

Modelo real de planificación y rutas bi-objetivo. Equilibrio entre costes y preferencias de clientes

Martínez Puras, Amaya. E-mail: amayamar@ubu.es
Pacheco Bonrostro, Joaquín. E-mail: jpacheco@ubu.es
Departamento de Economía Aplicada
Universidad de Burgos

RESUMEN

En este trabajo se ha desarrollado una metodología *ad hoc* basada en Búsqueda Tabú para la resolución de un problema bi-objetivo en el contexto del PVRP (*Periodic Vehicle Routing Problem*). El problema consiste en diseñar las rutas diarias de una empresa de análisis químicos de Salamanca a lo largo de un horizonte de planificación y la asignación de los calendarios de visita a sus clientes. El objetivo es doble: reducción de los costes y reducción de las modificaciones sobre los calendarios actuales de los clientes.

Desde el punto de vista metodológico, la estrategia empleada para la resolución de este nuevo modelo es una adaptación del procedimiento MOAMP (*MultiObjective Adaptive Memory Procedure*) desarrollado por Caballero *et al* (2013) para problemas multiobjetivo. Con el fin de examinar la ‘bondad’ de los resultados obtenidos, se comparan los mismos con una adaptación de un algoritmo de referencia como es NSGA-II (*Non-Dominated Sorting Genetic Algorithm*). Para ello se ha tomado una instancia con datos reales y se han generado otras instancias basadas en estos datos (instancias pseudo reales).

La conclusión final es que la estrategia MOAMP obtiene curvas de eficiencia más densas y pobladas que las obtenidas con NSGA-II. Además, todas las soluciones obtenidas con NSGA-II son dominadas por soluciones MOAMP.

Palabras claves:

Rutas; Planificación; PVRP; Prioridades de clientes; MOAMP; NSGA-II; Búsqueda Tabú.

Área temática: A2 - MÉTODOS CUANTITATIVOS EN UN ENTORNO CON CERTIDUMBRE.

1. INTRODUCCIÓN

El problema que se resuelve consiste en diseñar las rutas diarias de una empresa y la asignación de los calendarios de visita a los clientes.

Cuando una empresa pide que se racionalice su planificación de productos/servicios a clientes a lo largo de un horizonte temporal, normalmente se requiere la reducción de los costes de las operaciones (coste de las rutas). Sin embargo, hemos observado que esta nueva planificación suele ser significativamente diferente de la empleada hasta ese momento. Es decir, para muchos clientes supone la modificación de sus calendarios o fechas de entrega.

Estas modificaciones no suelen ser bien recibidas por los clientes, ya que conllevan cambios en sus hábitos o costumbres. Indirectamente, estos cambios pueden repercutir de forma negativa sobre la imagen de la empresa suministradora de los servicios. Por consiguiente, al plantear el problema debemos tener en cuenta, además de la reducción de los costes, la repercusión de los cambios realizados o la minimización de éstos.

Así planteado, el problema es claramente bi-objetivo: reducción de los costes y reducción de las modificaciones sobre los calendarios actuales de los clientes. De hecho el problema se puede considerar como una variante del conocido *Periodic Vehicle Routing Problem* (PVRP) con estos dos objetivos, y como se explicará en la siguiente sección, sin restricciones de capacidad. Una formalización más detallada se puede ver en la sección 4.

Por otra parte, el PVRP es un problema *NP-hard* y por tanto requiere de técnicas heurísticas para su resolución para poder abordar instancias tamaño (como las reales

que más adelante se usaran) en un tiempo de cálculo razonable. Por este motivo se propondrá un método basado en las estrategias MOAMP y búsqueda tabú.

El trabajo se estructura de la siguiente manera: en la sección 2., se realiza una descripción del problema, su contexto y características; la sección 3. hace referencia a trabajos que presentan ciertas analogías con el que aquí se aborda; en la sección 4. se definen las variables y objetivos del problema, así como las estrategias empleadas en su resolución; posteriormente, se aborda la metodología (sección 5.) empleada mostrándose la jerarquía de procedimientos que componen la estrategia de solución y una breve descripción de los mismos. Para finalizar las dos últimas secciones están dedicadas a los resultados computacionales y conclusiones.

2. DESCRIPCIÓN Y CONTEXTO DEL PROBLEMA REAL

El modelo que se plantea viene motivado por un problema real que consiste en el diseño de las rutas de una empresa de Análisis Químicos situada en Salamanca. La empresa suele realizar unos 1200 servicios al mes que son efectuados por 10 técnicos, quienes conduciendo su propio vehículo son los encargados de realizar las rutas y el servicio. Las rutas comienzan y finalizan en la empresa (Salamanca).

Los servicios se realizan de lunes a viernes y son de tres tipos: recogida de muestras (A), control de plagas (B) y control de calidad (C) con una duración aproximada de 15, 30 y 60 minutos respectivamente. El horizonte de planificación es mensual, 20 días (4 semanas) ya que sábados y domingos no se trabaja y la jornada laboral es como máximo de 9 horas e incluye el tiempo de conducción y de servicio.

El número de clientes que demandan el servicio oscila entre 97 y 104, ahora bien para la resolución del problema se han considerado 100 clientes ya que alguno de ellos es ocasional. También debemos tener presente que clientes con la misma frecuencia de visitas pueden disponer de distintos calendarios. Por otro lado, la capacidad de los vehículos es ilimitada ya que las muestras apenas ocupan espacio.

Estas características que acabamos de ver son las que tenemos que considerar para conseguir dos objetivos: la asignación de los calendarios de visita a los clientes tratando de respetar sus prioridades y el diseño de las rutas diarias.

En la figura 1, se muestra el mapa con las localizaciones de los clientes representados en distintos colores según su frecuencia de visitas. El significado de los colores es: rojo (empresa); azul (todos los días); amarillo (3 días a la semana); verde (2 días a la semana); naranja (1 vez a la semana); gris (1 vez cada 15 días) y negro (1 vez al mes).

