

Creación de celdas cooperativas en un grupo de tiendas de una misma franquicia

Manuel Mateo

UPC, Barcelona, España, manel.mateo@upc.edu

María Rocío De La Torre

UPC, Barcelona, España, maria.rocio.de.torre@upc.edu

Mayra D'Armas

UNEXPO, Puerto Ordaz, Venezuela, mdarmas@unexpo.edu.ve

RESUMEN

En este trabajo se resuelve un problema de creación de celdas cooperativas en un grupo de tiendas que forman parte de una misma franquicia, que permitan evitar las posibles roturas de stocks derivadas de incrementos puntuales de la demanda. El problema de las celdas cooperativas se inscribe en el marco del *Vehicle Routing Problem* (VRP). Para la resolución del problema se aplicaron dos procedimientos basados en el estudio de las distancias entre los componentes de las celdas. El primero (GCSI) sigue dos fases: en primer lugar se genera una solución inicial intuitiva y posteriormente se aplica un procedimiento de mejora fundamentado en el *Variable Neighborhood Search* (VNS) con aportaciones propias del *Multi-Start Local Search* (MSLS). En el segundo (GCGRASP) se aplica un GRASP. Los dos procedimientos se experimentaron en un caso real que consta de 116 tiendas, todas pertenecientes a la misma cadena de franquicias y ubicadas en la zona sur de Inglaterra. Los resultados demuestran que el procedimiento GCGRASP puede alcanzar valores de la distancia media por celda inferiores a los que se obtienen aplicando el procedimiento GCSI. No obstante, los tiempos de cálculo en aplicar el procedimiento GCSI son muy inferiores a los tiempos obtenidos para el procedimiento GCGRASP.

Palabras claves: Vehicle Routing Problem, Celdas Cooperativas, GRASP, Variable Neighborhood Search.

ABSTRACT

This paper solves a problem of creation of cooperative cells in a group of sales-points within a franchise, in order to avoid any stockout when the demand suddenly increases. This problem of cooperative cells belongs to the Vehicle Routing Problem (VRP). Two procedures were applied to solve the problem, both based on the distances between the components of the cells. The first one (GCSI) follows two phases: first, an intuitive initial solution is generated and then a local search, based on the Variable Neighborhood Search (VNS) with some ideas from Multi-Start Local Search (MSLS). In the second one (GCGRASP) a GRASP is applied. Both procedures were tested on a real case with 116 sales-points in the same franchise chain and located in the south of England. The results demonstrate that GCGRASP can reach a mean distance per cell lower than using GCSI. However, the computing times with GCSI are much lower than those from GCGRASP.

Keywords: Vehicle Routing Problem, Cooperative Cells, GRASP, Variable Neighborhood Search.

1. INTRODUCCIÓN

En este estudio se presenta una propuesta de creación de celdas cooperativas en un grupo de tiendas que forman parte de una misma franquicia. Una celda cooperativa supone establecer unos acuerdos especiales entre tiendas para que si en alguna o en varias de ellas tiene lugar un incremento de la demanda las que dispongan del exceso de stock de otras tiendas de la celda sin necesidad de esperar un nuevo lote procedente del almacén central. El objetivo es determinar cuáles deben ser estas celdas cooperativas (cuántas y qué tiendas las integran) para paliar

las posibles roturas de stocks derivadas de incrementos puntuales de la demanda.

La actual política de gestión de stocks de las tiendas es una mezcla de aprovisionamiento periódico, ya que se sirve cada determinado período T , y de gestión por punto de pedido, ya que si se llega a cierto nivel de stock antes de alcanzar el período T se realiza un pedido extra. Este sistema puede funcionar de forma eficiente en cuanto a costes siempre y cuando no se produzcan picos puntuales de demanda. En cambio, si se produce esta última situación, las tiendas se encuentran en dificultades para satisfacer la demanda, teniendo dos posibilidades: o bien realizar un pedido al almacén más cercano y que éste llegue urgentemente o esperar a que se sirva el siguiente pedido, con lo cual es muy fácil perder posibles nuevas ventas.

Por este motivo se diseñan las llamadas celdas cooperativas, que serán subconjuntos de puntos de venta cercanos, que establecerán relaciones de intercambio de mercancías en determinadas situaciones para evitar rupturas de stocks (si aún se tardará en recibir el siguiente pedido). Las celdas cooperativas se crearán mediante dos procedimientos basados en el estudio de las distancias entre los componentes de las celdas. El primero de ellos sigue dos fases: en primer lugar se aplica una heurística adaptada al problema y con posterioridad se aplica un procedimiento de mejora sobre la solución alcanzada en la fase anterior. En el segundo se aplica un procedimiento GRASP, es decir, un procedimiento metaheurístico.

