



UNIVERSIDADE DA CORUÑA

FACULTADE DE CIENCIAS ECONOMICAS E EMPRESARIAIS

DEPARTAMENTO DE ECONOMIA APLICADA II

**UN ESTUDIO DE LA IMPLEMENTACION
DEL METODO DEL CONJUNTO ACTIVO EN
PROCEDIMIENTOS DE ASIGNACION DE
REDES DE TRAFICO**

TESIS DOCTORAL PRESENTADA POR

JOSE ANTONIO SEIJAS MACIAS

BAJO LA DIRECCION DE

PROF. Dr. D. LUIS P. PEDREIRA ANDRADE

JUNIO 1997

UNIVERSIDADE DA CORUÑA

FACULTADE DE CIENCIAS ECONOMICAS E EMPRESARIAIS

DEPARTAMENTO DE ECONOMIA APLICADA II

**UN ESTUDIO DE LA IMPLEMENTACION
DEL METODO DEL CONJUNTO ACTIVO EN
PROCEDIMIENTOS DE ASIGNACION DE
REDES DE TRAFICO**

TESIS DOCTORAL PRESENTADA POR:

JOSE ANTONIO SEIJAS MACIAS

BAJO LA DIRECCION DE :

PROF. Dr. D. LUIS P. PEDREIRA ANDRADE

JUNIO 1997.

Trabajo original de investigación dirigido por el profesor Dr. D. Luis Pedro Pedreira Andrade, Catedrático Numerario del Departamento de Economía Aplicada II de la Universidad de A Coruña y presentado por Don José Antonio Seijas Macias para la obtención del Título de Doctor.

AGRADECIMIENTOS:

A Luis P. Pedreira Andrade por su dirección, amistad y confianza depositada sin la cual este trabajo no hubiera sido posible.

A mi familia por su comprensión y por su apoyo constante.

A los profesores de la Universidad de A Coruña, D. Luis C. Currais Nunes y D. Manuel A. Gómez Suárez por su ayuda y colaboración en todos los momentos.

A todos los miembros de los departamentos de Economía Aplicada II y Análisis Económico de la Universidad de A Coruña que de una u otra forma me han ayudado durante la realización de este trabajo.

Ab Fusi e Aria

INDICE

UN ESTUDIO DE LA IMPLEMENTACION DEL METODO DEL CONJUNTO ACTIVO EN PRODECIMIENTOS DE ASIGNACION DE REDES DE TRAFICO

| | |
|--|----|
| 1. INTRODUCCION..... | 1 |
| 2. OPTIMIZACION DE FUNCIONES..... | 7 |
| 2.1. OPTIMIZACIÓN SIN RESTRICCIONES..... | 9 |
| 2.1.1. <i>Caso Univariante.</i> | 9 |
| 2.1.2. <i>Caso Multivariante.</i> | 10 |
| 2.2. OPTIMIZACIÓN SUJETA A RESTRICCIONES..... | 11 |
| 2.2.1. <i>Problema de optimización con restricciones de igualdad.</i> | 12 |
| 2.2.2. <i>Problema de optimización con restricciones de desigualdad.</i> | 20 |
| 2.2.3. <i>Método del Conjunto Activo.</i> | 22 |
| 2.2.3.1. Elección del punto y conjunto de trabajo iniciales. | 27 |
| 2.2.3.2. Cálculo de la dirección de búsqueda..... | 29 |
| 2.2.3.3. Cálculo de la longitud de paso..... | 32 |
| 2.2.3.4. Añadir una restricción..... | 34 |
| 2.2.3.5. Eliminar una restricción..... | 35 |
| 2.2.4. <i>Optimización sujeta a restricciones de acotación.</i> | 36 |
| 2.2.4.1. Restricciones de acotación..... | 36 |
| 2.2.4.2. Restricciones de acotación y lineales. | 37 |
| 3. OPTIMIZACION DE REDES..... | 39 |
| 3.1. REDES DE TRÁFICO..... | 39 |
| 3.1.1. <i>Equilibrio en redes de tráfico.</i> | 43 |
| 3.1.1.1. Matrices de Viajes y Matrices Origen - Destino (O-D) | 43 |
| 3.1.1.2. Definición de Equilibrio..... | 45 |
| 3.1.2. <i>Equilibrio del Usuario.</i> | 45 |
| 3.2. FUNCIÓN DE REPRESENTACIÓN..... | 47 |
| 3.2.1. <i>Función de Costes.</i> | 54 |
| 3.2.2.1. Costes a Corto Plazo. | 55 |

| | |
|--|-----|
| 3.2.2.2. Costes a Largo Plazo. | 57 |
| 3.2.2. ESTIMACIÓN DEL COSTE TOTAL DE USUARIO..... | 60 |
| 3.3. ALGORITMOS DE REDES DE TRÁFICO. | 68 |
| 3.3.1. <i>Redes con flujo estático.</i> | 71 |
| 3.3.1.1. Método de las combinaciones convexas (Frank-Wolfe). | 75 |
| 3.3.2. <i>Redes con flujo variable no estocástico.</i> | 77 |
| 3.3.2.1. Método de Frank-Wolfe para redes con flujo variable. | 79 |
| 3.3.2.2. Método de modificación de la red. | 81 |
| 3.3.3. <i>Redes con efectos de interacción.</i> | 83 |
| 3.3.3.1. Redes con interacción simétrica..... | 84 |
| 3.3.3.2. Redes con interacción asimétrica. | 87 |
| 3.3.3.3. Implementación del método del Conjunto Activo..... | 91 |
| 3.4. APROXIMACIONES ALTERNATIVAS. | 97 |
| 3.4.1. <i>Modelos de asignación dinámica.</i> | 98 |
| 3.4.2. <i>Modelos Continuos.</i> | 100 |
| | |
| 4. TEORIAS ECONOMICAS DEL PEAJE. | 102 |
| 4.1. <i>Peajes de Congestión.</i> | 102 |
| 4.1.1. <i>Congestión y Tráfico.</i> | 105 |
| 4.1.2. <i>Peajes Optimos.</i> | 108 |
| 4.1.2.1. Modelos de Estado Estacionario..... | 112 |
| 4.1.2.2. Modelos Dinámicos. | 113 |
| 4.2. IMPLEMENTACIÓN PRÁCTICA..... | 117 |
| 4.2.1. <i>Características de los Sistemas de Cargo por la Utilización de Infraestructuras.</i> 117 | |
| 4.2.1.1. Equidad del sistema. | 118 |
| 4.2.1.2. Privacidad..... | 119 |
| 4.2.1.3. Forzamiento del Pago..... | 120 |
| 4.2.1.4. Efectos a Largo Plazo. | 121 |
| 4.2.1.5. Otros aspectos relevantes..... | 122 |
| 4.2.2. <i>Diferentes Tipos de Sistemas de Cargo.</i> | 123 |
| 4.2.3. <i>Viabilidad del Sistema de Peajes.</i> | 125 |
| 4.2.4. <i>Ejemplos Reales.</i> | 128 |
| 4.2.4.1. Noruega. | 129 |
| 4.2.4.2. Singapur. | 130 |
| 4.3. POLÍTICAS ALTERNATIVAS. | 132 |
| 4.3.1. <i>Críticas al Sistema de Peajes.</i> | 132 |
| 4.3.2. <i>Alternativas de Tarifación.</i> | 134 |
| 4.3.2.1. Pagos por Aparcamiento. | 135 |
| 4.3.2.2. Subvención al Transporte Público..... | 137 |
| 4.3.2.3. Usos Alternativos. | 140 |

| | |
|--|-----|
| 5. IMPLEMENTACION EN REDES DEL METODO DEL CONJUNTO ACTIVO..... | 144 |
| 5.1. OPTIMIZACIÓN DE LA RED..... | 144 |
| 5.1.1. <i>Concepto de Flujo</i> | 145 |
| 5.1.1.1. Un sólo origen y sólo destino..... | 147 |
| 5.1.1.2. Múltiples orígenes y destinos..... | 148 |
| 5.1.1.3. Eliminación de Arcos..... | 150 |
| 5.1.2. <i>Problema de Optimización</i> | 152 |
| 5.1.3. <i>Restricciones del Problema</i> | 153 |
| 5.1.4. <i>Función de Coste</i> | 155 |
| 5.2. APLICACIÓN DEL MÉTODO DEL CONJUNTO ACTIVO..... | 162 |
| 5.2.1. <i>Descripción del Algoritmo del Conjunto Activo para Asignación en Redes</i> | 163 |
| 5.2.1.1. Preliminares..... | 165 |
| 5.2.1.2. Elección del punto inicial..... | 166 |
| 5.2.1.3. Cálculo de la dirección factible..... | 167 |
| 5.2.1.4. Longitud del paso..... | 168 |
| 5.2.1.5. Adaptación de la matriz B..... | 171 |
| 5.2.1.6. Restricciones de no negatividad..... | 172 |
| 5.2.1.7. Criterio de Optimalidad..... | 174 |
| 5.2.1.8. Test de Parada..... | 176 |
| 5.2.2. <i>Implementación en Mathematica</i> | 179 |
| 5.3. CÁLCULO DEL PEAJE OPTIMO..... | 188 |
| 5.3.1. <i>Aproximación Tradicional al Peaje Optimo</i> | 189 |
| 5.3.2. <i>Propuesta de Peaje Alternativo</i> | 193 |
| 5.3.3. <i>Ejemplo de Red con Peajes</i> | 202 |
| 5.4. EVALUACIÓN RESULTADOS OBTENIDOS CON EL MÉTODO DEL CONJUNTO ACTIVO..... | 206 |
| 5.4.1. <i>Ejemplo 1</i> | 207 |
| 5.4.2. <i>Ejemplo 2</i> | 211 |
| 5.4.3. <i>Ejemplo 3</i> | 218 |
| | |
| 6. CONCLUSIONES..... | 227 |
| | |
| BIBLIOGRAFIA..... | 233 |

APENDICES.

| | |
|---|------|
| APENDICE A: FACTORIZACION DE MATRICES | A-1 |
| 1. ELIMINACIÓN GAUSSIANA..... | A-2 |
| 2. FACTORIZACIÓN LU..... | A-7 |
| 3. FACTORIZACIÓN DE CHOLESKY. | A-10 |
| 4. FACTORIZACIÓN QR. | A-16 |

| | |
|---|------|
| APENDICE B: MATHEMATICA. | B-1 |
| ESTRUCTURA DEL PROGRAMA MATHEMATICA. | B-2 |
| COMANDOS. | B-4 |
| FUNCIONES. | B-8 |
| OPERADORES..... | B-10 |
| LISTAS..... | B-11 |
| PACKAGES. | B-14 |
| ITERACIÓN Y RECURSIVIDAD..... | B-15 |
| PROGRAMACIÓN EN MATHEMATICA. | B-20 |
| BIBLIOGRAFÍA SOBRE MATHEMATICA. | B-26 |
| PACKAGE: CATIVO.M..... | B-27 |

| | |
|--|------|
| APENDICE C: RESULTADOS | C-1 |
| 1. EJEMPLO DE RED 5.1 (TOMADO DE SHEFFI, 1985)..... | C-2 |
| 2. EJEMPLO DE RED (TOMADO DE HEARN & RIBERA , 1980)..... | C-5 |
| 3. EJEMPLO DE RED (TOMADO DE NGUYEN & DUPUIS, 1984) | C-15 |
| 4. EJEMPLO DE REDES CON PEAJE. | C-30 |
| 4.1 <i>Peaje Tradicional.</i> | C-30 |
| 4.2 <i>Peaje Alternativo.</i> | C-35 |
| 5. EJEMPLO DE RED CON COSTES CON INTERACCIÓN ASIMÉTRICA..... | C-39 |

CAPITULO 1

INTRODUCCION

1. INTRODUCCION.

La optimización de una función consiste en encontrar su valor máximo o mínimo, sujeta o no a un conjunto de restricciones que acotan el dominio de búsqueda de dicha función. La existencia de métodos numéricos para abordar este problema ha sido un área objeto de un amplio estudio por parte de los matemáticos, y las soluciones propuestas han variado de forma considerable en función de los diversos supuestos iniciales del problema planteado. Dentro del campo de estudio sobre optimización nos hemos enfocado en el caso de funciones no lineales sujetas a restricciones lineales.

Uno de los principales métodos de optimización que existe en la literatura actual es el denominado método del conjunto activo. Este método forma parte del amplio conjunto de métodos iterativos que intentan alcanzar la solución óptima a partir de un punto inicial factible y desplazándose a través de diversos puntos factibles a lo largo de las diversas iteraciones. La principal particularidad de este método es la consideración del conjunto de restricciones activas, esto es, que se cumplen con signo de igualdad, como base de definición del espacio de búsqueda en cada una de las distintas etapas del proceso.

El algoritmo del conjunto activo es un método de carácter numérico. A la hora de abordar el estudio de ejemplos teóricos es preciso la utilización de otros métodos más habituales en la literatura matemática y económica, tal como el método de Lagrange y las condiciones de Karush-Kuhn-Tucker que se utilizan para definir el valor óptimo de la función a optimar. No obstante, hay que señalar que el óptimo obtenido a través del algoritmo del conjunto activo cumple todas las especificaciones de las condiciones anteriores.

Uno de los campos donde habitualmente se ha considerado el análisis de optimización ha sido la teoría de grafos, en particular las redes, y el flujo asignado a los arcos de los mismos. Existen diversos algoritmos de carácter genérico que abordan temas como el establecimiento del camino más corto (coste mínimo) entre dos nodos de una red o la asignación óptima de flujo de la red.

El estudio de redes de tráfico analiza la gestión óptima de las infraestructuras viarias mediante el empleo de mecanismos que reduzcan los costes asociados a la existencia de las mismas. Entre otros aspectos se ha promovido la utilización de la teoría de grafos en el estudio de la representación de las redes de transporte, esta rama ha profundizado en el avance de los procesos de optimización de redes (network optimization). El proceso de planificación del transporte agrupa diversas etapas. Una de ellas es el procedimiento de asignación del flujo de tráfico circulante por la red a los distintos arcos, de forma que se cumpla el criterio de coste mínimo de funcionamiento de la red en su conjunto. Existen numerosos algoritmos que se pueden emplear para la realización de esta asignación. La utilización de cada uno de ellos es una cuestión dependiente de las condiciones particulares establecidas para los distintos supuestos básicos sobre el funcionamiento y configuración de cada red particular. Básicamente, hay dos tendencias a la hora de establecer la asignación: un primer grupo constituido por aquellos métodos basados en procesos de asignación dinámica, y un segundo grupo que considera procedimientos de asignación estática, bien sea con criterios de temporalidad, consideración de la división del tiempo en periodos, o bien con criterios atemporales. En ambos grupos se pueden considerar situaciones de flujo estocástico o no. Nuestro enfoque se centra sólo en aquellos modelos de asignación estática no estocástica, sin consideración de criterios de temporalidad.

Uno de los principales problemas que presenta la asignación de flujo a los diversos arcos de la red de tráfico es el derivado de la existencia del fenómeno de la congestión. La congestión del tráfico se produce por ser la infraestructura un bien público cuyo uso por parte de los individuos no tiene carácter de apropiación, esto es,

no impide directamente que otros individuos puedan utilizarlo, pero presenta una capacidad de uso limitada. Esto supone que cuantos más personas deseen acceder a la utilización de una infraestructura concreta, mayor será el coste que los usuarios de la misma deberán soportar y, por tanto, menor el beneficio neto obtenido de la utilización de dicha infraestructura viaria. El sistema tradicional en economía para abordar este tipo de situaciones consiste en el establecimiento de un peaje que tienda a igualar el coste privado que cada usuario soporta por circular por la red, con el coste marginal al cual el conjunto de la sociedad tiene que hacer frente.

Los peajes han sido ampliamente estudiados en la literatura sobre economía del transporte, y se encuentran, en la actualidad, en un estado de debate permanente entre partidarios de su aplicación en el entorno de ciudades y áreas urbanas congestionadas y aquellos otros autores que abogan por el establecimiento de medidas alternativas sobre la gestión global del transporte urbano: estudios sobre aparcamientos, transporte público, configuración de la ciudad, etc.

En el presente estudio se ha tratado de combinar ambas situaciones y analizar el grado de aplicación del método del conjunto activo como método de asignación del flujo de tráfico en redes. Se ha considerado un problema de minimización del coste de usuario y del coste de sistema de la red. Dado que ambos problemas, bajo determinados supuestos, son equivalentes, hemos planteado el desarrollo de un método del conjunto activo como procedimiento de asignación de tráfico en redes. Los resultados obtenidos utilizando dicho algoritmo se han comparado con los resultantes del empleo de otros algoritmos alternativos, ampliamente comentados en la literatura.

Para poder efectuar la comparación con otros métodos hemos utilizado el programa de cálculo simbólico Mathematica. Se ha construido un paquete de funciones específicas para la realización de las iteraciones del método del conjunto activo y como método alternativo de comparación hemos considerado el método de Frank-Wolfe, lo que ha supuesto, también, la necesidad de crear un paquete con

funciones específicas para dicho método. Estas funciones, así como otras funciones estándar incluidas en el programa, han sido utilizadas para la ejecución de las diversas iteraciones y la obtención de resultados parciales que mostrasen el camino seguido durante el proceso de solución.

Dada la amplitud del tema considerado, se ha hecho necesaria una limitación del mismo, para lo cual se han establecido una serie de supuestos restrictivos sobre la configuración y desarrollo de la red, así como las características particulares de la demanda y temporalidad. La relajación de estas condiciones abrirá nuevas vías de investigación que creemos deberán ser abordadas en estudios posteriores.

El esquema que hemos seguido en el presente trabajo considera los siguientes capítulos:

En el capítulo 2, a continuación, realizamos una introducción a la optimización de funciones, en un principio sin restricciones y, en un segundo apartado, incorporando conjuntos de restricciones, ya sean de igualdad o desigualdad. Establecemos las condiciones que definen el óptimo y mostramos los fundamentos teóricos del método del conjunto activo, asimismo consideramos el caso de la optimización sujeta a restricciones de acotación, cuando se utiliza el método del conjunto activo.

El capítulo tres se centra en el estudio de las redes de tráfico. En el primer apartado presentamos los fundamentos para el desarrollo de las redes de tráfico, así como el establecimiento del equilibrio. El segundo apartado se centra en el estudio de la función de coste asociada a la utilización de los arcos de la red y, en el establecimiento de la función objetivo que servirá para configurar el coste total del usuario. El tercer apartado recogerá algunos de los algoritmos tradicionales planteados para el análisis de redes de tráfico de acuerdo con las especificaciones comentadas y con flujo estático y variable. Asimismo, en este capítulo también se realiza una aproximación a la existencia de interacción entre las funciones de costes de

los diversos arcos de una red. Se considera, en particular, una red con interacción asimétrica y se analiza la posible utilidad del método del conjunto activo para abordar el análisis de este tipo de redes que no pueden ser realizados empleando otros métodos tradicionales. Por último, se comenta la existencia de otras aproximaciones alternativas al análisis de redes: modelos de asignación dinámica y modelos continuos.

El cuarto capítulo presenta un aspecto fundamentalmente económico y pretende presentar una breve revisión del estado actual del debate sobre la implementación de los peajes por la utilización de infraestructuras. Después de una breve introducción a la teoría económica sobre peajes, se realiza un estudio sobre las características del peaje ideal, según han establecido los numerosos expertos económicos que han estudiado el tema con anterioridad. Por último, se comentan algunos de los sistemas de peaje que existen en diversos países destacando los resultados derivados de su implantación; así como una referencia a los posibles sistemas alternativos al peaje que defiende otros autores.

El quinto capítulo considera cuatro apartados. En el primer apartado, se establecen los supuestos sobre redes que se van a utilizar a la hora de diseñar nuestros modelos de redes de tráfico; el segundo apartado desarrolla un método del conjunto activo que se implementará en el lenguaje del programa de cálculo simbólico Mathematica, adaptado a las particulares características establecidas para las redes objeto de estudio. En este apartado se indican cuáles son las funciones generadas y cómo se pueden emplear desde el programa Mathematica para llevar a cabo las distintas iteraciones del algoritmo. El tercer apartado considera el establecimiento del peaje óptimo tradicional y muestra que, bajo ciertas circunstancias, la implementación de un sistema de tarifas por la utilización de los arcos de la red afecta al patrón de flujos óptimos de la misma, y por tanto, supone que el valor de la función objetivo no sea mínimo. Se considera un sistema de cálculo del peaje alternativo que proporciona un valor final de la función objetivo más próximo al mínimo que el obtenido a través de la utilización del peaje óptimo tradicional. Para la realización de todos los cálculos en este apartado se utilizará nuestro algoritmo del conjunto activo aplicado a redes

con peaje. Por último, el cuarto apartado muestra los resultados obtenidos al aplicar a una red sin peajes el algoritmo del conjunto activo que hemos diseñado, estos resultados se comparan con los obtenidos a través de la utilización del método de Frank-Wolfe aplicado a la optimización de redes, tal y como lo han establecido varios autores. Básicamente, nuestro análisis se centrará en dos parámetros: velocidad de convergencia hacia la solución óptima y valor final de la función objetivo.

Por último, y para finalizar nuestro estudio, recogemos en el capítulo seis las conclusiones obtenidas a lo largo del mismo, así como las posibles vías de estudio de posteriores desarrollos que creemos podrían derivarse de la presente investigación.

CAPITULO 2

OPTIMIZACION DE

FUNCIONES

2. OPTIMIZACION DE FUNCIONES.

Consideremos un problema general de optimización dado por:

$$\begin{aligned} & \underset{x \in \mathbb{R}^n}{\text{Min}} F(x) && (1) \\ & \text{sujeto a} \\ & c_i(x) = 0 \quad i = 1, 2, \dots, j \\ & c_i(x) \geq 0 \quad i = j + 1, j + 2, \dots, m \end{aligned}$$

donde $F(x)$ es una función definida en $A = \text{Dom}(F(x)) \subset \mathbb{R}^n$ que deseamos optimizar y se le denomina *función objetivo*; en el problema anterior deseamos hallar, concretamente, un mínimo de dicha función, y $c_i(x)$ constituyen un grupo de funciones, llamadas *restricciones* que acotan el posible espacio de búsqueda del mínimo de dicha función; por su parte, n es el número de variables, y m es el número de restricciones del problema.

La solución a un problema de este tipo se produce en el llamado *punto factible* x' , que es aquel punto donde i) la función $F(x)$ está definida y ii) se cumplen las condiciones impuestas por las restricciones. Evidentemente, no tenemos garantizado que exista siempre un punto factible único, de ahí que debemos dotarnos de un método para elegir, entre el conjunto de puntos factibles, aquél que tiene el valor óptimo global de todos ellos. Al punto factible que cumpla esta condición lo denominaremos *punto óptimo* x^* . La obtención de dicho punto es el objetivo de todos los métodos de optimización existentes.

Debemos distinguir la existencia de dos categorías de puntos óptimos, los llamados puntos óptimos locales y los denominados puntos óptimos globales. El punto óptimo local de una función puede ser, a su vez, de dos tipos:

1) x^* se denomina un mínimo local fuerte si $\exists \delta > 0$ tal que

i) $F(x)$ está definido en la bola abierta de centro x^* y radio δ :

$$N(x^*, \delta) \cap \text{Dom}(F(x)).$$

ii) $F(x^*) < F(y)$, $\forall y \in N(x^*, \delta) \cap \text{Dom}(F(x))$, $y \neq x^*$.

2) x^* se denomina un mínimo local débil si $\exists \delta > 0$ tal que

i) $F(x)$ está definido en $N(x^*, \delta) \cap \text{Dom}(F(x))$.

ii) $F(x^*) \leq F(y)$, $\forall y \in N(x^*, \delta)$, $y \neq x^*$.

iii) $\exists y \neq x^*$ tal que $F(x^*) = F(y)$.

Por su parte un punto óptimo, x^* , se denomina global si: no existe $y \in \text{Dom}(F(x))$ tal que y cumpla las condiciones de las restricciones y $F(y) < F(x^*)$. En este caso el punto x^* sería aquel donde la función alcanza su valor mínimo global.

La resolución del problema (1) consistiría en estudiar todos los puntos factibles de la región acotada y ver cual es el mínimo. Lógicamente este proceso podría resultar realmente arduo si existiese un número considerable de puntos factibles y, por lo tanto, sería interesante dotarnos de algún algoritmo mediante el cual podamos obtener dicho punto óptimo sin necesidad de recurrir al estudio de todos los puntos factibles. A lo largo de la historia de la matemática se han realizado numerosos intentos de establecer un algoritmo general de resolución del problema; no obstante, pronto se comprobó que todos los algoritmos obtenidos tenían un carácter parcial y, por lo tanto, tan sólo eran aplicables a determinados tipos concretos de problemas de optimización. La primera distinción importante surge a la hora de considerar si el problema de optimización está sujeto a restricciones o no y, en una segunda instancia, si es univariante o multivariante, esto es, si la función objetivo es una función de una sola variable, o por el contrario tiene una dimensión superior a 1.

2.1. Optimización sin restricciones.

2.1.1. Caso Univariante.

El problema de optimización sin restricciones en el caso univariante presenta la siguiente formulación:

$$\underset{x \in \mathbb{R}^1}{\text{Min}} F(x) \quad (2)$$

Si F es dos veces diferenciable y continua entonces x^* será un mínimo local débil si:

- i) $F'(x^*)=0$; con lo que x^* se denomina *punto estacionario*.
- ii) $F''(x^*) \geq 0$.

Estas condiciones, lógicamente, no garantizan que el punto x^* sea el óptimo; existen dos condiciones, denominadas condiciones suficientes, que garantizan que un punto estacionario sea un mínimo local fuerte:

- i) $F'(x^*)=0$;
- ii) $F''(x^*) > 0$.

El problema a la hora de la comprobación de las condiciones señaladas se plantea por el hecho de la exigencia de la diferenciabilidad dos de las funciones objetivo, lo que no siempre será posible, y lleve a la necesidad de plantear métodos alternativos que no tengan en cuenta las derivadas de la función.

2.1.2. Caso Multivariante.

El caso multivariante responde a un problema del tipo:

$$\underset{x \in \mathbb{R}^n}{\text{Min}} F(x) \quad (3)$$

Las condiciones necesarias y suficientes de mínimo local son muy similares a las del caso univariante, considerando la diferenciabilidad y continuidad apropiada de la función objetivo. Así pues, x^* será un mínimo local débil si:

- i) $\|g(x^*)\| = 0$; donde $g(x)$ es la función gradiente.
- ii) $G(x^*)$ es semi-definida positiva; donde $G(x)$ es la matriz Hessiana¹.

Por su parte, las condiciones suficientes, para que x^* sea un mínimo local fuerte serán:

- i) $\|g(x^*)\| = 0$.
- ii) $G(x^*)$ es definida positiva.

La obtención de un óptimo global resulta un poco más compleja y requiere que la función objetivo presente ciertas propiedades bondadosas que nos ayuden a la hora de su tratamiento. Así pues, tanto en el caso univariante como multivariante, diremos que:

Un punto estacionario x^* es un mínimo global si $F(x)$ es una función convexa².

¹ La matriz será seimidefinida positiva si todos los menores principales, de todos los órdenes son no negativos.

² $F(x)$ es una función convexa, caso univariante, si y sólo si:

2.2. Optimización sujeta a restricciones.

La optimización de una función objetivo sujeta a restricciones debe comenzar por la consideración sobre la naturaleza de la función objetivo y las restricciones. La mayoría de los algoritmos existentes abordan el caso de una función objetivo general sujeta a restricciones lineales, siendo menos frecuente la existencia de algoritmos de resolución para problemas con restricciones no lineales. El caso más sencillo sería el de una función objetivo lineal sujeta a restricciones lineales, para cuya resolución se utilizan habitualmente el método del Simplex, o el método de eliminación gaussiana, ambos ampliamente difundidos en la literatura sobre optimización [68, 134]. En el presente estudio nos centraremos en funciones objetivo no lineales lo que nos plantea la necesidad de otros métodos alternativos.

Las restricciones, sean lineales o no, pueden tener signos de igualdad o de desigualdad representándose como:

$$\begin{aligned} a'x - \beta &= 0 \\ a'x - \beta &\geq 0 \\ a'x - \beta &\leq 0 \end{aligned} \tag{4}$$

donde a es el vector columna del gradiente de la restricción (si ésta es univariante, es un escalar correspondiente al diferencial de la restricción) y β es un escalar, llamado coeficiente del lado derecho.

$$\begin{aligned} F(\lambda x^1 + (1-\lambda)x^2) &\leq \lambda F(x^1) + (1-\lambda)F(x^2) \\ \forall \lambda &\in [0,1] \\ \forall x^1, x^2 &\in \text{Dom}(F(x)) \end{aligned}$$

Si tenemos varias restricciones se pueden representar mediante una matriz A , cuyas filas sean los vectores columnas a , y un vector columna b , formado por los coeficientes del lado derecho de las distintas restricciones.

El caso particular en el que sólo exista una variable dentro de la restricción se denomina cota simple de la variable, y lo que nos indica es que dicha variable está acotada, distinguiéndose tres casos:

- i) Si $x_i = \beta$, entonces la variable x_i es una constante.
- ii) Si $x_i \leq \beta$, entonces tiene una cota superior.
- iii) Si $x_i \geq \beta$, entonces tiene una cota inferior.

Consideremos ahora las distintas situaciones bajo la perspectiva de la existencia de restricciones lineales.

2.2.1. Problema de optimización con restricciones de igualdad.

El problema de optimización de una función objetivo general, sea lineal o no lineal, con restricciones lineales de igualdad se escribe, en forma abreviada, como:

$$\begin{aligned} & \underset{x \in \mathbb{R}^n}{\text{Min}} F(x) & (5) \\ & \text{sujeto a } Ax = b \end{aligned}$$

La técnica más intuitiva para resolver un problema de este tipo consiste en la eliminación de m variables (donde m es el número de restricciones) mediante la resolución del sistema de ecuaciones que forma el conjunto de restricciones, convirtiéndolo de esta forma en un problema de optimización sin restricciones de $n-m$ variables, donde n es el número de variables del problema original. Un método de este tipo

supone que tanto $F(x)$ como las restricciones son funciones continuas y diferenciables y que $m < n$. Estas técnicas de resolución se fundamentan en la existencia del siguiente teorema:

Teorema I (véase [12]): Sea $F(x)$ continuamente diferenciable en un abierto $X \subset \mathbb{R}^n$ y $Ax-b=0$, donde A es de orden $m \times n$, con $m < n$ y $\text{rango}(A) = m$. Supongamos que x' es un mínimo local de $F(x)$ que satisface el sistema de restricciones. Entonces existe un vector único $\lambda = (\lambda_1, \dots, \lambda_m)$, donde $\lambda_i \in \mathbb{R}$, $\forall i$, denominado vector de multiplicadores, tal que:

$$-F(x') = \sum_{i=1}^m \sum_{j=1}^n \lambda_i a_{ij} x'_j - b_i \quad (6)$$

esto es:

$$-\left[\frac{\partial F(x')}{\partial x_j} \right] = \sum_{i=1}^m \sum_{j=1}^n \lambda_i a_{ij} = \sum_{i=1}^m \lambda_i a_{ij}, \quad j = 1, \dots, n \quad (7)$$

o lo que es lo mismo:

$$-g(x') = \sum_{i=1}^m \lambda_i a_{ij}, \quad j = 1, \dots, n \quad (8)$$

en otras palabras, el vector gradiente de la función F en x' es una combinación lineal de los vectores gradientes de las restricciones en dicho punto. En nuestro caso, con restricciones lineales, resulta una combinación lineal de los coeficientes de la matriz A .

La relación establecida en este teorema nos permite la construcción de la denominada *función de Lagrange* o lagrangiana, definida como:

$$L(x, \lambda) = F(x) + \sum_{i=1}^m \lambda_i r_i(x) \quad (9)$$

donde $r_i(x)$, $i=1, \dots, m$ son las restricciones del problema, que en nuestro caso serán funciones lineales. A los multiplicadores λ_i se les denomina *multiplicadores de Lagrange*.

Establecida la función de Lagrange podemos considerar las condiciones necesarias y suficientes para que un punto factible sea un mínimo local:

a) Condiciones necesarias:

- i) $Ax^* - b = 0$, o lo que es lo mismo, $r_i(x^*) = 0, \forall i=1, \dots, m$.
- ii) $\nabla(L, x^*) = \theta$, lo que supone que el gradiente de la función de Lagrange, se anula en el punto x^* .

b) Condiciones suficientes:

- i) Cumple las condiciones necesarias y además
- ii) $z^T G(L, x^*) z \geq 0, \forall z \in \mathbb{R}^n$ tal que $(\nabla(r_i(x^*)))^T z = 0; i=1, \dots, m$.

donde $G(L, x^*)$ es la matriz Hessiana de la función de Lagrange.

El método de la función de Lagrange es el más habitual dentro de la literatura económica como método de resolución de problemas de optimización, puesto que dentro de la denominada Programación Cóncava, permite el establecimiento de las condiciones de Kuhn - Tucker, como condiciones necesarias y suficientes de optimización [12, 107].

Otras aproximaciones al método de eliminación de variables se basan en la existencia del espacio nulo de la matriz de restricciones. Así, dado nuestro problema de optimización (5), si existe un punto x' que satisface $Ax'=b$, cualquier punto $x \in \mathbb{R}^n$, tal que $x=x'+d$ satisfará $Ax=b$ si y sólo si d es un vector del espacio nulo de A (ver [114])³.

En esta situación podemos reducir el problema (5) a un problema sin restricciones del tipo:

$$\text{Min}_{y \in \mathbb{R}^{n-m}} f(y) \quad (10)$$

donde $f(y) = F(x'+Zy)$, siendo Z una matriz del espacio nulo de A , esto es, $A^t Z=0$. La resolución del problema (10) permite la aplicación de métodos de optimización sin restricciones.

El cálculo de la matriz Z se puede realizar mediante varios métodos alternativos:

a) Método de Eliminación de Variables.

En este método se divide la matriz A en dos $[A^D, A^I]$, donde A^D tiene rango m , siendo m es el número de restricciones, y es una matriz cuadrada $m \times m$, mientras que A^I es una matriz de orden $(n-m) \times m$, y n es el número de variables.

En esta situación se puede calcular la matriz del espacio nulo Z , como :

$$Z = \begin{pmatrix} -(A^D)^{-1} A^I \\ I \end{pmatrix} \quad (11)$$

donde I es una matriz identidad de orden $(n-m)$.

³ Se dice que d es un vector del espacio nulo de A , si $A \cdot d=0$

Este método implica que la matriz A^D es no singular, y por tanto, posee matriz inversa. En caso de que esto no se cumpliera sería necesario utilizar otro método de cálculo.

b) Método de la Matriz Ortogonal.

Consiste en calcular la factorización QR de la matriz A^t , de forma tal que podamos escribir:

$$QA^t = \begin{pmatrix} R \\ 0 \end{pmatrix} \quad (12)$$

donde la matriz $Q=(Q_1 \ Q_2)$ es de orden $n \times n$, y las submatrices Q_1 y Q_2 son, respectivamente, de orden $m \times n$ y $(n-m) \times n$.

La matriz del espacio nulo, $Z=Q_2^t$. Esto es, la traspuesta de la matriz que forman las $(n-m)$ filas de la matriz Q .

Otra posibilidad es emplear la factorización LQ, donde Q es una matriz ortonormal de orden $n \times n$ y L es una matriz triangular inferior de orden $m \times m$. Siendo:

$$AQ = (L \ 0) \quad (13)$$

En este caso, la matriz Z sería la formada por las $n-m$ últimas columnas de la matriz Q .

Podemos calcular el gradiente y la Hessiana de la nueva función $f(y)$ a partir de los valores del gradiente y Hessiana de la función anterior F , de forma que obtenemos:

- a) Gradiente reducido: $\nabla_y f(y) = Z^T g(x)$.
- b) Hessiana reducida: $\nabla_y^2 f(y) = Z^T G(x) Z$.

Las condiciones de optimalidad de un punto y^* serán:

- a) Condiciones necesarias:

$$\nabla_y f(y) = 0$$

$\nabla_y^2 f(y)$ semidefinida positiva.

- b) Condiciones suficientes:

$$\nabla_y f(y) = 0$$

$\nabla_y^2 f(y)$ definida positiva.

Gill, Murray & Wright [68] plantean un algoritmo para la resolución de problemas de optimización de funciones con restricciones lineales de igualdad, consistente en generar una secuencia de iterantes, puntos factibles, a partir de un punto factible inicial de forma tal que deberán cumplir la denominada *condición de ortogonalidad*:

Si x_k es el punto factible en la iteración k -ésima, y p_k es la dirección de búsqueda en dicha iteración, entonces se cumple que $A p_k = 0$, o lo que es equivalente, $p_k = Z p_z$.

El algoritmo planteado por los autores tendría los siguientes pasos:

0) Cálculo del punto inicial factible x_0 .

1) Test de Convergencia: Si x_k cumple las condiciones de convergencia, entonces se detiene el proceso operativo y se considera $x_k = x^*$ como el punto óptimo.

2) Cálculo de una dirección factible: Calcular un vector p_z de orden $(n-m)$ distinto de cero, tal que el vector de dirección de búsqueda de una solución mejorada será: $p_k = Zp_z$.