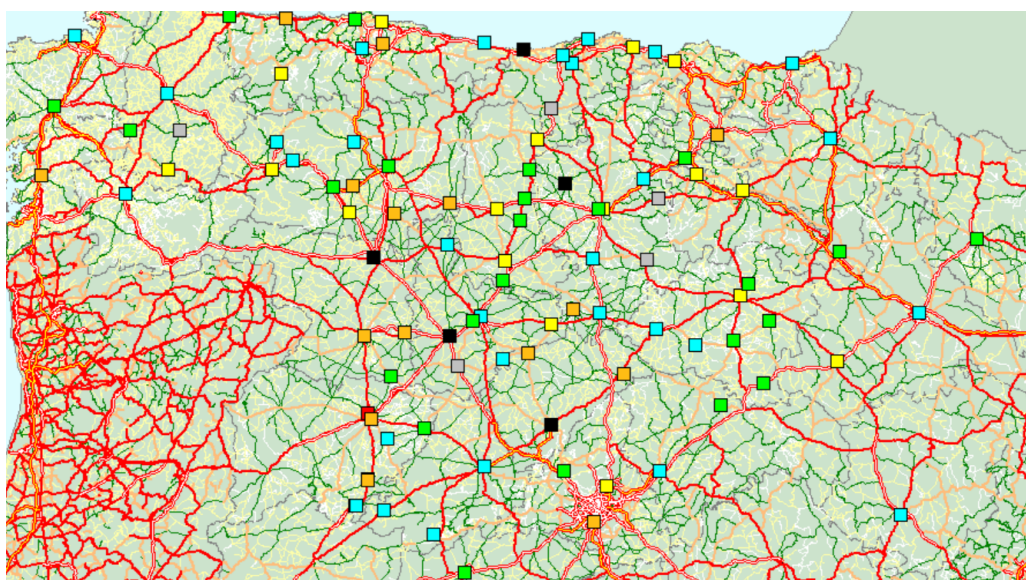


Figura 1: Localizaciones de los clientes.

Las frecuencias de visita y el tipo de servicio que es demandado por los clientes se distribuyen como aparece en la siguiente tabla.

FRECUENCIAS	CLIENTES	SERVICIO
Todos los días	18	A
3 días a la semana	31	A
2 días a la semana	27	A ó B
1 día a la semana	14	B
1 vez cada 15 días	5	B ó C
1 vez al mes	5	C

Tabla 1: Frecuencias de visita y tipo de servicio.

3. PRIMEROS INDICIOS EN LA LITERATURA

3.1. Trabajo de Gulczynski, Golden and Wasil (2011)

El objetivo de minimizar o controlar las diferencias en las asignaciones a calendarios respecto a los de una solución preestablecida en el PVRP solamente ha sido tratado en el trabajo de Gulczynski, Golden and Wasil (2011), desde nuestro conocimiento. No obstante en este trabajo no se plantean estas diferencias como una nueva función objetivo sino como una restricción. Al modelo resultante lo denominan PVRP-RC (*PVRP with Reassignments Constraints*).

3.1.1. Variantes de PVRP-RC

Concretamente, consideran 3 variantes:

- PVRP-RCH (con restricciones de reasignaciones fuertes): se busca minimizar el coste con la restricción adicional de que el número de asignaciones a calendarios diferentes de la solución preestablecida sea menor que una constante W prefijada.
- PVRP-RCS (con restricciones suaves): en este caso, se añade a la función objetivo del coste una penalización por asignación diferente. Esta penalización depende del número de veces que cada cliente es visitado a lo largo del periodo del horizonte temporal.
- PVRP-RCR (con asignaciones limitadas): permite asignaciones diferentes en clientes que son visitados sólo una vez. Estos clientes son los que generan menos problemas.

3.1.2. Reflexiones

Estos autores creen que el PVRP-RC es un modelo más realista que el tradicional PVRP.

En la práctica, existen razones de tipo económico, logístico o de contrato que impiden a las compañías construir nuevas rutas partiendo de 0. Cuando los clientes son asignados a nuevas rutas en nuevos días, la compañía puede incurrir en un coste. Algunas compañías tienen miles de vehículos en sus flotas y visitan millones de clientes anualmente por lo que construir nuevas rutas para estas compañías llevaría tiempo y además sería muy costoso.

También, algunos clientes que cuentan con el servicio en varios días, ven las asignaciones como un inconveniente. Por consiguiente, en la práctica, sería deseable minimizar el número de clientes reasignados en orden para mantener buenas relaciones con ellos.

3.2. Trabajo de García, Pacheco y Álvarez (2013)

Otro trabajo que presenta ciertas similitudes con el que aquí se trata es el de García, Pacheco y Álvarez (2013). En este trabajo se resuelve un problema bi-objetivo aunque no es tratado como un PVRP, podría estudiarse como un caso más particular de éste. Se analiza el problema de una empresa de repostería que debe planificar la entrega de pedidos a delegaciones situadas en el norte de España. El período de planificación es semanal de lunes a viernes, en vez de mensual como en nuestro problema. Cada pedido viene determinado por una fecha de entrega inicial (día de la semana en que se efectúa el pedido) y el número de pallets a entregar.

Con la finalidad de minimizar el coste de las operaciones, se permite que los pedidos sean entregados en la fecha inicial o el día anterior (excepto los del lunes que no pueden adelantarse al domingo). Así, se producen stocks pero controlados. El problema, por tanto, consiste en determinar la fecha definitiva de entrega de cada pedido y diseñar las rutas diarias con dos objetivos: minimizar la distancia total y el stock.

Este problema puede verse como un caso más particular de un PVRP. Se puede considerar que cada cliente dispone de dos posibles calendarios: la fecha inicial o el día anterior excepto los del lunes que sólo tienen un calendario. La diferencia de asignación de calendario con la solución actual vendría ponderada por el nº de pallets que se adelantan.