2. EL VEHICLE ROUTING PROBLEM

El problema de las celdas cooperativas se inscribe en el marco del *Vehicle Routing Problem* (VRP), problema inmensamente tratado en la literatura y del cual encontramos varias definiciones de referencia (Angelelli y Speranza, 2002; Kytöjoki et al., 2007; Goel y Gruhn, 2007).

El VRP se puede definir como el problema de determinar de manera óptima las rutas entre una serie de puntos en los cuales se entrega o se recoge productos. De entre estos puntos destaca uno llamado depósito, siendo el resto de puntos los clientes que se encuentran dispersos geográficamente.

Sea $G = (V, E)$ un grafo no orientado donde V es el conjunto de los $n+1$ nodos y E un conjunto de aristas con que tienen un valor de tiempo asociado. De entre los $n+1$ nodos, una vez descartado el depósito, cada nodo i de los n restantes presenta una demanda q_i . Las rutas entre el depósito y los clientes dependen de las órdenes de demanda que planteen los clientes. La visita de cada uno de los clientes en una ruta se realiza por un vehículo o conjunto de vehículos de capacidad Q , donde la cantidad servida en la ruta no debe superar la capacidad del vehículo $\sum q_i \leq Q$. En cuanto a la capacidad de los vehículos, se puede disponer de una flota de vehículos homogénea o heterogénea para atender las solicitudes de transporte, es decir con igual o diferente capacidad entre ellos. No obstante, como las rutas son diferentes, aunque los vehículos sean iguales la carga puede variar de acuerdo a las exigencias de cada conjunto de demandas.

Una solución es una ruta de transporte, donde se especifica mediante un conjunto no vacío los puntos de recogida, entrega y/o centros de servicio (almacenes) que deben ser visitados durante la secuencia (ruta determinada). Es decir, la ruta se inicia en una ubicación inicial, normalmente un depósito o almacén, y llega hasta una ubicación final que puede ser o no la inicial, pasando por los puntos que pertenecen a cada una de las solicitudes de transporte asignadas el vehículo. En nuestro caso, consideraremos que la ubicación final coincide con la inicial. También llamaremos a los lugares de entrega del producto tiendas o puntos de servicio.

El VRP es un problema del tipo NP (Lenstra y Rinnooy Kan, 2006). Para resolverlo se han propuesto gran cantidad de algoritmos, básicamente heurísticas y metaheurísticas. Los algoritmos exactos sólo tienen capacidad para resolver problemas de hasta unos 50 clientes en un tiempo de resolución razonable (Toth y Vigo, 2002).

Los procedimientos utilizados para la resolución del VRP se pueden dividir en tres categorías diferentes según Kytöjoki et al. (2007): heurísticas de construcción, heurísticas de mejora y metaheurísticas.

- Las heurísticas de construcción generan soluciones factibles mediante la realización de secuencias de

movimientos que minimicen o maximicen un determinado criterio (Renaud et al., 1996).

- Las heurísticas de mejora buscan optimizar de manera iterativa una solución factible generada por una heurística de construcción, mediante la reubicación de nodos y clientes en las diferentes iteraciones. Las iteraciones de mejora se llevan a cabo siempre y cuando se produzcan mejoras en el valor de la función objetivo (Ergun et al., 2004). Una de las heurísticas que se empleará en la determinación de las celdas cooperativas en este estudio es una adaptación de la heurística propuesta por Savelsbergh (1991), que estaría dentro de este grupo.
- Las metaheurísticas, entendidas como procedimientos de resolución más complejos que las heurísticas de construcción y mejora, se basan en procedimientos iterativos que intentan hallar soluciones mejores en un menor tiempo. Diversos autores han propuesto la aplicación de metaheurísticas al VRP: por ejemplo, Mester y Bräysy (2005) aplican *Tabu Search* (TS); Cordeau et al. (2002) prefieren *Greedy Randomized Adaptive Search Procedure* (GRASP); Baker y Ayecrew (2003) emplean algoritmos genéticos; Tang et al (2010) optan por *Scatter Search*.

Como se ha comentado, existen numerosas referencias a este problema, que presenta gran cantidad de variantes. Algunas de estas (*Inventory Routing Problem, Dynamic Vehicle Routing, Capacitated Routing Problem, Orienteering Problem*), así como los diferentes procedimientos de resolución, son tratadas en Golden et al (2008).