3) Cálculo de la longitud de paso: Calcular un escalar $\alpha_k > 0$, que nos determine la longitud máxima que nos podemos desplazar en la dirección factible sin violar las restricciones del problema y tal que $F(x_k + \alpha_k p_k) < F(x_k)$.

4) Actualizar el punto factible:

$$x_{k+1} \leftarrow x_k + \alpha_k p_k ; k \leftarrow k+1, \text{ y volver al paso 1.}$$

El algoritmo (Ilustración 1) podría describirse gráficamente del siguiente modo: Nos situamos en un punto inicial factible, a continuación determinamos una posible dirección de búsqueda, que deberá ser descendente puesto que estamos minimizando⁴, de forma tal que garanticemos una reducción en el valor de la función objetivo $F(x)$; la cuantía de dicha reducción vendrá, además, determinada por la longitud del paso que podemos dar en la dirección de búsqueda, la longitud del paso, asimismo, deberá permitirnos obtener una variación significativa en el valor de la función.

Esta idea de la variación significativa nos permite establecer un test de convergencia en función de las diferencias entre puntos factibles, así si dado un umbral ϵ , se cumple que $|Z^T g(x_k)| < \epsilon$, se detendrá el proceso de búsqueda y se considera el punto x_k como el punto óptimo.

⁴ La posible dirección de búsqueda deberá cumplir que $g_k^T Z p_z < 0$.

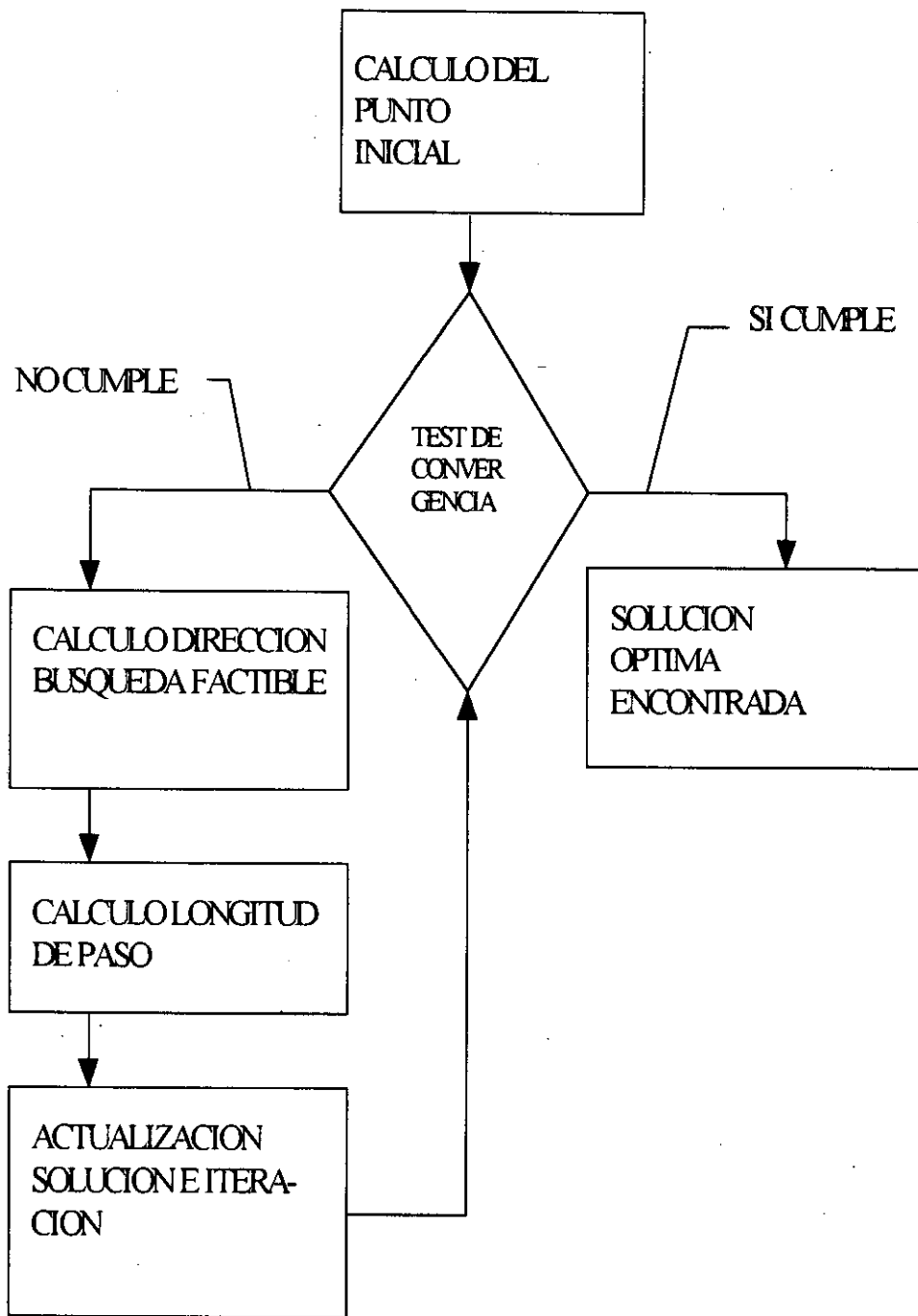


Ilustración 1: Diagrama de Flujo

2.2.2. Problema de optimización con restricciones de desigualdad.

El problema de optimización general con restricciones lineales de desigualdad se puede escribir:

$$\begin{aligned} & \text{Min}_{x \in \mathbb{R}^n} F(x) \\ & \text{sujeto a} \\ & Ax \geq b \end{aligned} \tag{14}$$

donde A es la matriz formada por los vectores fila a_i , donde i es el índice de la restricción ($i=1, \dots, m$), x es el vector columna de variables x_j ($j=1, \dots, n$), y el vector b recoge los coeficientes del lado derecho del conjunto de restricciones.

Dentro de este problema de optimización podemos distinguir entre dos grupos de restricciones. Aquellas que para un punto factible dado, x' , se cumplen con signo de igualdad $a_i x' = b_i$, denominadas *restricciones activas*, y las que se cumplen con signo de desigualdad, $a_i x' > b_i$, denominadas *restricciones no activas* (o inactivas).

Esta diferente categoría de las restricciones va a jugar un papel importante a la hora de optimizar. Las restricciones activas van a limitar las posibilidades de movimiento a partir de un punto factible hacia a otro punto, puesto que modera el movimiento en las distintas direcciones. Así, suponiendo que tenemos una restricción activa $a_i x' = b_i$, si un vector p , cumple que $a_i p = 0$, entonces podemos desplazarnos a lo largo de la dirección marcada por el vector p , puesto que la restricción permanecerá activa $\forall \alpha \in \mathbb{R}$ tal que $a_i(x + \alpha p) = b_i$; por otro lado, si p cumple que $a_i p > 0$ entonces $a_i(x + \alpha p) = a_i x + \alpha a_i p = b_i + \alpha a_i p > b_i$, con lo que la restricción se vuelve inactiva.

Las restricciones inactivas, por su parte, permiten una amplia movilidad, al no limitar el movimiento desde el punto factible en ninguna dirección. Así, para cualquier

vector p , $x^* + \epsilon p$ será un punto factible con respecto a la restricción inactiva para un ϵ de tamaño apropiado.

Las condiciones necesarias y suficientes de optimización se pueden establecer considerando una matriz \hat{A} con los coeficientes de las restricciones activas en el punto x^* , tal que $\hat{A}x^* = b$. Sea Z la matriz cuyas columnas forman una base del conjunto de vectores ortogonales a las filas de \hat{A} , tales que $\forall p$ se cumple que $\hat{A}p=0$, entonces:

a) Condiciones necesarias:

- i) $Ax^* \geq b$, con $\hat{A}x^* = b$.
- ii) $Z^t g(x^*) = 0$, o lo que es equivalente: $g(x^*) = \hat{A}\lambda^*$.
- iii) $\lambda_i^* \geq 0$, $i=1, \dots, t$.
- iv) $Z^t G(x^*) Z$ es semidefinida positiva.

donde λ_i^* son los multiplicadores de Lagrange asociados a cada restricción activa, y $Z^t G(x^*) Z$ es la matriz Hessiana proyectada.

b) Condiciones suficientes:

- i) $Ax^* \geq b$, con $\hat{A}x^* = b$.
- ii) $Z^t g(x^*) = 0$, o lo que es equivalente: $g(x^*) = \hat{A}\lambda^*$.
- iii) $\lambda_i^* > 0$, $i=1, \dots, t$.
- iv) $Z^t G(x^*) Z$ es definida positiva.

Uno de los métodos de resolución de los problemas de optimización de funciones no lineales con restricciones lineales, es el empleo de los *algoritmos del conjunto activo*.

2.2.3. Método del Conjunto Activo.

El método del conjunto activo permite la generalización de los métodos de resolución de problemas de optimización con restricciones de igualdad a las situaciones con restricciones de desigualdad siempre y cuando el número de restricciones del problema sea inferior al número de variables. En este método se considera un subconjunto del conjunto restricciones que se cumplen con signo de igualdad, y que se denomina el *conjunto activo* A . En cada iteración de este método tratamos de localizar un punto factible x^k como resultado de resolver un problema de restricciones de igualdad:

$$\begin{aligned} & \text{Min}_d f(x^k + d) \\ & \text{sujeto a } a_i^t d = 0 \quad i \in A \end{aligned} \tag{15}$$

La idea del Conjunto Activo ha llevado a diversas implementaciones del mismo [63, 68, 196]. Un posible algoritmo de resolución sería el que plantean Gill, Murray & Wright [68,69], que parte de considerar un grupo de las restricciones del problema y empezar a trabajar con las mismas, a estas restricciones las denominaremos *conjunto de trabajo*. Evidentemente, el conjunto de trabajo no tiene porque coincidir con el conjunto activo, esto es, nos deberemos dotar de algún método mediante el cual poder incluir o retirar restricciones del conjunto de trabajo según sea necesario.

El modelo de algoritmo propuesto para la resolución de este tipo de problemas es un método iterativo de búsqueda de una solución factible a partir de la consideración de un cierto número de restricciones activas (el conjunto de trabajo). Se fundamenta en la idea que ya comentada de la eliminación de variables en el caso de problemas con restricciones de igualdad. Si denotamos por k el número de iteración actual, tendremos las siguientes notaciones:

- I_k : conjunto de índices de las restricciones que forman el conjunto de trabajo.
- t_k : número de restricciones del conjunto de trabajo.
- x_k : solución obtenida en la k-ésima iteración.
- \hat{A}_k : coeficientes de las restricciones del conjunto de trabajo = conjunto de trabajo.
- b_k : vector de coeficientes del lado derecho de las restricciones del conjunto de trabajo.
- Z_k : base del subespacio de vectores ortogonal a las filas del conjunto de trabajo.

Partimos, así pues, de un punto inicial: x_0 , obtenido mediante algún procedimiento inicial. A partir de dicho punto los pasos que sigue el algoritmo propuesto (Ilustración 2) son los siguientes:

- Paso 0: Solución inicial.

Calculamos una solución inicial x_0 , y un conjunto de trabajo inicial.

- Paso 1: Test de convergencia.

Se necesita la existencia de un test que nos permita decidir si continuamos el proceso o consideramos que el valor obtenido en la k-ésima iteración se aproxima lo suficiente al valor óptimo, como para detener el proceso de búsqueda y considerarlo la solución al problema. Para comprobar el avance en cada iteración se analizará que el valor

del gradiente de de la función en el punto obtenido es significativamente distinto de cero, o como test alternativo que la diferencia en el valor de la función objetivo entre dos iterantes es significativa.

- Paso 2: Elección de la medida a seguir.

Se plantean dos opciones posibles:

a) Continuar maximizando en el subespacio actual (esto es, mantener el conjunto de trabajo) ; vamos al Paso 3.

b) Borrar una restricción del conjunto de trabajo (cambiar el subespacio de búsqueda); vamos al Paso 6.

- Paso 3: Cálculo de la dirección de búsqueda factible.

Calculamos un vector p_z de orden $(n-t_k)$ cuyas componentes sean distintas de cero; y obtenemos de esta forma la dirección de búsqueda como:

$$p_k = Z_k p_z \quad (16)$$

- Paso 4: Determinar la longitud del paso.

Calculamos α , la máxima longitud de paso factible ($\alpha \geq 0$) a lo largo de la dirección de búsqueda (p_k). Se deberá cumplir que:

$$F(x_k + \alpha_k p_k) < F(x_k) \quad (17)$$

donde $\alpha_k \leq \alpha$.

A continuación optaremos en función de la siguiente alternativa:

- a) Si $\alpha_k = \alpha$, entonces seguimos en el Paso 5.
- b) Si $\alpha_k < \alpha$, entonces vamos al Paso 7.

- Paso 5: Añadir una restricción al conjunto de trabajo.

Si α_k es el paso que activa la restricción con índice r , añadimos r al conjunto de índices I_k , y modificamos el resto de las cantidades de forma oportuna. Pasamos a continuación al Paso 7.

- Paso 6: Eliminar una restricción del conjunto de trabajo.

Elegimos una restricción (s) y la borramos del conjunto de trabajo, esto supone la realización de un conjunto de cambios oportunos en la matriz Z . Actualizamos el conjunto de índices y las variables afectadas. Volvemos al Paso 1.

- Paso 7: Actualizamos la solución estimada.

Actualizamos los valores de la solución estimada en la k -ésima iteración y el número de estimación.

$$x_{k+1} = x_k + \alpha_k p_k \tag{18}$$
$$k = k + 1$$

y retornamos al Paso 1.

Gráficamente, el diagrama de flujo de este algoritmo es el siguiente:

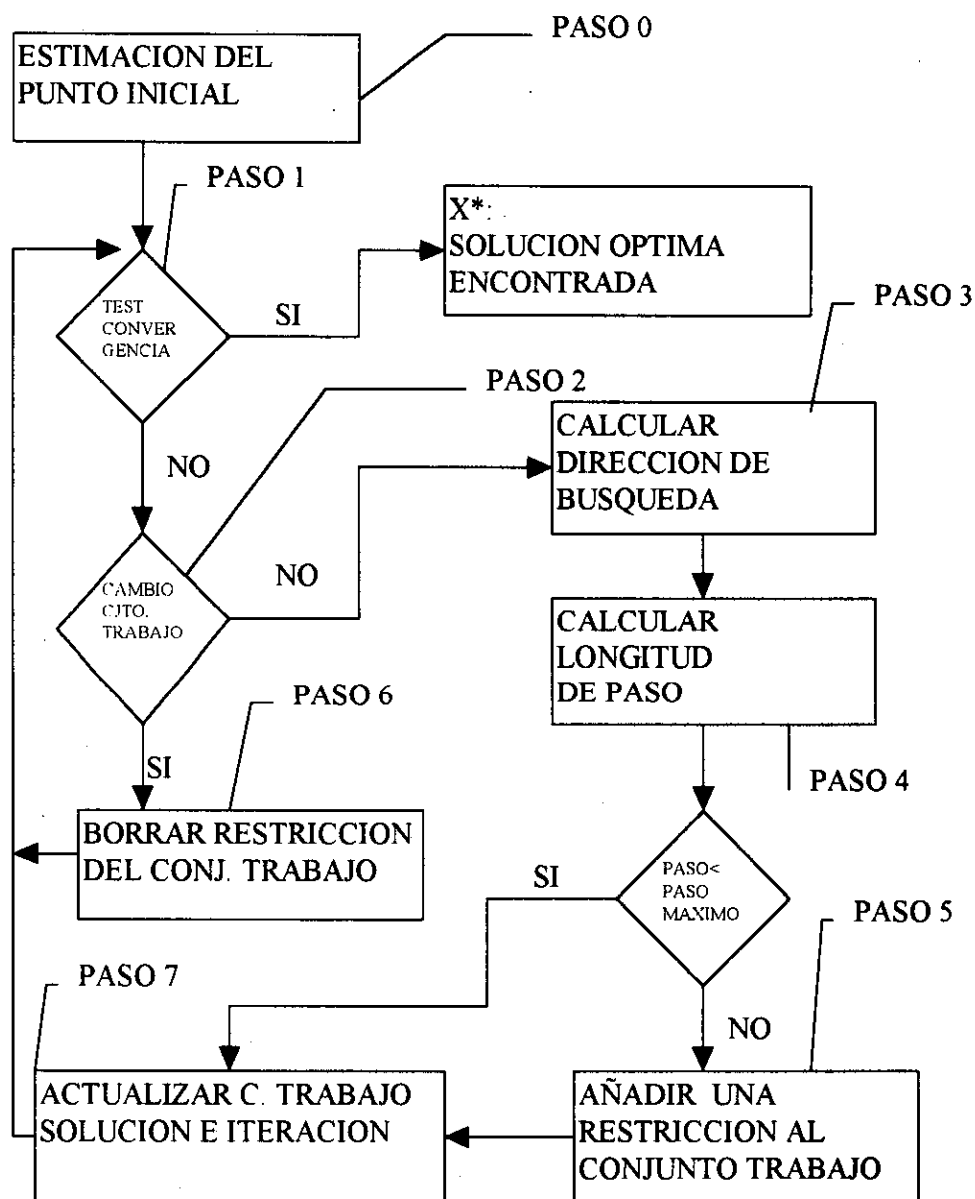


Ilustración 2: Diagrama de Flujo del Método del Conjunto Activo

La descripción más detallada de los pasos a seguir es la siguiente:

2.2.3.1. Elección del punto y conjunto de trabajo iniciales.

El primer paso (Paso 0) del método del conjunto activo consiste en la elección de un punto inicial para proceder a la realización de las iteraciones pertinentes que nos aproximen al punto solución, y en la determinación de nuestro primer conjunto de trabajo, esto es, el conjunto de restricciones activas iniciales. Se convierte este paso, así pues, en un paso clave, puesto que una buena elección inicial puede condicionar de forma considerable la eficacia del método. El caso ideal sería obtener en este primer paso la solución, pero nos conformaremos con la obtención de un punto de partida lo más ajustado posible a la misma, de forma tal que, combinado con una buena dirección de búsqueda y una longitud de paso adecuada, aumente de forma considerable la eficiencia en el resultado del desarrollo del algoritmo.

Fletcher [63] propone la elección de un conjunto de trabajo arbitrario dentro del conjunto de restricciones (m restricciones, donde $m < n$, n variables) y la determinación de un punto inicial x^1 , donde x^1 es la solución del sistema planteado por el conjunto de trabajo. No existe limitación al número de restricciones a incluir en el conjunto de trabajo, por lo que podemos optar por la inclusión del global de las mismas con el objetivo de obtener la mayor información posible sobre la acotación del área de optimización.

Gill, Murray & Wright [68], por su parte, consideran que la elección del conjunto de trabajo inicial es un hecho de mayor trascendencia puesto que una elección apropiada redundará en la eficiencia de los cálculos posteriores; estos autores denominan "*conjunto crash*" para el conjunto de trabajo inicial. En un principio dentro del conjunto inicial es obligatorio incluir todas las restricciones de igualdad linealmente independientes que existan dentro del conjunto de restricciones; a partir de aquí, en el caso de utilizar, posteriormente, en el proceso de optimización algún método del espacio nulo, será conveniente que el conjunto inicial incluya cuantas más de las restantes restricciones sea posible.

La obtención del punto inicial requiere la resolución de un problema de optimización lineal con restricciones lineales. El método para la elección del punto factible sigue la técnica del método simplex. Supongamos que tenemos m restricciones, funciones lineales, de desigualdad de la forma:

$$A x \leq b \quad (19)$$

entonces construimos una función objetivo lineal:

$$\bar{F}(x) = \sum_{i \in V} (b_i - a_i^T x) \quad (20)$$

donde V es el conjunto de índices de las restricciones no factibles en el punto x . Esta función objetivo es la suma en módulo de las restricciones violadas; de ahí que la solución se obtenga mediante la resolución del problema de minimización:

$$\begin{aligned} & \text{Min } \bar{F}(x) && (21) \\ & \text{sujeto a:} \\ & a_i^T x - b_i \geq 0 \quad \forall i \in V \end{aligned}$$

La resolución de este problema se puede realizar mediante el método simplex de Dantzig [43], que figura implementado en numerosos paquetes informáticos de investigación operativa, así como programas de cálculo simbólico; o bien, mediante la utilización de otros métodos iterativos como sugiere Fletcher [63]. La aplicación del método simplex requiere que el número de restricciones sea superior al de variables, esto es, $m > n$, de forma tal que obtengamos un punto factible inicial mediante la resolución de un sistema de n restricciones linealmente independientes seleccionadas para el "conjunto crash". Si $m < n$ entonces el método simplex no puede ser aplicado y se deben utilizar otros métodos como el descrito por Gill, Murray & Wright [68].

De esta forma obtendremos una solución que nos permite determinar un primer conjunto activo (conjunto de trabajo) y una solución inicial; este primer conjunto activo estará constituido por aquellas restricciones que son activas para la solución inicial. Ahora, el siguiente paso es establecer una forma de considerar una dirección de búsqueda y calcular la longitud del paso que debemos tomar.

2.2.3.2. Cálculo de la dirección de búsqueda.

Para proceder al cálculo de la dirección de movimiento es necesario obtener un vector p_z que será de orden $(n - t_k)$, donde n es el número de variables del problema y t_k es el número de restricciones activas en la k -ésima iteración⁵. La dirección de búsqueda vendrá dada por:

$$p_k = Z p_z \quad (22)$$

Para obtener este vector existen varios métodos disponibles, en [68] se proponen diversos tipos, en particular, los métodos más habituales en la literatura son: el método del mayor paso descendente, el método de Newton y los métodos Quasi - Newton. Estos métodos se basan en la idea de construir la información sobre la curvatura de la función objetivo utilizando la información obtenida a partir de la observación de F , g y G .

⁵ En este procedimiento es necesario el cálculo de una matriz auxiliar Z . La obtención de dicha matriz se produce a través de la factorización LQ de la matriz \hat{A} .

$$\hat{A}Q = (L \ 0)$$

donde Q es una matriz de orden $n \times n$ y L es una matriz triangular inferior de orden $t \times t$, donde $n > t$. Z será la matriz formada por las $(n-t)$ últimas columnas de Q .

a) Método Newton.

El método de Newton requiere el cálculo de la matriz Hessiana de la función objetivo, lo que conlleva la obtención de las segundas derivadas de dicha función, opción no siempre fácilmente disponible.

En el método de Newton, una vez calculado un punto inicial x_0 , procedemos a la evaluación del valor del gradiente en dicho punto g_0 , y de la hessiana G_0 ; de esta forma podemos calcular la dirección de búsqueda mediante la resolución de:

$$G_z p_z = -g_z \quad (23)$$

donde $g_z = Z^T g_k$, $G_z = Z^T G_k Z$, para la iteración k .

El nuevo punto solución será:

$$x_k^* = x_{k-1}^* + p_{k-1} \quad (24)$$

para $p_{k-1} = Z p_z$.

El método de Newton es bastante sencillo de aplicar, y la única condición exigida es que la matriz G_z sea definida positiva lo que garantizaría que p_{k-1} sea una dirección descendente.

b) Métodos Quasi-Newton.

Los métodos Quasi-Newton se fundamentan en la llamada condición Quasi-Newton que se expresa como:

$$B_{k+1} s_k = y_k \quad (25)$$

donde B_k es una matriz que aproxima a la matriz Hessiana, y s_k es el cambio obtenido en la k -ésima iteración en el punto x , esto es: $s_k = x_{k+1} - x_k$, que dado que estamos trabajando con un método del conjunto activo nos permite afirmar que la distancia entre dos soluciones de iteraciones sucesivas deberá ser: $s_k = \alpha_k p_k$; mientras que $y_k = g_{k+1} - g_k$ recoge el cambio en el gradiente.

La curvatura de F a lo largo de s_k viene dada por :

$$s_k^T G_k s_k \approx (g(x_k + s_k) - g_k)^T s_k \quad (26)$$

utilizando sólo la información de las derivadas de 1º orden⁶. Si consideramos que al principio de la k -ésima iteración tenemos una aproximación⁷ de la Hessiana B_k , entonces podemos calcular la dirección de búsqueda p_z a través de la resolución del sistema:

$$B_z p_z = -g_z \quad (27)$$

Cada vez que obtengamos un nuevo punto o iterante tendremos que realizar una nueva aproximación de la Hessiana en dicho punto, podemos entonces emplear para ello la siguiente formulación, basada en la adaptación de la Hessiana anterior⁸, mediante la fórmula de Broyden-Fletcher-Goldfarb- Shanno (BFGS):

- Si $s_k = x_{k+1} - x_k = \alpha_k p_k$, recoge el cambio de la solución en la k -ésima iteración tenemos que $s_z = Z^T s_k = \alpha_k p_k$, y asimismo si $y_k = g_{k+1} - g_k$ recoge el cambio en el gradiente tal que $y_z = Z^T y_k$; entonces la aproximación de la Hessiana actualizada será:

⁶ La no disponibilidad de las primeras derivadas supone un problema a la hora de la aplicabilidad de estos métodos que puede superarse mediante la utilización de aproximaciones tal y como se sugieren en [163] y [67].

⁷ Consideramos que $B_0 = I$, donde I es la matriz identidad.

⁸ La fórmula de adaptación se basa en la existencia de una matriz de adaptación U_k , tal que $B_{k+1} = B_k + U_k$.

$$B_{k+1} = B_k + \frac{1}{g_k^T p_z} g_k g_k^T + \frac{1}{\alpha_k y_z^T p_z} y_z y_z^T \quad (28)$$

donde $g_z = Z^1 g_k$; $p_z = -g_z/B_z$; $B_z = Z^1 B_k Z$.

La primera iteración en este método es equivalente a la resultante del empleo del método del mayor paso descendente y consiste en la resolución del sistema (27), de forma tal que la dirección de búsqueda viene dada a través de $p_0 = Zp_z$.

Las iteraciones sucesivas implican recalcular el valor del gradiente en el nuevo punto, donde $x_k = x_0 + p_0$, para $k > 0$; y aplicar la fórmula de adaptación (28), para obtener la proyección de la matriz hessiana aproximada actualizada, aplicando los nuevos resultados al sistema (27).

2.2.3.3. Cálculo de la longitud de paso.

Dado que consideramos que las restricciones del problema original son desigualdades, a la hora de desplazarnos es necesario tener en cuenta que no nos situemos fuera de la región factible; es necesario determinar una longitud máxima de paso que no suponga la violación de ninguna de las restricciones, y que puede afectar al carácter activo o no de las mismas.

Consideremos las restricciones no activas $i \notin I_k$. Si $a_i^1 p_k \leq 0$ entonces cualquier movimiento a lo largo de p_k no violará la restricción. Si se cumple para todo $i \notin I$, entonces no existen restricciones fuera del conjunto activo que limiten la longitud de paso,

en este caso el tercer sumando de (28), se vuelve cero, puesto que se asume que α tiende a infinito.

Por contra si $a_i^t p_k > 0$, entonces existe una longitud de paso máxima γ_i donde la restricción i , no activa, pasa a convertirse en una restricción activa, esto es:

$$a_i^t(x_k + \gamma_i p_k) = b_i \quad (29)$$

El valor de γ_i deberá pertenecer al conjunto:

$$\gamma_i = \left\{ \frac{b_i - a_i^t x_k}{a_i^t p_k} / i \notin I, a_i^t p_k > 0 \right\} \quad (30)$$

Si existen varios γ_i entonces consideramos un valor, que nos determine la máxima longitud de paso:

$$\alpha = \begin{cases} \min \{ \gamma_i \} & \text{si } a_i^t p_k < 0 \text{ para algun } i \notin I \\ +\infty & \text{si } a_i^t p_k \geq 0 \quad \forall i \notin I \end{cases} \quad (31)$$

El valor de α es el máximo paso factible no negativo que se puede tomar a lo largo de p . Según el valor de la longitud de paso obtenido tendremos que tomar una decisión sobre el mantenimiento o no del conjunto activo:

a) Si $\alpha < \alpha$ entonces el conjunto activo no sufre alteraciones.

b) Si $\alpha = \alpha$ entonces existe una nueva restricción que se ha vuelto activa y que deberá incorporarse al conjunto de restricciones activas. (En caso de que sean varias las restricciones que se vuelven activas sólo añadiremos una de ellas).

A la hora de determinar el valor de la longitud de paso en este método del conjunto activo deberemos localizar un valor de α que suponga una reducción en el valor de F significativa de acuerdo con el criterio de reducción suficiente establecido por el "principio Goldstein-Armijo"⁹ que establece:

$$0 < -\mu_1 \alpha_k g_k' p_k \leq F_k - F_{k+1} \leq -\mu_2 \alpha_k g_k' p_k \quad (32)$$

donde $0 < \mu_1 \leq \mu_2 < 1$, y α_k es la longitud de paso. Para la obtención de la longitud de paso de acuerdo con el criterio (32) se considera un paso inicial α^0 y un escalar w ; de forma tal que el valor de α_k será el primer miembro de la serie $\{w^j \alpha^0\}$, $j=0, \dots$, para el cual se satisface (32) para algún valor apropiado de μ_1 y μ_2 . Normalmente se considera como valor inicial $\alpha^0 = 1$, puesto que esta elección nos produce la mejor convergencia en el método Quasi-Newton. No obstante, en el caso de que no exista ningún punto que cumpla el criterio estableceremos el valor de α , como el valor dado por α , incluso aunque éste tampoco satisfaga (32).

Cuando las variables estén acotadas, bien superior o bien inferiormente¹⁰, hay que tener en cuenta que dichas cotas no integran el conjunto de restricciones, pero sí determinan la longitud de paso factible, de forma tal que se deberán considerar las posibles violaciones de las mismas a la hora de determinar la longitud de paso.

2.2.3.4. Añadir una restricción.

Añadir una restricción supone incorporar una nueva fila a la matriz \hat{A}_K (en general se añadirá al final); además es necesario realizar una serie de modificaciones a la matriz auxiliar Z . Para ello debemos proceder de la siguiente forma:

⁹ Otros criterios alternativos se pueden ver en [68], pgs. 100-102.

¹⁰ El caso más habitual es restringir la variables a sus valores positivos.

Si estamos empleando la factorización LQ de la matriz \hat{A} para la matriz Z , a la hora de incorporar una nueva restricción podemos utilizar el siguiente método de actualización de la factorización. Supongamos que tenemos la matriz:

$$\bar{A} = \begin{pmatrix} \hat{A} \\ \hat{a}' \end{pmatrix} = \begin{pmatrix} L & 0 \\ \hat{a}' & Q \end{pmatrix} = (\bar{L} \quad 0)\bar{Q}' \quad (33)$$

donde \hat{a}' es la r -ésima fila de la última restricción añadida. La nueva matriz Q se puede calcular a partir de la matriz Q original y de una matriz de Householder (H) construida de forma que anule las componentes desde $r+2$ hasta n de la fila \hat{a}' Q , mientras que no afecte a las componentes de 1 a r .

Así pues,

$$\bar{Q} = QH_{r+1} \quad (34)$$

En la nueva matriz Q , las r primeras columnas son iguales a las de la anterior matriz Q , mientras que las columnas desde la $r+1$ hasta la n son combinación lineal de las columnas correspondientes en la antigua Q .

2.2.3.5. Eliminar una restricción.

Cuando quitamos una restricción (s) del conjunto activo tenemos que:

$$\bar{A}Q = (M \quad 0) \quad (35)$$

donde las filas desde 1 a $s-1$ de M son en forma de matriz triangular inferior y el resto de las filas tienen un elemento extra en la diagonal. Para reducir M a la forma de la matriz L ,

el procedimiento consiste en aplicar una serie de rotaciones a la derecha de Q. Estas rotaciones no afectarán a las últimas n-s columnas de Q.

En esta situación :

$$\bar{Z} = (Z \hat{z}) \quad (36)$$

donde \hat{z} es una combinación lineal de las primeras t columnas de Q.

2.2.4. Optimización sujeta a restricciones de acotación.

Para acabar este capítulo comentaremos la situación especial que se produce cuando las restricciones no son funciones lineales sino simplemente funciones de acotación. Podemos distinguir dos situaciones: en la primera todas las restricciones serían de este tipo, y en la segunda, existirían ambos tipos de restricciones.

2.2.4.1. Restricciones de acotación.

Los problemas de este tipo se plantean como:

$$\begin{aligned} \text{Min}_{x \in \mathbb{R}^n} F(x) \\ \text{sujeto a} \\ i \leq x \leq s \end{aligned} \quad (37)$$

En este problema el valor de las variables presenta una cota inferior (i) y un cota superior (s), si consideramos el número de restricciones de desigualdad éstas serán 2n,

donde n es el número de variables. En esta situación el método del conjunto activo no es directamente aplicable puesto que el número de restricciones es superior al número de variables. No obstante, dado que la variable sólo puede ser igual a una de sus cotas, podemos establecer que el conjunto de restricciones activas nunca será superior a n , una restricción será activa si la variable es igual a alguna de sus cotas. En este sentido, el número de restricciones del conjunto activo, nos da el número de variables que son iguales a una de sus cotas, y por tanto las filas de la matriz A_k serán simplemente filas de la matriz identidad correspondiente a las variables acotadas, y las columnas de Z_k serán como las filas de la matriz identidad correspondiente a aquellas variables no igualadas a sus cotas.

A la hora de establecer la dirección de búsqueda ésta afectará sólo a las $n-t$ variables libres en cada momento, esto es, aquéllas no iguales a sus cotas. A partir de este punto es posible la aplicación de los métodos de búsqueda de la dirección factible habituales, teniendo en cuenta que las matrices proyectadas serán simplemente las submatrices considerando aquellas variables libres, esto es, $g_{FR} = Z^t \cdot g$, donde g_{FR} , son los elementos del vector gradiente correspondientes a las variables no acotadas. Asimismo, se simplifica de forma considerable los procesos de cambio en el conjunto de trabajo, puesto que añadir una restricción supone acotar una variable, es decir, igualarla a una de sus cotas, mientras que borrar una restricción equivale a liberar a dicha variable.

2.2.4.2. Restricciones de acotación y lineales.

En esta situación nos encontramos problemas donde se combinan ambos tipos de restricciones¹¹. El conjunto de trabajo en este tipo de problemas incluirá una mezcla de restricciones de acotación (r) y lineales ($t-r$), donde t será el total de restricciones en el conjunto de trabajo; en esta situación es posible obtener algunas economías a la hora de aplicar el método del conjunto activo.

¹¹ Si las únicas restricciones de acotación son las restricciones de no negatividad de las variables, el problema se puede resolver como un caso de restricciones generales lineales aplicando el método del conjunto activo descrito en el epígrafe anterior.

La dirección de búsqueda se puede calcular usando la matriz:

$$Z = \begin{pmatrix} Z_{FR} \\ 0 \end{pmatrix} \quad (38)$$

donde Z_{FR} es una matriz $(n-r)*(n-t)$ cuyas columnas se obtiene a partir de la factorización de A_{FR} , formada por las variables libres. Empleando el método de Newton la dirección de búsqueda satisfará la ecuación (39):

$$Z'_{FR} G_{FR} Z_{FR} p_{FR} = - Z'_{FR} g_{FR} \quad (39)$$

En este sentido cabe señalar que se produce una reducción en la dimensión del problema equivalente al número de variables igualadas a sus cotas.

En cuanto a la adaptación del conjunto de trabajo al añadir o eliminar restricciones no se produce realmente una ventaja excesiva como en el caso anterior, de ahí que la mayoría de los programas de implementación del método del conjunto activo consideren este caso, de acuerdo con la situación habitual.

CAPITULO 3

OPTIMIZACION DE

REDES

3. OPTIMIZACION DE REDES.

3.1. Redes de Tráfico.

La representación esquemática de una red de calles de una ciudad, o de carreteras de una zona, se puede realizar de forma simple mediante la aplicación de los conceptos de la teoría de grafos. Podemos considerar que una red es un grafo dirigido (véase [60]) $G=\{N,A\}$, donde N es el conjunto de nodos, que representan los puntos de intersección de las calles o carreteras, y A es el conjunto de arcos, que representan las calles de la ciudad o las carreteras por donde circula el tráfico. Dentro del conjunto de nodos se puede considerar la existencia de dos nodos privilegiados que reciben el nombre de *centroides*:

- *Nodo origen*: Es aquel nodo donde se genera el tráfico que va a circular por la red, presenta la particularidad de que no existen arcos de entrada a dicho nodo. Trabajaremos con redes donde existan uno ó más nodos origen.

- *Nodo destino*: Es aquel nodo a donde se dirige todo el tráfico existente en la red, por lo tanto, todos los arcos que confluyen a dicho nodo son arcos de entrada. En nuestros ejemplos, por simplicidad operacional, la mayoría de las redes consideradas tendrán un nodo destino único.

Los centroides son los únicos nodos que pueden generar tráfico en la red, mientras que el resto de nodos de la red simplemente soportan las distintas cantidades de tráfico asignadas a los arcos. Normalmente, en los estudios sobre tráfico se divide la red en zonas y a cada una de ellas se le asigna un nodo centroide, que actuará como

receptor/emisor de tráfico de la zona. En redes simples el centroide es un nodo real de la misma, mientras que en redes complejas, el centroide será un nodo ficticio y actuará como representante de la absorción/generación de tráfico de una zona.