4. ENUNCIADO Y METODOLOGÍA

4.1. Notación

Nuestro modelo se puede plantear de forma concisa de la siguiente forma. Se define:

T : el número de días del período de planificación

V : conjunto de puntos del problema; $V = \{0, 1, \dots, n\}$ donde 0 representa el depósito, y $\{1, \dots, n\}$ las localizaciones de los clientes. (Para simplificar de ahora en adelante identificaremos los clientes con sus respectivas localizaciones)

Λ_i : Conjunto de patrones o calendarios de visita posibles para el cliente i , $i = 1..n$

q_i : cantidad demandada por el cliente i

m : tamaño de la flota de vehículos

Q : Capacidad de cada vehículo

d_{ij} : distancias entre los puntos i y j , $i, j \in V$

Se define $nc_i = |\Lambda_i|$ y denotamos por λ_{ij} , $j = 1, \dots, nc_i$, a cada uno de los posibles calendarios del cliente i , es decir $\Lambda_i = \{ \lambda_{ij}, j = 1, \dots, nc_i \}$.

Denotemos por R_t un conjunto de rutas para cada día $t \in \{1, \dots, T\}$; y por $V(r)$ y $E(r)$ respectivamente el conjunto de puntos y arcos para cada ruta $r \in R_t$. El problema consiste en asignar un calendario $\lambda_{ij} \in \Lambda_i$, para cada cliente $i \in \{1, \dots, n\}$, y diseñar el conjunto de rutas R_t para cada $t \in \{1, \dots, T\}$ de forma que

- Cada ruta r empieza y finaliza en el depósito 0, $\forall r \in R_t, t \in \{1, \dots, T\}$
- $\forall i \in \{1, \dots, n\}$ y $t \in \lambda_{ij}$ existe una $r \in R_t$ verificando que $i \in V(r)$
- $\sum_{i \in V(r)} q_i \leq Q, \forall r \in R_t, t \in \{1, \dots, T\}$

Por tanto cada solución factible S vendrá determinada por la asignación de cada cliente i a un calendario $\lambda_i \in \Lambda_i$ y por el diseño del conjunto de las rutas diarias R_t que verifican las condiciones anteriores.

4.2. Objetivos

Los objetivos del problema son minimizar la función coste y la función distorsión.

La función coste se define como el coste total de las rutas, el cual es proporcional a la distancia total recorrida. Por tanto, la función a minimizar será la suma de la distancia recorrida por cada ruta diaria a lo largo del horizonte temporal:

$$\sum_{t=1}^T \sum_{r \in R_t} \sum_{(i,j) \in E(r)} d_{ij} \quad (1)$$

La función distorsión se define como la diferencia entre los calendarios de los clientes con la solución actual:

$$\text{dif}(S, S_0) = \text{número de clientes con calendarios diferentes en } S \text{ y } S_0. \quad (2)$$

Si se observa este segundo objetivo está acotado en el número de clientes. Estos dos objetivos no son comparables ni fácilmente agregables y además entran en conflicto. Por tanto, se buscan soluciones que equilibren estos dos objetivos.

El problema, así planteado, es claramente bi-objetivo y PVRP. Para cada cliente se determinan los días de visita y se diseñan las rutas diarias de acuerdo a la asignación de calendarios de visita a los clientes.

Para su resolución, se determina el ‘conjunto de puntos eficientes’, es decir, el conjunto de soluciones no dominadas. Una solución no dominada es aquella para la cual no se conoce otra solución que mejore su valor en relación a alguno de los objetivos sin empeorar otros.

Dada una solución cualquiera S denotemos por $f_1(S)$ al valor de la función objetivo “económica”, es decir, la definida por (1) y $f_2(S)$ la función objetivo definida por (2). Sean S y S' dos soluciones, se dice que S domina a S' si:

- a) $f_1(S) \leq f_1(S')$
- b) $f_2(S) \leq f_2(S')$ y
- c) $f_1(S) < f_1(S')$ o bien $f_2(S) < f_2(S')$

Una solución es eficiente si no existe otra que la domina. El problema que aquí se plantea consiste en encontrar soluciones lo más cercanas posibles al conjunto de soluciones eficientes del problema que se denomina ‘curva de eficiencia’.

4.3. Estrategia de solución

La estrategia que se propone para resolver el nuevo modelo planteado es una adaptación del procedimiento MOAMP (*Multiobjective Adaptive Memory Procedure*) para problemas multiobjetivo.

MOAMP fue desarrollado por Caballero, Molina y Rodríguez (2003) y está basado en dos principios:

- Principio de Proximidad de los puntos eficientes, según el cual, en un entorno o vecindad de una solución eficiente se puede encontrar otra solución eficiente.
- La solución que minimiza la distancia L_∞ normalizada y/o ponderada al punto ideal es también eficiente.

La estrategia MOAMP se desarrolla en 3 fases creando una aproximación a la curva de eficiencia enlazando la ejecución de una serie de procedimientos de búsqueda tabú con diferentes funciones objetivo. En la primera fase, usa las funciones objetivo originales y en la segunda combinaciones de las originales. Finalmente, realiza una intensificación explorando el vecindario de las soluciones encontradas hasta ese momento.

4.4. Algoritmo NSGA-II

Con el fin de poder examinar la ‘bondad’ de los resultados obtenidos por el algoritmo diseñado para la resolución del problema, se van a comparar con los resultados obtenidos por una adaptación del conocido NSGA-II (*Non-Dominate Sorting Algorithm*). Fue propuesto por Deb y sus alumnos (2002) y está encuadrado dentro de los algoritmos evolutivos. NSGA-II usa elitismo y un operador de comparación (crowding) en función de la proximidad entre sí de las diferentes soluciones.

5. METODOLOGÍA.

Para facilitar la descripción de los procedimientos desarrollados para la resolución del problema formalicemos primero la representación de las soluciones. Cada solución S viene definida por la asignación de clientes a calendarios y el diseño de las correspondientes rutas diarias.