3. PROCEDIMIENTOS DE RESOLUCIÓN

3.1 HEURÍSTICAS BASADAS EN LA BÚSQUEDA LOCAL

Un problema de optimización puede formularse de la siguiente manera: $\min \{f(s) \mid s \in S, S \subset X\}$, siendo X el espacio de soluciones, S la región factible, s una solución factible y f una función real. Alrededor de dicha solución s se define un entorno $N(s)$. Cada $y \in N(s)$ será una solución vecina de s . Una vecindad de una solución s es un conjunto $N(s) \subset X$. Cada solución $s' \in N(s)$ se determina a partir de una solución previa s . Normalmente, dos soluciones vecinas difieren sólo en unos pocos elementos. Un vecindario, en ocasiones, también puede contener soluciones factibles que no pertenezcan a X . Se define una solución s^* como óptimo local respecto a un vecindario determinado $N(s)$ si $f(s^*) \leq f(s), \forall s \in N(s^*)$ (Anderson, 1992).

Uno de los procedimientos empleados para resolver problemas de optimización es la búsqueda local, conocida como *Local Search* (LS). Esta permite mejorar la solución encontrada, aplicando una heurística de mejora de forma iterativa. Consiste en ir sustituyendo de manera sucesiva la solución en curso por otra que mejore las condiciones de evaluación en el contexto de un procedimiento *greedy*. Este procedimiento finaliza cuando el valor de la función a evaluar en las nuevas soluciones vecinas no mejora, es decir, no hay mejor solución en el vecindario actual.

El principal problema que presenta esta heurística es que una vez se llega a un óptimo local la búsqueda no continúa, siendo necesario generar otra solución inicial y repetir de nuevo el proceso si se quiere mejorar la solución obtenida. Este es uno de los motivos por los que se presenta como alternativa algunas metaheurísticas, entre las que podemos destacar *Variable Neighborhood Search* (VNS) por su simplicidad y efectividad (Pelegrín-Pelegrín 2004).

3.2 MULTI-START LOCAL SEARCH (MSLS) Y VARIABLE NEIGHBORHOOD SEARCH (VNS)

La búsqueda de nuevos vecindarios puede ser implementada, entre otros, usando los procedimientos *Multi-Start Local Search* (MSLS) o bien *Variable Neighborhood Search* (VNS).

La MSLS resulta adecuada en aquellos casos en que los óptimos locales se distribuyen al azar en el espacio de soluciones. Sin embargo, obtener el óptimo global utilizando este procedimiento es más complicado, ya que es

necesario generar gran cantidad de soluciones aleatorias que conduzcan a él en las iteraciones de la búsqueda (Pelegrín-Pelegrín, 2004).

Para mejorar el esquema de la LS, con el objetivo de que el algoritmo no se quede en un óptimo local, se presenta también VNS. Se trata de estudiar varias estructuras de entorno, ya que un óptimo local para un determinado entorno no ha de serlo necesariamente para otro. Así pues, la búsqueda podrá continuar hasta obtener una solución que sea óptimo local para todas las estructuras del entorno consideradas, es decir, llegar al óptimo global (Pelegrín-Pelegrín, 2004).

En cuanto al problema de estudio, se sigue esta segunda estrategia, es decir, se investigan sucesivos vecindarios generados a partir de la solución en curso. Con este procedimiento, se estudia un mayor número de entornos, es decir, se generan vecindarios que permite cubrir una mayor superficie del espacio de soluciones.

3.3 PROCEDIMIENTOS GRASP

Dentro de las diferentes metaheurísticas, se ha optado por aplicar un *greedy randomized adaptative search procedure* (GRASP). La metaheurística GRASP (Feo y Resende, 1989; Resende y Gonzalez Velarde, 2003; Resende y Ribeiro, 2008; Resende y Silva, 2009) se basa en múltiples inicios para problemas de optimización combinatoria en los cuales se utiliza heurísticas de resolución *greedy* para su resolución. La introducción de la aleatoriedad permite diversificar el espacio de exploración. Cada iteración en un GRASP se compone básicamente de dos fases: la construcción de una solución inicial y la búsqueda local a partir de ésta.

En la fase de construcción se determina una solución factible y en la fase de búsqueda local se investigan diferentes soluciones vecinas de la solución inicial hasta llegar a un mínimo local. La mejor solución global se mantiene como resultado. En la construcción de la solución parcial, el conjunto de elementos que se pueden incorporar se consideran candidatos. El elemento seleccionado se determina a partir de la evaluación de todos los candidatos de acuerdo a una función de evaluación. Se crea una lista de posibles candidatos, cuya incorporación a la solución parcial implica una mejora en la función objetivo. De entre este listado, se selecciona al azar el elemento que finalmente se incorpora a la solución parcial. Una vez dicho elemento se ha incorporado a la solución parcial, se actualiza el conjunto de elementos candidatos y son revaluados los costes adicionales.