En redes muy complejas, representaciones esquemáticas de ciudades, se distingue una tercera clase de nodos, son los *conectores*, a través de los cuales la red estudiada se conecta con el resto del grafo general de la ciudad, presentan la particularidad de que pueden suponer tanto la salida como la entrada de tráfico a la red.

Un grafo de red no sólo debe ser un grafo dirigido, sino que además deberá de ser un grafo ponderado; esto es, cada arco deberá presentar una función que nos represente el coste de utilización de dicho arco; en otras palabras, deberá de existir alguna forma de representación matemática que permita al usuario conocer el coste asociado a circular por una determinada calle o carretera. Asimismo, por cuestiones evidentes, es necesario que el grafo sea conexo, de tal forma que podamos establecer un conjunto de caminos, formados un conjunto de arcos, que nos conecten el nodo origen con el nodo destino. Esta propiedad permitirá al usuario la elección de uno de los posibles caminos en función del coste asociado al mismo¹.

La representación de una red real mediante un grafo no es única, tal y como muestra Sheffi [153] las intersecciones entre arcos pueden representarse de formas diversas, bien mediante un nodo único, o bien de forma más detallada, lo que supone el empleo de más nodos, así como de un mayor número de arcos. La representación detallada presenta la ventaja de permitir una aproximación más fiel al coste de la utilización de los distintos caminos, puesto que nos permite asignar distintos costes a los distintos arcos que componen la intersección.

¹ El coste asociado a un camino se calculará en función de la suma de los costes asociados a cada uno de los arcos que configuran dicho camino.

Uno de los puntos claves en el análisis de redes de tráfico es la obtención de la función de coste asociada a cada arco. La primera aproximación a la función de coste se basa en la llamada *función de representación*. Esta función recoge el tiempo necesario para realizar la travesía de un arco de la red, en función del flujo existente en dicho arco. La función de representación (véase Ilustración 1) será una creciente respecto al flujo, y asintóticamente tenderá a infinito según nos aproximemos a la capacidad máxima del arco².

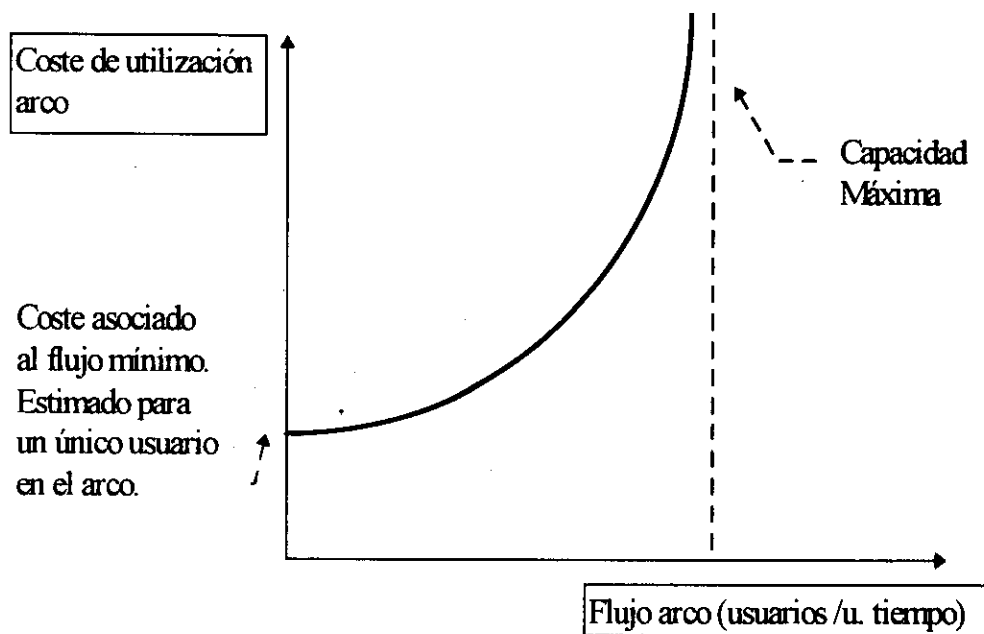


Ilustración 1: Función de Representación

El carácter asintótico de la función recoge un fenómeno típico de las redes de tráfico, esto es la idea de la *congestión*. La congestión supone la presencia de una externalidad sobre la función, de forma tal que cuando el nivel de utilización de un arco es muy elevado, el coste asociado a dicho uso tiende a infinito, recogiendo por tanto un coste marginal infinito, lo que supone que la incorporación de un nuevo

² La idea de capacidad máxima de un arco representa un parámetro técnico que nos mida el número máximo de usuarios que un arco puede soportar por unidad de tiempo.

usuario a una red congestionada elevaría el coste de utilización del arco para todos los usuarios existentes a niveles muy altos.

La función de coste asociada al arco ha sido tratada por diversos economistas³, siendo una de las funciones más estudiadas dentro de la teoría económica. La mayoría de los autores consideran la existencia de dos costes dentro de dicha función:

- *Coste Directo*: Sería el coste imputable de forma directa al usuario, esto es el coste de utilización del medio de transporte, así como el coste temporal asociado al tiempo mínimo necesario para atravesar el arco.

- *Coste Externo (Externalidad)*: Representa el coste de congestión; es decir, el coste que la incorporación de un nuevo usuario representa en incremento del tiempo de viaje con respecto al conjunto de usuarios de la red.

La idea básica de la optimización de redes de tráfico sería poder establecer los patrones óptimos de uso de la red, de forma tal que redujesen al mínimo factible los costes externos (congestión) asociados a dicha red. Este hecho nos faculta la posibilidad de intentar aplicar el conocimiento sobre optimización matemática al mundo económico.

El proceso de gestión y control de tráfico está compuesto de cuatro fases o subprocesos (ver [169]):

1) *Generación de Viajes*: Estimación de los viajes producidos y atraídos por la zona en estudio.

³ Small [158] presenta un estudio muy completo del estado del estudio de la función de coste.

- 2) *Distribución de Viajes*: Determinación de la matriz de frecuencias de viajes entre las zonas de producción y atracción.
- 3) *Reparto Modal*: División de los viajes entre los diferentes modos de transporte.
- 4) *Asignación de Tráfico*: Asignación de matrices de viaje de la zona origen a la zona destino de acuerdo con la red de transporte existente.

En el presente trabajo nos centraremos tan sólo en el apartado 4, el proceso de asignación del tráfico, suponiendo que todos los viajes se realizan en un único modo de transporte: el automóvil privado⁴.

3.1.1. Equilibrio en redes de tráfico.

El proceso de asignación del tráfico en una red consiste en determinar el valor de las matrices de viajes que representen una situación de equilibrio. Esto supone que nos debemos dotar de un procedimiento que nos permita la determinación de una situación de equilibrio a la hora de proceder a asignar las distintas cargas de tráfico a los arcos de la red.

3.1.1.1. Matrices de Viajes y Matrices Origen - Destino (O-D)

Las matrices O-D (véanse [14, 90]) representan, en filas, los arcos que componen cada uno de los caminos entre los distintos origen y destinos de una red. Estarán formadas por ceros y unos, tal que un cero indica que el arco no forma parte del camino y un uno indica que dicho arco está contenido en el camino.

⁴ Para un análisis más detallado de las restantes fases del proceso se pueden consultar varios autores, entre ellos: [33, 95, 169]

Las matrices de Viajes nos darán el valor del tráfico que circula por cada uno de los arcos de la red, de acuerdo con los distintos caminos que componen la misma. Lógicamente, la matriz de viajes representa una visión estática de la red, puesto que dado el carácter inherentemente dinámico del tráfico, las matrices de viaje estará cambiando en cada momento, por ello, normalmente, este tipo de matrices nos dará la carga que tendrá que soportar la red en un período concreto de tiempo: una hora, un día, etc.

En el proceso de estimación de las matrices podemos distinguir tres tipos de aproximación:

- 1) La primera aproximación está basada en los modelos de generación de viajes, esto es los modelos de gravedad y atracción entre zonas (ver [33, 95]).
- 2) La segunda trata de introducir los principios de la probabilidad y la generación estocástica de viajes entre zonas. Se basa en teorías como las de minimización de la información y maximización de la entropía (ver [174]), o la estimación máximo-verosímil (en [15, 90]), o la probabilidad bayesiana [108].
- 3) La tercera considera la formulación del equilibrio matemático sometido a las restricciones de capacidad de los arcos [153].

Nosotros seguiremos esta tercera aproximación puesto que nos permitirá la aplicación directa de los métodos de optimización matemática considerados en el capítulo anterior.

3.1.1.2. Definición de Equilibrio.

La definición clásica de equilibrio fue establecida por Wardrop en 1952; se basa en la idea de que los conductores escogerán aquella ruta de menor coste, considerando como medida principal del coste, el valor del tiempo empleado en recorrer dicha ruta entre el origen y el destino. Dado que el coste de atravesar un arco viene dado por el tiempo, y éste a su vez es función de la velocidad, podemos establecer una relación entre el coste y el flujo del arco, a través de la relación entre la velocidad y el flujo. Esta relación entre coste y flujo (tal y como se especifica en la función de representación) supone que los conductores se repartirán entre los distintos caminos de la red buscando aquéllos de menor coste. *Habremos alcanzado una situación de equilibrio cuando ningún conductor pueda mejorar el coste de su viaje mediante un cambio unilateral de su ruta.* A este tipo de equilibrio se le denomina *Equilibrio del Usuario*.

Esta aproximación supone que los usuarios poseen información completa del coste de atravesar un arco de la red. Frente a esta visión se pueden considerar enfoques alternativos como el propuesto por las teorías estocásticas sobre el comportamiento de los conductores; en este caso, el equilibrio se alcanzará cuando *el conductor cree que no es posible mejorar el coste de su viaje mediante un cambio unilateral en su ruta* [41].

3.1.2. Equilibrio del Usuario.

La idea del equilibrio del usuario se basa en el supuesto matemático de que el conjunto de individuos actúa con un criterio de minimización; y que el conjunto del sistema está sujeto al supuesto de que la demanda de usuarios debe quedar cubierta.

El problema puede escribirse como:

$$\begin{aligned} & \text{Min}_{x \in \mathbb{R}^n} F(x) \\ & \text{sujeto a:} \\ & q = \sum_{i=1}^n x_i \end{aligned} \quad (1)$$

donde $F(x)$ es la función de coste, q es la demanda total y x_i es el flujo de usuarios que circula por cada arco i , de la red.

Podemos escribir el problema de forma más detallada si consideramos las siguientes variables:

x_i : representa el flujo en el arco i .

c_i : representa el coste de atravesar el arco i .

q_{ab} : es la demanda de viajes entre los nodos a y b , unidos por uno o más caminos j .

c_j^{ab} : es el coste de ir entre a y b a través del camino j .

$$\begin{aligned} & \text{Min}_{x \in \mathbb{R}^n} F(x) = \sum_i \int_0^{x_i} c_i(w) dw \\ & \text{sujeto a:} \\ & \sum_j f_j^{ab} = q_{ab} \quad \forall a, b \\ & f_j^{ab} \geq 0 \end{aligned} \quad (2)$$

El primer grupo de restricciones representan la ecuación de flujo, mientras que el segundo son las restricciones de no negatividad.

Debemos tener en cuenta:

$$x_i = \sum_a \sum_b \sum_j f_j^{ab} \delta_{i,j}^{ab}$$

donde

$$\delta_{i,j}^{ab} = \begin{cases} 1 & \text{si el arco } i \text{ forma parte del camino } j \text{ entre } a \text{ y } b. \\ 0 & \text{en otro caso.} \end{cases} \quad (3)$$

Los valores de δ constituirán los elementos de la matriz O-D.

En esta aproximación se trata de resolver un problema de optimización de una función F , no lineal, sujeta a restricciones lineales. Por otro lado, debemos señalar que dicha visión no considera la división temporal de la red; esto es, podemos calcular el óptimo en un momento dado, que se toma como período de referencia temporal. Una visión temporal puede verse en Nemhauser y Merchant [116,117], Ran, Boyce et al [148]; o bien, en su vertiente de programación dinámica, en Agnew [1], o Ran y Boyce [149].

3.2. Función de Representación.

La función de representación (Ilustración 1) refleja el incremento percibido en los costes por el usuario a la hora de circular a través de un arco de la red, en función del flujo que atravesase dicho arco. Es por tanto crucial en nuestro problema puesto que constituirá la función objetivo.

La idea subyacente tras esta función es que el usuario percibe un mayor coste temporal al atravesar un arco, cuanto mayor sea la cantidad de tráfico (el flujo) que

circule por el mismo, basándonos en el hecho de que se produce una disminución de la velocidad a la que se puede circular. Este supuesto ha sido ampliamente reflejado en la literatura y se conoce como *congestión del arco*. Gráficamente (Ilustración 2), la mayoría de los autores sostienen que la relación entre el flujo y la velocidad sería una curva, de forma tal que entre A y B indicaría una relación inversa, esto es, a mayor flujo menor velocidad, donde A representa el punto de máxima velocidad, que corresponde a un flujo mínimo, y B representa el flujo máximo que es capaz de soportar el arco; y a partir del punto B, la curva reflejaría una relación directa entre ambas variables. Esta rama de la curva se denomina rama de *hipercongestión*, aunque no todos los autores están de acuerdo en cuanto a su posible existencia.

Veamos a continuación las distintas alternativas consideradas por varios autores como forma factible de la curva flujo (x) - velocidad (v):

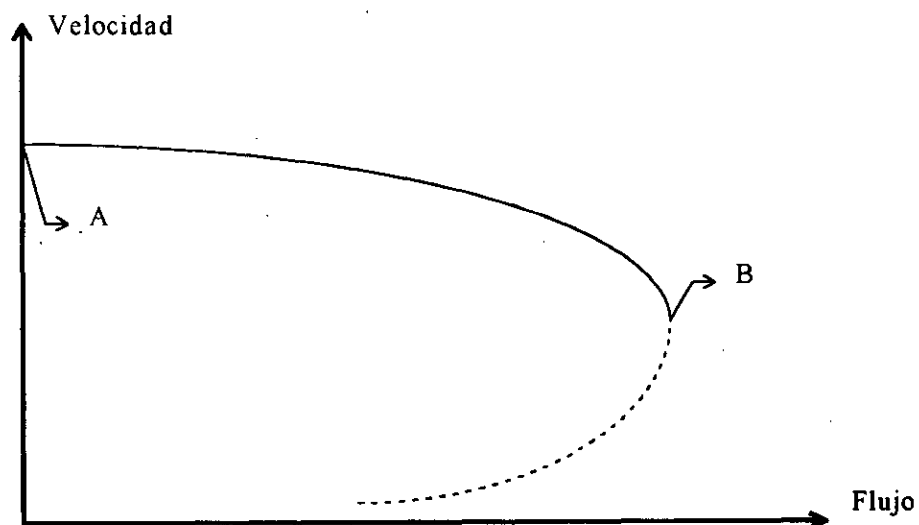


Ilustración 2: Relación Velocidad - Flujo

- Boardman y Lave (1977):

Estos autores presentan la siguiente relación entre el flujo y la velocidad⁵:

$$x = a - b(v - c)^2$$

Donde a, b y c serían parámetros particulares para cada arco estudiado. Los autores trabajan con los valores siguientes:

$$a=2490; b=.523; c=35.34$$

Obteniendo la siguiente relación gráfica:

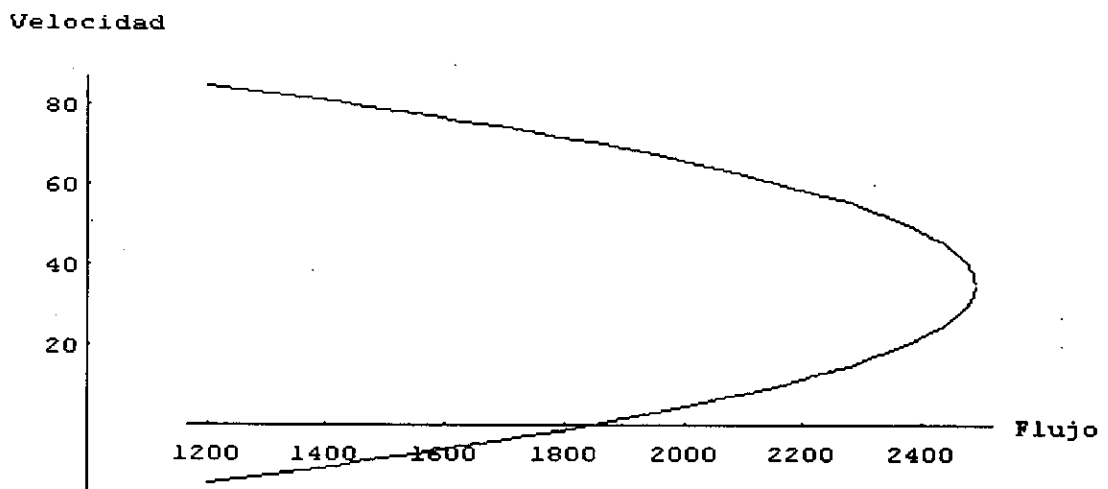


Ilustración 3: Boardman y Lave

Donde existen dos ramas dentro de la curva, alcanzando un valor máximo de flujo en $a = 2490$.

⁵ La velocidad figura en millas por hora en todas las gráficas consideradas, mientras que el flujo indica el número de vehículos en una unidad de tiempo.

- Inman (1978):

Propone una relación flujo - velocidad reflejada mediante una función generalizada que recoja el conjunto de funciones estimadas hasta ese momento:

$$\frac{x^\beta - 1}{\beta} = a - b \left(\frac{(v - c)^\alpha - 1}{\alpha} \right)$$

Dicha ecuación estimada con los parámetros facilitados por Small [158]⁶ nos proporciona la siguiente gráfica:

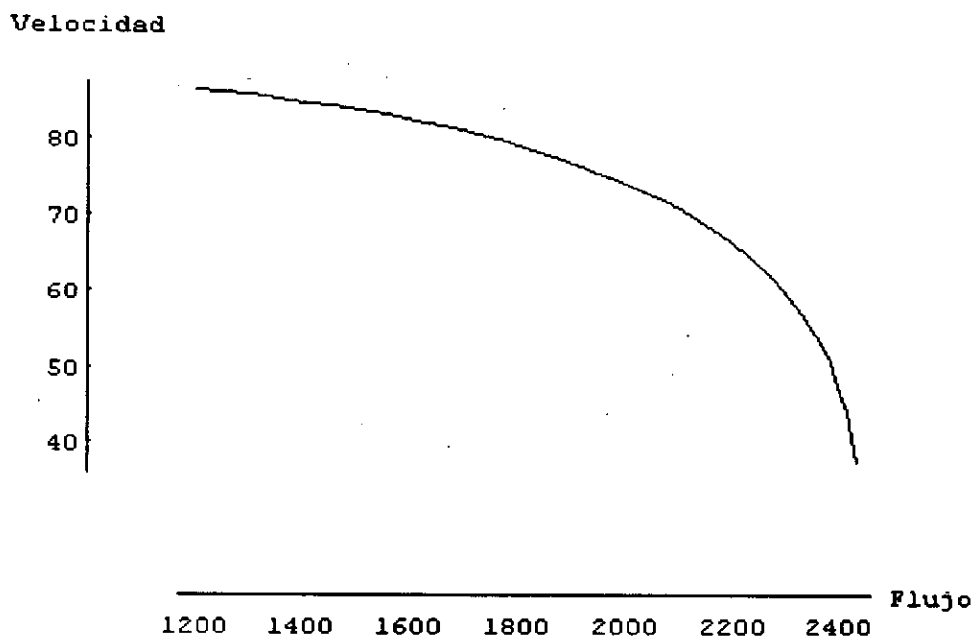


Ilustración 4: Inman

La principal particularidad de la función de Inman es que no posee rama de hipercongestión, y que la velocidad tiende a 0; según el flujo se aproxima asintóticamente al flujo máximo.

⁶ a=3.351*10⁹; b=231.4; c=7.2; α=4.06; β=2.95

• Coleman (1961):

Este autor presenta la particularidad de que considera que la función flujo-velocidad debe incorporar la capacidad máxima del arco, que es un parámetro de carácter técnico que nos indica cuál es el flujo máximo que puede soportar un arco determinado. La relación queda establecida por:

$$\frac{x}{C} = a - b \left(\frac{1}{v} - \frac{1}{c} \right)^2$$

donde C representa la capacidad máxima del arco.

Estimando dicha ecuación para los siguientes parámetros:

$$a = .89; b = 288; c = 13.0$$

obtenemos la siguiente gráfica que nos relaciona la velocidad con el grado de cobertura de la capacidad del arco (x/C):

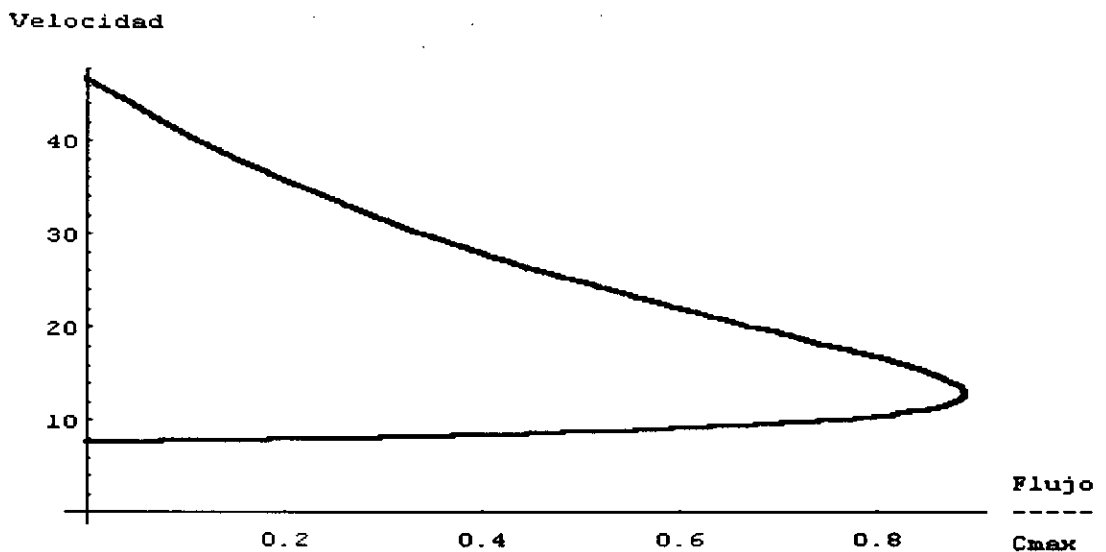


Ilustración 5: Coleman

En dicha gráfica se observa la fuerte caída de la velocidad según se aproxima el flujo en el arco a la capacidad máxima; y que antes de alcanzar la misma, a una ocupación del 89%, se entra en la rama de hipercongestión, donde la pendiente de la curva se vuelve mucho más suave.

- Keeler y Small (1977):

Al igual que Coleman estos autores consideran dentro de la relación la capacidad máxima del arco. La forma funcional es similar a la de Coleman:

$$\frac{x}{C} = a - b(v - c)^2$$

Con los parámetros:

$$a=0.8603; b=0.001923; c=45.68.$$

Obtienen la siguiente gráfica:

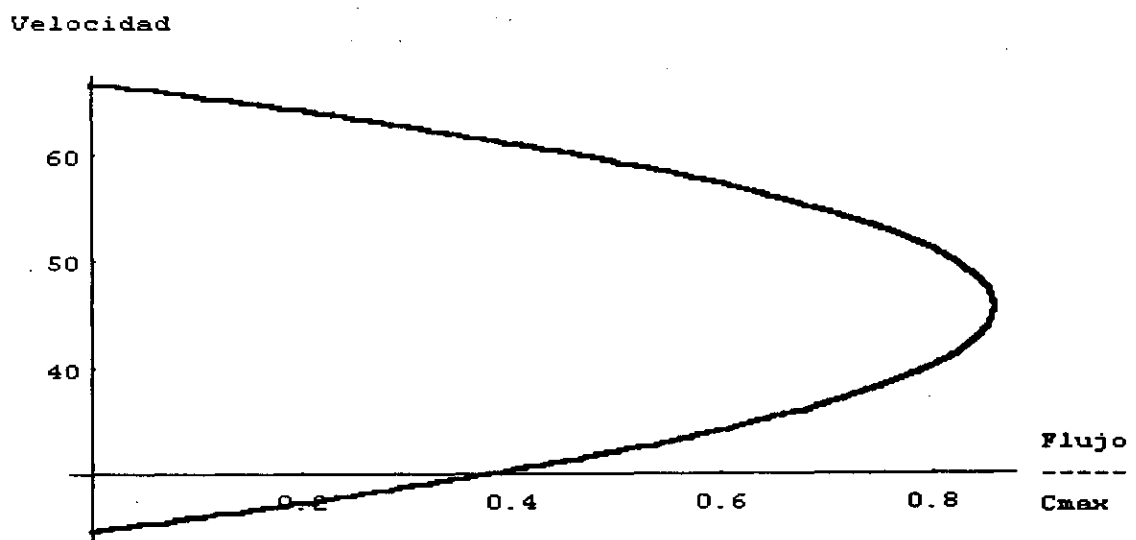


Ilustración 6: Keeler y Small

Se observa que presenta una fuerte similitud con Coleman, pero la rama de hipercongestión presenta una pendiente similar a la rama de congestión. Hay que señalar que en esta representación el flujo máximo es inferior a la capacidad máxima, como ocurría en el caso anterior, e incluso inferior al flujo máximo obtenido por Coleman, concretamente el valor máximo de ocupación es del 86% de la capacidad máxima.

- *Thomas (1991):*

Este autor presenta la relación más genérica de todas las comentadas:

$$x = \frac{av - v^2}{b}$$

donde a : es el valor de la velocidad máxima (en km/h) con flujo libre, $b=a^2/4C$ y C es la capacidad máxima.

Operando podemos establecer la relación entre la Velocidad/Velocidad máxima y el Flujo/Capacidad máxima:

$$\frac{x}{C} = 4 \left(\frac{v^2}{a^2} - \frac{v}{a} \right)$$

Gráficamente la curva estimada es muy similar a la del resto de los autores, presentando dos ramas:

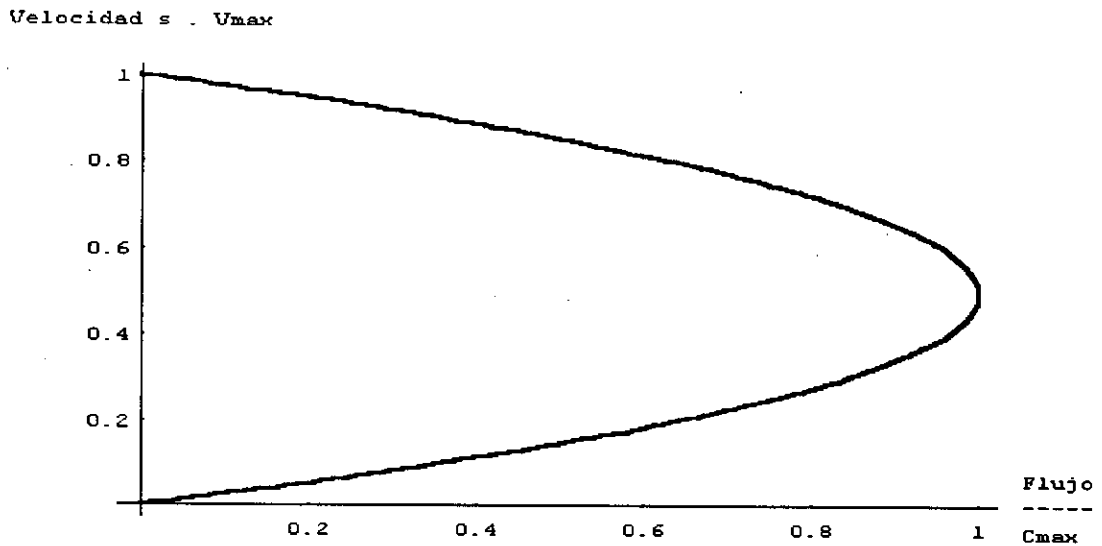


Ilustración 7: Thomas

Esta relación es la única que nos permite que el flujo máximo sea igual a la capacidad máxima.

La visión práctica de la curva flujo-velocidad debe limitarse a la zona superior de la curva; esto es, la rama de congestión, puesto que la mayoría de los autores asumen que es imposible, en la realidad, que el flujo y la velocidad decrezcan al mismo tiempo, tal y como postulan los modelos teóricos. La elección de la curva flujo-velocidad debe tener en cuenta las características particulares del arco estudiado, de hecho todos los autores aquí referidos han realizado sus curvas en función de datos obtenidos de la observación real del tráfico en determinadas vías.

3.2.1. Función de Costes.

A la hora de considerar una función de coste para el transporte que nos permita ponderar la utilización de los arcos de la red debemos considerar la muy diferente naturaleza de los diversos componentes de dicha función.

3.2.2.1. Costes a Corto Plazo.

La función de costes a corto plazo estaría constituida por un total de 8 categorías de coste:

- a) Costes Directos:
 - i) Costes de Funcionamiento: Serían los costes que soporta de una forma más directa el usuario; esto es, combustible, mantenimiento del vehículo, etc. Este tipo de costes serán función directa de la distancia recorrida.
 - ii) Costes de Capital: Son los costes asociados al capital invertido en la adquisición del vehículo.
 - iii) Costes Temporales: Estimación en función del valor del tiempo (ver [55, 110]), de los costes asociados al tiempo empleado en realizar el recorrido.
 - iv) Costes de Retrasos Horario: Introducidos por Small [158] y Arnott, De Palma et al. [6, 7, 8, 9], intentan reflejar los costes que para un usuario representa el llegar a su destino con adelanto o retraso sobre el horario previsto.

- b) Costes Sociales:
 - i) Costes asociados a los gastos producidos por los accidentes de tráfico.
 - ii) Costes de aparcamiento: costes necesarios para la provisión de plazas de parking.
 - iii) Costes de servicios públicos para automóviles.
 - iv) Costes medioambientales.

Consideramos que los costes directos serán los más importantes a la hora de que los usuarios opten entre las diversas rutas, dando especial prioridad a los costes temporales dentro de su función de coste.

La estimación de la función de coste ha sido tratada por varios autores en la literatura económica reciente⁷, nosotros optaremos por una aproximación atemporal, del tipo de estado estacionario, basándonos en la relación flujo - coste que se deduce de las relación flujo - velocidad comentadas anteriormente.

La relación flujo - coste considerada es la establecida por el Bureau of Public Roads [27]:

$$C_{ii} = C_{fi} (1 + a(x_i / K_i)^b) \quad (4)$$

donde C_{ii} expresa el coste temporal de atravesar el arco i ; C_{fi} representa el coste temporal mínimo de atravesar dicho arco, esto es, con flujo libre, x_i es el flujo actual en el arco i , y K_i es la capacidad máxima del arco; por último, a y b son dos constantes⁸.

La función general de coste a corto plazo de atravesar el arco se puede expresar como:

$$C_i(x_i, K_i) = C_0 + C_{ii} \quad (5)$$

donde C_0 sería el coste independiente del tiempo, y por tanto independiente del grado de congestión del arco, que se corresponde con los recogidos en el apartado a) i), y

⁷ Jara-Díaz [91. 92] ofrece un buen resumen sobre el tema.

⁸ Los valores sugeridos para dichas constantes son: $a=0.15$ y $b=4$.

C_{ti} es el coste temporal⁹. Si sustituimos (4) en (5), obtenemos la función de coste a corto plazo de atravesar un arco i :

$$C_i = C_0 + C_{fi}(1 + a(x_i / K_i)^b) \quad (6)$$

C_i sería el coste asociado a utilizar el arco i en una ruta, y lo utilizaremos como función de representación en nuestro problema. Gráficamente, si representamos ambas funciones veremos que son continuas, crecientes y diferenciables tal y como se mostraba en la Ilustración 1.

3.2.2.2. Costes a Largo Plazo.

La principal diferencia con la función de costes a corto plazo es la consideración en esta función de los costes de capital asociados a la construcción de la infraestructura viaria de la red. El principal estudio que trata el tema de la provisión de infraestructuras fue realizado por Meyer, Kain y Wohl en [120] y considera el coste de capital de provisión de una infraestructura como una función lineal del número de carriles de dicha vía. Así:

$$C_k = a + bw \quad (7)$$

donde a y b son dos parámetros técnicos y w es el número de carriles.

⁹ Suponemos que el valor del coste temporal incluye la valoración económica del tiempo, si quisiéramos un modelo donde sólo se incluye el tiempo, sería necesario multiplicar el valor obtenido en (4) por una estimación económica del valor del tiempo [49].

Estudios posteriores, Keeler y Small [98] y Kraus [102], proponen la adopción de funciones no lineales, lo que permite la consideración de distintos tipos de economías de escala. Small [158] propone la adopción de una función similar a la de Meyer et al. [120], pero utilizando como variable independiente la capacidad de la vía y no el número de carriles. La función resultante es:

$$C_k = f_0 + f_1 K \quad (8)$$

donde f_0 y f_1 son dos parámetros.

La integración de los costes de capital en la función de costes supone la necesidad de considerar un factor de recuperación del capital basado en el tipo de interés y la vida media esperada de la infraestructura. Meyer et al [120] proponen considerar un coeficiente de capitalización:

$$\rho = \frac{r}{1 - \delta^T}$$

donde r es la tasa de interés, T es la vida media esperada y $\delta=1/1+r$.

Así pues, el coste a largo plazo de un arco i será función de su capacidad:

$$C_{LP} = \text{Min}_K x_i \left(c_0 + c_{\beta} \left(1 + a \left(x_i / K_i \right)^b \right) \right) + \rho (f_0 + f_1 K_i) \quad (9)$$

esta función recoge los costes a largo plazo del arco i en un momento dado, lógicamente si consideramos varios momentos discretos en el tiempo habrá que tener en cuenta las posibles variaciones en el flujo temporal y así (9) se convierte en:

$$C_{LP} = \text{Min}_K \sum_h x_i^h \left(c_0 + c_{\beta} \left(1 + a \left(x_i^h / K_i \right)^b \right) \right) + \rho (f_0 + f_1 K_i) \quad (10)$$

donde h recoge los distintos periodos de tiempo considerados, si éstos fuesen de duración diferente habría que introducir la duración de cada uno de ellos dentro de la ecuación (10) mediante un factor q_h . El resultado será:

$$C_{LP} = \text{Min}_K \sum_h q_h x_i^h \left(c_0 + c_f \left(1 + a \left(x_i^h / K_i \right)^b \right) \right) + \rho (f_0 + f_1 K_i) \quad (11)$$

Para obtener el coste a largo plazo necesitamos resolver un problema de minimización sin restricciones, para lo que derivamos e igualamos la primera derivada a cero con el objeto de hallar el mínimo de la función.

$$f_1 \rho - \frac{abc_f \sum_h q_h x_i^h \left(\frac{x_i^h}{K_i} \right)^b}{K_i} \quad (12)$$

Para un período h , donde el volumen servido es $q_h x_i^h$ obtenemos como valor óptimo de K_i :

$$K_i^* = x_i \left(\frac{abc_f q}{\rho f_1} \right)^{\frac{1}{b+1}} \quad (13)$$

La función de coste a largo plazo para un periodo de tiempo h , se obtiene sustituyendo (13) en (11):

$$C(x_i; q) = \rho f_0 + x_i q (c_0 + c_f) + C_g(x_i; q) \quad (14)$$

donde C_g es el coste de congestión a largo plazo:

$$C_g = c_g x_i q^{\frac{1}{1+b}} \quad (15)$$

donde c_g :

$$c_g = (abc_{fi})^{\frac{1}{1+b}} (\rho f_i)^{\frac{b}{1+b}} \left(1 + \frac{1}{b}\right) \quad (16)$$

Al optimizar una red para un momento concreto del tiempo debemos considerar la función de costes a corto plazo, y no a largo plazo. Algunos autores (Nemhauser & Merchant [117, 118], Carey [32], Ran et al. [148, 149]) trabajan con adaptaciones dinámicas, considerando la red dividida en periodos de tiempo; en este caso la función de representación a elegir debería ser la función de costes a largo plazo.

3.2.2. Estimación del coste total de usuario.

La estimación del coste total de usuario se basa en la consideración del coste temporal que soporta éste, junto a un coste directo en forma de peaje que el usuario debería pagar por la utilización de la infraestructura en función del grado de congestión que presente la misma.

La idea de considerar un precio por la utilización de una carretera congestionada tiene su base en los economistas clásicos; siendo probablemente Dupuit durante la primera mitad del siglo XIX, el primer autor que considera la posibilidad de la instalación de un peaje por la utilización de determinadas obras de la infraestructura viaria; posteriormente, Pigou [146] y Knight [101] retoman el tema desde el punto de

vista del bienestar social e individual. No obstante, será a partir de 1950 cuando el tema adquiera una mayor dimensión teórica a partir de las consideraciones de Beckaman et al. [13] y del artículo de Walters [182] en el que se considera el establecimiento de un pago por el uso de una infraestructura en horas punta dado un nivel mínimo de congestión. Este artículo será el desencadenante de una serie sobre el tema que considera tanto los aspectos económicos como instrumentales sobre la posible implantación de peajes en carreteras, autopistas, etc; citar entre otros¹⁰: Mohring y Harwitz [123]; Vickrey [177, 178, 179, 180, 181]; Mohring [125, 126]; Keeler y Small [98]. El tema del peaje sufrirá un cierto abandono a partir de finales de los setenta, aunque volverá a ser retomado desde nuevos puntos de vista a finales de los años ochenta, momento en el que surgen nuevas aproximaciones que relanzan la idea del peaje: Arnott et al. [6, 9]; Braid [25]; Tabuchi [166]; Yinger [198].