5.1. Esquema general y Jerarquía de procedimientos

Como ya se ha comentado, MOAMP se desarrolla en tres fases. En la figura 2 se muestra la jerarquía de todos los procedimientos empleados.

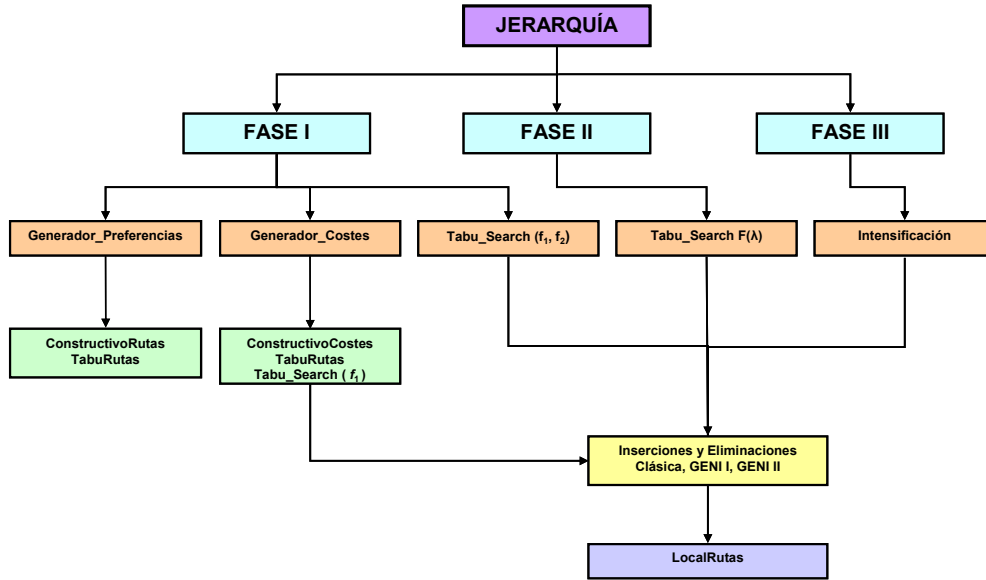


Figura 2: Jerarquía de procedimientos.

A continuación se hace una descripción más detallada de las 3 fases de MOAMP incluyendo los pseudocódigos que componen el cuerpo principal. para ello se define F_λ de la siguiente forma:

$$F_\lambda(S) = \max \left\{ \lambda \cdot \frac{f_1(S) - f_1^{\min}}{f_1^{\max} - f_1^{\min}}, (1 - \lambda) \cdot \frac{f_2(S) - f_2^{\min}}{f_2^{\max} - f_2^{\min}} \right\}$$

donde $\lambda \in (0,1)$, $f_1^{\min}$ y $f_1^{\max}$ son el mínimo y máximo valor encontrado para f_1 en el conjunto de soluciones no dominadas, y $f_2^{\min}$ y $f_2^{\max}$ como el mínimo y máximo valor encontrado para f_2 en el conjunto de soluciones no dominadas.

Se va a denotar por Set_ND el conjunto de soluciones no dominadas encontradas durante la búsqueda. El conjunto Set_ND final será la aproximación a la curva de eficiencia obtenida. La estrategia de solución se compone de las tres fases siguientes:

FASE I

1. Hacer $Set_ND = \emptyset$.

2. Ejecutar **Generador_Preferencias**(S) para obtener una solución S teniendo en cuenta las preferencias de los clientes (f_2).
 3. Ejecutar **Tabu_Search**(f_1, S), tomando f_1 como función objetivo ($f_2 \rightarrow f_1$)
 4. Ejecutar **Generador_Costes**(S) para obtener una solución S minimizando la distancia (f_1).
 5. Ejecutar **Tabu_Search**(f_2, S) ($f_1 \rightarrow f_2$)
 6. Actualizar Set_ND con todas las soluciones no-dominadas encontradas
-

FASE II

Repetir

7. Generar $\lambda \in (0,1)$ aleatoriamente.
 8. Ejecutar **Tabu_Search**(F_λ, S) usando F_λ como función objetivo.
 9. Actualizar Set_ND
- hasta** que trascurren max_iter_faseii iteraciones sin cambios en Set_ND
-

FASE III

Repetir

10. Identificar las soluciones S de Set_ND cuyos vecindarios no han sido explorados (todas en la primera iteración)
 11. Para cada una de estas soluciones S explorar sus vecindarios y actualizar Set_ND .
 12. Identificar las nuevas soluciones de Set_ND y mejorarlas
- hasta** Set_ND se estabilice (no cambie)
-

Con la primera fase se quieren obtener las mejores soluciones con respecto a cada una de las funciones objetivos originales. En otras palabras: se obtienen los extremos de la (aproximación a la) curva de eficiencia y por tanto los valores de $f_1^{\min}, f_1^{\max}, f_2^{\min}$ y $f_2^{\max}$ que son usados en el cálculo de las funciones mixtas F_λ en la fase siguiente.

En la segunda fase se obtiene una muestra diversa de soluciones ‘de compromiso’; es decir, no las mejores soluciones con respecto a algún objetivo concreto, pero sí buenas soluciones en conjunto. Esta fase finaliza cuando transcurre un

cierto número de iteraciones (*max_iter_faseii*) sin cambios en *Set_ND*. Hay que indicar que, dentro de cada procedimiento de búsqueda tabú, se comprueba si cada solución visitada se incorpora a *Set_ND*. En otras palabras, el conjunto de soluciones no dominadas se actualiza con todas las soluciones visitadas durante cada ejecución de la búsqueda tabú en estas dos primeras fases. Finalmente, en la tercera fase se completa el conjunto de soluciones no dominadas explorando el vecindario de las encontradas hasta ese momento.