4. GENERACIÓN DE CELDAS COOPERATIVAS

Para determinar las agrupaciones de tiendas, que llamaremos celdas cooperativas, únicamente se han tenido en cuenta los costes asociados al transporte. Así pues, el proceso de formación de las celdas se ha establecido en base a un criterio geográfico, a partir de las distancias a recorrer para de satisfacer la demanda de cada punto de servicio o tienda.

Para la resolución del problema de generación de celdas cooperativas, se presentan dos métodos. El primero de ellos se basa en determinar una solución inicial intuitiva. Así pues, en la primera fase se establecen los criterios para la determinación de la solución inicial del conjunto de celdas y en la segunda se desarrolla el procedimiento de mejora de la solución inicial encontrada. El segundo método supone la aplicación de un algoritmo GRASP, cuya fase de mejora coincide con la del primer método.

Una solución se evalúa mediante la distancia a recorrer por parte del sistema de transporte de la compañía para servir a todos los integrantes presentes en cada celda cooperativa. Se entiende que la distribución, partiendo de los depósitos centrales de la franquicia, se realiza mediante una o varias rutas de transporte que también han sido optimizadas.

Dado N el número de total de tiendas o puntos de servicio, C es el número de celdas cooperativas que se forman al empaquetar el total de N tiendas. Para cada celda cooperativa c ($c=1, \dots, C$), se define N_c como número de tiendas de dicha celda y T_c como el conjunto de tiendas que forman parte de la misma. Por cuestiones prácticas de

funcionamiento interno de la celda, se establece que el valor máximo de integrantes de una celda será de ocho y el valor mínimo, de dos: $N_c = \{2, \dots, 8\}$. Se define d_c como la distancia media de una celda c :

$$d_c = \frac{2}{N_c \cdot (N_c - 1)} \sum_{i=1}^{N_c-1} \sum_{j=i+1}^{N_c} d_{i,j}$$

Así pues, se evalúa y se pretende minimizar la siguiente función z que expresa la media de todas las distancias medias de las C celdas:

$$z = \frac{1}{C} \sum_{c=1}^C \left[\frac{2}{N_c \cdot (N_c - 1)} \sum_{i=1}^{N_c-1} \sum_{j=i+1}^{N_c} d_{i,j} \right]$$

siendo $d_{i,j}$ la distancia entre dos tiendas i y j , que en este caso formarán parte del conjunto T_c de la celda c .

Además, debe considerarse la siguiente restricción del número de tiendas por celda:

$$2 \leq N_c \leq 8 \quad c=1, \dots, C$$

4.1 PROCEDIMIENTO DE GENERACIÓN DE LAS CELDAS CON SOLUCIÓN INICIAL INTUITIVA (GCSI)

4.1.1. FASE 1: DETERMINACIÓN DE LA SOLUCIÓN INICIAL

Esta fase del procedimiento GCSI tiene como objetivo la determinación del conjunto de las celdas cooperativas que conformarán una solución inicial. Para ello se aplica una heurística que sigue unos principios intuitivos y que analiza los datos de demanda como si esta fuera determinista.

Se define como paquete el conjunto de tiendas que forman parte de la misma celda. Se propone un empaquetamiento de las tiendas atendiendo a su proximidad geográfica y se imponen restricciones en el número de integrantes que formarán las celdas.

Esta parte del procedimiento sigue tres pasos:

- Se forman grupos entre las diferentes tiendas que se encuentran en la misma población.
- Se agrupan aquellas que se encuentran en municipios diferentes, pero separadas por la menor distancia posible.
- Finalmente, se asignan aquellas que han quedado separadas del resto de grupos debido a su aislamiento geográfico.

4.1.2. FASE 2: PROCEDIMIENTO DE MEJORA

Debido a que las soluciones generadas en la fase 1 no son necesariamente óptimas, a continuación se desarrolla el procedimiento de mejora. La exploración del entorno de las soluciones creadas al aplicar una serie de cambios con el objetivo de alcanzar una solución con menor valor en la función objetivo z . El algoritmo desarrollado en esta fase se fundamenta en el procedimiento VNS con aportaciones propias del MSLS.

Los puntos de servicio que conforman las celdas se modificarán en base a una adaptación a este sistema de estudio por parte del algoritmo para creación de rutas de distribución propuesto por Savelsbergh (1991). A diferencia del problema donde se aplica inicialmente el algoritmo, donde los clientes están agrupados en rutas, en el presente estudio se ha adaptado el algoritmo para resolver un sistema de celdas cooperativas.