La idea del peaje de congestión surge a partir de la consideración de la existencia de una diferencia entre el coste privado, esto es, el coste que un usuario percibe al incorporarse a la red, y el coste social, como el que la incorporación a la red de un nuevo usuario repercute sobre cada uno de los usuarios existentes hasta ese momento. La Ilustración 8 muestra gráficamente las diferencias existentes entre el coste de usuario = C_{Medio} y el coste social = C_{Marginal} . Vemos que el coste medio va parejo al coste marginal para niveles bajos de flujo, pero según estos niveles aumentan, esto es al incrementarse la congestión en la red, se produce una separación entre ambos; de forma que el equilibrio entre la demanda D y el C_{Me} (F_0) no coincide con el equilibrio entre D y C_{Mg} (F_1). Siendo el primero mayor que el segundo, se hace necesario el establecimiento de un peaje que suponga un aumento de precio al usuario, por tanto incremente el coste medio, y disminuya la demanda hasta el nivel de equilibrio entre D y C_{Mg} . Este peaje π deberá recoger, por tanto, la diferencia entre $C_{\text{Mg}} - C_{\text{Me}}$.

¹⁰ Un magnífico resumen de la literatura publicada hasta el momento se puede encontrar en el libro de Small [160].

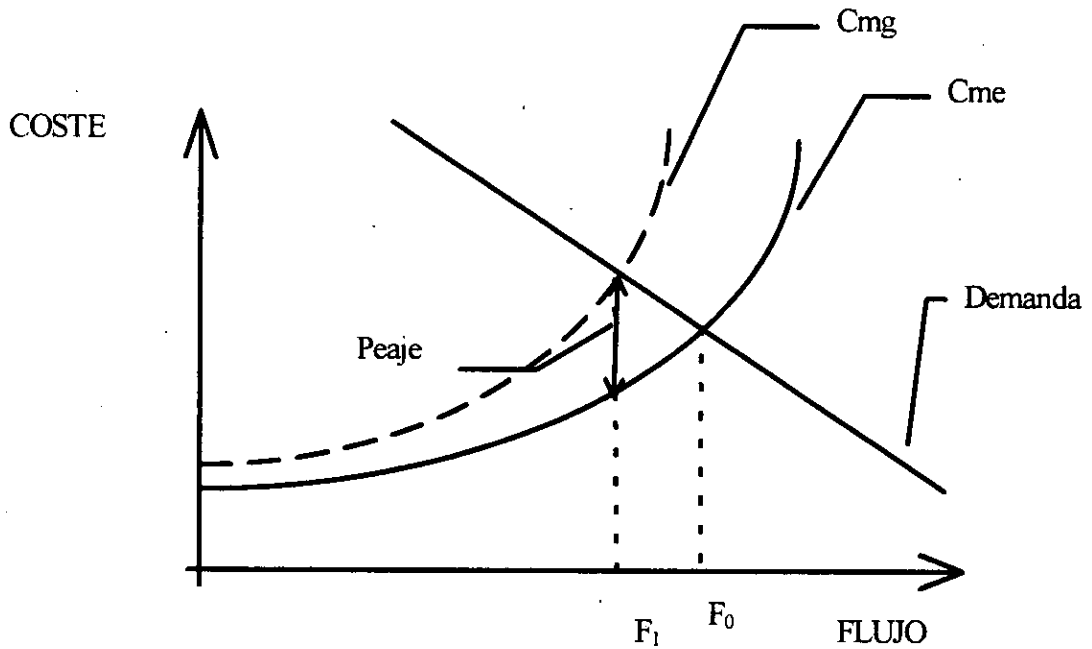


Ilustración 8: Determinación Gráfica del Peaje

Si consideramos la función de costes a corto plazo obtenida en (6) y el volumen de usuarios en un período tendremos:

$$CTCP = x_i \left(c_0 + c_{fi} \left(1 + a \left(\frac{x_i}{K_i} \right)^b \right) \right) \quad (17)$$

Si calculamos el Cme y el Cmg y los restamos obtendremos el valor del peaje asociado al volumen x_i :

$$Cmg = c_{fi} + c_0 + ac_{fi} \left(\frac{x_i}{K_i} \right)^b (1 + b)$$

$$Cme = c_{fi} + c_0 + ac_{fi} \left(\frac{x_i}{K_i} \right)^b \quad (18)$$

El peaje resulta:

$$\pi = abc_{fi} \left(\frac{x_i}{K_i} \right)^b \quad (19)$$

Con lo que obtenemos que el peaje óptimo será igual a b veces el coste de congestión.

Por lo tanto, el precio que soportarian los usuarios por el uso de la vía, sería el coste del usuario (6) más el peaje (19); esto es:

$$C_i = c_0 + abc_{fi} \left(\frac{x_i}{K_i} \right)^b + c_{fi} \left(1 + a \left(\frac{x_i}{K_i} \right)^b \right) \quad (20)$$

Lo que cumple el principio basado en la literatura económica de precio=coste marginal, como precio de equilibrio de mercado. Podemos expresar (20) como:

$$C_i = c_0 + c_{fi} + ac_{fi} \left(\frac{x_i}{K_i} \right)^b (b + 1) \quad (21)$$

Dado que tanto c_0 , c_{fi} , a , b son parámetros técnicos, el coste de usuario de atravesar un arco i será, por tanto, función del grado de congestión del mismo, expresado por la relación entre el flujo existente en dicho arco y la capacidad máxima de éste.

El importe del peaje sobre el coste mínimo de atravesar el arco (c_{fi}) con respecto a la relación flujo / capacidad se expresa gráficamente en la Ilustración 9:

Peaje/Coste Libre

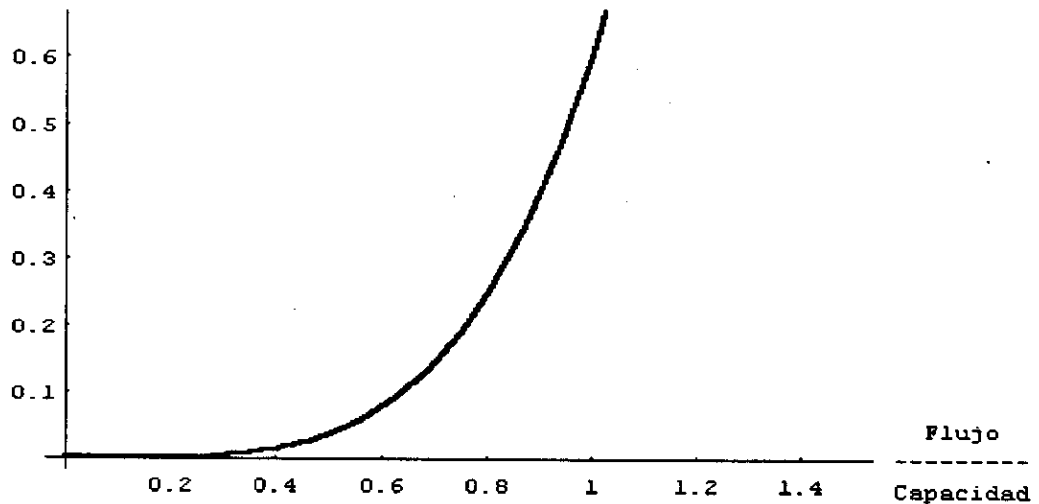


Ilustración 9: Relación Peaje - Flujo

Se observa en dicha gráfica¹¹ que la evolución del peaje es bastante lenta mientras no se ha alcanzado una tasa de congestión severa, sufriendo un fuerte incremento al superar esta tasa de congestión el valor 1. Por su parte, si representamos los costes y el peaje para un flujo dado (ver Ilustración 10) se observa claramente el carácter creciente de todas las curvas, así como el paralelismo existente entre el coste marginal y el peaje.

La consideración de las relaciones entre el flujo y la velocidad nos permite establecer una relación entre el coste del usuario y el peaje y la velocidad de la vía.

Si observamos la relación establecida por Boardman y Lave [21] obtenemos la siguiente relación entre flujo (x) y velocidad (v):

$$x = b_0 + b_1v + b_2v^2 \quad (22)$$

¹¹ La realización de ambas gráficas se ha hecho teniendo en cuenta los valores del BPR [27] para los parámetros a y b. Para el cálculo del coste medio y coste marginal se ha considerado un arco de capacidad máxima 5000, y con valores $c_0=5.5$ y $c_0=7.75$.

donde los parámetros estimados por los autores presentan los valores: $b_0=3673$; $b_1=73.93$; $b_2=-1.046$.

Sustituyendo (22) en (6) obtenemos la expresión de la relación entre el coste y la velocidad:

$$C = c_0 + c_{fi} \left(1 + a \left(\frac{b_0 + b_1 v + b_2 v^2}{K} \right)^b \right) \quad (23)$$

Calculando el Coste total como el producto de (23) por el flujo (x), y derivando mediante la regla de la cadena para obtener el Coste Marginal, podemos establecer el peaje que en este modelo resulta:

$$\pi = abc_{fi} \left(\frac{b_0 + b_1 v + b_2 v^2}{K} \right)^b \quad (24)$$

Si realizamos una estimación gráfica (Ilustración 11) del peaje¹² observamos que la curva presenta un primer tramo creciente, y una vez alcanza su valor máximo, la curva de peaje comenzaría a decrecer hasta anularse. Los puntos singulares de la curva se corresponden, en el caso del nivel cero, con el flujo mínimo y la velocidad máxima; y en el caso del nivel máximo de peaje con el flujo máximo. La parte a la derecha del máximo se correspondería con la rama de congestión de la curva de Boardman y Lave, mientras que la izquierda se corresponde con la rama de hipercongestión de dicha curva (Ilustración 3).

¹² Para la realización de la estimación se han empleado los valores: $c_0=5.5$; $c_{fi}=7.75$; $K=5000$.

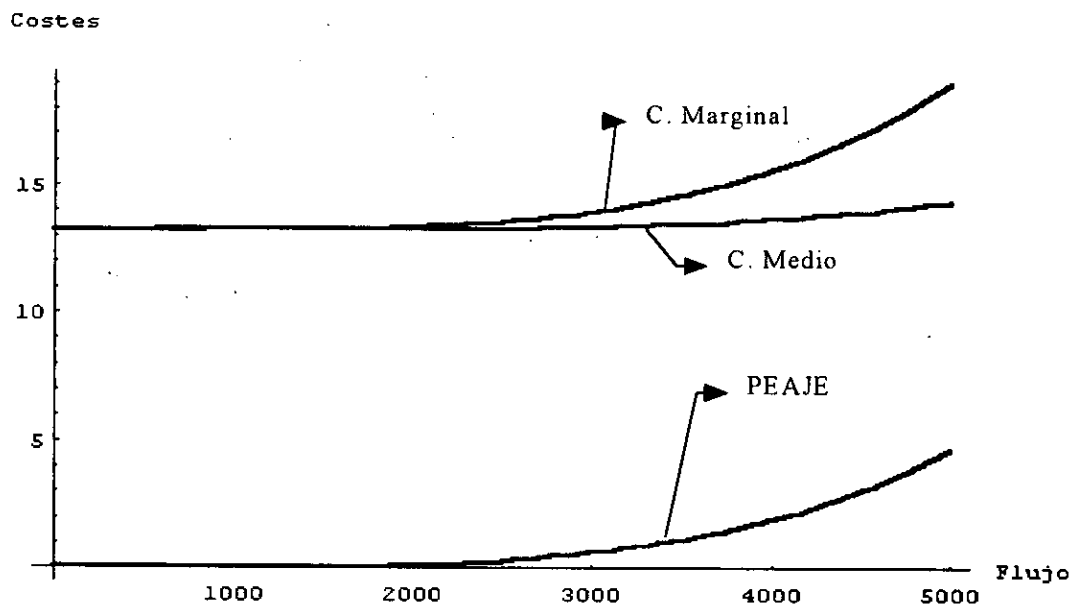


Ilustración 10: Peaje, Coste Medio y Coste Marginal.

Esta situación de hipercongestión no parece recoger una situación real de ahí que Inman [88] optase por considerar una curva flujo-velocidad con una rama asintótica una vez alcanzado el nivel máximo de flujo. La relación establecida por dicho autor sería:

$$\frac{x^\beta - 1}{\beta} = i_1 - i_2 \left(\frac{(v - c)^\alpha - 1}{\alpha} \right) \quad (25)$$

Donde el valor de los parámetros utilizados serían: $i_1=3.351 \cdot 10^9$; $i_2=231.4$; $c=7.2$; $\alpha=4.06$; $\beta=2.95$. Sustituyendo en (6) y operando como en el caso anterior podemos hallar el peaje asociado a esta curva flujo - velocidad:

$$\pi = abc_{fi} \left(\frac{\left(1 + \beta i_1 + \frac{\beta i_2 (1 - (v - c)^\alpha)}{\alpha} \right)^{\frac{1}{x}}}{K} \right)^b \quad (26)$$

La representación gráfica¹³ de (26) (Ilustración 12) muestra una curva de peaje mucho más suave y que una vez ha alcanzado su nivel máximo se vuelve plana, hasta alcanzar una velocidad de 7.2, a partir de la cual la curva de Inman con los parámetros establecidos no está definida. Esta curva de peaje representa una situación más proxima a la realidad puesto que no muestra dos ramas y un punto máximo, sino que, una vez alcanzado el peaje máximo, que se corresponde con el flujo máximo, éste ya no variaría, sino que se mantendría igual aunque ingresaran nuevas unidades a la red.

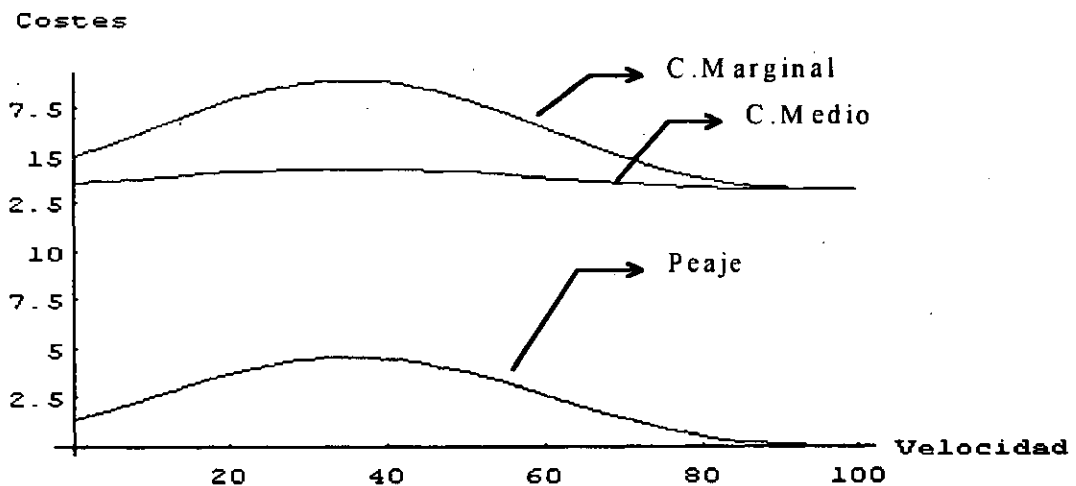


Ilustración 11: Coste basados en curva flujo-velocidad de Boardman y Lave

¹³ Para la realización de dicha curva se han empleado los mismos valores que en el caso de la curva de Boardman y Lave.

Asimismo, se observa al comparar ambas curvas, que al utilizar la curva establecida por Inman, existe una mayor proximidad entre el coste medio y el coste marginal y, por lo tanto, el valor del peaje es inferior al establecido por la curva de Boardman y Lave.

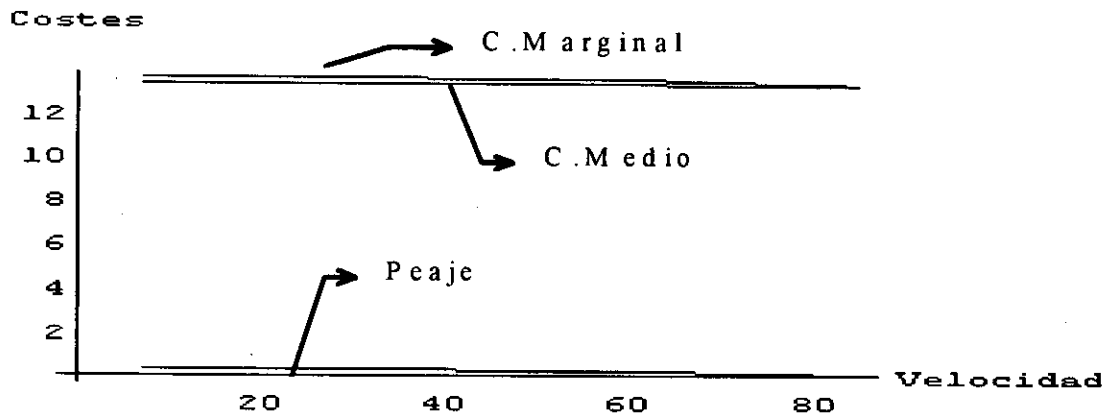


Ilustración 12: Costes basados en la relación flujo - velocidad de Inman

3.3. Algoritmos de Redes de Tráfico.

La elección de ruta por parte de los usuarios de una red se basa en el principio de la ruta percibida como de coste mínimo. No obstante, las diferentes elecciones por parte de los usuarios se pueden explicar a través de dos grupos de factores: aquellos usuarios que eligen distintas rutas debido a que poseen una percepción diferente de los costes asociados a la utilización de la vía; y aquéllos que utilizan una ruta distinta de aquella que hubieran elegido ante la ausencia de tráfico debido a la congestión, que hace que dicha ruta ya no sea considerada como la más económica.

La consideración del coste de elección de la ruta ha propiciado diversos estudios sobre los distintos factores que influyen en los usuarios a la hora de considerar esta elección. Thomas [169] apunta una relación de los principales estudios efectuados hasta esa fecha. Nuestro enfoque considera que los usuarios se enfrentan a un coste directo temporal, debido al coste que para ellos supone el tiempo empleado en recorrer la ruta escogida, junto a unos costes fijos debidos a la elección del automóvil privado como modo de transporte; asimismo, y dependiendo del grado de congestión existente en la ruta, existirá un peaje de acuerdo con la función establecida en el epígrafe anterior.

Los procedimientos matemáticos empleados en la resolución del problema de asignación dependerán de los supuestos considerados a la hora de establecer las características del sistema. Así pues, podemos establecer los siguientes modelos en función de dichas características:

a) *Modelos estáticos:*

Consideran la asignación del tráfico en un momento concreto del tiempo, así por tanto, éste no constituye una variable relevante. Refleja un modelo de asignación a corto plazo, por lo que emplean las funciones de coste a corto [3, 20, 41].

b) *Modelos dinámicos:*

Tratan la asignación [19] a lo largo de un periodo de tiempo pudiendo existir dos posibilidades:

1. *Modelos dinámicos puros:* Utilizan criterios de optimación dinámica [1, 23, 149]

2. *Modelos con ventanas temporales*: Consideran la división del período de tiempo tratado en h momentos o ventanas de tiempo de longitud igual o diferente. Utilizan los denominados algoritmos de asignación dinámica [31, 32, 117, 118, 148]

Un segundo esquema de clasificación de los algoritmos de asignación considera la existencia de diferencias a la hora de la elección de la ruta por parte de los usuarios que comentábamos al principio de este capítulo. Los usuarios que presentan distintas percepciones del costes presentan una elección de rutas que se considera regida por factores estocásticos; de ahí que se les denomine usuarios estocásticos. Por contra, aquéllos que consideran los efectos de la congestión actúan en función de la existencia de restricciones a la capacidad. Podemos reflejar los siguientes modelos:

a) *Modelos estocásticos puros*: consideran la elección de la ruta por factores estocásticos sin tener en cuenta los efectos propiciados por la existencia de restricciones a la capacidad [17, 18]

b) *Modelos estocásticos con consideración de las restricciones de capacidad*: introducen factores de ambas aproximaciones. Constituyen la formulación del equilibrio de usuario estocástico.[20, 153]

c) *Modelos de restricción de la capacidad*: sólo consideran los factores asociados a la existencia de congestión. Los métodos de resolución se denominan métodos de equilibrio del usuario. Son los modelos que consideraremos en nuestro estudio. Dentro de los mismos se pueden distinguir dos grupos:

1. *Modelos con flujo estático*: son aquéllos donde el número de usuarios que desean desplazarse entre el origen y el destino está predeterminado, y el modelo se limita a realizar una asignación estática de los mismos entre las diversas

rutas que comunican ambos nodos, de forma tal que minimice el total de costes de la red.

2. *Modelos con flujo variable no estocástico*: en este caso el número de usuarios dispuestos a desplazarse entre el nodo origen y el nodo destino vendrá determinado por una función de demanda, que presentará una relación inversa con el coste asociado al desplazamiento. En estos modelos existirá, no obstante, una cota superior a la demanda establecida en función de la población existente en ambos nodos.

Por último, podemos considerar un supuesto más a la hora de establecer las características de la red y el tipo de algoritmo de asignación a emplear; nos referimos concretamente al hecho de considerar la posible interacción que se puede producir entre los distintos arcos de la red, de forma tal que el coste asociado a atravesar un arco no sólo sea función del número de usuarios que utilizan dicho arco, sino que considere también los usuarios existentes en los arcos adyacentes.

3.3.1. Redes con flujo estático.

El método de resolución de una red con flujo estático responde al criterio del denominado equilibrio de usuario, basado en el principio de Wardrop.

Para el empleo de dicho criterio es necesario establecer una función de coste para cada arco de la red; así como un conjunto de nodos origen y un conjunto de nodos destino. El primero será el único punto de la red donde se generará el flujo, y el segundo será el punto a donde converge la totalidad de dicho flujo que desea utilizar la red. Supondremos que la red está compuesta por un conjunto de caminos entre ambos conjuntos de nodos, de forma que nos permita establecer la matriz origen - destino (O-D).

En estas circunstancias, el problema de asignación consiste en determinar el patrón de flujos de cada arco, de acuerdo con la demanda dada y de forma que minimice la suma de los costes de los usuarios de la red. Así pues, en términos económicos buscamos el punto de equilibrio entre la demanda de desplazamientos y la oferta de infraestructura viaria entre el conjunto de nodos origen y el de nodos destino, para una red dada.

El *principio de Wardrop* de optimización de la red se fundamenta en los siguientes puntos:

1. El coste de todas las rutas utilizadas es menor o igual que el coste que soportaría cualquier vehículo que emplease alguna de las rutas no utilizadas.

2. El coste medio del viaje en todas las rutas utilizadas es el mínimo posible.

A partir de este principio general, Beckman et al. [13] plantean la consideración de la función objetivo a minimizar como:

$$Z = \sum_i \int_0^{x_i} c_i(\omega) d\omega \quad (27)$$

donde i es el arco, y x_i es el flujo dentro de dicho arco, siendo c_i la función de coste asociada al mismo, la cual será creciente respecto al flujo.

Las restricciones serán:

1. La *ecuación de flujo*: nos dice que la suma de flujos a través de todas las rutas entre dos nodos ha de ser igual a la demanda de desplazamientos entre ambos nodos:

$$\sum_j f_j^{ab} = q^{ab} \quad (28)$$

donde j es el índice del camino y a y b son los nodos origen y destino; siendo f , el flujo total entre a y b a través del j -ésimo camino y q la demanda total entre a y b .

2. La *ecuación de no negatividad*: nos indica que el flujo por cada ruta ha de ser positivo o nulo.

$$f_j^{ab} \geq 0 \quad (29)$$

Debemos también tener en cuenta que el flujo por cada arco x_i cumple:

$$x_i = \sum_{ab} \sum_j f_j^{ab} \delta_{i,j}^{ab} \quad (30)$$

donde δ aparece definido en (3).

En Sheffi [153] se demuestra que esta formulación cumple el principio de Wardrop. A partir de la formulación de [13] se ha producido la resolución del problema mediante varios algoritmos. En [153, 169] se presentan varios algoritmos basados en procesos heurísticos. No obstante, el algoritmo más habitual en la resolución de este problema fue formulado por Frank y Wolfe en 1956, aunque ha sufrido numerosas modificaciones en años posteriores. La posible justificación de la utilización de los procedimientos de asignación estática puede comprobarse mediante la utilización de dos medidas propuestas por van Vliet [173]:

E_1 sería la diferencia entre el coste total en la ruta observada y el coste total en la ruta de coste mínimo, dividida entre el coste total en la ruta de coste mínimo; esto es:

$$E_1 = (\sum_i x_i C_i(x_i) - \sum_i y_i C_i(x_i)) / \sum_i y_i C_i(x_i) \quad (31)$$

donde y_i es el flujo asignado al arco i empleando un algoritmo "todo o nada" y el camino de coste mínimo.

E_2 sería el coste total en equilibrio menos el coste total con flujo libre, dividiendo el resultado entre el coste total con flujo libre.

$$E_2 = (\sum_i x_i C_i(x_i) - \sum_i z_i C_i(x_i)) / \sum_i z_i C_i(x_i) \quad (32)$$

donde z_i es el flujo cuando existe flujo libre.

Si $E_2 > E_1$ entonces la congestión es el principal factor a la hora de determinar la ruta; se debe aplicar en este caso el principio de Wardrop. Por contra, si $E_2 < E_1$ entonces el factor dominante en la red son las variaciones percibidas por el usuario en el coste, y por lo tanto, no sería aplicable el principio anterior. En general, se propone que si $E_2 > 10\%$ entonces la aplicación del principio de equilibrio del usuario sería la correcta; frente a esta situación, si $E_2 < 5\%$, entonces deberíamos considerar un algoritmo de asignación estocástica con demanda estática. El caso intermedio requeriría la aplicación del principio del equilibrio estocástico.

3.3.1.1. Método de las combinaciones convexas (Frank-Wolfe).

El algoritmo de Frank - Wolfe, también conocido como método de las combinaciones convexas, es un procedimiento iterativo que nos permite encontrar la solución de equilibrio de usuario en una red. La versión que nosotros hemos considerado en el presente estudio ha sido tomada de [153]. Basándonos en la formulación de [13] el algoritmo constaría de los siguientes pasos:

Paso 0: Desarrollar un procedimiento de asignación “todo o nada” basándonos en el coste de atravesar un arco con flujo libre, c_{fi} . Esto nos proporcionará los flujos iniciales en cada arco x_i^1 . Inicializaremos el contador de iteraciones a uno.

Paso 1: Calcular el coste asociado a cada arco $c_i(x_i; K_i)$ para los flujos hallados en la iteración anterior:

$$c_i^n = c_i^n(x_i^n; K_i); \quad \forall i$$

Paso 2: Desarrollar un algoritmo de asignación “todo o nada” con los costes obtenidos en el Paso 1; esto nos proporciona un segundo vector de flujos $y = \{y_i\}$.

Paso 3: Determinar la longitud de paso. Encontrar un α_n tal que resuelva el problema de optimización sin restricciones:

$$\text{Min}_{0 \leq \alpha \leq 1} \sum_i \int_0^{x_i^n + \alpha(y_i^n - x_i^n)} c_i(\omega) d\omega$$

Paso 4: Moverse a través de la dirección descendente y en base a la longitud de paso obtenida en el Paso 3, y establecer el nuevo vector de flujos:

$$x_i^{n+1} = x_i^n + \alpha_n (y_i^n - x_i^n); \quad \forall i$$

Paso 5: Test de Convergencia.

Si la solución obtenida en $n+1$ iteraciones cumple el criterio de convergencia establecido, entonces parar el algoritmo y mostrar solución óptima encontrada; en caso contrario continuar ejecución, actualizar iteración y volver a Paso 1.

Gráficamente el diagrama de flujo de este algoritmo queda:

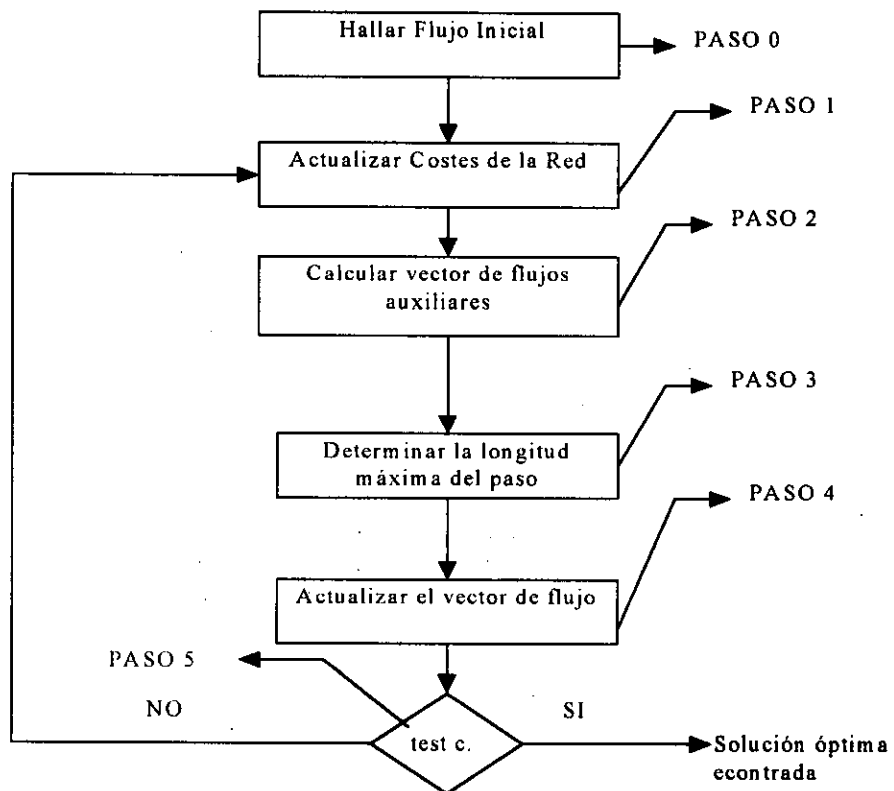


Ilustración 13 : Algoritmo de Frank-Wolfe.

Uno de los pasos más discutidos en este algoritmo es la relevancia del criterio de convergencia establecido, puesto que determinadas elecciones pueden condicionar el resultado final del algoritmo (ver [169]).

Nuestra elección se ha basado en aprovechar la propiedad del algoritmo de Frank-Wolfe del carácter descendente de la longitud de paso en iteraciones sucesivas, lo que permite establecer una cota ϵ , donde ϵ es un número lo suficientemente pequeño, de forma tal que si $\alpha < \epsilon$, entonces se considera que la solución hallada es óptima y se detiene el proceso de búsqueda.

Asimismo, hay que señalar que el número de iteraciones requeridas aparece afectado de forma significativa por el nivel de congestión de la red; esto se debe a que en el caso de redes poco congestionadas estamos trabajando en la parte plana de las curvas de coste, mientras que en redes donde la congestión ha alcanzado niveles severos nos desplazamos hacia la parte creciente de dicha curva. Lógicamente, la elección de la curva de coste se convierte de esta forma en un hecho que puede afectar al funcionamiento del algoritmo, tal y como señala Boyce et al. [23].

3.3.2. Redes con flujo variable no estocástico.

Este tipo de redes constituyen una aproximación más realista, al tener en cuenta la existencia de una relación entre el coste del viaje y el número de posibles usuarios que desean desplazarse entre los nodos origen y los nodos destino. La relación más frecuente es considerar una función de demanda que mostraría una relación inversa con el coste del viaje. Así pues, el número total de usuarios que desean desplazarse entre dos nodos a y b sería:

$$q_{ab} = D_{ab}(c_{ab}^*) \quad (33)$$

donde c^* sería el coste mínimo¹⁴ del desplazamiento entre los nodos origen y destino.

Si seguimos la formulación de [13] podemos escribir nuestro problema de optimización como:

$$\begin{aligned} \text{Min } z(x; q) &= \sum_i \int_0^{x_i} c_i(\omega) d\omega - \sum_{ab} \int_0^{q_{ab}} D_{ab}^{-1}(\omega) d\omega \\ \text{sujeto a:} \\ \sum_j f_j^{ab} &= q_{ab} \\ 0 \leq q_{ab} &\leq \bar{q}_{ab} \\ f_j^{ab} &\geq 0 \end{aligned} \quad (34)$$

donde D^{-1} es la función inversa de demanda de viajes entre los nodos a y b, x es el vector de flujos y q es el vector de demandas, siendo \bar{q} el número máximo de viajes que se producirían entre dos nodos a y b en función de la población de los mismos y la distancia existente entre ambos. Esta formulación del problema se puede establecer puesto que las funciones de coste con las que trabajamos son separables, positivas y crecientes.

Para enfocar la solución al problema (34) tenemos dos posibles actuaciones. Podemos trabajar con el método de Frank-Wolfe adaptado al nuevo problema, o bien considerar la reformulación planteada por Gartner [65] que convierte el problema en uno del tipo de demanda fija en una red aumentada.

¹⁴ Algunos autores consideran la posibilidad de trabajar con una función de demanda dependiente de algún tipo de medida de la desutilidad asociada a la utilización del arco.

3.3.2.1. Método de Frank-Wolfe para redes con flujo variable.

La aproximación mediante el método de Frank-Wolfe mejorado intenta encontrar la solución al problema (34) mediante una reformulación del mismo en términos de una única variable de flujo. Dado (30) y supuesto que:

$$q_{ab} = q_{ab}(f^{ab}) = \sum_j f_j^{ab} \quad \forall a, b \quad (35)$$

podemos escribir el problema original como:

$$\begin{aligned} \text{Min } z(f) &= \sum_i \int_0^{x_i(f)} c_i(\omega) d\omega - \sum_{ab} \int_0^{q_{ab}(f^{ab})} D_{ab}^{-1}(\omega) d\omega \\ \text{sujeto a:} \\ f_j^{ab} &\geq 0; \quad \forall j, a, b \\ \sum_j f_j^{ab} &\leq \bar{q}_{ab}; \quad \forall a, b \end{aligned} \quad (36)$$

La resolución de este problema se realiza mediante el método de Frank-Wolfe, a través de la búsqueda de un vector descendente en cada iteración que resuelva una aproximación lineal de la función objetivo. La descripción por pasos sería:

Paso 0: Encontrar un patrón inicial factible de flujos y demandas (x_i, q_{ab}).
Establecer el número de iteración a 1.

Paso 1: Calcular los costes asociados al flujo encontrado en el paso anterior y la función inversa de demanda que se corresponde con el valor de q hallado en ese mismo paso.

Paso 2: Calcular un camino m , en función del algoritmo del camino más corto, entre cada par de nodos a y b , considerando que cada arco aparece ponderado por el valor de c , en la iteración anterior:

$$\frac{\partial z[x(f^n); q(f^n)]}{\partial f_i^{m,n}} = g_i^{m,n}(f^n) - D_{mn}^{-1}(f^n) \quad (37)$$

En función del resultado obtenido:

$$\text{Si } g_m^{ab} < D_{ab}^{-1}(q_{ab}^n) \Rightarrow g_m^{ab} = \bar{q}_{ab} \text{ y } g_k^{ab} = 0 \quad \forall k \neq m.$$

$$\text{Si } g_m^{ab} > D_{ab}^{-1}(q_{ab}^n) \Rightarrow g_k^{ab} = 0 \quad \forall k.$$

Establecemos:

$$y_i^n = \sum_{ab} \sum_j g_j^{ab} \delta_{i,j}^{ab} \quad \forall i, j, a, b$$

$$v_{ab}^n = \sum_j g_j^{ab} \quad \forall j, a, b$$

(38)

Paso 3: Encontrar el valor de α que resuelva el siguiente problema de optimización sin restricciones:

$$\text{Min}_{0 \leq \alpha \leq 1} z(\alpha) = \sum_i \int_0^{x_i^n + \alpha(y_i^n - x_i^n)} c_i(\omega) d\omega + \sum_{ab} \int_0^{q_{ab}^n + \alpha(v_{ab}^n - q_{ab}^n)} D_{ab}^{-1}(\omega) d\omega$$

(39)

Paso 4: Actualizar las soluciones obtenidas en función del valor de la longitud de paso, α_n obtenida en (39).

$$x_i^{n+1} = x_i^n + \alpha_n (y_i^n - x_i^n)$$

$$q_{ab}^{n+1} = q_{ab}^n + \alpha_n (v_{ab}^n - q_{ab}^n)$$

Paso 5: Si se cumple el criterio de convergencia, entonces la solución encontrada es la óptima y el algoritmo se detiene, en caso contrario, actualizar la iteración y volver al Paso 1.

3.3.2.2. Método de modificación de la red.

Gartner [65] plantea la resolución del problema mediante una conversión en un modelo de demanda fija dentro de una red aumentada.