A continuación se describen muy brevemente los procedimientos que componen este método. Por problemas de espacio se hace una descripción muy concisa. No obstante, está a disposición de los lectores interesados la tesis de Amaya Martínez Puras donde se dan más detalles.

5.2. Procedimientos principales

Los procedimientos principales son *Generador_Preferencias*, *Generador_Costes* y *Tabu_Search*. Mediante *Generador_Preferencias* se obtiene una solución *S* usando los calendarios de la solución pre-establecida *S₀*. Posteriormente, se optimiza el coste de las rutas mediante *ConstructivoRutas* y *TabuRutas*.

Generador_Costes construye la solución que minimiza la función coste. Este problema es equivalente al PVRP clásico con algunos matices. En nuestro caso, *Generador_Costes* sigue una estrategia iterativa de tipo multi-arranque. En cada iteración, *ConstructivoCostes* construye una solución. Después, sin modificar los calendarios asignados a los clientes, se mejora el conjunto de rutas para cada día con *TabuRutas*. Finalmente, se intenta mejorar aún más la solución obtenida aplicando a la función coste el algoritmo *Tabu_Search*.

TabuSearch es una metaheurística basada en búsqueda tabú. En cada iteración se mueve de la solución actual a la mejor solución vecina. En este caso, un movimiento viene definido por el cambio de calendario asignado a un cliente. Para que el proceso no cicle determinados movimientos son declarados tabú. A su vez, este criterio tabú puede

ser ignorado en determinadas condiciones (“Criterio de Aspiración”). El proceso finaliza cuando se alcanza un criterio de parada.

5.3. Procedimientos de rutas

En esta sección se describen los procedimientos de construcción y mejora de las rutas diarias usados en los procedimientos generales explicados en la sección anterior. El objetivo siempre es minimizar el coste. Para ello, se desarrollan tres procedimientos: ConstructivoRutas, LocalRutas y TabuRutas.

ConstructivoRutas toma como parámetro de entrada el subconjunto de puntos a visitar ese día y como salida se obtiene el conjunto de las rutas, para ello utiliza inserciones.

LocalRutas es una heurística de búsqueda local que intenta mejorar el conjunto de rutas de un determinado día. Los datos de entrada y salida son la propia variable Rutas. Se define el conjunto $NR(Rutas)$ como el vecindario de Rutas, es decir, todos los conjuntos de rutas que se pueden obtener realizando en *Rutas* cambios o movimientos. Estos cambios consisten en intercambios de cadenas.

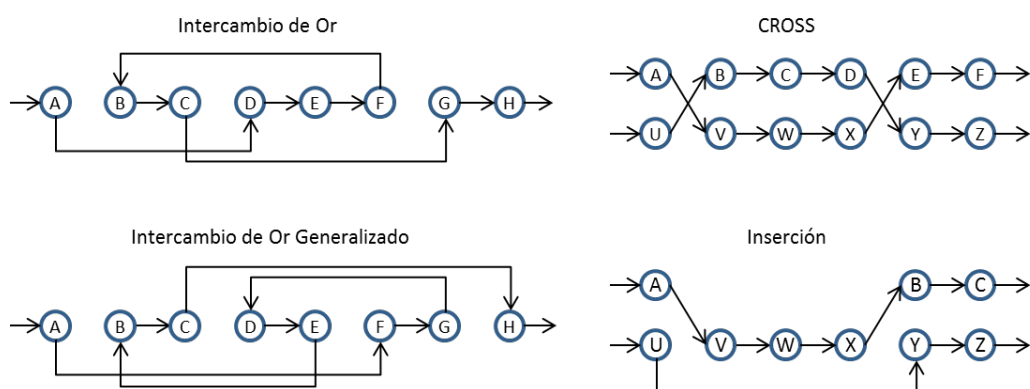


Figura 3: Tipos de movimientos.

LocalRutas mejora cada conjunto de rutas inicial hasta que no hay mejoras posibles. Posteriormente, se ejecuta TabuRutas que intenta mejorar los resultados obtenidos por LocalRutas ya que permite exploraciones más allá del primer óptimo local encontrado. Para ello se permiten movimientos a conjuntos de rutas peores. Los

movimientos consisten en la eliminación e incorporación de una serie de arcos y para evitar que el algoritmo cicle, algunos movimientos se declaran tabú aunque pueden ser admitidos si dan lugar a un conjunto de rutas mejor.

5.4. Inserciones y eliminaciones

Para terminar la descripción de todos los métodos que componen la estrategia de solución, se comentan los tipos de eliminaciones e inserciones usadas por los procedimientos antes descritos. Las eliminaciones son usadas por Tabu_Search y TabuRutas, las inserciones por ConstructivoCostes, ConstructivoRutas, Tabu_Search y TabuRutas.

Concretamente, se contemplan 3 tipos de eliminaciones: Normal o Clásica, GENI I y GENI II y sus análogas inserciones. GENI I y GENI II fueron introducidas por Gendreau *et al.* (1992).

Además de los procedimientos descritos se han usado diferentes estrategias de aceleración para reducir los tiempos de computación. Los lectores interesados pueden preguntar a los autores o consultar el trabajo de Martínez (2013).

6. EXPERIMENTOS COMPUTACIONALES

En esta sección se realizan un aserie de pruebas para chequear el funcionamiento de nuestro método MOAMP. La primera subsección muestra un ejemplo del funcionamiento de las 3 fases MOAMP. En la segunda se compara MOAMP con el método NSGA-II a través de un conjunto de instancias pseudo-reales. En la tercera subsección se analizan los resultados de ambas estrategias con una instancia real y se comparan con la solución usada por la empresa. Finalmente, en la cuarta subsección se analizan las diferentes opciones que nos da la aproximación a la curva de eficiencia obtenida por MOAMP con estos datos reales.