Savelsbergh (1991) expone tres funciones para la recolocación de los clientes que conforman una ruta, buscando minimizar la distancia a recorrer en servir a todos los integrantes de la ruta de distribución. Las tres funciones del

algoritmo de Savelsbergh son las siguientes: *Relocate*, *Exchange* y *Cross*. En el caso de la primera función, *Relocate*, se basa en la reubicación de un integrante de una ruta que se encuentra en la ruta $r1$ a otra ruta vecina $r2$. En el caso de la función *Exchange*, se intercambian de manera simultánea los nodos integrantes de dos rutas distintas $r1$ y $r2$. Finalmente, la función *Cross* evita el solapamiento de dos rutas $r1$ y $r2$.

En nuestro caso, en lugar de considerar ramificaciones lineales desde el depósito (rutas de distribución), los parámetros de evaluación de la mejora se especifican para un diseño en red (celdas cooperativas). Se propone un reagrupamiento de las tiendas de acuerdo a la citada función *Relocate*. Esta función ahora se basará en la reubicación de un cliente i que forma parte de una celda $c1$ a otra celda vecina $c2$. El movimiento se realiza si la distancia de la tienda al centro de gravedad de su propia celda es mayor que la distancia al centro de gravedad de la celda alternativa, de manera similar a lo que propone Savelsbergh (1991).

En la Figura 1, se presenta un esquema donde se observa la situación de partida (a la izquierda la tienda pintada a rayas forma parte de la celda $c1$; a la derecha la misma tienda forma parte de la celda $c2$).

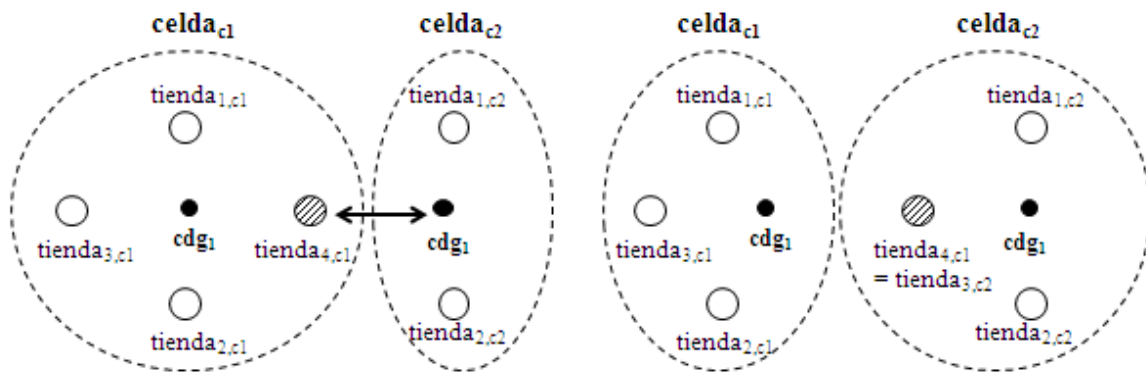


Figura1: Esquema de la función *Relocate*

En nuestro caso, la función *Exchange* intercambia uno de los elementos de una celda $c1$ por otro elemento que pertenece a otra celda vecina $c2$. Finalmente, la función *Cross* evita que el área de influencia de una celda $c1$ se superponga al de otra celda vecina $c2$. Esta última función tiene sentido si la solución inicial se genera al azar, pero en cambio no tiene tanto si se opta por una solución inicial basada en distancias.

4.2 PROCEDIMIENTO DE GENERACIÓN DE LAS CELDAS MEDIANTE GRASP (GCGRASP)

Este procedimiento halla una solución inicial, sobre la cual se aplicará un procedimiento de mejora de la solución análogo al Procedimiento GCSI. Por ello, no se detallará de nuevo en este apartado.

Para determinar una solución inicial, se realiza los siguientes pasos:

1. Utilizando el procedimiento *random*, se determina el número de puntos de servicio que formarán cada una de las celdas, el cual oscila entre ocho y dos como ya se comentado.
2. Se aplica de nuevo el procedimiento *random* para determinar cuántos candidatos se consideran cada vez que un nuevo punto de servicio debe integrarse a una de las celdas.
3. Del listado de tiendas no asignadas, se selecciona una de ellas al azar.
4. Una vez la celda contiene esta primera tienda, se evalúa la función de distancias medias recorridas por las tiendas como resultado de incluir esta nueva tienda candidata. De entre todas las tiendas aún no incluidas en la celda, se seleccionan las cuatro con menor valor en la función evaluada.