Si \bar{q}_{ab} es el valor máximo de la función de demanda $D^{-1}(c^*_{ab})$; entonces definimos el exceso de demanda del flujo entre los nodos a y b como $e_{ab} = \bar{q}_{ab} - q_{ab}$. Por tanto, podemos escribir la integral original de la función de demanda como una suma de integrales:

$$\int_0^{q_{ab}} D_{ab}^{-1}(\omega) d\omega = \int_0^{\bar{q}_{ab}} D_{ab}^{-1}(\omega) d\omega - \int_{q_{ab}}^{\bar{q}_{ab}} D_{ab}^{-1}(\omega) d\omega \quad (40)$$

La primera integral del lado derecho de (40) es una constante, y por tanto, se puede eliminar de la función objetivo sin que por ello se vea afectado el problema de minimización planteado.

La segunda integral se puede expresar en términos del exceso de demanda mediante un cambio de variable:

$$e_{ab} = \bar{q}_{ab} - q_{ab}$$

La nueva variable de integración sería: $v = \bar{q}_{ab} - \omega$, y por tanto:

$$-\int_{q_{ab}}^{\bar{q}_{ab}} D_{ab}^{-1}(\omega) d\omega = -\int_{\bar{q}_{ab}-q_{ab}}^0 D_{ab}^{-1}(q_{ab}-v)(-dv) = -\int_0^{e_{ab}} W_{ab}(v) dv \quad (41)$$

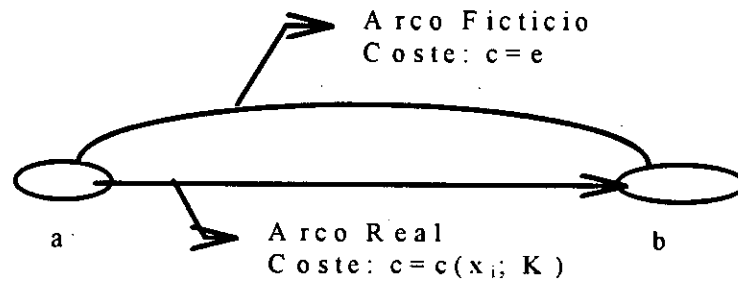
donde $W_{ab}(e_{ab})=D_{ab}^{-1}(q_{ab})$, para todo a y b .

Con (41) podemos reformular el problema (34) como:

$$\begin{aligned} \text{Min } z(x, e) &= \sum_i \int_0^{x_i} c_i(\omega) d\omega + \sum_{ab} \int_0^{e_{ab}} W_{ab}(v) dv \\ \text{sujeto a:} \\ \sum_j f_j^{ab} + e_{ab} &= \bar{q}_{ab} \quad \forall j, a, b \\ f_j^{ab} &\geq 0 \quad \forall j, a, b \\ e_{ab} &\geq 0 \quad \forall a, b \end{aligned} \quad (42)$$

siendo un problema de flujo estático que puede resolverse fácilmente con el método de Frank-Wolfe; no obstante, hay que tener en cuenta que Gartner apunta que el valor de \bar{q}_{ab} debe ser lo más pequeño posible; en este sentido, podemos tomar los valores producidos por los arcos con flujo libre. Sheffi [153] sugiere, por contra, que en cada iteración debemos intentar conseguir que los valores de \bar{q}_{ab} sean lo más próximos posibles a q_{ab} , y muestra claramente que la elección del valor \bar{q}_{ab} no influye en el resultado obtenido, siempre y cuando $\bar{q}_{ab} > q_{ab}$.

Gráficamente la aproximación sugerida por este método supone la consideración de la existencia de un arco extra que conecta los nodos origen y destino, tal y como mostramos en la figura:



3.3.3. Redes con efectos de interacción.

Esta última aproximación a las redes de tráfico considera la posible interacción que se puede producir entre los arcos de una red. Hasta ahora, habíamos asumido que el coste de un viaje a través de un arco dependía tan sólo del número de usuarios de dicho arco; en esta nueva aproximación consideramos que pueden existir interacciones entre el flujo en un arco i y el coste del viaje a través del arco j . Podemos distinguir dos tipos de interacción:

a) Simétrica: El efecto marginal del flujo de x_i sobre el coste c_j es el mismo que el efecto marginal del flujo de x_j sobre c_i .

b) Asimétrica: En este caso no se producen los mismos efectos marginal entre ambos arcos. Sobre este tema puede verse [39, 40, 64, 131]

3.3.3.1. Redes con interacción simétrica.

Las interacciones simétricas entre arcos son el supuesto más fácil de tratar en el análisis de redes de tráfico. El caso más simple es considerar una red con dos arcos i, j donde las funciones de coste de ambos arcos serían:

$$c_i = c_i(x_i / K_i; x_j / K_j)$$

$$c_j = c_j(x_j / K_j; x_i / K_i)$$

Según dichas funciones el coste de atravesar un arco i será función del grado de congestión de dicho arco, así como del grado de congestión¹⁵ del arco j .

El caso más sencillo es el que considera la interacción que se produce entre dos direcciones. Así sea i' el arco en dirección opuesta al arco i . Las funciones de coste de cada arco serán:

$$c_i = c_i(x_i / K_i; x_{i'} / K_{i'})$$

$$c_{i'} = c_{i'}(x_{i'} / K_{i'}; x_i / K_i)$$

En estos casos el supuesto de simetría resulta mucho más pausable que entre arcos en general. Desde un punto de vista formal este supuesto implica:

$$\frac{\partial c_i(x_i / K_i; x_{i'} / K_{i'})}{\partial x_{i'}} = \frac{\partial c_{i'}(x_{i'} / K_{i'}; x_i / K_i)}{\partial x_i} \quad (43)$$

Dado (43) la función objetivo del problema de minimización a resolver se puede escribir como:

¹⁵ La consideración del grado de congestión como medida de la influencia puede sustituirse por el flujo existente en dicho arco.

$$\text{Min } z(x) = \frac{1}{2} \sum_i \left(\int_0^{x_i} c_i(\omega, x_i) d\omega + \int_0^{x_i} c_i(\omega, 0) d\omega \right) \quad (44)$$

Esta aproximación supone que la función de integración de cada arco se divide en dos partes: por un lado, una primera integral que recoge el valor de la función de coste con respecto al flujo del arco cuando el flujo en dirección opuesta se mantiene constante, y una segunda integral que recoge el coste cuando el flujo en dirección opuesta es cero.

Los algoritmos para resolver el problema de optimización serán los mismos que para el caso del equilibrio sin interacciones. El algoritmo de Frank-Wolfe, por ejemplo, simplemente requiere la siguiente adaptación a la hora de calcular el valor de α :

$$\text{Min}_{0 \leq \alpha \leq 1} Z(\alpha) = \sum_i \int_0^{x_i^n + \alpha(y_i^n - x_i^n)} c_i(\omega, x_i^n) + \alpha(y_i^n - x_i^n) d\omega + \sum_i \int_0^{x_i^n + \alpha(y_i^n - x_i^n)} c_i(\omega, 0) d\omega$$

Problema de minimización fácilmente resoluble a través del cálculo de la primera derivada e igualando a cero. Esta aproximación sólo se mantiene en el caso de interacción simétrica entre arcos; es es, la influencia del flujo en j sobre un arco i es la misma que la influencia del flujo en i sobre el arco j .

Un ejemplo típico de interacción de este tipo lo constituyen el caso de la confluencia de dos arcos en un tercero. Sheffi [153] plantea el caso ilustrativo de este ejemplo como una red con tres arcos y cuatro nodos, (véase Ilustración 14) donde existen dos nodos origen y un nodo destino. En una red de este tipo, es evidente que ambos arcos confluyentes deben tener una interacción simétrica.

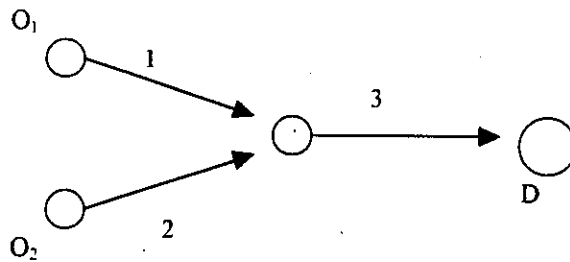


Ilustración 14: Red con interacción Simétrica

La condición de simetría en este problema viene dada por:

$$\frac{\partial c_1}{\partial x_2} = \frac{\partial c_2}{\partial x_1}$$

La resolución de esta red es muy sencilla puesto que una vez establecida la función objetivo se puede resolver de forma directa mediante el cálculo de la primera derivada e igualando a cero. Para obtener la función objetivo deberemos tener en cuenta que el flujo que circula por el arco 3, cumple: $x_3 = x_1 + x_2$, y $x_1 = q_{0D} - x_2$, donde q , es la demanda total de usuarios que desean desplazarse desde los orígenes al destino. Podemos comprobar la unicidad de la solución obtenida mediante el cálculo de la matriz Hessiana de la función objetivo. Si ésta resulta definida positiva, entonces la solución es única.

Las redes con interacción simétrica son fáciles de resolver puesto que la existencia de la condición de simetría nos permite utilizar los algoritmos habituales en los casos de redes sin interacción, simplemente modificando la función objetivo. La comprobación matemática de este hecho se recoge en [153].

3.3.3.2. Redes con interacción asimétrica.

La principal característica de la interacción asimétrica es la pérdida de la correspondencia entre los efectos del flujo de unos arcos sobre otros. Este hecho se refleja en la no verificación de (43) y por consiguiente la matriz jacobiana de costes del arco será asimétrica¹⁶.

La principal consecuencia de la pérdida de la condición de simetría es la no aplicabilidad del principio del equilibrio del usuario, puesto que la existencia de una matriz jacobiana no simétrica, no garantiza la unicidad de la solución [131].

Esta situación ha supuesto que no exista ningún algoritmo de solución basado en la programación matemática; no obstante, existen algoritmos "ad-hoc" basados en las ideas de la optimización y programación matemática. Sheffi [153] presenta dos algoritmos para la resolución de este tipo de problemas.

a) Algoritmo I.

Supongamos que en la n-ésima iteración conocemos el vector de flujos $x=(x_1, \dots, x_n)$, donde x_i se refiere al flujo que circula por el arco i . En esta situación se puede plantear el problema de minimización siguiente:

$$\text{Min } \tilde{z}^n(x) = \sum_i \int_0^{x_i} c_i(x_1^n, \dots, x_{i-1}^n, \omega, x_{i+1}^n, \dots, x_n^n) d\omega$$

sujeto a :

$$\sum_j f_j^{ab} = q_{ab} \quad \forall a, b, j$$

$$f_j^{ab} \geq 0$$

(45)

¹⁶ En el caso de la interacción simétrica esta matriz era simétrica.

donde (45) presenta el coste de cada arco como dependiente tan sólo del flujo que circula por dicho arco, puesto que el resto de variable permanecen fijas. Este problema se conoce como *problema diagonalizado*, ya que supone que la Hessiana de (45) se convierta en una matriz diagonal. Denotamos por \tilde{c}_i la función de coste del arco i -ésimo, donde todas las variables, excepto la x_i , permanecen asignadas a los valores fijos obtenidos en la n -ésima iteración. El algoritmo de solución planteado tiene los siguientes pasos:

Paso 0: Inicialización. Encontrar unos valores iniciales de flujos x^n , donde $n=0$.

Paso 1: Resolver el problema (45) utilizando el método de minimización del equilibrio de usuario. Esto nos producirá un vector de equilibrio x^{n+1} .

Paso 2: Si se cumple el test de convergencia, entonces hemos encontrado la solución óptima, en caso contrario establecer $n:=n+1$ y volver al paso 1.

Gráficamente el diagrama de flujo del Algoritmo I es:

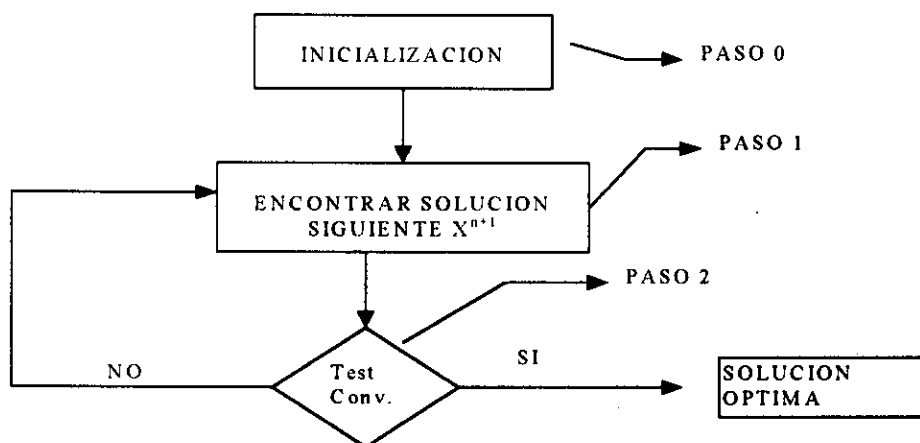


Ilustración 15: Diagrama de Flujo Algoritmo I.

El test de convergencia viene dado por el siguiente criterio:

$$\text{Max}_i \left\{ \frac{|x_i^{n+1} - x_i^n|}{x_i^n} \right\} \leq k$$

donde k es un número lo suficientemente pequeño.

b) Algoritmo II:

Este segundo algoritmo utiliza la diagonalización sólo en la resolución necesaria para obtener la longitud del paso a dar entre iteraciones [144]. Se basa en el método de Frank-Wolfe, pero tan sólo realiza una iteración, lo que supone que la única asignación se realiza mediante un procedimiento "todo-o-nada".

El algoritmo tiene los siguientes pasos:

Paso 0: Inicialización.

Establecer $n:=0$; y encontrar un valor inicial de flujos factible x^n .

Paso 1: Actualizar costes en la red.

$$c_i^n = c_i(x^n), \quad \forall i$$

Paso 2: Encontrar dirección descendente.

Hacer una asignación "todo-o-nada" basándonos en los valores obtenidos en c_i^n . Esto nos proporcionará el valor de flujos auxiliares y^n .

Paso 3: Encontrar la longitud de paso.

$$\text{Min } \tilde{z}(\alpha) = \sum_i \int_0^{x_i^n + \alpha(y_i^n - x_i^n)} c_i(x_1^n, \dots, x_{i-1}^n, \omega, x_{i+1}^n, \dots, x_n^n) d\omega$$

Paso 4: Actualizar la solución.

$$x_i^{n+1} = x_i^n + \alpha_n (y_i^n - x_i^n) \quad \forall i$$

Paso 5: Test de convergencia.

Si se cumple el test de convergencia entonces hemos encontrado la solución óptima, en caso contrario actualizamos iteración y volvemos al Paso 1.

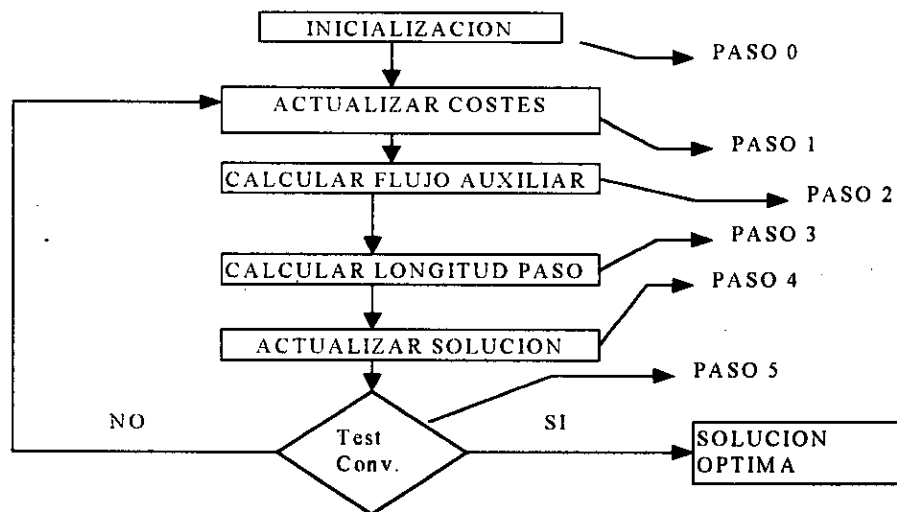


Ilustración 16: Diagrama de Flujo Algoritmo II.

Sheffi [153] sostiene que este segundo algoritmo presenta un mayor número de iteraciones antes de converger que el primer modelo de algoritmo establecido, pero

por contra, el gasto computacional del mismo es mucho menor, puesto que el algoritmo I requiere en media, unas tres asignaciones “todo-o-nada” para cada subproblema, mientras que el algoritmo II tan sólo exige una asignación “todo-o-nada” por cada subproblema planteado.

3.3.3.3. Implementación del método del Conjunto Activo.

Nuestra aproximación mediante el método del conjunto activo¹⁷ a redes con efectos de interacción se limitará a aquellos casos de existencia de costes asimétricos. Intentaremos comprobar la eficiencia del método en la resolución de este tipo de problemas frente al algoritmo II planteado por [153].

La implementación sugerida para la resolución de problemas de redes con interacción de costes asimétrica mediante el método del conjunto activo es la siguiente:

Paso 0: Inicialización.

Calculamos la matriz A de restricciones, hallamos la factorización LQ de dicha matriz y obtenemos la matriz Z del espacio nulo de A, formada por las (n-m+1) últimas columnas de la matriz Q; donde n es el número de variables del problema (= nº de arcos) y m es el número de restricciones del problema (= nº de ecuaciones de flujo). Hallamos la matriz auxiliar $B=I_n$. Asimismo, obtenemos una solución inicial factible a partir del conjunto de restricciones: x^0 , y ponemos el contador de iteraciones a cero, $n:=0$.

¹⁷ Una primera implementación aparece en [143].

Paso 1: Calcular la función objetivo.

$$F^n = \sum_i \int_0^{x_i} c_i(x_1^n, \dots, x_{i-1}^n, \omega, x_{i+1}^n, \dots, x_n^n) d\omega \quad (46)$$

Calculamos el gradiente de la función objetivo, g^n , y hallamos el valor en el punto x^n .

Si $n > 1$ entonces Paso 6, en caso contrario Paso 2.

Paso 2: Dirección de Búsqueda.

Obtenemos la dirección de búsqueda, a partir de:

$$g_z = Z \cdot g^n$$

$$B_z = Z' \cdot B^n \cdot Z$$

$$p_z = -(B_z)^{-1} \cdot g_z$$

$$p^n = Z \cdot p_z$$

donde p^n nos proporciona la dirección de búsqueda en la n -ésima iteración.

Paso 3: Longitud de Paso.

Determinamos la longitud de paso α .

$$\alpha^n = \text{Min} \left\{ \frac{x_i}{p_i^n}, 1 \right\} \quad \forall i$$

Paso 4: Actualizar la solución.

$$x^{n+1} := x^n + \alpha^n p^n$$

$$n := n + 1$$

Paso 5: Test de Convergencia.

Si la solución hallada cumple el test de convergencia, entonces hemos encontrado la solución óptima, en caso contrario, volver al Paso 1.

Paso 6: Actualizar la matriz auxiliar B.

Para actualizar la matriz auxiliar B, se utiliza la fórmula BFGS, lo que nos permite hallar la matriz B^n , una vez hallada volvemos al Paso 2.

Gráficamente el algoritmo planteado presenta el siguiente diagrama de flujo:

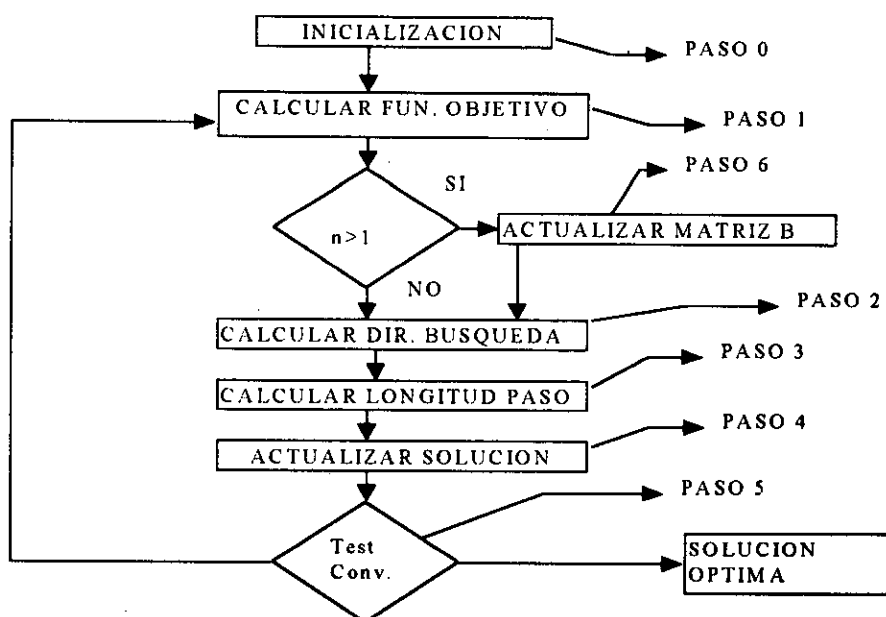


Ilustración 17: Diagrama de Flujo del Algoritmo Método Conjunto Activo

Para ver la eficacia del algoritmo podemos usar el siguiente ejemplo tomado de [153].

Ejemplo: Consideremos una red formada por dos arcos y dos nodos, uno origen y otro destino. Supongamos que las funciones de coste de cada nodo vienen dadas por las siguientes ecuaciones:

$$c_1 = 2 + 4x_1 + x_2$$

$$c_2 = 4 + 3x_2 + 2x_1$$

Considerando un modelo donde la demanda es fija; la ecuación de flujo resultante será: $x_1 + x_2 = 5$. El test de convergencia se establece para un valor de $k=0.01$.

Los resultados del algoritmo II se recogen en la siguiente tabla:

| Iteración (n) | x_1 | x_2 | Convergencia |
|---------------|-------|-------|--------------|
| 0 | 0 | 5 | -- |
| 1 | 1.714 | 3.286 | ∞ |
| 2 | 2.449 | 2.551 | 42.88 |
| 3 | 2.764 | 2.236 | 12.86 |
| 4 | 2.899 | 2.101 | 4.88 |
| 5 | 2.957 | 2.083 | 2.00 |
| 6 | 2.981 | 2.019 | 0.81 |
| 7 | 2.992 | 2.008 | 0.37 |
| 8 | 2.997 | 2.003 | 0.15 |
| 9 | 2.999 | 2.001 | 0.07 |
| 10 | 2.999 | 2.001 | 0.03 |
| 11 | 2.000 | 3.000 | 0.01 |

Aplicando al mismo ejemplo nuestro algoritmo los resultados se recogen en la siguiente tabla:

| Iteración (n) | x_1 | x_2 | Convergencia |
|---------------|-------|-------|--------------|
| 0 | 0 | 5 | -- |
| 1 | 5 | 0 | ∞ |
| 2 | 2.6 | 2.4 | ∞ |
| 3 | 3 | 2 | 0.2 |
| 4 | 3 | 2 | 0 |

Se observa que hemos alcanzado el mismo resultado que el algoritmo de Sheffi pero utilizando tan sólo 4 iteraciones de nuestro algoritmo, lo que supone una fuerte reducción de las iteraciones necesarias para alcanzar la convergencia con respecto a los algoritmos propuestos por Sheffi.

Gráficamente se puede observar las curvas de convergencia (Convergencia - N° de iteraciones) de los tres algoritmos comentados:

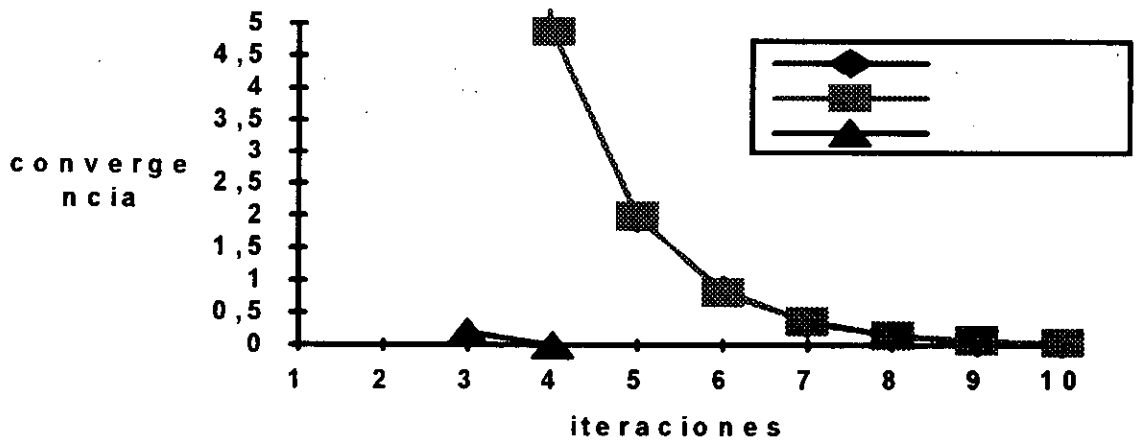


Ilustración 18: Gráfico de Convergencia de los Algoritmos planteados

Se observa la fuerte tasa de convergencia de nuestro algoritmo frente a los otros dos alternativos, también podemos ver que la tasa de convergencia del Algoritmo I y del Algoritmo II es prácticamente la misma, ya que ambas curvas se superponen en gran medida. Si consideramos la tasa de convergencia medida con respecto al valor final en cada iteración, veremos en que medida nos vamos aproximando a la solución óptima en cada una de las iteraciones realizadas, tal como se muestra en el gráfico (véase Ilustración 19). Dado de nuestro algoritmo emplea muchas menos iteraciones se observa una mayor tasa de convergencia a la hora de medir la aproximación a la solución óptima; no obstante, destaca el hecho de que los otros algoritmos empiezan la convergencia antes que el nuestro, a partir de la primera iteración, mientras que el algoritmo III no alcanza una buena convergencia hasta la segunda iteración. Este hecho pone de manifiesto que aunque el algoritmo III es más lento a la hora de encontrar la línea correcta de búsqueda, es muy superior una vez ésta ha sido hallada, puesto que utiliza de forma eficiente la información aportada por el gradiente de la función objetivo, la cual se va actualizando a lo largo del problema, de forma que en cada iteración recoge de forma más precisa la línea de búsqueda. Asimismo se beneficia de la información aportada por el conjunto de restricciones del problema, el cual no varía a lo largo del proceso de solución.

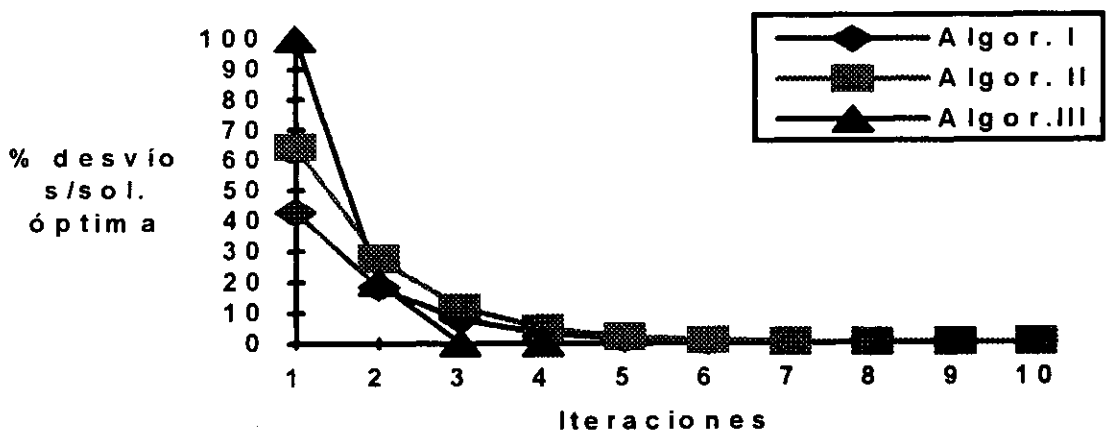


Ilustración 19: Convergencia respecto a la solución óptima.

La otra medida de eficacia es el coste operacional del algoritmo. Mientras el algoritmo I requiere la resolución de un problema auxiliar de minimización completo, los algoritmos II y III tan sólo requieren la resolución de una iteración de dicho problema auxiliar, dentro de cada una de las etapas del algoritmo principal. Sheffi sostenía la mayor eficacia del algoritmo II en base al menor número de asignaciones “todo-o-nada” a realizar: en el algoritmo I se ejecutaban 3 asignaciones por iteración, en el algoritmo II era tan sólo 1 asignación por iteración, frente a ellos el algoritmo III nada más que realiza una asignación “todo-o-nada” en el paso 0 de inicialización.

En el apéndice mostramos otro ejemplo donde nuestro algoritmo sigue mostrando una tasa de convergencia muy superior, aunque para la tasa considerada el resultado es ligeramente inferior al obtenido por el algoritmo II. Este hecho pone de manifiesto la necesidad de ser cuidadosos a la hora de la elección de la tasa de convergencia, puesto que un valor de k relativamente grande puede distorsionar fácilmente la solución del algoritmo.

3.4. Aproximaciones alternativas.

Por último, para finalizar este capítulo comentaremos otras aproximaciones alternativas al análisis de redes de tráfico que no serán consideradas en el presente estudio y que suponen distintas aproximaciones al problema de optimización de la red. Podemos considerar tres grandes grupos de aproximaciones alternativas.

En un primer grupo estarían los autores que han optado por una aproximación dinámica, bien sea en su versión de optimización dinámica [1] donde el tiempo es tratado como un *continuum*, o bien sea la utilizada por aquellos autores que emplean el recurso de la división del tiempo en espacios o ventanas temporales

discretas permitiendo la consideración de distintas características sobre los parámetros en cada ventana.

Un segundo grupo esta formado por aquellos autores que tratan las variables de flujo como variables estocásticas, trabajando con modelos probabilísticos a la hora de intentar estudiar la demanda de flujo de la red. Esta aproximación aparece recogida por varios autores:[153, 167, 169].

Por último, un tercer grupo de autores han optado por la consideración de aproximaciones más específicas sobre los parámetros o condiciones de funcionamiento de la red. Arnott, De Palma et al. [6, 7] consideran el caso de la existencia de los llamados “cuellos de botella” como factor fundamental dentro del análisis de la red, retomando en este sentido los trabajos de Vickrey [180]. Small [157] y Wilson [190] trabajan con funciones de utilidad basadas en el estudio de los horarios de los usuarios. El estudio de las colas y los retrasos originados en cruces y semáforos es el objetivo de autores como Garwood [66] o Arnott [5].

3.4.1. Modelos de asignación dinámica.

Los modelos de asignación dinámica fueron introducidos por Merchant y Nemhauser [117, 118]. Estos autores consideran la existencia de una red y un conjunto de pares de nodos origen / destino. Para cada par de nodos O/D establecen una función $R_{ij}(t)$, $0 \leq t \leq T$, que estima el número de usuarios que se desplazan del nodo i al nodo j en el momento t , y donde T es el horizonte de planificación. En este modelo los nodos intermedios de un camino pueden ser fuentes de flujo, cosa que no se producía en los anteriores casos analizados.

El análisis planteado considera la división del horizonte temporal de planificación en periodos de tiempo pequeños de igual tamaño, y donde $F_t(q)$ es el número de usuarios que se incorporan en el período t al nodo q desde fuera de la red.

Asimismo, cada arco i tiene una función de coste $c_{ii}(x)$ y una tasa de salida de usuarios $g_i(x)$, donde x es el flujo en el arco i al comienzo del período t ; por su parte, x_{ii} es el número de usuarios en el arco i al comienzo del período t , y d_{ii} es el número de usuarios que acceden al arco i en el período t .

Para cada nodo definimos dos conjuntos: $A(q)$ sería el conjunto formado por todos aquellos arcos que salen del nodo q , y $B(q)$ sería el conjunto formado por todos aquellos arcos que entran en el nodo q .

El problema planteado sería:

$$\text{Min } \sum_{t=1}^T \sum_{i=1}^a c_{ii}(x_{ii})$$

sujeto a:

$$x_{t+1,i} = x_{ti} - g_i(x_{ti}) + d_{ti} \quad t = 0, \dots, T-1, \forall i$$

$$\sum_{i \in A(q)} d_{ti} = F_t(q) + \sum_{i \in B(q)} g_i(x_{ti})$$

$$x_{0i} = R_i \geq 0$$

$$d_{ti} \geq 0, x_{ti} \geq 0$$

Carey [32] y Carey y Srinivasan [31] consideran diversas variantes de este problema. La más sencilla es el llamado "Problema S", que representa una red como un modelo de flujo estático y un único destino.

$$\text{Min } Z = \sum_{i \in A} x_i c_i(x_i)$$

sujeto a:

$$\sum_{i \in A(q)} x_i = F(q) + \sum_{i \in B(q)} x_i$$

$$x_i \geq 0$$

Posteriormente complican el modelo considerando la existencia de flujos dinámicos y peajes e introduciendo una variable v que recoge la capacidad del arco i , medida por la tasa de salida de usuarios de un arco en un período de tiempo t , generando el denominado “Problema C”:

$$\text{Min } Z = \sum_{t=1}^T \sum_{i \in A} c_{it}(x_{it})$$

sujeto a:

$$g_i(x_{it}) \geq v_{it}$$

$$x_{t+1,i} = x_{t,i} - v_{it} + d_{it} \quad t = 1, \dots, T-1, \forall i$$

$$\sum_{i \in A(q)} d_{it} = F_i(q) + \sum_{i \in B(q)} g_i(x_{it})$$

$$x_{1i} = \bar{x}_{1i} \geq 0$$

$$d_{it} \geq 0, v_{it} \geq 0, F_i(q) \geq 0$$

La resolución de este problema se realiza mediante la consideración de la Función de Lagrange y la búsqueda de las condiciones de Kuhn-Tucker.

3.4.2. Modelos Continuos.

Los modelos dinámicos puros realizan una aproximación al problema considerando la naturaleza continua del tiempo y del fenómeno de la congestión.

Agnew [1] presenta un modelo dinámico cuya resolución se realiza mediante los métodos de control óptimo.

Su modelo determina un peaje de control del arco en función de la variación del número de usuarios que acceden a dicho arco y con la función objetivo de maximizar el beneficio neto asociado a la utilización del mismo. Su comparación con la teoría estática muestra que sólo la aproximación dinámica es capaz de recoger el efecto del peaje sobre las cargas futuras de la red.

Otra aproximación de carácter dinámico se puede establecer a partir del estudio realizado por Beltrami [14] para el análisis de la velocidad de circulación de los automóviles en función del número de usuarios de la vía, esto es, de la densidad. Este estudio fundamenta la relación entre densidad y velocidad en los modelos de transmisión de ondas empleados en física, y que envuelven la utilización de ecuaciones diferenciales parciales de primer orden no lineales; siendo su objetivo el estudio de la ecuación característica del movimiento. Esta aproximación resulta bastante compleja desde un punto de vista formal y todavía no se ha utilizado en el desarrollo de análisis sobre los efectos de la introducción de costes directos (peajes) dentro del tráfico.

Por último, dentro del análisis dinámico debemos hacer referencia a los artículos de Bertsimas y Van Ryzin [17, 18] que trabajan con una modelo genérico de asignación de tráfico, de carácter dinámico y con demandas estocásticas. Una de sus conclusiones más interesantes es que mientras que en el caso que se considera que los vehículos en la red no tienen restricciones de capacidad implica que la condición de estabilidad del sistema será independiente de la geometría del mismo; en el caso de existir restricciones a la capacidad se produce una fuerte influencia entre la condición de estabilidad y la geometría del sistema.

CAPITULO 4

TEORIAS ECONOMICAS

DEL PEAJE

4. TEORIAS ECONOMICAS DEL PEAJE.

4.1. Peajes de Congestión.

La teoría económica tradicional establece que el precio competitivo de equilibrio de un bien es aquel que iguala el coste marginal de producir dicho bien. Desde el punto de vista de la literatura económica, si consideramos un mercado con competencia perfecta, el precio que los consumidores tendrán que pagar por los bienes de desean será el precio de equilibrio y, por lo tanto, será independiente del número de consumidores existentes en el mercado.

La provisión de un servicio o bien público también se puede considerar como un acto de consumo donde el consumidor, en vez de adquirir un bien, obtiene el derecho a la utilización y disfrute de un determinado servicio que le proporciona el proveedor; en este sentido el precio que el consumidor tendrá que pagar para poder disfrutar de dicho servicio vendrá determinado por el coste marginal asociado a la realización del mismo. No obstante, existe una diferencia sustancial entre la provisión de bienes al mercado y la provisión de servicios propiciada por la aparición del fenómeno de la congestión. La aparición de congestión en bienes públicos tiene como consecuencia un detrimento de la calidad general del servicio que reciben los usuarios de dicho bien. Por el contrario, los bienes de provisión privada son apropiados por los ciudadanos y por lo tanto, no existe el fenómeno de la congestión.

A la hora de estudiar la congestión debemos considerar que el término define a varias situaciones concretas en lo que se refiere al estudio del tráfico [58]. El artículo

de Vickrey sobre la congestión [180] y las inversiones en transporte recogían un total de 6 tipos de congestión, de los cuales destacamos los cuatro siguientes:

a) Interacción Simple: dos unidades se aproximan de forma tal que una de ellas deberá retrasar su marcha para evitar una colisión entre ambas.

b) Interacción múltiple: en estas situaciones la densidad de tráfico es alta, de forma tal que la velocidad de los usuarios es función del flujo: $s=f(x)$.

c) Cuellos de Botella: se refiere a aquel segmento de una ruta con una capacidad fija sustancialmente menor que la demanda de tráfico en los segmentos anteriores o posteriores al mismo.

d) Triggerneck: se refiere a la situación en la que las colas formadas ante los cuellos de botellas interactúan con el flujo de tráfico global y no sólo con aquella pendiente utilización de la infraestructura colapsada.

e) Control del Tráfico: en esta situación existe una regulación del tráfico mediante dispositivos físicos; semáforos, vallas, conos, etc.