6.1. Las 3 fases de MOAMP

Tras ejecutar MOAMP sobre las instancias generadas, en cada fase se obtiene un conjunto de soluciones. A modo de ilustrar los resultados que se obtienen en cada una de las 3 fases de MOAMP, la figura 4 muestra los resultados obtenidos para una instancia de tamaño 100. En ella se puede apreciar como en cada fase se mejora el conjunto de soluciones obtenidas en la fase anterior. Además, a medida que aumenta el tamaño de las instancias se obtienen conjuntos más densos y con un mayor número de soluciones.

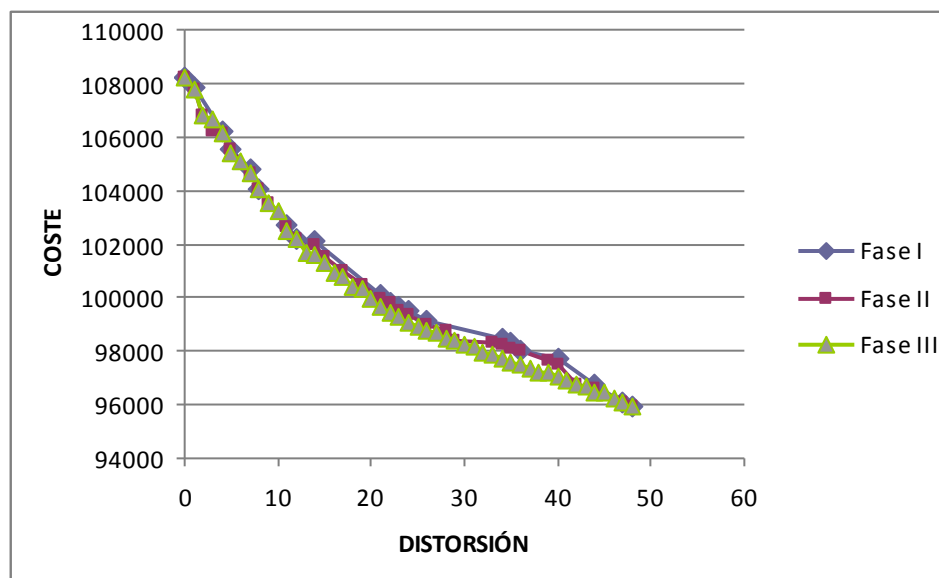


Figura 4: Resultados de las 3 fases de MOAMP.

6.2. Comparación de MOAMP con NSGA-II

En esta sección se comparan los resultados obtenidos por ambas estrategias. Para ello se generan instancias seudoreales de tamaño $n = 20, 40, 60, 80$ y 100 . Concretamente se generan 5 instancias de cada tamaño. Inicialmente se ejecuta el procedimiento MOAMP y se registra el número de soluciones no dominadas ($|Set_ND|$) y el tiempo de computación empleado (en segundos). Entonces se ejecuta NSGA- II tomando $n_pob = 2 \cdot |Set_ND|$ y $p_mut = 0.02$

El criterio de parada de NSGA-II se alcanza cuando tras finalizar una iteración (o “generación”) se supera el tiempo de computación empleado por MOAMP. El ajuste de estos parámetros se ha obtenido mediante diferentes experimentos basándonos además en los resultados obtenidos en Gómez *et al.* (2013) y Pacheco *et al.* (2013) en problemas similares.

En la tabla 3, se recogen los resultados de la media del número de soluciones MOAMP y NSGA-II, también aparece el número medio de soluciones MOAMP no dominadas por soluciones NSGA-II y la media de soluciones NSGA-II no dominadas por soluciones MOAMP. Si nos fijamos, vemos que a partir de las instancias de tamaño 60 no hay ninguna solución NSGA-II que no esté dominada por alguna MOAMP.

MEDIA DE	TAMAÑO DE INSTANCIAS				
	20	40	60	80	100
Soluciones MOAMP	9,8	20,3	28,8	41,4	52,2
Soluciones MOAMP no dominadas por NSGA II	8,8	20,3	28,8	41,4	52,2
Soluciones NSGA II	6,1	11,4	15,8	21,7	24,8
Soluciones NSGA II no dominadas por MOAMP	1,6	0,2	0	0	0

Tabla 3. Resumen (media) de soluciones dominadas por cada estrategia.

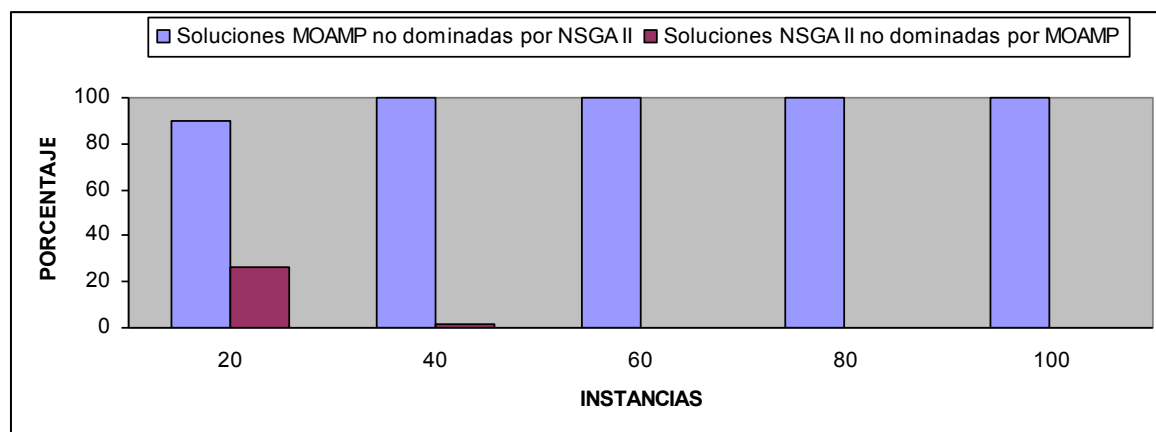


Figura 5: Porcentaje de soluciones no dominadas.