- Entre las cuatro tiendas candidatas a formar parte de la celda, se selecciona una de ellas al azar, que pasará a formar parte de dicha celda. Al ser seleccionada para formar parte de una celda, se elimina de la lista de tiendas no asignadas.
- Dado el número de elementos elegido en el paso 1, si la celda aún no está completa, se vuelve al paso 4 (y se sigue con el paso 5) hasta completar dicho número de elementos. Si ya se ha completado la celda, se continúa en el paso 7.
- Si aún no se han asignado a las celdas todas las tiendas, o sea N , se vuelve al paso 1. Por tanto, a la hora de crear una nueva celda, se repite el procedimiento desde el paso 1, aunque en esta nueva iteración se han eliminando del listado inicial de tiendas aquellas ya asignadas.

5. RESULTADOS

El sistema de estudio consta de 116 tiendas, una de las cuales hace las veces de depósito (punto desde el que se sirve la mercancía al resto de establecimientos) y punto de venta, todas ellas pertenecen a la misma cadena de franquicias y están ubicadas en la zona sur de Inglaterra. Para la resolución del problema se ha seleccionado un solo tipo de vehículo de transporte, es decir, la franquicia de estudio cuenta para la distribución de mercancías con una flota de vehículos homogénea.

En los siguientes apartados se presentan los resultados obtenidos en la creación de celdas cooperativas mediante los procedimientos detallados en los apartados 4.1 y 4.2 respectivamente.

5.1 RESULTADOS OBTENIDOS DE LA APLICACIÓN DEL PROCEDIMIENTO GCSI

En este apartado se presentan los resultados obtenidos en aplicar el procedimiento GCSI para el empaquetamiento de las tiendas en celdas cooperativas. La solución inicial, que se determina mediante la heurística intuitiva (Sección 4.1.1), presenta una agrupación de celdas con una distancia media entre tiendas por celda de 13,982 Km. A partir de esta solución se aplica el procedimiento de mejora (Sección 4.1.2).

La Figura 2 representa la evolución de los resultados alcanzados. En ella se observa para cada iteración la mejor solución obtenida hasta el momento, que siempre sigue una tendencia descendente ya que sólo se considera la mejor solución hasta ese momento. Cada escalón de la Figura 2 corresponde a una iteración en que se produce la mejora del valor de la solución. La solución inicialmente obtenida (distancia media a recorrer entre los clientes de una celda de 13,982 km) puede verse reducida, después de aplicar el procedimiento de mejora, hasta un valor de la de 13,7667 km.

La Tabla 1 indica el mejor resultado obtenido, así como el tiempo de ejecución necesario, para el procedimiento GCSI.