Por su parte, el Institution of Highways and Transportation (IHT) considera en una publicación de 1992 cuatro tipos principales de congestión [89] en función del origen de la misma:

a) Congestión Prevista: aquella relacionada con los cuellos de botella y habitual en ciertas zonas, debida a las limitaciones físicas de la capacidad de la infraestructura de tráfico.

b) Congestión Ocasional: relacionada con demandas estacionales u obras planificadas.

c) Congestion Excepcional: promovida por circunstancias no habituales pero predecibles.

d) Congestión Inesperada: relacionada con accidentes, obstáculos en la vía, o condiciones meteorológicas adversas.

Dentro de los cuatro tipos, la más frustrante y más habitualmente estudiada es la del primer tipo, puesto que aunque sabemos que va existir el problema no parecen existir formas eficaces de superarlo.

Cuando los bienes públicos pasan a la esfera privada, como sucede en el caso de la desregulación del transporte y la provisión privada de dicho servicio, se establece un control de la calidad del servicio vía precio, que evita la aparición de congestión de una forma continuada. Numerosos autores consideran que una forma de evitar la congestión sería la privatización del servicio (Gomez-Ibañez & Meyer [74]); lo que disminuiría el gasto público y reduciría los niveles de congestión.

La alternativa de la privatización choca con el principio de libre circulación y movimiento de las personas que se establece en las regulaciones occidentales, y ha sido criticada por sus efectos sobre aquellas personas de rentas inferiores. En general, casi todos los países han optado por fórmulas mixtas donde se combina la existencia de infraestructuras privatizadas con alternativas públicas gratuitas. En el caso español, la existencia de autopistas de peaje ha estado vinculada a la presencia de una alternativa libre (Ribas & Montllor [150])¹. Esta fórmula no es aplicable, sin embargo, en los casos donde la congestión suele ser más severa, como es en el transporte urbano, puesto que supondría la duplicidad de infraestructuras en una situación donde la gran problemática surge por la escasez de suelo disponible.

¹ La primera vía de España con peaje fue el tunel de Guadarrama y se concedió en 1960.

4.1.1. Congestión y Tráfico.

La congestión se produce cuando un grupo de usuarios desean hacer uso de un mismo servicio en un mismo momento determinado del tiempo. Ante esta situación se produce un problema de colusión entre los distintos usuarios, que puede incidir o no en el coste que soportan por la utilización del servicio. Esta variación en el coste marginal permite el establecimiento de dos nuevas magnitudes. Por un lado, estaría el coste marginal privado, que es aquél que soporta el usuario por la utilización del servicio; por otro, el coste marginal social que es la variación en el coste marginal privado que la incorporación de un nuevo usuario supone sobre el conjunto de usuarios ya existentes. Esta última magnitud aparece ligada a la externalidad provocada por la incorporación de un nuevo usuario. La congestión es una de las externalidades más habituales asociadas al consumo de un bien público junto a otros tipos como es el caso de la contaminación.

La congestión presenta la propiedad de “no separabilidad” [50], esto es, el consumo de la externalidad es inseparable de la participación en la actividad que la causa. En este sentido, el consumo total de congestión de los individuos es función directa de la utilización del bien o servicio causante:

$$C_i = bx_i; \quad \forall i$$

Esta idea de no separabilidad supone que la congestión presenta naturaleza autolimitante; por tanto, alcanzado cierto nivel, los usuarios de un bien congestionado se limitarán en su uso, y así, el grupo de usuarios toma en consideración los costes totales externos que soporta a la hora de utilizar o no un servicio público. De esta forma, podemos hablar de un coste social, distinto al coste privado, que produce una diferencia entre el coste percibido por el usuario del servicio y el coste que la sociedad en su conjunto soporta realmente por la provisión del servicio, de modo tal que,

cuanto mayor es el gap entre ambos costes, mayor es el diferencial entre el precio y el coste real del servicio.

El ejemplo más habitual de congestión de un bien público es el caso del tráfico en carreteras y autopistas, especialmente dentro del entorno urbano. El continuo crecimiento de las ciudades ha propiciado que las necesidades de transporte de sus ciudadanos hayan aumentado de forma considerable, propiciando el incremento de usuarios de las vías existentes. Especialmente en los países con mayor grado de desarrollo, así como en aquéllos con una alta densidad poblacional, este problema ha ocupado muchas horas del estudio de economistas, ingenieros y otros expertos que han intentado hacer frente a la congestión sin que parezca que las medidas habitualmente adoptadas: - ampliación de las infraestructuras, creación de nuevas vías rápidas, mejoras del transporte público, encarecimiento del aparcamiento, etc. -, no hayan hecho otra cosa sino empeorar la situación existente.

El coste asociado a la utilización de una carretera se considera que viene dado por el tiempo que la persona utiliza para recorrer dicha vía desde su origen hasta su destino. Este tiempo es una magnitud física que dependerá fundamentalmente de la velocidad a la que circule el usuario, la cual a su vez dependerá de una serie de condiciones controlables por él mismo: tipo de coche, su pericia al conducir, etc. Al mismo tiempo será función de una serie de factores ajenos: el tipo de vía, el estado de la misma, el clima, y el número de usuarios que comparten dicha vía. Se ha podido demostrar que a mayor número de usuarios disminuye la velocidad a la que se circula por una carretera puesto que existe una interacción entre ellos (Herman [82]), que afecta al resultado final, esto es a la velocidad de circulación, y por tanto, al tiempo que los individuos utilizarán para alcanzar su destino. Dado que medimos el coste marginal a través del tiempo, observamos que el mismo varía cuando se modifica el número de usuarios del servicio, aunque desde un punto de vista económico todos los usuarios harán frente al mismo coste marginal. La visión del economista establece que el coste privado de un usuario está compuesto por la suma de sus costes monetarios y

su costes de tiempo, y por su parte, el coste social también sumará a los anteriores el coste externo.

Otros autores, como Button [29], consideran que la congestión no es la única externalidad asociada al transporte que debe tenerse en cuenta. En concreto, sostiene que siempre existirá un nivel mínimo de congestión de forma tal que hay que considerar los costes de congestión puros -el mayor consumo de carburantes así como el tiempo extra gastado en el desplazamiento-, y aquellos costes externos asociados a la congestión -el incremento de la contaminación-. En esta situación aboga por el calculo de un coste marginal superior al coste marginal social habitual, calculado en función del coste total.

En general, dentro del tráfico, el coste social suele ser superior al coste privado, lo que ha llevado a varios autores [158] a considerar el precio que los usuarios soportan individualmente por la utilización de la infraestructura de comunicación es inferior al precio real de provisión de la misma por parte de la sociedad. En esta línea abogan por el establecimiento de un peaje que reduzca el diferencial existente entre ambas magnitudes. Esta idea del peaje no parece ser aceptada satisfactoriamente por los decisores políticos de su aplicación, de ahí que por ahora, la mayoría de los países implicados en el tema hayan optado por retrasar su aplicación mediante la realización de otro tipo de medidas. No obstante, a nivel internacional existe un cierto acuerdo sobre la inevitabilidad del pago por la utilización de ciertas infraestructuras (véase [56]).

La aparición de la congestión lleva la no aplicabilidad de la teoría económica tradicional a la hora de la fijación de precios. El esquema alternativo habitual es el denominado *principio de Ramsey* que establece el vector de precios en función de un proceso de maximización de la suma del excedente del consumidor y el beneficio del empresario, sujeto a que el beneficio del empresario supere un determinado valor constante. La posible aplicación de este principio al tráfico urbano ha sido rechazada por Arnott y Krauss [4]; que muestran que en el caso del transporte la función de

coste de producción no sólo depende de la tecnología, sino que además presenta una relación de dependencia con el nivel de precios establecido, no siendo, por tanto, la solución establecida por el esquema de Ramsey, una solución de segundo óptimo; de ahí se deriva la necesidad de una correcta especificación de la función de costes.

Otros estudios inciden en la necesidad de determinar un cargo por la utilización de la infraestructura que cubra también los posibles desgastes de la misma (Newbery [136, 137] y Shah [148]), así como el intento de determinar el grado de capacidad óptima [159], de la vía, tanto desde el punto de vista de la congestión, como de la inversión (véase [38, 77, 190, 193]).

4.1.2. Peajes Optimos.

Los dos métodos más habituales para hacer frente a la congestión de una carretera son: la ampliación de la capacidad de la misma y el establecimiento de un peaje. La primera alternativa resuelve parcialmente el problema puesto que aproxima los costes medios y marginales, y por tanto, reduce el diferencial; pero su aplicabilidad es reducida, puesto que, como ha demostrado Downs [53] en su famosa ley sobre la congestión, el número de usuarios de una vía en hora punta crece hasta alcanzar la capacidad máxima de la vía. En este sentido, el peaje parece presentar una mejor solución.

Holden [86] establece un ejemplo claro del efecto perverso que supone en ciertos casos la mejora de una infraestructura para el conjunto de usuarios de la misma. Considera una situación donde existen dos modos de desplazamiento, el coche y el ferrocarril; en la situación inicial existe un equilibrio entre ambos modos que determina un flujo de usuarios x_1 y un tiempo medio de viaje t_1 (vease Ilustración 1). Si suponemos que el gobierno desea mejorar esta velocidad y procede a una mejora de la carretera, la nueva situación supone que el equilibrio se desplazará a una

situación donde el flujo de usuarios es mayor, x_2 , y el tiempo de viaje es el mismo, t_1 . Ante esta situación el modo de transporte alternativo perderá usuarios, lo que posiblemente lleve a una reducción del servicio y a un empeoramiento del tiempo de viaje, ante esta situación el equilibrio final se producirá para un flujo x_3 , y un tiempo de viaje $t_2 > t_1$. El resultado final es que la situación ha empeorado.

La idea de que una mejora de infraestructuras no siempre redonda en una mejora del bienestar colectivo nos lleva a la formulación del Tercer Principio de Wardrop [86], que se uniría a los comentados en el capítulo anterior:

“ El sistema tiende a un equilibrio en el cual ningún viajero puede reducir su tiempo de viaje eligiendo una nueva ruta”.

El establecimiento de un peaje por la utilización de una infraestructura aparece recogido en la literatura económico por primera vez en un escrito de Dupuit en 1849 que plantea la necesidad de un cobro para poder cruzar a través de un puente. Posteriormente, el tema será retomado por Marshal [109] y Piguou [146] junto con Knight [101]. No obstante, la primera aproximación moderna al tema fue establecida por Walters [183] que plantea la necesidad del establecimiento de un peaje considerando las funciones de coste. Walters considera la aplicación del principio del coste marginal al pago por servicios en las autopistas de Estados Unidos, llegando a la conclusión de que dicho pago está infravalorado con respecto al coste social, y que es preciso un aumento de las cargas a los usuarios a través de un incremento de los impuestos asociados al consumo de combustibles, así como el establecimiento de peajes especiales que gravasen las vías congestionadas. Numerosos autores utilizan este artículo de Walters como fuente de sus estudios sobre los peajes de congestión. Otros sin embargo, trabajan con funciones de utilidad en vez de con funciones de coste a la hora de establecer los peajes oportunos. En este caso plantean un problema de maximización de la utilidad, tales como los trabajos de Vickrey [179, 180], Mohring [125, 126], Kraus, Mohring and Pinfeld [103].

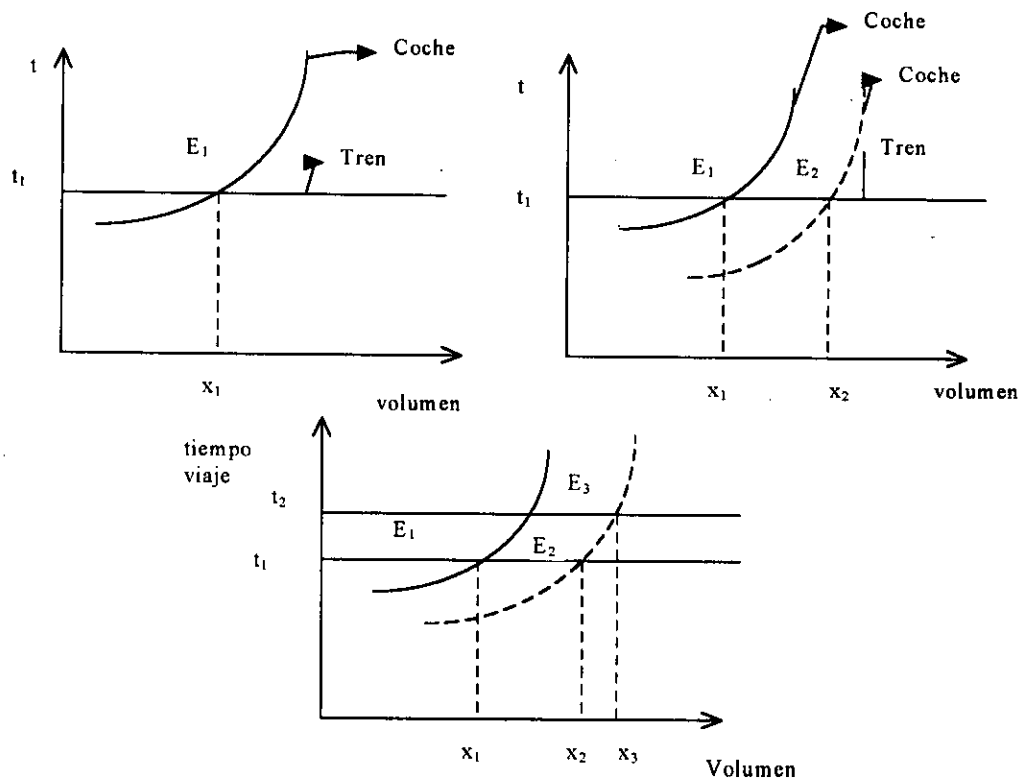


Ilustración 1: Tercer Principio de Wardrop

La idea de la implementación en la práctica de este tipo de peajes se recoge de forma explícita por primera vez en el denominado informe Smeed² de 1964, donde se estudia las posibilidades reales, con los niveles técnicos alcanzados hasta la fecha, de establecer un sistema de peajes sobre la utilización de carreteras en Gran Bretaña. No obstante, y aunque en el informe se establecía la posibilidad técnica de llevar a cabo tal medida, sus costes, tanto monetarios como político-sociales, llevaron a la no realización de las ideas establecidas en el mismo.

El aumento de los índices de congestión a finales de los años 80, que supuso una fuerte reducción de la velocidad media en numerosos entornos urbanos, así como

² Ministry of Transport (1964), Road Pricing: The Economic and Technical Possibilities (The Smeed Report), London, HMSO. [121]

la mayor preocupación por los costes medioambientales asociados, lleva a que el tema de los peajes urbanos sea retomado por las autoridades políticas de numerosos países que manejan seriamente la necesidad de establecer algún control que limite la utilización del automóvil privado en entornos urbanos. Un ejemplo del problema generado por la congestión se ve claramente si consideramos que los costes económicos [51] provocados por la misma en Londres se estimaban en unos 10 billones de libras esterlinas³, o las 250.000 horas de trabajo perdidas a diario en Madrid⁴ en el año 1992.

Todos los autores consideran como punto de partida de su estudio la existencia de una diferencia entre el coste marginal privado y el coste marginal social, de forma tal que el segundo es superior al primero. En este sentido proponen el establecimiento de una carga que grave la utilización de la infraestructura de forma tal que se aproximen el coste social y el coste privado. A esta diferencia entre el coste marginal social y el coste marginal privado es a lo que denominaremos peaje de congestión.

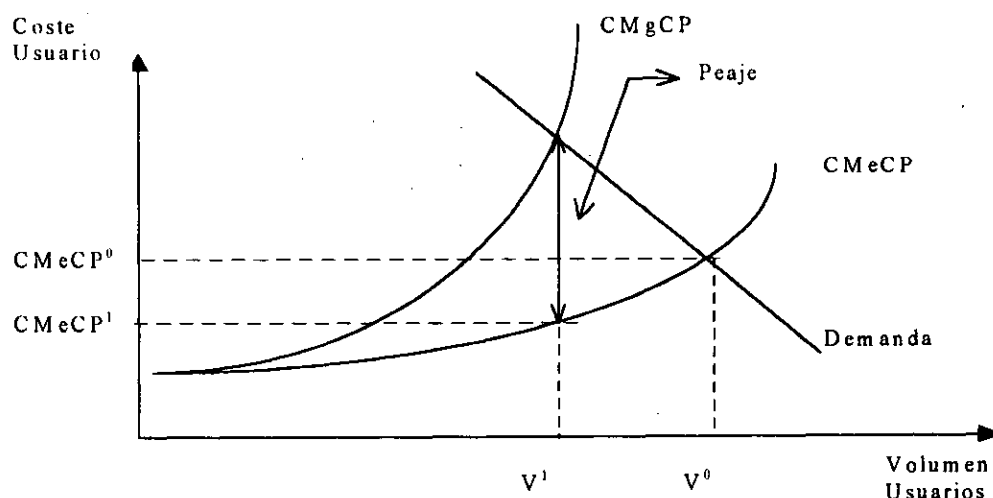


Ilustración 2: Establecimiento de un Peaje.

³ ECMT. 97th Round Table, Paris 1994, p. 111.

⁴ El País, Negocios, nº 462, 4 - Septiembre - 1994. p. 3

Desde el punto de vista económico los individuos que utilizan una infraestructura perciben como coste asociado al uso de la misma el Coste Medio a Corto Plazo (CMeCP), mientras que el coste social asociado a dicha utilización es el Coste Marginal a Corto Plazo (CMgCP); la diferencia entre ambos será el peaje de congestión que permita que el coste que realmente perciba el usuario sea el CMgCP, en el supuesto de que no existan más externalidades. Gráficamente la situación se muestra en la Ilustración 2. Desde el punto de vista del usuario el peaje supone un mayor coste, lo que incidirá en una caída en el volumen de personas que desean utilizar el servicio, y por tanto en el coste medio que éstos deben soportar [45, 54, 71,83, 128, 137, 161, 194].

Podemos establecer dos grandes grupos de teorías a la hora de abordar el problema del establecimiento del peaje óptimo. Por un lado, están aquellos autores que consideran un modelo independiente del tiempo, denominados de estado estacionario, por otro estarían los modelos dinámicos donde el tiempo es relevante a la hora de establecer el peaje óptimo.

4.1.2.1. Modelos de Estado Estacionario.

En los modelos de estado estacionario consideramos que el total de tiempo se ha dividido en H periodos de longitud q_h cada uno de ellos, y con un flujo de usuarios x_h . Durante cada período, los usuarios hacen frente a un coste de atravesar la infraestructura c_h , que será función del flujo y de la capacidad de dicha infraestructura.

El coste marginal se calcula mediante la derivada del coste total ($q_h x_h c_h$) respecto a un nuevo usuario ($q_h x_h$), resultando:

$$CMgCP = \frac{\partial(x_h q_h c_h)}{\partial(q_h x_h)} = c_h + x_h \frac{\partial c_h}{\partial x_h}$$

Dado que el peaje es la diferencia entre el CMeCP y el CMgCP, en este tipo de aproximación el peaje resultante será:

$$\text{Peaje: } \pi_h = x_h \frac{\partial c_h}{\partial x_h}$$

Por lo tanto, según este modelo, es necesario calcular para cada período de tiempo un peaje, que será igual al número de usuarios de dicho período por la parcial de la función de coste respecto al flujo que, dada una función de coste creciente, será un valor ≥ 0 . El resultado concreto dependerá del tipo de función de coste considerada.

La principal crítica a este modelo se centra en la falta de consideración de la interacción que se produce entre el establecimiento del peaje y el flujo de vehículos que demandan la utilización de la infraestructura, puesto que el modelo considera un movimiento a lo largo de la función de demanda y no un desplazamiento hacia la izquierda de la misma.

4.1.2.2. Modelos Dinámicos.

Estos modelos constituyen la aproximación más reciente al estudio del peaje. Aunque sus ideas básicas se apoyan en el artículo de Vickrey [180], su realización práctica hubo de esperar hasta los artículos de Arnot, De Palma y Lindsey [5, 6, 8, 9]. Estos autores plantean un modelo de “cuello de botella”, donde consideran tan sólo que la congestión se produce en un punto determinado de la vía, siendo el tráfico

libre en el tramo anterior y posterior a dicho punto. En esta situación el análisis de la congestión se limita al mencionado punto del trazado.

En su modelo básico estos autores consideran que $N(p)$, donde p representa el precio, personas desean llegar a un punto de destino común a una hora determinada, pero no es posible que todos alcancen el mismo punto a la misma hora, debido a la presencia de un cuello de botella que se congestiona para volúmenes altos de tráfico. Esto provoca que los individuos vayan ajustando su salida de acuerdo con sus necesidades de tiempo, y con el valor de llegar tarde o temprano a su destino.

El valor del coste privado que soportan los ciudadanos usuarios de la vía está constituido por tres tiempos (tiempo de viaje, tiempo de adelanto en la llegada al trabajo o tiempo de retraso en la llegada al mismo) cuya valoración es diferente. Por un lado, el tiempo necesario para el viaje, que es función de la hora de salida, puesto que dependiendo de la misma nos encontraremos con una mayor o menor congestión en el cuello de botella. Además estaría el coste asociado a llegar tarde al trabajo, o bien el coste asociado a llegar temprano.

Establecemos $D(t)$ como longitud de la cola en el cuello de botella que depende de la función de salida del origen, del momento en el que surge la cola \hat{t} , y del instante del tiempo t .

Así:

$$D(t) = \int_{\hat{t}}^t r(u) du - k(t - \hat{t})$$

donde $r(t)$ es la función de salida en función del tiempo (t) y k es la capacidad del cuello de botella. Establecemos el tiempo de viaje como:

$$Tv(t) = \frac{D(t)}{k}$$

con lo que el coste privado al que tiene que hacer frente el usuario es:

$$c(t) = \alpha Tv(t) + \beta(\max[0, t^* - t - Tv(t)]) + \gamma(\max[0, t + Tv(t) - t^*])$$

donde t^* es la hora de llegada deseada.

El precio asociado será el coste $c(t)$ más el peaje establecido. Resolviendo el modelo, establecemos el coste total del viaje como:

$$CT = \delta \left(\frac{N^2}{k} \right) \text{ donde } \delta = \frac{\beta\gamma}{\beta + \gamma}$$

A diferencia de otros estudios, la variable clave para establecer el coste, así como el peaje, será la hora de salida y no el flujo de usuarios de la vía. Se pueden establecer tres tipos de peaje:

a) Peaje uniforme: es aquel que se establece de forma constante para el conjunto del día. Puesto que supone añadir una tasa constante al precio que pagan los usuarios, las pautas de salida de los mismos no se ven alteradas. El peaje óptimo sería igual al coste medio total del viaje.

$$\text{Peaje: } \pi = \delta \left(\frac{N}{k} \right)$$

b) Peaje fino: este tipo de peaje varía con el tiempo y, por lo tanto, en función del grado de congestión. Cada viajero pagará el coste marginal social que iguala dos veces el coste medio de viaje.

$$\text{Peaje: } \pi = 2\delta \left(\frac{N}{k} \right)$$

c) Peaje intermedio: en esta situación los usuarios pagan un peaje de forma tal, que el peaje medio iguala al coste medio del viaje calculado sobre un coste total dado por:

$$CT = \frac{\delta}{4} \left[3 - \frac{(\gamma - \alpha)\beta}{(\beta + \gamma)(\alpha + \beta)} \right] \frac{N^2}{k}$$

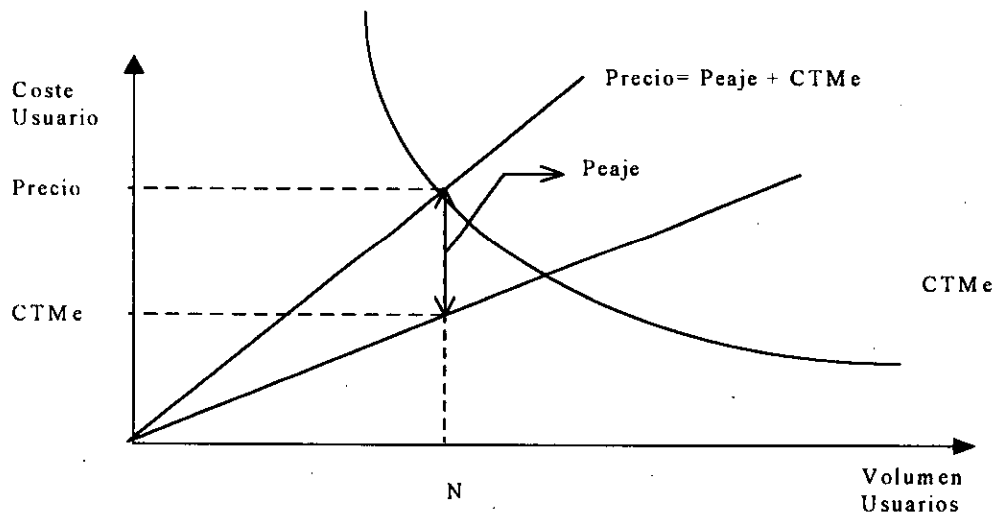


Ilustración 3: Peaje Optimo Dinámico

La característica más interesante que se deduce del modelo de estos autores es la separación del peaje óptimo de congestión en dos partes, una primera componente dependiente del tiempo, y una segunda componente invariante respecto al mismo. Una vez que consideramos que los usuarios actuarán teniendo en cuenta tan sólo la componente dependiente del tiempo, el problema del peaje óptimo será equivalente al modelo de estado estacionario.

4.2. Implementación Práctica.

Una vez considerada la posibilidad de la implementación de un peaje como medida correctora del gap entre coste marginal y coste medio que soportan los usuarios de una infraestructura, surge la discusión sobre la optimalidad del mismo; de forma tal que la inclusión de un peaje en los costes del usuario no suponga un abandono del criterio óptimo de fijación de precios; posteriormente se procede a su implementación real sobre los usuarios de forma eficiente, tal que no suponga la creación de obstáculos al libre tráfico que aumenten el tiempo de viaje y, por consiguiente, el coste soportado por los mismos.

Prácticamente todos los países presentan un sistema de cargo por la utilización de infraestructuras basado en el cobro de un impuesto sobre el combustible. Este sistema, que resulta de muy fácil implementación y recaudación presenta, no obstante, desde un punto de vista económico un efecto perverso, puesto que grava en igual medida a todos los usuarios, sin establecer la diferenciación pertinente entre las características de la vía utilizada o incluso la franja horaria. En este sentido, la aparición del cobro de peajes de forma directa se presenta como la alternativa más próxima al óptimo de tasación.

4.2.1. Características de los Sistemas de Cargo por la Utilización de Infraestructuras.

Los principios que deben regir la fijación de un sistema de cargo por la utilización de infraestructuras fueron expuestos por varios autores a lo largo de la 97th Round Table of European Conference of Ministers of Transport, ECMT celebrada en París en 1994. En ella se llegó a un consenso en relación al

establecimiento de una serie de principios comunes que deberían cumplir cualquier sistema de peaje:

4.2.1.1. Equidad del sistema.

El sistema deberá tener presente el principio de equidad, de forma que se grave en la misma medida a todos los potenciales usuarios. En este sentido se deberá considerar la necesidad de establecer controles sobre los usuarios que se desplacen a vías alternativas que les permitan evadir los peajes, frente a otros usuarios que carecen de esta posibilidad. Se ha detectado en algunos ejemplos prácticos que ciertos usuarios, dada la red existentes de vías gravadas, quedan ajenos al impuesto y, por lo tanto, gozan de una cierta ventaja sobre los otros usuarios que soportan los peajes. Particularmente, este caso se produce en los sistema de licencias de acceso al centro de las ciudades existente en Noruega.

La presencia de distintos tipos de usuarios también debe ser tenida en cuenta dentro de este principio. Así se deberá distinguir entre los usuarios habituales de la vía con peaje, y aquéllos otros que la emplean de forma ocasional. Esta situación afecta particularmente a los sistemas en los cuales los usuarios tienen que hacerse cargo de la instalación de los dispositivos electrónicos de tarificación en sus vehículos particulares; en el caso de usuarios muy poco habituales, el coste de dicha instalación puede resultar excesivo y, por tanto, suponer un mayor gravamen por la utilización de la vía. La alternativa a esta situación obliga a que los sistemas de control electrónico requieran la necesidad del mantenimiento de los sistemas tradicionales de barrera para aquellos usuarios en estas circunstancias.

Un aspecto de más difícil solución desde el punto de vista de la equidad, es el concerniente al tema de la equidad fiscal de los peajes. En general, se ha utilizado como elemento contra la implementación de peajes la idea de que aquellas personas con menores rentas sufrirían de forma más negativa el impacto del establecimiento de

los mismos. La mayoría de los defensores de los peajes sostienen la necesidad de que la fijación de la cuantía de los mismos tienda a tener en cuenta estos hechos. En particular, se ha argumentado que dada las bajas elasticidades de demanda del transporte privado, los efectos sobre los usuarios serían reducidos, puesto que sólo un grupo muy minoritario prescindiría de la utilización de los vehículos privados ante la existencia de un peaje.

Algunos autores sostienen que el establecimiento de un peaje no cumpliría las condiciones de óptimo de Pareto desde un punto de vista fiscal; puesto que existiría un grupo de usuarios que se vería perjudicados⁵ y no es fácilmente detectable que el conjunto se beneficiase. El sistema tradicional aplicado en esta situación es el del llamado “segundo-óptimo” que promueva la subsidiación de algún tipo de transporte público alternativo, esto es el autobús o el ferrocarril. Frente a este sistema en [140] se propone la implementación de un sistema de peaje utilizando los impuestos sobre el combustible como una variable proxy; de esta forma estaríamos ante una situación que se denomina de “tercer-óptimo”, aunque Mckee y West [116] consideran que es realmente una política de óptimo paretiano.

4.2.1.2. Privacidad.

La privacidad es otra de las características necesarias del sistema. Los sistemas de cargo deberán tener en cuenta la necesidad de mantener el anonimato del usuario del mismo, de forma tal que la información necesaria para el cobro del peaje no pueda ser utilizada de forma inadecuada por terceros.

En estos sistemas se deben de poder establecer las cuentas de cargo de forma anónima, de modo que no sea accesible la información sobre un usuario concreto o su utilización del sistema de forma identificada. En este sentido, los sistemas de pago con barreras, o mediante ticket que no identifican al usuario resultan más privados,

⁵ Un estudio empírico sobre los efectos en el conjunto de usuarios se puede ver en [56] pp. 122-123.

pero presentan la desventaja de que inciden de una forma más negativa sobre el flujo de tráfico de la vía.

4.2.1.3. Forzamiento del Pago.

Uno de los principios más comentados es la necesidad de dotar al sistema de algún tipo de capacidad que le permita registrar a aquellos usuarios que intentan no pagar y actuar de forma pausable contra los mismos. Normalmente se han considerado dos posibilidades como medios para forzar el pago de los usuarios:

- a) La detención del vehículo.
- b) El registro del número de matrícula.

El primer sistema, si bien parece de una mayor efectividad práctica, presenta el problema de afectar al flujo de tráfico existente de una forma más directa, impidiendo el discurrir del mismo y, por lo tanto, aumentando el grado de congestión. Por su parte, los requerimientos técnicos de este procedimiento obligan a la implementación de algún tipo de dispositivo que bloquee al infractor y no permita su movilidad hasta haber satisfecho el pago o respondido a los requerimientos del sistema. En caso contrario, la detención manual obligaría a la presencia de personal del staff técnico de forma continuada en la zona de peajes.

Los sistemas del segundo tipo resultan menos agresivos con los usuarios infractores y suelen consistir en la realización de un registro fotográfico de la placa de matrícula de los vehículos que se han saltado el control, para actuar en un momento posterior contra sus propietarios. Este sistema presenta, no obstante, algunos problemas técnicos para la correcta verificación del vehículo infractor, especialmente en el caso de más de un carril. A su vez pueden existir ciertas restricciones legales a este tipo de dispositivos en algunos países, que dificulten el empleo generalizado de estos sistemas de chequeo electrónico.

4.2.1.4. Efectos a Largo Plazo.

Los efectos a largo plazo del peaje es otra de las características a estudiar a la hora de su implementación. Se trata básicamente de responder a dos preguntas:

- ¿ Será políticamente factible mantener el valor real del cargo y, si es necesario incrementarlo en función del incremento que experimente la renta ?
- ¿ Cuáles serán las implicaciones a largo plazo sobre la utilización de la tierra ?

Las respuestas a estas preguntas permanecen abiertas. Se ha especulado sobre los posibles efectos de la utilización de la tierra en función de la existencia de algún tipo de peaje en la zona y como esto podría afectar a la distribución de actividades económicas, especialmente en políticas del segundo óptimo (véase [164]). Se llegó a la conclusión de que una tarificación por zonas es una política de segundo óptimo eficiente; y superior a los impuestos unitarios, que se muestran como relativamente ineficientes. En general, los efectos previstos más relevantes serían una redistribución de las actividades económicas de las zonas con peaje hacia aquellas localizaciones libres de tarifa, lo que normalmente supondría una salida de las actividades de negocio del centro de las ciudades hacia la periferia, y como consecuencia un repoblamiento del centro de las ciudades, y un incremento de los costes de transporte debido a la mayor dispersión. Un debate entre las economías de aglomeración y los costes de congestión se abre a la hora de considerar respuestas a este tipo de preguntas [100, 182, 187, 192].

Otros autores (véase [129, 142,157]) consideran que la necesidad del establecimiento de mecanismos que aproximen los costes de los usuarios obliga a la adopción de medidas de control y tarificación del tráfico, bien sea de forma directa a través de impuestos y peajes, o de forma indirecta a través del establecimiento de las

llamadas políticas del segundo óptimo, articuladas normalmente en torno a subvenciones a medios de transporte alternativos.

Por último, el establecimiento de sistemas de peaje flexibles respondería a los problemas aquí planteados de una forma eficiente. Se han propuesto sistema con alto grado de flexibilidad de forma que el peaje se estableciese en función del grado de congestión en un momento determinado y en un área concreta, es el denominado “sistema de peajes borroso” propuesto en Cambridge en el Reino Unido⁶.

4.2.1.5. Otros aspectos relevantes.

Existen otros aspectos cuya contemplación se hace necesaria a la hora de establecer sistemas de tarificación, entre otros podemos citar:

a) La provisión adecuada de medios de transporte alternativos al vehículo privado, especialmente a través de los sistemas de interconexión conjunta de bus y ferrocarril; o bien la mejora de las condiciones de desplazamiento de peatones y carriles bici en el centro de la ciudad.

b) La introducción de mecanismos que permitan a los ciudadanos replantearse sus horarios de desplazamiento de forma que se evite la aparición de congestión en horas punta coincidentes con los desplazamientos al trabajo. Se proponen medidas como la flexibilización de horarios, la redistribución de centros de trabajo, o el fomento de los denominados teletrabajadores.

c) El estudio de los problemas secundarios derivados de la desaparición de la congestión: la mayor accidentalidad existente provocada por el aumento de la velocidad, así como la necesidad de considerar la creación de infraestructuras de aparcamiento en los puntos de destino mayoritarios; o el incremento de la

⁶ vease [56], pp. 126-127.

contaminación derivado de la mayor utilización de los vehículos y las mayores velocidades.

4.2.2. Diferentes Tipos de Sistemas de Cargo.

Existen en la actualidad diversos sistemas de cobro por la utilización de infraestructuras, algunos de los cuales ya están siendo utilizados en algunas zonas, frente a otros que tan sólo se han propuesto como alternativas posibles [59, 78, 81]. Entre ellos podemos identificar diferencias:

a) Tasas sobre el carburante.

Si bien constituyen el método más extendido, como ya hemos comentado, resulta poco apropiada para reducir la congestión y aproximar la diferencia entre costes de usuario. Se podría considerar que cumple un papel meramente fiscal, gravando el consumo de un determinado bien. En este sentido se aproxima a los impuestos generales al consumo.

Se han propuesto mejoras en la equidad del sistema, tales como gravar el consumo de combustible en áreas congestionadas, y otras similares, pero no parecen ser excesivamente satisfactorias.

No cumplen los criterios de equidad.

b) Tasas sobre vehículos.

A diferencia de las tasas sobre carburante, en este caso el objeto del cargo es directamente el vehículo y, por lo tanto, el usuario se hace mucho más consciente del coste extra añadido por la utilización del mismo. Existen dos tipos de aproximaciones:

1. Tasas Independientes de la Utilización de la Infraestructura. Gravan la posesión del vehículo independientemente de su uso. No responden a los criterios de equidad, ni de privacidad.

2. Tasas Relacionadas con la Utilización de la Infraestructura. Son sistemas que gravan al usuario en el caso de utilizar una infraestructura concreta. El ejemplo más extendido es el de las Autopistas de Peaje, donde la presencia de barreras a la entrada y salida de las mismas obliga al pago de un cargo por su utilización. En la actualidad existen varias alternativas a este sistema de barreras físicas, muchas de ellas todavía en proceso de experimentación:

- Venta de Licencias (vignette): En este sistema, el usuario compra mediante el pago de una cantidad fija, una licencia para poder acceder a una zona determinada durante un período concreto de tiempo. En caso de que no desee utilizar la infraestructura no tendría que pagar.

