011.
- DA SILVA, R. F. y URRUTIA, S. A general VNS heuristic for the traveling salesman problem with time windows. *Discrete Optimization*, vol. 7(4), páginas 203–211, 2010.
- SINTEF. Sintef pdptw pick up and delivery problem with time windows. 2017. Disponible en <http://www.sintef.no/projectweb/top/vrptw/homberger-benchmark> (último acceso, Agosto, 2016).
- SOLOMON, M. M. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, vol. 35(2), páginas 254–265, 1987.
- SOLUTIONS, S. Spider. 2016. Disponible en <http://spidersolutions.no/english/about-spider/theory-and-practise> (último acceso, Agosto, 2016).
- STÜTZLE, T. y DORIGO, M. A Short Convergence Proof for a Class of Ant Colony Optimization Algorithms. 2006.
- STÜTZLE, T. y HOOS, H. MAX-MIN Ant System and local search for the traveling salesman problem. En *Evolutionary Computation, 1997., IEEE International Conference on*, páginas 309–314. 1997.
- SWAGATAM DAS, A. A. y KONAR, A. *Metaheuristic Clustering*. Springer Berlin Heidelberg, 2009.
- SWAGATAM DAS, A. K., AJITH ABRAHAM. *Metaheuristic Clustering*. Springer Berlin Heidelberg, 2009. ISBN 978-3-540-92172-1.
- TAILLARD, É., BADEAU, P., GENDREAU, M., GUERTIN, F. y POTVIN, J.-Y. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, vol. 31(2), páginas 170–186, 1997.
- TALBI, E.-G. *Metaheuristics*. Wiley-Blackwell, 2009.

- THANGIAH, S., NYGARD, K. y JUELL, P. GIDEON: a genetic algorithm system for vehicle routing with time windows. En *[1991] Proceedings. The Seventh IEEE Conference on Artificial Intelligence Application*. Institute of Electrical and Electronics Engineers (IEEE), 1991.
- THANGIAH, S. R. An adaptive clustering method using a geometric shape for vehicle routing problems with time windows. En *Proceedings of the Sixth International Conference on Genetic Algorithms* (editado por L. Eshelman), páginas 536–543. Morgan Kaufmann, San Francisco, CA, 1995.
- TRAUTSAMWIESER, A. y HIRSCH, P. Optimization of daily scheduling for home health care services. *Journal of Applied Operational Research*, vol. 3, páginas 124–136, 2011.
- UCHOA, E., PECIN, D., PESSOA, A., POGGI, M., VIDAL, T. y SUBRAMANIAN, A. New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research*, 2016.
- UNITED NATIONS. Political declaration and Madrid international plan of action on ageing. 2002. Disponible en http://www.un.org/en/events/pastevents/pdfs/Madrid_plan.pdf (último acceso, Online; accessed 24 de septiembre de 2018).
- UNITED NATIONS. World economic and social survey, 2007: development in an ageing world. *Choice Reviews Online*, vol. 45(04), páginas 45–2157–45–2157, 2007.
- UNITED NATIONS. The 2015 revision of the un’s world population projections. *Population and Development Review*, vol. 41(3), páginas 557–561, 2015. ISSN 1728-4457.
- UTAH, U. O. The oldest homo sapiens: Fossils push human emergence back to 195,000 years ago. 2005. Disponible en <http://www.sciencedaily.com/releases/2005/02/050223122209.htm> (último acceso, June 15, 2017).
- VIDAL, T., CRAINIC, T. G., GENDREAU, M., LAHRICHI, N. y REI, W. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research*, vol. 60(3), páginas 611–624, 2012.
- WAGSTAFF, K. y CARDIE, C. Clustering with instance-level constraints. En *Proceedings of the Seventeenth International Conference on Machine Learning, ICML ’00*, páginas 1103–1110. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000. ISBN 1-55860-707-2.
- WAGSTAFF, K., DAVIDSON, I. y BASU, S., editores. *Constrained Clustering*. Chapman and Hall/CRC, 2008.
- WARD, J. H. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, vol. 58(301), páginas 236–244, 1963.
- WARK, P. y HOLT, J. A repeated matching heuristic for the vehicle routing problem. *J Oper Res Soc*, vol. 45(10), páginas 1156–1167, 1994.
- WILSON, E. *The future of life*. Alfred A. Knopf, 2002. ISBN 9780679450788.

- WILSON, E. O. *The Insect Societies*. Belknap Press of Harvard University Press, 1971.
- WIRNITZER, J., HECKMANN, I., MEYER, A. y NICKEL, S. Patient-based nurse rostering in home care. *Operations Research for Health Care*, vol. 8(Supplement C), páginas 91 – 102, 2016. ISSN 2211-6923.
- WORLD HEALTH ORGANIZATION. Active Ageing: A Policy Framework. 2002. Disponible en http://apps.who.int/iris/bitstream/10665/67215/1/WHO_NMH_NPH_02.8.pdf (último acceso, Online; accessed 24 de septiembre de 2018).
- WORLD HEALTH ORGANIZATION. Home care across Europe, current structure and future challenges. European Observatory on Health Systems and Policies. 2012. Disponible en http://www.euro.who.int/__data/assets/pdf_file/0008/181799/e96757.pdf (último acceso, Online; accessed 24 de septiembre de 2018).
- WORLD HEALTH ORGANIZATION. Global strategy on people-centred and integrated health services. 2015. Disponible en <http://www.who.int/servicedeliverysafety/areas/people-centred-care/global-strategy/en/> (último acceso, Junio, 2015).
- WORLD HEALTH ORGANIZATION. World report on ageing and health. 2015. Disponible en http://apps.who.int/iris/bitstream/10665/186463/1/9789240694811_eng.pdf (último acceso, Online; accessed 24 de septiembre de 2018).
- WORLD HEALTH ORGANIZATION. The global strategy and action plan on ageing and health 2016–2020: towards a world in which everyone can live a long and healthy life. 2016. Disponible en http://apps.who.int/gb/ebwha/pdf_files/WHA69/A69_R3-en.pdf (último acceso, Agosto, 2015).
- YU, B. y YANG, Z. Z. An ant colony optimization model: The period vehicle routing problem with time windows. *Transportation Research Part E: Logistics and Transportation Review*, vol. 47(2), páginas 166 – 181, 2011. ISSN 1366-5545.
- ZHOU, R., NEE, A. y LEE, H. Performance of an ant colony optimisation algorithm in dynamic job shop scheduling problems. *International Journal of Production Research*, vol. 47(11), páginas 2903–2920, 2009.