Con el objetivo de ilustrar estos resultados las figuras 6 muestra las curvas obtenidas por ambos métodos para una instancia de tamaño 100. Se observa como con MOAMP se consigue una curva de eficiencia con un mayor número de soluciones,

menos dominadas, más densas y con un comportamiento mejor en los extremos. Entendemos que las figuras 5 y 6 son suficientemente claras y que por tanto son innecesarios realizar test de dominancias.

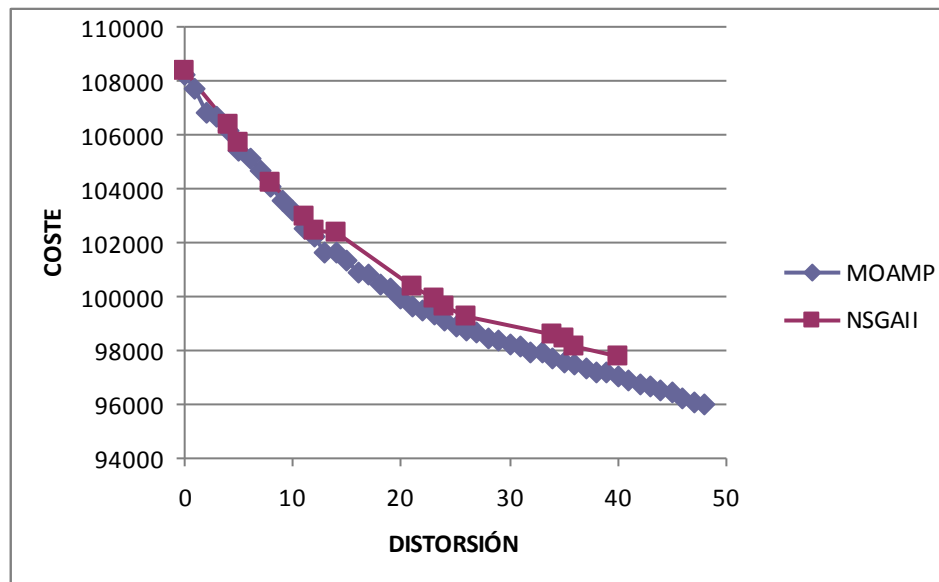


Figura 6: Conjunto de soluciones no dominadas.

6.4. Comparación de MOAMP, NSGA-II y la solución real (empresa)

En la figura 7 se representan para la instancia con datos reales, las soluciones obtenidas con MOAMP, NSGA-II y la solución empleada por la empresa con estos datos.

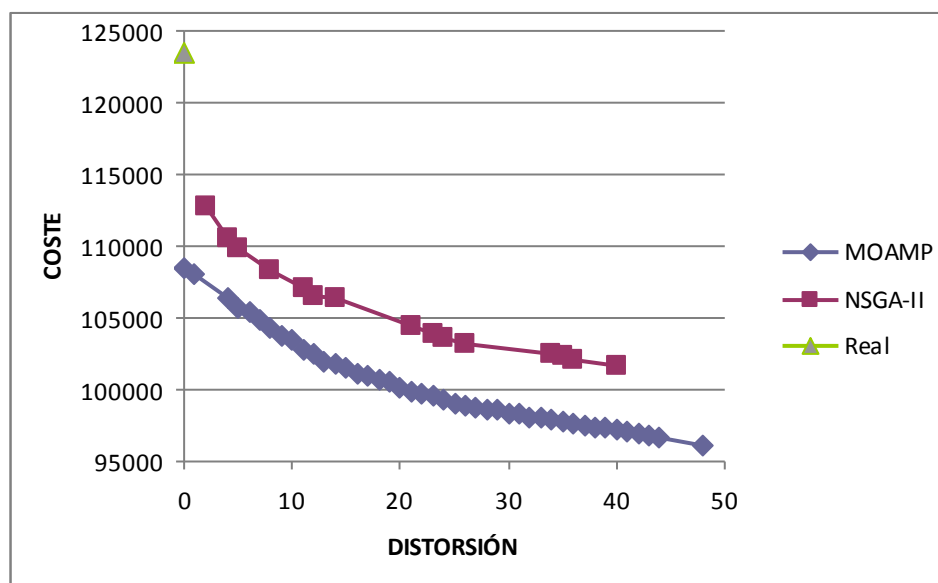


Figura 7: Soluciones MOAMP, NSGA-II y actual.

Como se observa, es obvio que la solución actual tiene que tener distorsión 0 ya que son los calendarios que actualmente tienen asignados los clientes y por los cuales tienen preferencia. Sin embargo, la estrategia MOAMP obtiene una solución con menor coste que respeta dichos calendarios. Esto es debido a que la empresa ni siquiera tiene optimizadas las rutas diarias. También, podemos apreciar como al permitir modificar la asignación de calendarios a los clientes, tanto MOAMP como NSGA-II obtienen soluciones que reducen considerablemente el coste.

Está claro que ninguna solución MOAMP es dominada por soluciones NSGA-II y que la ‘curva de eficiencia’ obtenida por MOAMP es más densa y está más poblada que la obtenida por NSGA-II.

6.5. Análisis de la curva obtenida por MAOMP

Creemos conveniente hacer un análisis más detallado de la curva de soluciones obtenida con MOAMP.