- Contadores: El sistema de contadores consiste en la instalación en el vehículo de un dispositivo que registra los kilómetros que dicho vehículo ha circulado a través de una vía determinada; posteriormente se puede proceder a la lectura de dicho contador y se le carga el importe al propietario del vehículo.

- Sistemas de Pago automático (Automatic Debiting Systems - ADS)⁷: En estos sistemas, se registra el paso del vehículo por una zona determinada y se procede a efectuarle un cargo en la cuenta de dicho vehículo. Los requerimientos técnicos en esta modalidad son muy elevados, puesto que es necesario un sistema que identifique al vehículo de forma eficiente, así como un mecanismo de registro donde figure dicho vehículo y conste con una cuenta asociada para efectuar el cargo.

⁷ Existe un proyecto de standarización en Europa del sistema ADS, recogido dentro del proyecto DRIVE (Dedicate Road Infrastructure for Vehicle safety in Europe).

4.2.3. Viabilidad del Sistema de Peajes.

Los objetivos últimos que lleven a la implantación de un sistema de peaje pueden ser de tres tipos:

- a) Incrementar los ingresos del gobierno.
- b) Reducir la congestión.
- c) Mejorar el medio ambiente.

Cada uno de estos objetivos propicia un diferente tipo de peaje, y supone una reacción más o menos adversa por parte de los ciudadanos. La viabilidad del sistema se enfrenta, por tanto, a este tipo de reacción adversa que puede condicionar de una forma clara su realización.

El tema de la aceptación por parte del público de los peajes ha sido muy discutido a lo largo de los últimos años. En 1990, en un estudio realizado en Londres sobre una muestra de 489 habitantes (NEDO [134]), se llegó a la conclusión de que los ciudadanos se oponían en un cierto grado a la implementación de un peaje, de forma que menos de la mitad, el 43%, eran favorables a los mismos. Las razones que se esgrimían contra los peajes eran cuatro⁸:

- 1) Problemas con respecto a la confidencialidad en ciertos sistemas.
- 2) Temor al exceso de tecnología en la conducción, y a los posibles fallos del sistema.
- 3) Oposición por creer que supone un nuevo impuesto a los conductores.
- 4) Un sentimiento de oposición a la intromisión del estado en la esfera privada del ciudadano. Según señala David Banister⁹ existe “un resentimiento contra

⁸ véase [56], pp. 123-126.

⁹ [56], pp. 126.

pagar por algo que no quieren y que ya ha sido considerado a la hora de tomar la decisión de emprender un viaje”, refiriéndose al tiempo gastado por culpa de la congestión.

Esta oposición manifiesta de los ciudadanos al establecimiento de un sistema de peajes lleva a que en numerosos países existan fuertes recelos políticos que impiden el establecimiento de los mismos¹⁰. Por su parte, el empresariado tampoco se muestra favorable a la implantación de peajes puesto que repercutiría en sus costes de transporte, y por lo tanto encarecería la producción. Al mismo tiempo, podría tener ciertos efectos en términos de una relocalización de las actividades productivas hacia las zonas libres de peaje.

No obstante, algunos autores [160] apuntan un cierto cambio de actitud por parte de los usuarios, que hasta ahora habían apostado por la mejora de las infraestructuras como forma de remediar la congestión existente. Parece deducirse de este cambio de actitud que los ciudadanos empiezan a comprender la denominada ley de Downs, que supone que toda infraestructura viaria tiende al colapso; esto es, una mejora de las infraestructuras no parece producir a largo plazo una mejora de los niveles de congestión. En un estudio realizado en Gran Bretaña, se afirma que, incluso con un incremento del 50% en el gasto real en carreteras, la congestión seguiría empeorando, puesto que el incremento del tráfico se expandiría aún más rápidamente¹¹.

Este cambio de actitud va acompañado de un conjunto de reivindicaciones que los usuarios creen que debería cumplir el sistema para ser aceptable:

¹⁰ En Londres esta parece ser la principal causa de que aun no existan (*The Economist*, 12-11-1994, pp 45-46).

¹¹ *The Economist*, 3-12-1994, p.39.

- La recaudación obtenida a través del peaje debería revertir en la mejora de las infraestructuras de forma que sirviese, tanto, para realizar mejoras en las mismas, como para mejorar las zonas afectadas por la implantación del peaje.

- El peaje deberá tener un efecto redistribuidor de la renta de tal forma que aquellas zonas cuya actividad económica o desarrollo urbano presente mayores niveles de infradotación se vean beneficiadas de las rentas obtenidas por el gobierno.

- La mayor sensibilidad de los ciudadanos hacia el medio ambiente también se apunta como una de las causas que ha favorecido este cambio de actitud. En concreto, en varios países occidentales, la mayor preocupación medioambiental ha puesto de manifiesto la necesidad de establecer sistemas que penalicen el daño que el automóvil provoca mediante la contaminación atmosférica. En concreto, se ha propuesto la creación de un sistema combinado que conlleve un incremento del impuesto sobre carburantes complementado con el establecimiento de un peaje en determinadas zonas, que sirva para repercutir sobre el usuario los costes ambientales¹².

- Proposición de los peajes dentro de una visión global de la política de transportes que incluiría entre otros: el establecimiento de subsidios al transporte público, control de las plazas de aparcamiento en las ciudades; planificación del uso de las distintas zonas urbanas.

En estos momentos, pues, la viabilidad de los sistemas de cargo por la utilización de infraestructuras parece encontrarse tan sólo con el fuerte costo político que una medida de este tipo podría tener para el gobierno; de hecho, en aquellos países donde existe un mayor grado de responsabilidad social, caso de los países escandinavos, o bien donde el régimen político no se ve presionado por las

¹² The Economist, 17-6-1995, p. 20, y The Economist, 29-10-94, p. 48-49.

necesidades electorales, como sucede en Singapur o Hong-Kong, este tipo de medidas han sido llevadas a cabo.

4.2.4. Ejemplos Reales.

Existen en la actualidad varios ejemplos de aplicación real de peajes a infraestructuras. Como ya hemos comentado, el sistema más extendido es el de creación de un peaje para una infraestructura concreta: una autopista, un tunel o un puente. En la actualidad se está procediendo a la implementación de peajes generalizados en algunas ciudades, existiendo varios proyectos en curso.

En la Unión Europea el establecimiento de peajes se aborda dentro del programa DRIVE, en un proyecto denominado CASH [Co-ordinating of Activities for The Standardization of HADES (High-level Automatic Debiting system Specification)]. Su principal objetivo es el establecer los estándares en el campo de los sistemas de pago automáticos (ADS)¹³ que se aplicarían a toda la Unión; agrupa a operadores privados de Austria, Francia, Italia y España, así como a las administraciones de carreteras de Holanda, Noruega, Suecia y la Universidad de Newcastle (Reino Unido). En España, existe un antecedente de estudio para la implantación de peajes realizado por González Blanch, et al [75], centrado en el establecimiento de un sistema de tarificación para el transporte pesado por carretera. Estos sistemas de tarificación del transporte de mercancías por carretera persigue básicamente una mayor equidad de costes con respecto a otros tipos de medios de transporte, especialmente el ferrocarril, cuyas infraestructuras tienen unos costes mayores que no pueden ser repartidos entre todos los ciudadanos, a diferencia de los costes de las infraestructuras de carreteras y autovías. Se busca, por tanto, el reducir la subsidiación encubierta que se realiza del transporte de mercancías por carretera.

¹³ La descripción de este proyecto figura en *Description of the Automatic Debiting System, DRIVE-CASH, September 16, 1992, Revision Date: May, 3, 1993.*

Una aproximación más reciente centrada en la ciudad de Barcelona se puede ver en [10].

Otros proyectos próximos a la realidad son el sistema de pago electrónico (ERP) establecido en Hong-Kong, cuyos resultados de funcionamiento provisionales aparecen en [160], y el futuro sistema de pago en horas puntas que se pretende instaurar en algunas ciudades de los Países Bajos a finales de la década de los noventa (ver [56]).

En cuanto a los ejemplos reales, ya implantados hoy en día, se destacan principalmente dos:

4.2.4.1. Noruega.

En Noruega se ha establecido un sistema de peaje electrónico conocido como "Tronhdheim Toll Ring", que responde a la tecnología más avanzada hoy en día, para cobrar un peaje a aquellos automovilistas que desean acceder a un área concreta de la ciudad. El sistema identifica al usuario y procede a cargar en su cuenta el importe del peaje. Como el procedimiento es electrónico, los usuarios no necesitan detenerse para abonar el mismo. Para su funcionamiento es necesaria la instalación de un dispositivo electrónico en el automóvil; si no se desea se puede pagar de forma directa en los peajes tradicionales, bien a los empleados o bien en la oficina postal.

El sistema funciona sólo entre las 6:00 a.m. y las 5:00 p.m.; siendo gratuito el acceso a otras horas del día. Esta situación supone que en las horas fronterizas exista un mayor volumen de coches que desean entrar o salir del área, lo que provoca un cierto grado de congestión. La tarifa de pago manual es de 1.25 ECU por coche; y la tarifa de pago electrónica es un poco más económica 1.00 ECU entre las 6:00 -- 10:00 a.m. y entre las 2:00 -- 5.00 p.m.), y con diferentes descuentos en el resto del horario. Existe un límite al número de pagos que un usuario puede realizar al mes, 75,

que en caso de ser rebasado supone que los viajes extras no sean cobrados. A través de esta medida se pretende que el sistema no sea excesivamente gravoso para aquellos usuarios frecuentes, como pueden ser repartidores, mensajeros, etc. que se ven obligados a entrar y salir con cierta frecuencia del área dentro de las horas tarifadas.

El principal objetivo de este sistema es fiscal y, por lo tanto no se centra en reducir el grado de congestión, sino en la obtención de fondos para su posterior empleo en la mejora de las infraestructuras de la ciudad; no sólo del área restringida, sino del conjunto del área urbana. El nivel de ingresos anuales del sistema es de 8.75 millones ECU, ligeramente por debajo de las necesidades del plan general de infraestructuras del área de Trondheim, estimadas en 12.5 millones de ECU anuales.

El sistema de peajes parece funcionar adecuadamente y es muy pequeño el número de usuarios que alcanzan el número máximo de pagos mensuales. La mayoría de los usuarios optan por los sistemas de pago electrónico, bien en cuentas directas, o través de transferencia bancaria.

4.2.4.2. Singapur.

En la ciudad de Singapur existe un peaje para permitir el acceso a la ciudad. El sistema sigue el modelo de licencias que se pagan de forma previa y que dan derecho a acceder con el vehículo en determinadas horas del día. Es uno de los modelos de peaje más antiguos y fue introducido en 1975. Se conoce como el nombre de Area Licensing Shceme (ALS).

Todos los usuarios que desean acceder al centro de la ciudad de Singapur deben de comprar un billete (la licencia) que pegan en el parabrisas de sus vehículos. Esta licencia autoriza el acceso entre las 7:30-10:15 a.m. de lunes a sábado y de 4:30-6:30 p.m. de lunes a viernes. Existe un grupo de personas que controlan que los

coches lleven la licencia; si no lo hacen, se les toma nota de la matrícula para proceder al posterior envío de una multa a sus domicilios.

Las tarifas han variado a lo largo de los años para irse adaptando a la situación particular del momento; en [61] se señalan unas tarifas que oscilan en función del tipo de vehículo, entre los 0.6 dólares americanos (\$) y los 3.5 \$ para el billete de un día, y los 11.6 \$ y 70 \$ para el abono mensual.

El principal objetivo del sistema implantado en Singapur, a diferencia del sistema noruego, es reducir la congestión existente en el área central de la ciudad, el conocido como distrito central de comercio y, al mismo tiempo fomentar la utilización del transporte público e incrementar el número de usuarios de cada vehículo privado mediante el fomento del denominado "car-pooling", consistente en que varias personas compartan un mismo coche para efectuar sus desplazamientos al lugar de trabajo; de hecho, hasta fechas recientes los automóviles con más de 4 pasajeros estaban exentos de pagar la licencia para acceder al área restringida.

Los objetivos planteados parece que han sido logrados en un grado considerable con las medidas tomadas: la congestión se ha reducido de forma considerable, durante las horas de la mañana el tráfico ha caído en un 73%; al mismo tiempo, se ha incrementado la velocidad media de circulación, que ha pasado de menos de 20 Km/h a 33 Km/h. Otra consecuencia positiva ha sido la redistribución de los usuarios entre los distintos modos de transporte. Se ha incrementado el número de usuarios del autobús y del sistema de coche compartido, y ha disminuido el uso del coche privado. Como consecuencia negativa debemos citar el incremento de la congestión que se ha producido en las zonas alrededor del área restringida, motivada por el aumento de usuarios que utilizan la zona para evitar el centro de la ciudad a las horas punta.

4.3. Políticas Alternativas.

Los detractores del sistema de peajes también han ido creciendo y refinando sus conceptos a lo largo de los años [119]. Lo que en principio era una oposición frontal basada en el excesivo coste de los sistemas de peaje, así como en el hecho de que tenderían a incrementar la congestión más que a aliviarla, ha dejado paso a nuevas críticas y a propuestas alternativas que pasamos a examinar.

4.3.1. Críticas al Sistema de Peajes.

La principal crítica contra la posibilidad de cobrar un peaje por la congestión se basa en la consideración de la ley de Downs que como ya hemos comentado, establece el principio de congestión de cualquier bien público. Este hecho, conocido en la literatura económica como “tragedia de los comunes”, se sostiene sobre la base de la existencia de una demanda latente del uso de las carreteras. Esto es, existe un grupo de usuarios que no demanda la utilización de una carretera en las condiciones iniciales, pero si se produjese una variación en las mismas que conllevara la mejora de su funcionamiento, éstos demandarían la utilización de la carretera y, por lo tanto, reducirían la efectividad de las medidas establecidas para mejorar el rendimiento de la carretera. Así pues, toda medida, y no sólo el establecimiento de peajes, promovida para mejorar el nivel de congestión existente no tienden a producir una reducción significativa de la misma; y por tanto, no parece que deba ser implementada en el caso de que los costes asociados sean elevados. Esta idea sugerida para todos aquellos bienes públicos con precios reducidos y con capacidad de congestión se puede establecer en las carreteras en función de las respuestas esperadas a un sistema de cobro de peajes. Según figura en el informe [56] ante un sistema de peajes los ciudadanos reaccionarían con las siguientes consecuencias:

- Se produciría una redistribución del número de viajes existentes entre los diferentes modos de transporte, de forma tal que aquéllos con menores costes se

verían beneficiados frente a aquellos cuyos costes se incrementarían de forma sustancial al establecer el peaje de congestión.

- Asimismo, existiría una redistribución entre las rutas utilizadas, las horas, y los destinos, buscando las mayores posibilidades de reducción de costes. Este hecho supondría, entre otras consecuencias, que algunos usuarios buscasen su vivienda en las proximidades del centro de trabajo, o en aquellas zonas con mejor dotación de medios de transporte público alternativos, incrementando el valor de la vivienda en estas zonas y reduciendo la densidad de las zonas más alejadas e infradotadas.

- Por último, una tercera opción sería la supresión de algunos de los viajes programados, lo que según algunos autores afectaría al conjunto de la actividad económica puesto que ciertos viajes de ocio o con destino a centros comerciales se verían afectados y por lo tanto, reducirían las ventas de los comerciantes ubicados en dichas zonas.

Otras críticas buscan argumentos diferentes al establecido por Downs, considerando los siguientes puntos:

- Viajar se constituye un derecho básico y la imposición de cualquier medida que tienda a reducir los viajes afectaría a este derecho, y por tanto, introduciría un grave sesgo a la libertad individual.

- El establecimiento de un peaje de congestión presenta incentivos muy importantes para los gobiernos puesto que serían los mayores beneficiarios de la recaudación obtenida, mejorándose, de este modo, los ingresos públicos [57]. Al mismo tiempo, sería visto como una nueva forma de impuesto desde la óptica de los ciudadanos.

- Otro argumento muy generalizado es que aquellos ciudadanos de rentas más bajas se verían perjudicados por los peajes [157], aunque este hecho dependerá del tipo de congestión y tráfico existente¹⁴.

- La invasión de la privacidad es otro de las posibles consecuencias del establecimiento de un sistema de peajes, que como se ha comentado, debe ser objeto de especial cuidado a la hora de establecer las características del sistema de cargo.

- Por último, los efectos sobre el conjunto de la actividad económica pueden ser importantes puesto que algunas zonas sufrirían los efectos del peaje de una forma más importante que otras existiendo una posible discriminación entre áreas, lo que conllevaría un replanteamiento de la localización de las actividades.

Evans [57] considera que el peaje de congestión sólo es admisible en ciertas situaciones¹⁵ pero que no se puede tomar de forma generalizada como medida contra la congestión, puesto que otras medidas: discriminación entre modos de transporte mediante carriles reservados, o fomentar la mayor ocupación de vehículos, ...; tienen efectos menos perversos sobre el bienestar de los ciudadanos. No obstante; esta situación parece aplicable tan sólo a un análisis de estática comparativa [85], que no considere los efectos de doble dirección entre peajes y congestión.

4.3.2. Alternativas de Tarificación.

Existen numerosas alternativas a la política de peajes para hacer frente a la congestión en una ciudad [154, 172, 176, 188, 191]. En varios casos son medidas concretas aplicables a situaciones específicas; otras, al contrario, son medidas que

¹⁴ Evans [57] muestra que esta redistribución de rentas sería cero en el caso de cuellos de botella con tráfico homogéneo, pero sería positiva en el resto de posibles casos de congestión.

¹⁵ Considera el caso particular de los cuellos de botella.

tienen validez generalizada y por lo tanto son susceptibles de una aplicación más extendida. Nosotros nos limitaremos a comentar tres posibles alternativas [77, 94, 113] dentro de las que suelen emplearse de forma más habitual como formas de reducción de la congestión.

4.3.2.1. Pagos por Aparcamiento.

La política de establecer tarifas por aparcar los vehículos, bien sea en lugares cerrados, o bien en la vía pública es una de las más antiguas y aceptadas entre las medidas de reducción de la congestión y establecimiento de tarifas sobre el tráfico (Vickrey [177]). Normalmente las restricciones al aparcamiento han tenido como objetivo el desincentivar la utilización del automóvil como medio de acceso a ciertos destinos a través de dos tipos: restricción del número de espacios de parking existentes en la zona; o bien, reducir el tiempo permitido de aparcamiento.

La facilidad de implementación de los métodos de control, así como la limitada oposición pública ha llevado a que estas medidas se encuentren entre las más habituales dentro de las medidas públicas de control del tráfico urbano.

Existen varios estudios sobre el tema [28, 36, 70, 72, 186] que coinciden en valorar la viabilidad de las políticas de aparcamiento como fácilmente implementables, pero que destacan en muchos casos los fallos de este tipo de políticas como medidas del control de la congestión. Los principales argumentos contra esta alternativa son:

- La falta de un auténtico control de las plazas de parking regulado, así como la ausencia de una capacidad real para forzar a aquellos usuarios que incumplen la normativa al correcto cumplimiento de la misma.

- El incremento del tráfico que se produce en las zonas con regulación de aparcamiento, puesto que si el número de plazas es escaso muchos usuarios

entrarán en maniobras de tráfico de diversión con el objetivo de obtener una plaza en una zona más próxima a su destino final. Esta crítica supone que el control de aparcamientos no sólo no reduce la congestión sino que tiende a aumentarla, puesto que los usuarios reducirán de forma sustancial sus velocidades en las proximidades de sus lugares de destino incidiendo en el conjunto de usuarios de la red.

- Las tarifas establecidas normalmente no parecen tener incidencia sobre el número de vehículos aparcados, que parece responder de forma muy elástica a variaciones en los precios del parking. De este hecho parece deducirse que la mayoría de las tarifas establecidas son pequeñas [56] y que, por lo tanto, no cumplen con el objetivo final previsto de reducción de la congestión. El establecimiento de una tarifa correcta se convierte así en una de las principales dificultades del sistema, puesto que su fijación requiere un profundo estudio de la zona y hábitos de los usuarios. Gillen [70] sugiere que los individuos reaccionan cambiando sus lugares de aparcamiento ante subidas en áreas localizadas, y que por tanto, el incremento del precio del aparcamiento en una zona no redundará en una disminución de la congestión y sí en una recolocación de las plazas utilizadas.

- La necesidad de reducir el número de vehículos aparcados ilegalmente, así como de establecer los mecanismos necesarios para el control de los infractores, también supone una seria dificultad de los modelos de control del aparcamiento que no han sido capaces de evitar estas situaciones que limitan todavía más el correcto discurrir del tráfico en las zonas con aparcamiento regulado.

Estos fallos en el sistema, al que se suman a otros, ha llevado a algunos autores a afirmar que el control de aparcamientos tan sólo supone una política de redistribución de los usuarios, pero que no es válida como medida de control de la congestión. Glazer y Niskanen [72] sostienen que las políticas de costes de aparcamiento no son equivalentes a las políticas de costes de conducción, puesto que este principio sólo se cumpliría para aquellos usuarios que aparcan durante períodos de tiempo de longitud fija; por ejemplo, el tiempo de trabajo, pero no para aquéllos

otros que presentan longitudes de tiempo variable a la hora de aparcar. Si éstos últimos constituyen la mayoría del tráfico en la zona, entonces ésta no se verá favorecida por la existencia de tarifas sobre el aparcamiento, sino que además existirán efectos negativos sobre la congestión existente.

La comparación entre ambos métodos de control del tráfico parece quedar abierta puesto que los dos presentan ventajas y desventajas que deben ser analizadas rigurosamente de forma previa a la implementación de los mismos. Arnott et al. [7] realizan un análisis comparativo entre ambos sistemas concluyendo la validez de los sistemas de cobro por aparcamiento en el caso de áreas donde no existen sistemas de peaje por utilización de las infraestructuras, aunque los supuestos establecidos por los autores en su análisis resultan excesivamente restrictivos como para permitir una generalización de los resultados obtenidos.

Así pues, la posible regulación del aparcamiento en determinadas zonas de la ciudad parece no producir los efectos deseados en dichas zonas, excepto en circunstancias muy particulares, y por lo tanto, no parecen una buena medida de control de la congestión. A pesar de ello, las principales razones de su difusión en la actualidad se centran en: la escasa oposición por parte del público, el menor efecto redistribucional negativo sobre los usuarios de menores rentas (véase [73, 157]); así como la facilidad de implementación y la generación de ingresos para las autoridades pertinentes.

4.3.2.2. Subvención al Transporte Público.

Existen varios autores que proponen diversas medidas para reducir la congestión del tráfico urbano cuyo nexo común es el favorecer la utilización del transporte público alternativo, bien mediante autobuses o ferrocarriles. La elección de un modo u otro se basa principalmente en la disponibilidad del mismo, puesto que las

inversiones iniciales en el caso del ferrocarril son muy cuantiosas y suponen un fuerte coste. La medida habitualmente recomendada consiste en el establecimiento de un precio del billete en el transporte público menor del precio del mercado con el fin de atraer a un mayor número de usuarios (véase [175]); en este caso existe una subvención al precio de utilización de estos modos de transporte.

Diversos estudios han tratado de cuantificar el valor de dicha subvención de forma tal que exista un equilibrio económico entre modos alternativos. En el caso español existen estudios de De Rus [46, 47, 48] y estudios más genéricos sobre tarifas e infraestructuras se pueden consultar en [21, 24, 29, 123, 147, 155, 170].

El proceso de determinación del precio del transporte público se vuelve fundamental a la hora de establecer la utilidad del mismo como modo alternativo y, por lo tanto, sus posibles efectos sobre la congestión urbana. Cándido Carbo y Ginés De Rus [30] presentan un modelo de elección de precio para una red de líneas urbanas explotada en régimen monopolístico, que ha de cubrir costes. En su estudio concluyen que, bajo determinadas circunstancias, los incrementos de precio del billete no sirven para cubrir costes puesto que el detrimento del servicio reduce el número de usuarios; en esta situación, la reducción de costes permite ofrecer a los usuarios un servicio de mayor calidad sin incrementar los precios. En el caso español consideran que “la fijación de tarifas a un nivel relativamente bajo, estaría impidiendo que la cantidad del servicio responda a las demandas de los usuarios”. (Carbajo & De Rus, op. cit. p. 286).

El sistema tradicional de establecimiento del subsidio se realiza mediante el modelo aleatorio de tiempo de espera, que supone que los pasajeros llegan al punto de salida del servicio de una forma aleatoria. En este esquema, el coste marginal privado del usuario es igual para todos ellos e iguala al coste medio; no existiendo el gap entre ambos habitual en el transporte por motivo de la congestión. Los costes habitualmente considerados son el tiempo transcurrido entre servicios consecutivos o

bien el número de servicios por unidad de tiempo¹⁶. Para calcular el importe del subsidio se procede a la minimización del coste del usuario. En [170] propone la utilización de un modelo más complejo donde los usuarios puedan actuar de un modo planificado, obteniendo fuertes reducciones en el nivel de los subsidios establecidos con el modelo tradicional.

El análisis de Webster [185] sobre el nivel de subsidización del transporte público en Europa señala que en gran parte de los países de Europa Occidental este es muy elevado, cubriendo hasta un 70% de los costes de operación. Manifiesta, no obstante, el riesgo de estas políticas puesto que, si no existe una mejora de la calidad o si el transporte público presenta una tendencia decreciente, este tipo de situaciones supone la necesidad de incrementar cada año el valor de las subvenciones. La principal conclusión aboga por la necesidad de una mejora de la eficiencia y competitividad del sector, de forma tal que se pueda establecer una paulatina reducción de las subvenciones existentes, las cuales, en general, se consideran excesivas según la mayoría de los estudios realizados.

El estudio del transporte urbano no debe limitarse, así pues, a un mero estudio de la fijación de tarifas, sino que debe ir integrado en un análisis conjunto del desarrollo del mismo. Esto supone considerar la fijación de rutas, capacidad de autobuses, establecimiento de derechos de paso, frecuencias, etc.; de modo que se convierta en una auténtica alternativa al transporte privado.

La mayoría de los autores coinciden a la hora de señalar al transporte público como una alternativa al transporte privado cuando el destino es el trabajo; quedaría por resolver la situación de congestión en otros destinos, donde el papel de alternativa resulta mucho más complicado debido a la menor uniformidad de los desplazamientos. En este sentido, los desplazamientos de ocio u otro tipo presentan mayores dificultades puesto que no permiten el establecimiento de esquemas horarios sencillos,

¹⁶ En este modelo se asume que el transporte cumple los horarios establecidos.

a diferencia de lo que sucede con los desplazamientos al centro de trabajo, los cuales tienden a concentrarse en un conjunto de horas determinado, siendo los principales responsables de la congestión existentes en dichos horarios.

4.3.2.3. Usos Alternativos.

Por último, existe una alternativa a los peajes que afecta al conjunto de la idea de ciudad, así como la organización que actualmente existen en el conjunto del transporte urbano. Supone la realización de estudios más complejos sobre la estructuración y formación de los entornos urbanos que consideren las necesidades de transporte y permitan una planificación eficiente de los mismos. Se pueden establecer mejoras en el tráfico que disminuyan la congestión en las ciudades a través de dos alternativas de mejora de la situación actual:

a) Mejoras Urbanas.

Las políticas de mejora urbana se fundamentan en la idea de maximizar la accesibilidad de las ciudades [87, 122], de forma tal que todos los destinos principales sean fácilmente accesibles desde todos los puntos de la ciudad, sin necesidad de recurrir al transporte privado en automóvil. Asimismo, en un segundo lugar, se pretende minimizar la longitud media del viaje, de forma que los usuarios estén dispuestos a utilizar modos de transporte alternativos dados sus respectivas valoraciones del costo temporal.

Los planificadores argumentan que, bajo estas premisas, la reducción de la congestión sería suficiente como para evitar la necesidad del establecimiento del sistema de peajes. Un ejemplo práctico de esta situación se sitúa en Zürich (Suiza)¹⁷

¹⁷ vease FitzRoy & Smith [62]

donde se ha establecido una regulación especial respecto al transporte público que tienden a beneficiarlo frente al transporte privado y así fomentar el uso del primero.

El esquema de Zürich propone la creación de caminos especiales para autobuses y tranvías, siguiendo el esquema apuntado por Berglas et al. [16] sobre el establecimiento de derechos de paso exclusivos para el transporte público, de este modo ambos medios de transporte no estarían afectados por la congestión general de la ciudad y los tiempos de realización de los servicios mejorarían de forma sustancial. Las medidas establecidas en Zürich suponen la creación de carriles exclusivos, el establecimiento de abonos temporales, tiempo de espera cero en semáforos para el transporte público y mejora de las condiciones del servicio. Al mismo tiempo, se dificultaría el acceso de los vehículos privados a través de restricciones a su uso en determinadas zonas que se fomentarían como zonas peatonales y la reducción de las plazas de aparcamiento, lo que implicaría el progresivo abandono de los vehículos privados como medio de desplazamiento por parte de los ciudadanos.

Los autores señalan, no obstante, que objetivos tan ambiciosos, similares a los expuestos por Dimitrou & Hass-Klau [52] o Whitelegg, J. [189] chocan con la oposición frontal del “poderoso lobby de los automóviles”. Asimismo, su aceptación masiva por el público no siempre estará garantizada y dependerá de las circunstancias locales de cada entorno.

b) Mejoras en los Automóviles.

Un segundo grupo de alternativas aboga por la implementación de ciertas mejoras tecnológicas en los automóviles que faciliten tanto su conducción como sus prestaciones. Así pues, se sugiere al utilización del vehículos eléctricos de bajo consumo y reducido tamaño como forma de disminuir la contaminación e incrementar el espacio de aparcamiento en el centro de las ciudades.

La mejora de los automóviles también afecta a aspectos técnicos como la posible implementación de mecanismos informáticos de guía que encaminen a los vehículos por aquellas rutas donde el grado de congestión es menor, así como, que permitan determinar los tiempos medios de viaje previstos y controlar de forma eficiente la asignación de vehículos dentro de la red. Esta visión futurista es la que se recoge en dos proyectos de la Comunidad Europea; el programa DRIVE (Dedicated Road Infrastructure for Vehicle Safety in Europe), sobre el que existe un estudio de su aplicabilidad en España (Muñoz [130]), y PROMETHEUS (Programme for European Traffic with Highest Efficiency and Unprecedented Safety). Estos servicios son conocidos en inglés como servicios de RTI (Road Transport Informatics), y su objetivo es tanto la mejora de las prestaciones de las infraestructuras viarias, como el incremento de la seguridad y la disminución del número de accidentes.

Su principal justificación se basa en el supuesto del posible incremento que la demanda de transporte tendrá en los próximos años en el conjunto mundial. Según la Comisión Europea, el transporte representaba en 1991 más del 6% del PNB y suponía más del 10% del presupuesto medio de una familia; presentando además una tendencia creciente, lo que supone la inevitabilidad de la aparición de los “cuellos de botella” en el conjunto de infraestructuras europeas en los próximos años.

Las ideas apuntadas dentro de estos programas de mejora de los vehículos permitirían una reducción limitada de la congestión, centrándose la eficiencia del sistema en un mejor manejo del enrutamiento del tráfico, lo que a la larga redundaría en una mejora del sistema de transporte global, incluido el transporte urbano.

Las primeras implementaciones del sistema ya se están produciendo a través del sistema de radio RDS (Radio Data System) que combinan los canales especializados en información del estado del tráfico TMC (Traffic Message Channel) permitiendo un contacto directo entre los gestores del tráfico y los automovilistas a

través de sistemas de lectura cartográfica. Este sistema ya se ha implementado en algunas ciudades de Francia y dentro de algunos vehículos de la empresa Renault¹⁸.

Existe una cierta tendencia por parte de la industria automovilística a fomentar la aparición de mecanismos electrónicos e informáticos que tiendan a facilitar la conducción, aunque estrictamente hablando, no producen un efecto directo sobre la congestión urbana. Desde el punto de vista práctico, su único efecto sobre la misma sería una reducción del número de vehículos que se incorporen a las áreas congestionadas, siempre y cuando, los usuarios tuviesen alternativas disponibles. La gran dificultad a la que se enfrenta esta industria y por tanto la eficacia de los resultados dependerá de la reacción de los ciudadanos ante la proliferación de este tipo de vehículos (Smart Car¹⁹), y la adaptabilidad de los conductores al manejo de este instrumental. Puesto que, en general, se ha detectado entre los usuarios una cierta reacción contraria a la excesiva sofisticación que están sufriendo los automóviles en las últimas décadas.

Podemos terminar este capítulo concluyendo que, en general, la mayoría de las alternativas existentes al establecimiento de peajes resultan complejas a la hora de su implementación e introducción y, tampoco muestran una eficiencia superior a la situación de congestión actual, de ahí que la mayoría de los expertos consideren que dado que no parece previsible una disminución en la utilización de los automóviles privados, puesto que las cifras de propietarios de vehículos han ido creciendo a lo largo de las últimas décadas, especialmente en aquellos países que partían de niveles inferiores y en donde se espera que todavía sigan aumentando. Al mismo tiempo, parte de los recursos públicos sin que haya contribuido de forma sustancial a la mejora de la situación existente, el peaje se convierte, bajo estas circunstancias, en el método más eficiente para luchar contra la congestión urbana.

¹⁸ ECO, Setembro 1994, p. 90.

¹⁹ The Economist, 29-IV- 1995, pp. 113-114.

CAPITULO 5

IMPLEMENTACION EN

REDES DEL METODO

DEL CONJUNTO

ACTIVO

5. IMPLEMENTACION EN REDES DEL METODO DEL CONJUNTO ACTIVO.

El objetivo central de este capítulo es comprobar el comportamiento del método del conjunto activo como medio de asignación óptima en redes de tráfico con presencia de congestión y peajes. Como ya hemos comentado existen diversos estudios [42, 93, 97, 171, 196] que proponen la utilización de métodos basados en la idea del conjunto activo, o en la dirección del gradiente a la hora de establecer el nivel de tráfico óptimo en los arcos de una red. En este trabajo se plantea una propuesta similar, pero con una serie de criterios diferentes a la hora de establecer las características de la red que nos facilitará de forma sustancial el empleo del presente método.

5.1. Optimización de la Red.

Consideramos una red de tráfico formada por un conjunto de arcos que interconectan varios nodos. Dentro del conjunto de nodos podemos distinguir tres tipos:

a) Nodos origen: Son aquéllos de donde parten los usuarios para dirigirse a un destino. Presenta al menos dos arcos de salida, y pueden o no presentar arcos de entrada. Se caracterizan por ser inicio de dos o más caminos hacia el nodo destino para un conjunto de usuarios que se toma como dado y que se suele representar con la letra q^{xy} , donde "x" sería el nodo origen e "y" el nodo destino. Son los únicos nodos de la red que pueden generar tráfico.

b) Nodos destino: Serán los nodos hacia donde se dirige el flujo de tráfico que existe en la red. Tendrán dos o más arcos de entrada. Se caracterizan por ser el final de dos o más caminos procedentes de un mismo nodo origen. Son receptores de tráfico.

c) Nodos intermedios: Se ubican dentro de la red entre los nodos origen y destino. Sobre ellos deberán incidir al menos tres arcos. Actúan como distribuidores del tráfico existente, posibilitando la existencia de diversos caminos alternativos entre los nodos origen y los nodos destino.

Este conjunto de nodos figura interconectado por un conjunto de arcos, a través de los cuales circulan los usuarios de la red, constituyendo el flujo de la misma. Este flujo a través de la red será la variable de nuestro problema de optimización.

5.1.1. Concepto de Flujo.

El flujo, que como ya hemos comentado será la variable objetivo de nuestro problema de asignación, representa a los usuarios que atraviesan la red con sus automóviles para dirigirse desde los distintos orígenes a los diferentes destinos existentes. El concepto de flujo que empleamos en nuestro trabajo es el que se ha venido utilizando en el análisis de redes de transporte; según dicho enfoque consideramos el flujo como una medida instantánea, de forma tal que el flujo que entra en un arco es igual al flujo que sale del mismo, no existiendo valoración temporal alguna. Aun siendo conscientes de la existencia de diversas críticas contra este planteamiento [139]; consideraremos el flujo x_i como el total de usuarios que circulan a través de un arco i , esto es, el total de viajes que entre un origen y un destino que utilizan dicho enlace.

A la hora de hablar de flujo en una red podemos considerar dos visiones: el flujo de vehículos en un arco, o bien, el flujo de vehículos en una ruta. Arco es simplemente un nexo directo entre dos nodos, por lo tanto, el flujo que sale del nodo cabecera del arco será el mismo que el que llega al nodo objetivo de dicho arco; este principio se conoce como *principio de conservación del flujo de arco*. Ruta es un nexo entre un nodo origen y un nodo destino, y puede estar formada por uno o más arcos, y uno o más nodos intermedios¹. Las rutas también deben atenerse al principio de conservación del flujo, esto es, el flujo que entra en una ruta a través del nodo origen será el mismo que alcance el nodo destino a través de dicha ruta. Un arco puede formar parte de varias rutas.