Tabla 1: Mejores resultados de la ejecución del procedimiento 1

Procedimiento	z (km)	Celdas	Tiempo cálculo (s)
GCSI	13,77	31	2,13

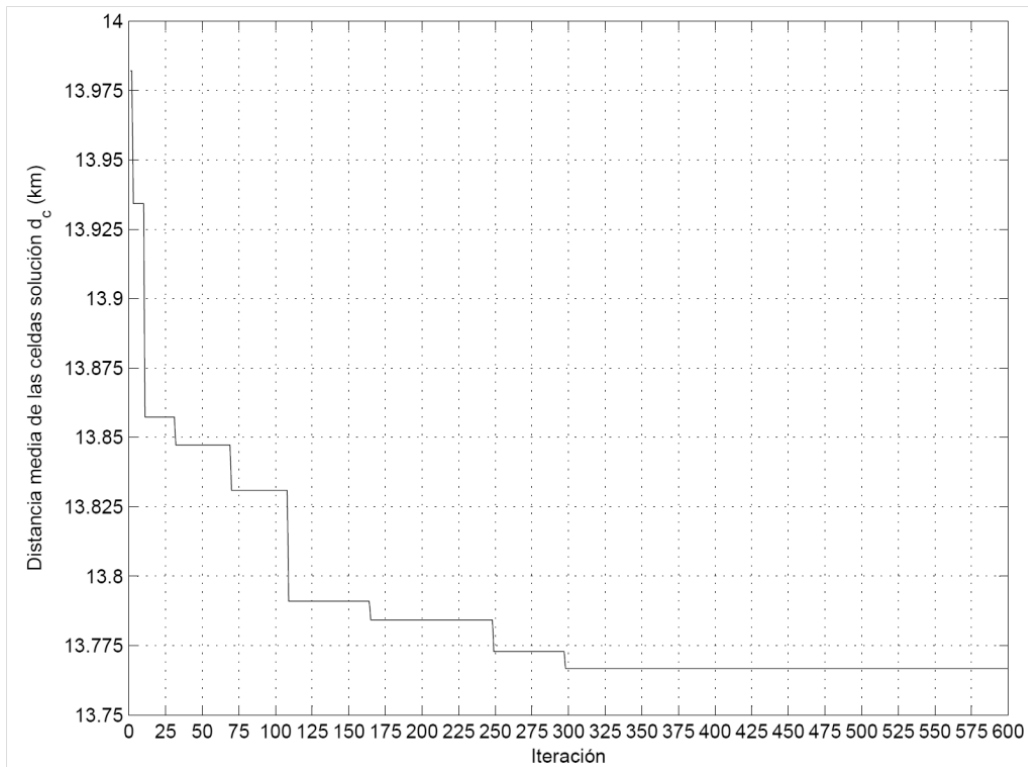


Figura 2: Resultado computacional del procedimiento GCSI

5.2 RESULTADOS OBTENIDOS DE LA APLICACIÓN DEL PROCEDIMIENTO GCGRASP

En este apartado se presentan los resultados obtenidos en aplicar el procedimiento GCGRASP para el empaquetamiento de las tiendas en celdas cooperativas. Tal y como se explicada en detalle en la Sección 4.2, el procedimiento consiste en hallar la solución inicial mediante la metaheurística GRASP y su posterior mejora. Se realiza este procedimiento global un total de n_{rep} iteraciones. Se han probado diferentes valores de n_{rep} , siendo los más representativos los que se hallan entre 40000 y 60000. En cada iteración se explora hasta un máximo de n_{iter}^{MAX} vecindarios de la solución en curso, que puede diferir de la mejor solución hallada hasta el momento.

La Figura 3 indica los resultados después de realizar las 3 réplicas más relevantes del algoritmo. El valor de la cota superior corresponde a la distancia media de la solución de partida. Los resultados obtenidos están reflejados en la Tabla 2.

Tabla 2: Mejores resultados de la ejecución del procedimiento 2

Ejecución	z (km)	Celdas	n_{iter}^{MAX}	n_{rep}	Tiempo de cálculo (min)
1	8,57	22	1000	40000	66,65
2	7,97	25	1000	50000	84,60
3	8,47	22	1000	60000	95,04