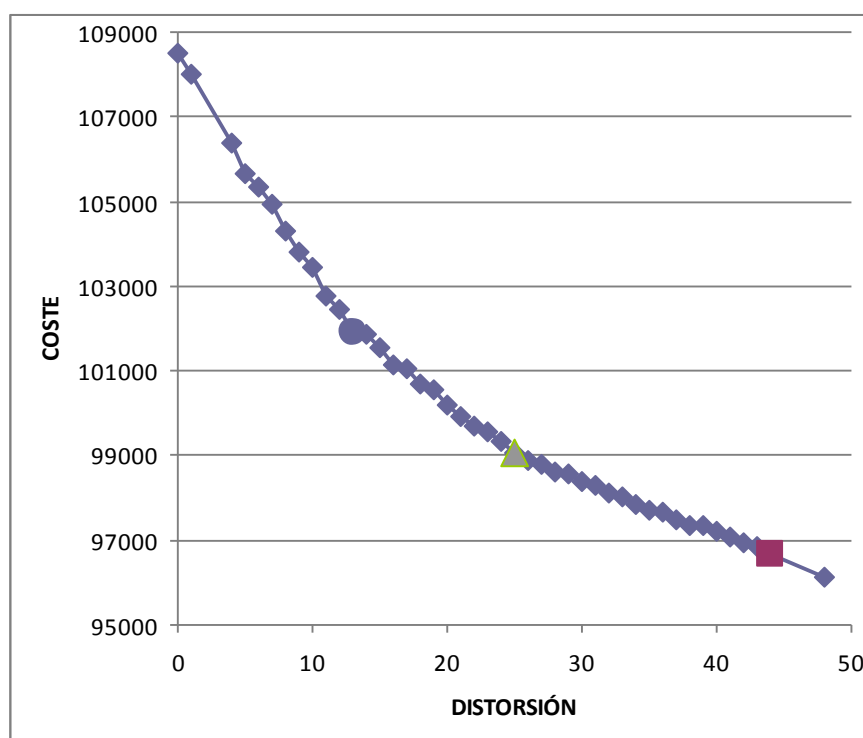


Figura 8: Análisis de la curva de soluciones.

La solución de distorsión 0 de MOAMP produce un ahorro del 12% sobre la solución actual de la empresa. En el otro extremo, la solución de mínimo coste tiene una distorsión de 48 y produce un ahorro de un 11,4% sobre la solución MOAMP de distorsión 0.

Entre estas dos soluciones nos encontramos con un amplio abanico de soluciones intermedias. Un análisis que puede resultar muy interesante es estudiarla desde distintas situaciones económicas.

Así, en época de crisis o de recesión, como la que vivimos en la actualidad, la empresa puede estar más interesada en minimizar el coste aunque esto le suponga aumentar la distorsión. Como ya he dicho, la solución de mínimo coste tiene distorsión 48. Ahora bien, partiendo de esta solución nos encontramos con que la siguiente solución con menos coste (resaltada en rojo) reduce la distorsión en un 8% sin apenas aumentar el coste. Incluso, podríamos llegar hasta la solución resaltada en verde en la que se reduce la distorsión hasta 25 produciéndose un aumento del coste de sólo el 2,4%. A partir de aquí, resulta algo más costoso reducir la distorsión.

Detengámonos ahora en el primer tramo que sería el más conservador en cuanto a distorsión y quizá más interesante en tiempos de expansión. En este tramo, la pendiente de la curva es más pronunciada por lo que un pequeño ahorro supone un aumento considerable de la distorsión. No obstante, podríamos llegar hasta la solución resaltada en amarillo que con distorsión 13 reduce en un 5,3% el coste.

Por consiguiente, se puede concluir que MOAMP facilita un conjunto de soluciones aunque la decisión final deberá tomarla la empresa según sus intereses y prioridades.

7. CONCLUSIONES

Se ha desarrollado un método para resolver un problema bi-objetivo y PVRP. El objetivo es doble: reducción de los costes y reducción de los cambios de calendario a los clientes. Estos objetivos no son fácilmente agregables ni comparables y además entran en conflicto.

El método desarrollado aporta un conjunto denso de soluciones con las que se puede encontrar un equilibrio entre los objetivos. Para contrastar los resultados de MOAMP, éstos se comparan con los obtenidos por una adaptación de NSGA-II obteniéndose con MOAMP una curva de eficiencia más densa, poblada y que no es dominada por la obtenida con NSGA-II.

Por consiguiente, concluimos que el diseño de estrategias MOAMP adaptadas al problema produce mejores resultados aunque conlleva un esfuerzo adicional de diseño e implementación. Destacar que el problema es nuevo en sí y es abordado desde una perspectiva que considera las costumbres adquiridas por los clientes como un nuevo objetivo y no como una restricción.

8. REFERENCIAS BIBLIOGRÁFICAS

- Bentley, J.L. (1992). "Fast Algorithms for Geometric Salesman Problems". *ORSA Journal of Computing*, 4, pp. 387-411.

- Caballero, R., Molin, J. y Rodríguez, M.V. (2003). “MOAMP: Programación Multiobjetivo mediante un Procedimiento de Búsqueda Tabú”. II Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados, MAEB, Gijón.
- Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002). “A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II”. IEEE Transactions on Evolutionary Computation 6, 2, pp. 182–197.
- García, I., Pacheco, J. y Álvarez, A. (2013). “Optimizing routes and stock”. Journal of Heuristics, 19, 2, pp. 157-177.
- Gendreau, M., Hertz, A. and Laporte, G. (1992). “New insertion and postoptimization procedures for the traveling salesman problem”. Operations Research, 40, pp. 1086-1094.
- Gómez, J.R., Pacheco, J. and Gonzalo-Orden, H. (2013). “A Tabu Search method for a Bi-objective Urban Waste Collection Problem”. To appear in Computer-Aided Civil and Infrastructure Engineering doi: 10.1111/mice.12031
- Gulczynski, D., Golden, B. and Wasił, E. (2011). “The period vehicle routing problem: New heuristics and real-world variants”. Transportation Research Part E 47, 5, pp. 648-668.
- Martínez, A. (2013). “Modelos reales de problemas simultáneos de rutas y planificación. Equilibrio entre prioridades del cliente y reducción de costes”. Tesis doctoral, Universidad de Burgos.
- Pacheco, J., Caballero, R., Laguna, M. and Molina, J. (2013). “Bi-objective Bus Routing: An Application to School Buses in Rural Areas”. Transportation Science 47, 3, pp. 397-411.