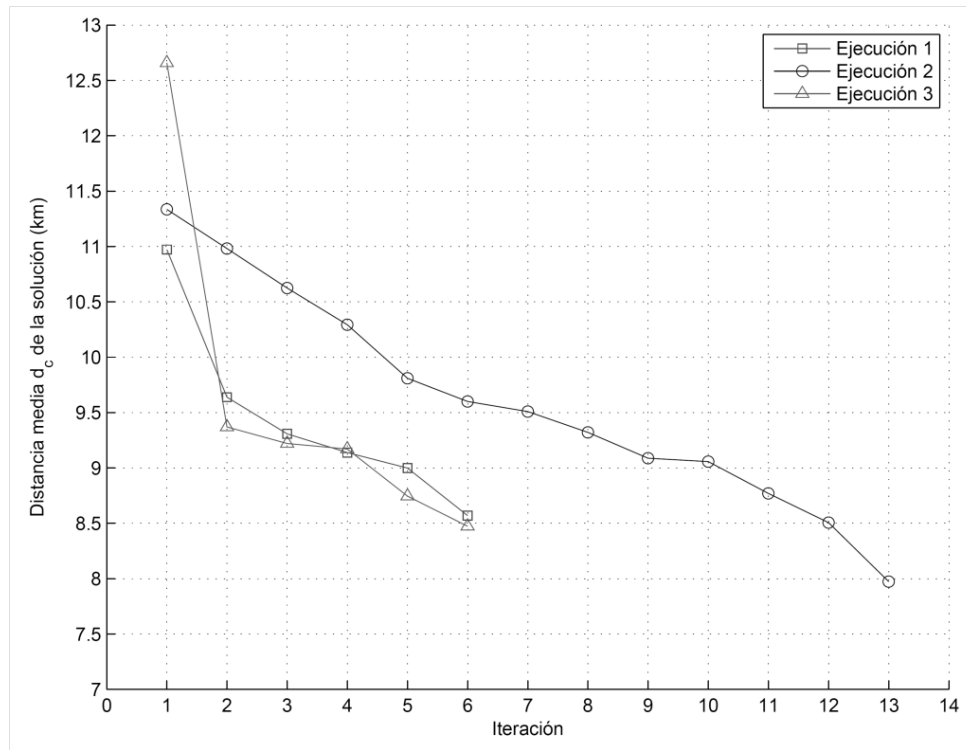


Figura 3: Resultados computacionales del procedimiento GCGRASP

6. CONCLUSIONES

El problema planteado consiste en la agrupación de tiendas de una franquicia para compartir stocks en caso de un incremento puntual de la demanda en alguna de las tiendas. Para ello se crean unas celdas cooperativas, que procuran evitar la rotura de stocks mediante intercambio del producto entre tiendas. Esta problemática se aplicó en un caso real, que consta de 116 tiendas, incluido un depósito. Por razones prácticas, se fijó el tamaño mínimo de la celda en 2 integrantes y el máximo en 8. Además, se cuenta con una flota de vehículos homogéneos, lo que indica que todas las celdas puedan ser consideradas igual. El criterio de evaluación de una solución es la distancia media recorrida entre cualquier par de tiendas que componen una celda cooperativa.

De los resultados obtenidos al aplicar los dos procedimientos descritos para la determinación de estas celdas, se extraen las conclusiones siguientes:

- El procedimiento GCSI es menos sofisticado, ya que la solución inicial se determina mediante una heurística intuitiva muy sencilla. En el procedimiento GCGRASP la aleatoriedad utilizada en un GRASP facilita mayor variedad de soluciones iniciales, alcanzando finalmente mejores resultados.
- El procedimiento GCGRASP puede alcanzar valores de la distancia media por celda inferiores (7,97 km) a los que se obtienen aplicando el procedimiento GCSI (13,77 km).
- No obstante, los tiempos de cálculo en aplicar el procedimiento GCSI (valores de segundos) son muy inferiores a los tiempos obtenidos para el procedimiento GCGRASP (hasta casi 100 minutos).

REFERENCIAS

- Angelelli E, Speranza M.G. The periodic vehicle routing problems with intermediate facilities. *European Journal on Operational Research*, 2002, 137, 233-247
- Baker B, Ayecrew M.A. A genetic algorithm for the vehicle routing problem. *Computers and Operations Research*, 2003, 30, 787-800
- Cordeau J-F, Gendreau M, Laporte G, Potvin J-Y, Semet F. A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 2002, 53,512–22
- Ergun Ö, Orlin JB, Steele-Feldman A. Creating very large scale neighborhoods out of smaller ones by compounding moves: a study on the vehicle routing problem. Working paper, Georgia Institute of Technology, USA; 2004
- Feo T.A., Resende M.G.C. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*,1989, 8, 67–71
- Goel A, Gruhn V. A general Vehicle Routing Problem. *European Journal of Operational Research*, 2008, 191, 650-660
- Kytöjoki J., Nuortio T., Bräysy O., Gendreau M. An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. *Computers & Operations Research*, 2007, 34, 2743-2757
- Lenstra J, Rinnooy Kan A. Complexity of vehicle routing and scheduling problems. *Networks* 1981, 11, 221–7
- Mester D, Bräysy O. Active guided evolution strategies for large scale vehicle routing problem. Working paper, University of Haifa, Israel; 2004
- Pelegrín-Pelegrín, B. *Avances en Localización de Servicios y sus Aplicaciones*. Servicio de publicaciones Universidad de Murcia, 2004
- Renaud J, Boctor FF, Laporte G. An improved petal heuristic for the vehicle routing problem. *Journal of the Operational Research Society* 1996, 47, 329–36
- Resende, M.G.C., Gonzalez Velarde, J.L. GRASP: Greedy Randomized Adaptive Search Procedures. *Revista Iberoamericana de Inteligencia Artificial*, 2003, 19, 61-76
- Resende, M.G.C., Ribeiro, C. GRASP. AT&T Labs Research Technical Report, 2008
- Resende, M.G.C., Silva, R. GRASP: Greedy Randomized Adaptive Search Procedures. AT&T Labs Research Technical Report, 2009
- Savelsbergh, M.W.P. The vehicle routing problem with time windows: minimizing route duration. *COSOR Memorandum* 91-03
- Tang J,Zhang J, Pan Z. A scatter search algorithm for solving vehicle routing problem with loading cost. *Expert Systems with Applications*. 2010, 37, 4073–4083
- Toth P, Vigo D. Branch-and-bound algorithms for the capacitated VRP. In: Toth P, Vigo D, editors. *The vehicle routing problem*. Philadelphia: SIAM; 2001. p. 29–52

Autorización y Renuncia

Los autores autorizan a LACCEI para publicar el escrito en los procedimientos de la conferencia. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que esta expresado en el escrito.