En base al principio de conservación del flujo podemos enunciar la siguiente proposición:

Proposición 1: El flujo de un arco será igual a la suma de los flujos de todas la rutas que utilicen dicho arco.

Sea f_j^{AB} el flujo que circula entre los nodos origen y destino A y B, respectivamente, a través de la J-ésima ruta. Entonces dado el principio de conservación del flujo, podemos escribir el flujo a través del i-ésimo arco de la ruta J-ésima ($i \in J$), donde J es el conjunto de índices de los arcos que componen dicha ruta, como:

$$x_i = \begin{cases} \sum_J f_j^{AB} & \text{si solo existe un origen y un destino} \\ \sum_A \sum_B \sum_J f_j^{AB} & \text{si existen multiples origenes y destinos} \end{cases}$$

Por lo tanto, existe una relación entre el flujo de ruta y el flujo de arco, permitiéndonos expresar el problema en cualquiera de las dos variables.

¹ En el caso de no existir nodos intermedios solo existe un arco y por lo tanto, no existirá diferencia entre arco y ruta.

Por simplicidad, suponderemos que todo el flujo que existe en la red se dirige de un nodo origen a un nodo destino; esto es, los nodos intermedios no tienen capacidad de crear o absorber flujo, son meros distribuidores. Asimismo, no consideramos la existencia de flujo circular en la red, puesto que dado que estamos trabajando con el coste de la ruta como función objetivo, y dado que el coste asociado a cada arco será positivo o cero, no será posible introducir reducciones en el coste de una ruta a través de movimientos circulares, de ahí que podamos prescindir de dichos movimientos a la hora de estudiar la asignación óptima.

Cuando realizamos el análisis de la red resulta fundamental la elección de la variable objetivo, bien el flujo de ruta o bien el flujo de arco. En general, en redes grandes (de n nodos), el número de arcos será probablemente un múltiplo del número de nodos (por ejemplo: $3n$), mientras que el número de rutas será igual a c^n , donde c suele ser un número próximo a 3. Así pues, en una red con 18 nodos, tendremos un total de aproximadamente 54 arcos, y frente a ellos habrá cerca de 3^{18} rutas; en general, el número de rutas suele ser muy superior al número de arcos, de ahí que en la mayoría de los problemas optemos por la utilización como variable del flujo de arco². Para ello sólo tendremos que utilizar la transformación planteada en la Proposición 1. Ahora bien, hay que tener en cuenta que el principio de conservación del flujo deberá cumplirse de igual forma una vez establecidas las variables de flujo de arco, lo que conlleva una formulación matemática más compleja.

5.1.1.1. Un sólo origen y sólo destino.

Si sólo existen un nodo origen y un nodo destino, y no existen nodos intermedios, sólo existirá una variable q^{AB} que recoja la demanda de tráfico entre el nodo origen A y el nodo destino B. Dado el flujo de ruta, la suma de flujos a través de cada ruta ha de ser igual a q^{AB} :

² Cowan & Fine [35] aportan un interesante estudio sobre el número de enlaces a establecer en un grafo que represente una realidad social.

$$\sum_J f_J^{AB} = q^{AB}$$

Para cada arco, existirá un conjunto de rutas que utilicen dicho arco, de forma que sumando el conjunto de arcos tales que:

$$\sum_i x_i = q^{AB}$$

constituirán la única restricción que acotará a la función objetivo de nuestro problema de optimización. La transformación de flujo de ruta a flujo de arco es directa puesto que cada arco constituye una ruta.

En el caso de que existan nodos intermedios, y por consiguiente arcos que los interconecten, la determinación de las restricciones se vuelve más compleja y deberá realizarse de acuerdo con lo que se establece en el apartado siguiente.

5.1.1.2. Múltiples orígenes y destinos.

Si consideramos como variable objetivo el flujo de ruta, existirán un conjunto de ecuaciones, a diferencia del caso anterior donde sólo había una ecuación; esto es, una por cada par de nodos origen-destino que podemos establecer. Cada una de las mismas se expresará a través de la suma del conjunto de flujos de ruta existentes:

$$\sum_J f_J^{AB} = q^{AB} \quad \forall A, B / q^{AB} > 0$$

Vemos, por tanto, que el conjunto de trabajo se incrementa en tamaño, ahora estará constituido por un mayor número de filas y, por tanto, aumentará la dimensión de las matrices utilizadas en la resolución del problema.

Ahora bien, si trabajamos como antes con el flujo de arco, habrá que tener en cuenta que un mismo arco no sólo formará parte de diferentes rutas, sino que además estas rutas pueden pertenecer a distintos pares origen-destino, lo que nos imposibilita la construcción de una sola restricción constituida por la suma de arcos de la red. Los mecanismos de construcción de las restricciones del problema deberán tener en consideración no tanto el flujo de red, sino el principio de conservación del flujo de arco, y el *principio de conservación del flujo de nodos intermedios* que se puede enunciar de la forma siguiente:

Proposición 2: El flujo de los arcos de entrada a un nodo intermedio de la red de transporte ha de ser igual al flujo de los arcos de salida de dicho nodo.

$$\sum_{i \in E(N)} x_i = \sum_{i \in S(N)} x_i$$

donde $E(N)$ es el conjunto de arcos que inciden en el nodo N , y $S(N)$ es el conjunto de arcos que parten del nodo N .

Cuando los arcos inciden en nodos destino o parten de nodos de entrada entonces la suma de flujo de los mismos ha de ser igual a la demanda generada por el nodo destino o a la oferta conjunta generada por los nodos origen.

Proposición 3: El flujo de arcos de entrada a un nodo destino de una red de transporte ha de ser igual a la suma de usuarios que demanda ese nodo como destino de sus viajes.

$$\sum_{i \in E(B)} x_i = q^B$$

donde $E(B)$ es el conjunto de arcos que entran en el nodo destino B , y q^B es la demanda total de usuarios que desean desplazarse a B .

Proposición 4: El flujo de arcos de salida de un nodo origen de una red de transporte ha de ser igual a la suma de usuarios que oferta ese nodo como origen de sus viajes.

$$\sum_{i \in S(A)} x_i = q^A$$

donde $S(A)$ es el conjunto de arcos que parten del nodo origen A , y q^A es la oferta total de usuarios que desean desplazarse desde A hacia los diferentes destinos factibles.

En general, tendremos una ecuación por cada nodo de la red, si estamos trabajando con flujos de arco, y una por cada ruta, si estamos considerando como variable objetivo el flujo de ruta. Por tanto, para redes con pocos arcos por nodo, el número de ecuaciones generadas por ambas variables será muy similar. En redes más densas el número de ecuaciones generadas por la variable de flujo de arco será sustancialmente inferior al generado por la variable de flujo de ruta. Este hecho se explica porque en este tipo de redes el número de rutas en las que participa cada arco es mucho mayor que en aquellas redes menos densas.

5.1.1.3. Eliminación de Arcos.

En determinadas redes podremos proceder a la eliminación de algunos flujos de arco como variables del problema a través de la sustitución de dichos flujos por la suma de otros. Así en el caso de que:

$$x_j = \sum_i x_i$$

la variable x_j se sustituirá en el problema por la suma de las x_i variables pertinentes. Estas sustituciones vienen propiciadas por el principio de conservación del flujo y no

sólo afecta a las restricciones, sino que asimismo, deberán tenerse en cuenta a la hora de establecer la función objetivo.

Podemos distinguir dos tipos de situaciones en las cuales es conveniente realizar la sustitución de variables, lo que se traduce a efectos prácticos, en una reducción del número de variables que constituyen la red.

a) Un sólo arco de entrada y un único arco de salida.

En este caso nos encontramos ante la situación más simple; tenemos un nodo donde incide un sólo arco y del que también sale un único arco de salida. Desde el punto de vista práctico, dado el principio de conservación del flujo, el flujo en el arco de entrada (x_i) tendrá que ser el mismo que el del arco de salida (x_j), $x_i = x_j$. En esta situación procedemos a sustituir la variable x_j por la variable x_i , en el conjunto de la red.

b) Varios arcos de entrada y un único arco de salida.

En este tipo de interconexión el flujo que circula a través de arco de salida (x_j) será la suma del flujo que circula por todos los arcos de entrada:

$$x_j = \sum_{m \in E(N)} x_m$$

donde $E(N)$ es el conjunto de arcos de entrada del nodo N .

Esta situación supone, además, que la variable x_j de la red se puede sustituir lo que reduce el número de variables del problema en estudio.

Por último, en el caso de que un arco tenga asignado un flujo de arco igual a cero, equivale a que dicho arco no forme parte de la red, y por tanto, se puede

considerar el estudio de la misma eliminando el arco, y asimismo, eliminando todos aquellos caminos de los que éste forme parte.

5.1.2. Problema de Optimización.

Desde un punto de vista económico existen dos enfoques para aproximarnos a una red con varios orígenes y varios destinos. En un primer enfoque podemos considerar el problema de minimizar el coste total de todos los viajes dadas las restricciones de conservación del flujo. Este problema se conoce en la literatura como “problema del transporte”, y ha sido ampliamente estudiado en diversos trabajos de investigación operativa y matemáticas; económicamente es equivalente al caso de una empresa multiplanta que desea enviar su producción a un conjunto de mercados. Dado que el producto fabricado es homogéneo, la solución consistirá en enviar la máxima cantidad posible del mismo hacia los mercados más cercanos a cada una de las diferentes plantas productivas.

Una segunda aproximación considera el esquema planteado habitualmente por los planificadores del tráfico. Aquí el objetivo es minimizar el coste total de funcionamiento de la red, dadas las restricciones de conservación del flujo. La visión economicista de este problema es distinta a la anterior y equivale a una empresa multiplanta que produce productos diferentes en cada una de las distintas plantas de producción y que su destino es un conjunto de mercados espacialmente distribuidos. En este caso todas las plantas tienen que enviar parte de su producción a los diferentes mercados en los cuales exista demanda. Ya no es aplicable, por tanto, el criterio de proximidad entre oferente y demandante; sino que ahora el objetivo será canalizar los envíos a través de la red al menor coste posible.

Nuestra aproximación recogerá sólo aquellos casos en los que la red asigna todo el tráfico existente a los diferentes arcos de forma tal que todas las restricciones

tienen signo de igualdad; en este sentido hay que señalar que siempre formarán parte del conjunto de trabajo o conjunto activo.

Podemos considerar el problema de optimización de la red como:

$$\begin{aligned} & \text{Min}_{x \in \mathbb{R}^n} f(x) \\ & \text{sujeto a:} \\ & Ax = q \\ & x_i \geq 0, \quad \forall i = 1, \dots, n \end{aligned}$$

Algunos autores³ consideran además la existencia de posibles cotas laterales a los valores de x ; esto es, valores máximos y mínimos para las variables de flujo de arco. En nuestro caso existirá una cota inferior para todos los arcos, y vendrá dada por las condiciones de no negatividad del problema considerado; en cuanto a la cota superior de flujo de arco, nosotros no establecemos un valor máximo, en este sentido la red admite cualquier demanda, pero a diferencia de otros autores consideramos una demanda de usuarios constante, lo que supone que la cota máxima de cada arco se autocontrolará a través del coste de congestión.

5.1.3. Restricciones del Problema.

Las restricciones que acotan el problema considerado son, como acabamos de ver, de dos tipos. Por un lado, las restricciones de no negatividad de los flujos; y por otro, están las restricciones de igualdad, que son funciones lineales de las variables consideradas en el problema.

Dada una red cualquiera, para calcular este conjunto de restricciones habrá que tener en cuenta los principios establecidos en las proposiciones 2 a 4. De forma tal

³ Toint & Tuytens [171] y Narbona en [97].

que todas las restricciones sean función del flujo de arco. En función de lo que hemos comentado en los apartados anteriores podemos establecer que el número de restricciones de igualdad en un problema de este tipo queda determinado por el número de nodos de la red, existiendo una restricción de igualdad a cero por cada nodo intermedio, y una de igualdad diferente de cero por cada de nodo origen o destino que se establezca. Como acabamos de señalar, el principio de conservación del flujo supone que a la hora de establecer las restricciones se ha de considerar el número de arcos que se pueden sustituir como suma de otros de la red; de forma tal que se garantice que el flujo que circula a través de los mismos será siempre igual al del arco suma.

No todos los flujos de arco existentes en la red integrarán las restricciones del problema. Habrá que considerar las posibles relaciones que se establezcan entre los mismos, de forma tal que se proceda a las sustituciones oportunas. Así, en el caso de que tengamos una sustitución podemos proceder a eliminar dicha variable de la restricción.

Proposición 5: Sea r la restricción formada por la suma de un conjunto de variables x_1, \dots, x_n , y sea $x_i = x_j$, entonces podemos escribir la nueva restricción r' como la suma de variables anteriores sustituyendo x_j por x_i . En el caso de que la variable x_j estuviese presente en varias restricciones procederíamos a su sustitución en todas ellas.

De igual modo procederemos cuando consideramos varios arcos de entrada y un arco de salida realizando la sustitución:

$$x_j = \sum_{m \in E(N)} x_m$$

En general, las restricciones estarán formadas por sumas de variables, un signo de igualdad y una constante positiva o cero. Las variables irán multiplicadas por 1 o -1, lo que supone que la matriz A , estará formada por 0, 1, o -1; siendo por tanto una

matriz poco densa, lo que facilita la factorización de la misma⁴. El número de filas de la matriz A será el número de restricciones y, por tanto, será equivalente al número de nodos del problema; en cuanto al número de columnas, vendrá determinado por el número de arcos o variables de flujo de arco que presente la red, una vez realizadas las sustituciones oportunas. En general A será una matriz de orden $m \times n$, donde m es el número de nodos y n es el número de arcos de la red. Para redes de tamaño grande, el número de arcos suele ser un múltiplo del número de nodos, así pues, el orden de la matriz A será $m \times (a \times m)$, donde "a" es una constante. El número de columnas será, habitualmente, "a" veces superior al número de filas; lo que nos garantiza la aplicabilidad del método del conjunto activo, puesto que siempre existirán columnas para determinar la dimensión del espacio nulo.

5.1.4. Función de Coste.

La función objetivo deberá ser una función que permita medir el coste asociado al funcionamiento de la red. Para ello tendremos que utilizar una función que nos mida el coste que soportan los usuarios por circular a través de los distintos arcos que componen la misma. En general, y de acuerdo con lo que ya hemos comentado en el capítulo 3, esta función de coste de arco deberá ser una función no lineal, continua, y creciente; dependiente directamente del flujo que circula por dicho arco (x_i) y de la capacidad técnica asignada al mismo (K_i).

De las diversas posibilidades de implementación de dicha función, una de las más frecuentes en la literatura es la establecida por el Bureau of Public Roads de Washington (E.U.A.) en 1964 y cuya formulación es:

$$c = tf(1 + a(x/K)^b)$$

⁴ Véase [111] para la implementación de un método de factorización QR de matrices no densas en MATLAB.

donde t_f es el tiempo necesario para atravesar el arco si no existiese ningún grado de congestión⁵, x es el flujo que circula a través del arco, K es la capacidad máxima asociada a dicho arco, y a y b son dos constantes con valores sugeridos de $a=0.15$ y $b=4$. Este tipo de función ha sido ampliamente utilizada y no suele presentar problemas a la hora de implementarla en los algoritmos de asignación habituales. Asimismo vemos que la función cumple las características deseadas:

a) Es una función no lineal y continua en todo \mathcal{R} , si $K \neq 0$.

b) Es una función creciente:

$$\frac{\partial c}{\partial x} = \frac{a b t_f x^{b-1}}{K^b} > 0$$

para $x \geq 0$, y dado que t_f y K son siempre valores positivos.

c) Es una función convexa:

$$\frac{\partial^2 c}{\partial x^2} = \frac{a(b-1)b t_f x^{b-2}}{K^b} > 0$$

para $x \geq 0$.

Una de las principales críticas establecidas con respecto a la función BPR es que no es asintótica, esto es, proporciona valores para todos los flujos independientemente de que éstos superen la capacidad máxima del arco. Davidson [44] propone la utilización de una función alternativa, asintótica, que tienda a infinito según nos aproximemos a la capacidad máxima. La función de Davidson se formula como:

$$c = t_f (1 + J(x / K - x))$$

⁵ También denominado tiempo con tráfico libre.

donde J es un parámetro específico de cada arco y que representa el grado de curvatura de la función de coste, de forma tal que, a mayores valores de J menor longitud del ángulo de curvatura, y viceversa. Vemos, asimismo, que la función de Davidson⁶ no está definida para valores de $x=K$.

El análisis de la primera y segunda derivada muestran que la función es creciente y convexa, al igual que la función BPR. La primera derivada resulta positiva para valores positivos de J, K y t_f , y no está definida para $x=K$.

$$\frac{\partial c}{\partial x} = \frac{JKt_f}{(-K+x)^2} > 0$$

y la segunda derivada:

$$\frac{\partial^2 c}{\partial x^2} = \frac{2JKt_f}{(K-x)^3} > 0$$

si $K > x$.

Nosotros nos decantaremos por la utilización de la función BPR, puesto que permite la existencia de flujos superiores a la capacidad, y por tanto no implica que el algoritmo de optimización tenga que controlar en todo momento la factibilidad de la solución obtenida. En general, la utilización de la función de Davidson supone una mayor complejidad de gestión del algoritmo y no parece implicar ningún tipo de mejora de la precisión del mismo. De ahí que algunos autores aboguen por la utilización de la formulación BPR.

Una vez hemos construido la función de coste, existen dos posibles criterios de optimización:

⁶ Existen reformas posteriores de esta función que afectan a la zona asintótica de la misma Branston [26], Akcelik [2].

- 1) Minimizar el coste para cada usuario. Este es el criterio que se sigue en el algoritmo de determinación del camino más corto de una red.
- 2) Minimizar el coste total para todos los usuarios.

En un problema de optimización de redes, y suponiendo que el coste de arco es función del flujo de arco, se puede demostrar que ambos criterios se cumplen para la función objetivo adecuada⁷.

En el segundo criterio, que es el que emplearemos nosotros, el problema de optimización resultante es:

$$\text{Min}_{x \in \mathbb{R}^n} \text{CT} = \sum_i x_i c_i(x_i)$$

sujeto a:

$$Ax = q$$

$$x_i \geq 0, \quad \forall i = 1, \dots, n$$

Si calculamos :

$$\frac{\partial \text{CT}}{\partial x_i} = t f_i (1 + a(x_i / K_i)^b) + t f_i a b (x_i / K_i)^{b-1} \quad (1)$$

el primer término de (1) es el coste marginal que soporta un nuevo viajero que se incorpore al arco i de la red; mientras que el segundo término viene a representar el coste marginal que el nuevo viajero impone al resto de usuarios de la red, y evidentemente, a sí mismo.

En el óptimo todas las rutas (constituidas por un arco) tendrán el mismo coste, puesto que si en alguno de ellos el valor del coste fuese menor, se produciría un desplazamiento de usuarios hacia dicha ruta. Por tanto en el equilibrio, el coste de

⁷ Véase [139], pp. 137-138.

atravesar todos los arcos que conecten de forma directa el origen y el destino ha de ser el mismo. Podemos expresar esta condición de equilibrio como:

$$\begin{aligned} C^*_i(x_i / K_i) &= C^*, \quad \forall i, x_i > 0 \\ \forall i, x_i = 0 &\Rightarrow C^*_i(0) = C_i(0) \geq C^* \end{aligned} \quad (2)$$

Vemos que, asimismo, si el flujo de un arco es igual a cero, entonces es que el coste de atravesar dicho arco con flujo cero, esto es el valor de t_f , es superior al valor de coste de equilibrio C^* ; de ahí que el arco permanezca sin flujo en la solución óptima.

Se puede demostrar [139] que este criterio de optimización no es apropiado para la resolución del problema puesto que sólo nos produce una respuesta óptima para el segundo criterio de asignación que hemos comentado; por lo tanto, el valor óptimo minimiza el coste total de funcionamiento de la red, pero no supone que todos los viajeros sigan la ruta de coste mínimo. Beckman et al. [13] demuestran que empleando un función objetivo adecuada podemos responder de forma precisa a ambos criterios de asignación obteniendo un óptimo válido para ambos. Su aproximación se basa en el planteamiento del siguiente problema de optimización:

$$\text{Min}_{x \in \mathbb{R}^n} \text{CT} = \sum_i \int_0^{x_i} c_i(w) dw$$

sujeto a:

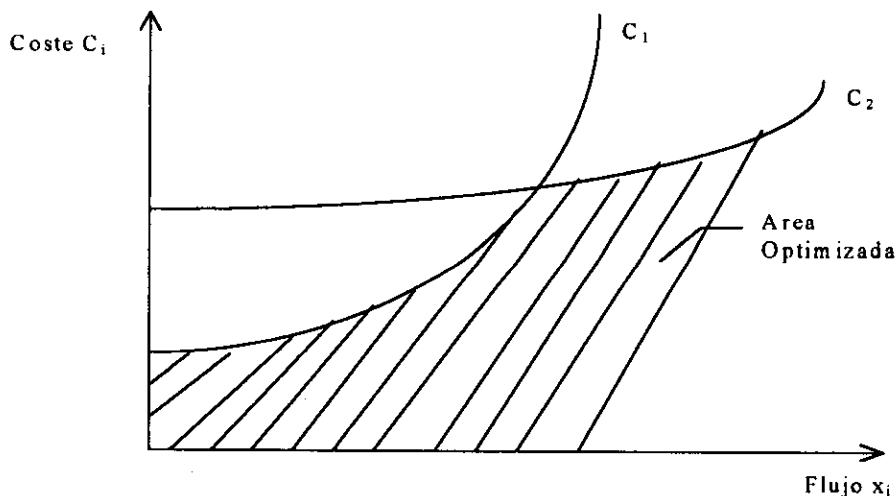
$$Ax = q$$

$$x_i \geq 0, \quad \forall i = 1, \dots, n$$

Si al igual que antes, calculamos la derivada de CT con respecto a x_i , obtendremos que ésta es igual C_i . Dado este resultado, la condición de equilibrio ya no es la expresada en (2), sino que se expresa como:

$$\begin{aligned} c_i(x_i) &= c && \text{en todas las rutas con } x_i > 0 \\ c_i(0) &\geq c && \text{en todas las rutas con } x_i = 0 \end{aligned} \quad (3)$$

Bajo esta condición los costes de todas las rutas utilizadas entre el origen y el destino serán iguales. En esta situación y para esta función objetivo, los dos criterios de asignación se cumplen; esto es, todos los usuarios seguirán la ruta de coste mínimo, y se minimiza el coste total de todos los viajeros. Desde un punto de vista gráfico nuestro objetivo consistirá en minimizar el área por debajo de la función de coste de atravesar cada arco:



La existencia de varios orígenes y varios destinos, junto a la presencia de varios arcos en común a varias, rutas supone una mayor complejidad analítica a la hora de plantear la función objetivo del problema de optimización.

Suponemos que nuestro objetivo, al igual que antes, tiene una doble finalidad:

- 1) Elegir la ruta de mínimo coste entre cada par Origen - Destino.
- 2) Minimizar el coste total asociado al flujo asignado a la red.

Ahora el coste asociado a atravesar cada arco dependerá del flujo que circula por una ruta que estará integrada por el coste de los arcos que formen parte de dicha ruta. La función objetivo de esta nueva aproximación resulta ser:

$$CT = \sum_i \int_0^{x_i} C_i(w) dw \quad (4)$$

siendo las restricciones del mismo:

$$\sum_J f_J^{AB} = q^{AB} \quad (5)$$

$$f_J^{AB} \geq 0 \quad (6)$$

donde J indica la ruta y el par AB el origen - destino.

Y la restricción que nos permite trasladar el flujo de ruta al flujo de arco:

$$x_i = \sum_A \sum_B \sum_J f_J^{AB} \delta_{i,J}^{AB} \quad (7)$$

donde δ es una variable cuyo valor es igual a 1, si el arco x_i forma parte de la ruta J-ésima que une A y B, e igual a cero en otro caso; tal y como ya hemos definido en el capítulo 3.

Si hemos realizado reducciones de arcos en la red, tendremos que tenerlas en cuenta a la hora de calcular la función de coste total CT, o función objetivo. Así, en el caso de que hayamos sustituido el flujo de dos arcos, $x_i = x_j$, la función objetivo resulta con la siguiente modificación:

$$\int_0^{x_i} C_i(w) + C_j(w) dw$$

Vemos que ahora ambos costes asociados a los dos arcos se integran respecto al valor de x_i .

Cuando hemos sustituido un arco por la suma del flujo que circula por varios, tendremos que realizar la suma del coste de atravesar dicho arco a todos los arcos anteriores al mismo:

$$\int_0^{x_m} C_m(w) + C_j(w) dw, \quad \forall m = i, \dots, l$$

Así pues, mediante la reducción del número de arcos propiciamos una disminución en el número de variables que facilita el tratamiento del problema, sin afectar al valor final de la función objetivo, puesto que éste se muestra invariante al considerar las transformaciones comentadas.

5.2. Aplicación del Método del Conjunto Activo.

Los algoritmos de asignación de redes han sido ampliamente estudiados en la literatura de investigación operativa. Se han descrito numerosos métodos para asignar de forma óptima el flujo de una red de tráfico, desde las aproximaciones más simples como puede ser el algoritmo del camino más corto formulado por Djisktra, hasta las versiones más sofisticadas implementadas en programas de ordenador, como es el programa SATURN del Instituto de Estudios del Transporte de la Universidad de Leeds, o los modelos de asignación del Departamento de Estadística e Investigación Operativa de la Universidad Politécnica de Cataluña⁸, u otras referencias disponibles

⁸ Puede obtenerse una visión sobre la situación actual de estos algoritmos en [165,169].

en [151, 197]. Nuestro objetivo en el presente trabajo se aproxima más a los objetivos modestos de algoritmos de asignación como los planteados por Frank-Wolfe o Sheffi [153] y se limitan a la aplicación de un algoritmo del conjunto activo sobre redes sencillas como método de asignación óptima de la red.

Desde el punto de vista informático hemos optado por implementar nuestro procedimientos mediante el lenguaje de programación del programa Mathematica (versión 2.1) construyendo una serie de funciones para el cálculo de los resultados del algoritmo. Todas estas funciones se han reunido en un "package" denominado "cactivo.m".

En el presente trabajo consideramos una red de tráfico donde las funciones de coste de los arcos son del tipo BPR, la demanda de flujo entre los orígenes y destinos está dada y es constante. No existen valoraciones temporales; es, por tanto, una aproximación puramente estática al problema. Tampoco hemos considerado la posible aproximación estocástica. En el óptimo, la red asignará todo el tráfico a los diversos arcos de forma tal que el coste global de funcionamiento de la red será el mínimo; asimismo, y siguiendo a [13], el coste de todas las rutas posibles será igual, de modo que ningún usuario estará dispuesto a modificar su ruta de forma individual puesto que no le supondrá una reducción el coste del viaje.

5.2.1. Descripción del Algoritmo del Conjunto Activo para Asignación en Redes.

Dado nuestro problema de optimización de la red como:

$$\text{Min}_{x \in \mathbb{R}^n} \text{CT} = \sum_i \int_0^{x_i} c_i(w) dw$$

sujeto a:

$$Ax = q$$

$$x_i \geq 0, \quad \forall i = 1, \dots, n$$

(8)

Tenemos una función objetivo no lineal y un conjunto de restricciones lineales de igualdad junto a las restricciones de no negatividad de las variables. Siguiendo los planteamientos de [63, 68], este tipo de problemas de optimización se pueden abordar mediante la utilización del método iterativo denominado del Conjunto Activo⁹. No obstante, el sistema habitual empleado en la resolución de este problema ha sido a través del método iterativo planteado por Frank - Wolfe en 1956, tal y como lo recogen diversos autores, veáanse [153, 169].

En nuestro modelo, la utilización del método del conjunto activo se beneficia de ciertas características del problema que facilitan de forma sustancial la implementación del mismo:

1. Dado que todas las restricciones son de igualdad, el conjunto de trabajo (o conjunto activo) estará formado por todas ellas en todo momento, no habiendo necesidad de considerar, en un principio, mecanismos de actualización del conjunto de trabajo, ni de las matrices auxiliares utilizadas en el desarrollo del algoritmo.
2. Emplearemos la factorización LQ de la matriz de restricciones A; a partir de la cual se calcula Z como matriz del espacio nulo de A; esto es, formada por las n-m últimas columnas de la matriz Q, donde n es el número de variables del problema (arcos) y m es el número de restricciones activas.
3. El número de restricciones de igualdad del problema, m, siempre será menor que el número de variables del mismo, n, dadas las características que hemos visto para la construcción de las mismas.

⁹ Aproximaciones al conjunto activo se pueden encontrar en [63, 68, 69, 133, 145, 163]

4. Nuestra aproximación emplea una línea de búsqueda quasi-newton, evitando de esta forma la necesidad de calcular la matriz Hessiana de CT.

5.2.1.1. Preliminares.

Dada la función objetivo $CT(x)$, donde x es un vector de \mathbb{R}^n , procedemos a calcular el gradiente de dicha función $g(x)=\nabla CT(x)$; y calculamos una matriz simétrica B , que será una aproximación a la Hessiana de $CT(x)$, $G=\nabla^2 CT(x)$. Inicialmente podemos comenzar trabajando con la función identidad de forma que $B=I_n$.

Asimismo procedemos a realizar la factorización de la matriz A , matriz de restricciones activas. El empleo de la factorización LQ, supone que A se puede escribir como producto de dos matrices: una matriz L , triangular inferior, y una matriz Q , matriz ortogonal.

El empleo del método del conjunto activo tal y como lo planteamos nosotros supone que el número de variables n , es mayor que el número de restricciones activas m . Dado que todas las restricciones que nosotros consideramos son restricciones de igualdad, esto supone que el número de restricciones del problema ha de ser inferior al número de variables, situación que tenemos garantizada.

Por tanto, dado que A es una matriz de orden $m \times n$, la factorización LQ, supone que Q es una matriz $n \times n$, y por tanto, L es una matriz $m \times m$ triangular inferior, produciéndose asimismo una matriz de ceros 0 , de orden $(n-m) \times m$. El resultado es:

$$A = (L \ 0)Q$$

A partir de la matriz Q , ortogonal de orden $n \times n$, procedemos a calcular la matriz Z , como matriz del espacio nulo de A , que estará formada por las $n-m$ últimas columnas de la matriz Q , siendo por tanto una matriz de orden $n \times (n-m)$.

5.2.1.2. Elección del punto inicial.

Tenemos que considerar una solución inicial x^0 , cuyo cálculo puede realizarse resolviendo el sistema de ecuaciones iniciales $Ax=q$, o bien mediante un método de asignación todo-o-nada que proceda a asignar todo el flujo entre dos nodos al arco con menor valor del coste libre t_f . Nuestra propuesta, no obstante, aborda una aproximación diferente.

Dado que vamos a utilizar como conjunto activo, aquél formado íntegramente por las restricciones de igualdad, esto supone que la utilización de un algoritmo de asignación todo-o-nada, no es factible, puesto que nos obliga a incorporar al conjunto activo todas aquellas restricciones de no negatividad correspondientes a las variables que hemos asignado un flujo igual a cero.

Para evitar esta situación emplearemos un sistema de asignación inicial que establezca un flujo distinto de cero para todos los arcos de la red. En principio, podemos además considerar una solución del tipo de la peor situación posible, esto es, asignando a todos los arcos un flujo igual a su nivel de capacidad K , si existe demanda suficiente, o bien mediante la asignación de las mayores cargas de flujo posible a aquellos arcos con mayor valor de t_f .

Este método de asignación empleado para el cálculo del punto inicial se basa en la idea de que la red está congestionada; esto es, existe una demanda de tráfico, q , suficiente como para poder asignar tráfico a todos los arcos de la red, de forma tal que en ningún momento nos encontramos en una situación donde el flujo de tráfico no

es suficiente para cargar toda la red. En el caso de que esto no fuera factible, y por lo tanto, no pudiésemos cargar de forma global todos los arcos de la red, el algoritmo del conjunto activo no plantea problemas, siempre y cuando consideremos las restricciones de no negatividad pertinentes dentro del conjunto de trabajo. No obstante, desde el punto de vista de nuestro estudio, creemos que estas situaciones no tienen cabida, puesto que nosotros suponemos de partida que la demanda de tráfico ha de ser suficiente como para producir la congestión de la red, de ahí la utilidad del algoritmo de asignación que estamos planteando. En redes no congestionadas, se impone, si cabe, el posible empleo de otros métodos de asignación ampliamente comentados en la literatura sobre este tema.

5.2.1.3. Cálculo de la dirección factible.

Una vez establecido un punto de inicio debemos dotarnos de una dirección de búsqueda factible para proceder a avanzar a lo largo la misma. Dado que nuestro problema es de minimización, ésta ha de ser descendente de forma tal que nos lleve a un menor valor de la función objetivo.

Supongamos que deseamos calcular una dirección factible p , y consideramos un modelo local de la función objetivo en x dado por la función cuadrática:

$$q(x + p) = CT(x) + g(x)' p + \frac{1}{2} p' Bp$$

entonces podemos plantearnos la dirección de búsqueda como un problema de minimización del tipo:

$$\text{Min}_p q(x + p)$$

sujeto a:

$$Ap = 0$$

considerando un cambio de variable $p=Zy$, y dado que $AZ=0$ entonces queda

$$q(x + Zy) = CT(x) + g(x)' Zy + \frac{1}{2} y' Z' B Z y$$

y el mínimo de esta función será:

$$Z' B Z y = -Z' g(x)$$

siendo por tanto la dirección de búsqueda y , igual a:

$$y = -Z' g(x) / Z' B Z$$

(9)

donde el numerador es el denominado gradiente reducido y el denominador es la aproximación de la Hessiana reducida.

En nuestra proposición de algoritmo emplearemos un método quasi-newton, por lo tanto, a la hora de trabajar con la Hessiana emplearemos una matriz de aproximación a la misma, B , en lugar del valor exacto de G . Esto nos evitará tener que calcular las segundas derivadas de la función objetivo. La matriz de aproximación B , que inicialmente se puede tomar como una matriz identidad de orden n , siendo n el número de variables, deberá ser en todo momento una matriz simétrica y definida positiva.

En esta situación, y si $Z' B Z$ es una matriz definida positiva, entonces la dirección obtenida a través de la resolución de (9) es una dirección de búsqueda descendente puesto que

$$g' Zy = -(Z' g)' (Z' B Z)^{-1} Z' g < 0$$

5.2.1.4. Longitud del paso.

Una vez determinada la dirección de búsqueda es necesario establecer una longitud de paso. Dado que trabajamos con restricciones activas en todo momento, la

longitud de paso deberá controlar que ninguna de las variables tome valores negativos, y por tanto, no se incumplan las condiciones de no negatividad.

Podemos entonces calcular la nueva dirección como:

$$x^{i+1} = x^i + \alpha p^i$$

donde α representa la longitud máxima de paso a dar, y p^i representa la dirección de búsqueda obtenida en el i -ésima iteración, siendo x^i la solución de la iteración anterior.

Para proceder a la estimación de α procederemos a estudiar las distintas componentes del vector de dirección p .

Supongamos que la i -ésima componente de dicho vector $p_i < 0$, entonces el valor máximo que podremos descender a lo largo de dicha dirección para la variables x_i vendrá dado por :

$$\alpha_i = \left| \frac{x_i^i}{p_i} \right|$$

de forma tal que en la nueva solución, la variable x_i se anularía, para dicho valor de α_i . En el caso de que esto sucediese habría que plantearse algunas modificaciones en el proceso iterativo del algoritmo.

Si la i -ésima componente de p es mayor que 0, entonces el valor de x_i se incrementará al desplazarnos a lo largo del vector dirección, y por lo tanto la máxima longitud del paso a dar deberá garantizar el cumplimiento de las restricciones que configuran el conjunto de trabajo, y con ello de las restricciones de igualdad del problema. Así, pues, el valor de α_i se determinará como el mínimo de:

$$\text{Min}_i \left\{ \left| \frac{x_i'}{p_i} \right|, 1 \right\}$$

para todas aquellas variables x_i , tales que $x_i > 0$.

Por último, en el caso de que $p_i = 0$, entonces la longitud de paso no aparece definida, y por tanto, no es necesario controlarla para la variable x_i .

Una política que permite establecer una cota genérica, considerando $p_i \neq 0$, para todas las variables de flujo implicadas en el problema cuyo valor sea estrictamente mayor que cero, viene dada por:

$$\alpha = \text{Min}_i \left\{ \left| \frac{x_i'}{p_i} \right|, 1 \right\} \quad (10)$$

De esta forma tenemos garantizado el cumplimiento de las condiciones de no negatividad en todo momento. No obstante, esta aproximación presenta el problema de que no siempre nos va a permitir dar el paso en su máxima longitud factible. Podemos entonces considerar una cota genérica más cuidadosa, que sólo consideraría aquellos casos donde el valor de p_i es negativo; siendo por tanto el valor del paso:

$$\alpha = \text{Min}_{\substack{i \\ p_i < 0}} \left\{ \left| \frac{x_i'}{p_i} \right|, 1 \right\} \quad (11)$$

En este caso, hay que notar que el algoritmo del conjunto activo tiene una mayor propensión a la realización de zig-zags, que en el caso de emplear la cota anterior, debido a que en este caso las variables tienden a anularse lo que implica la necesidad de introducir en el conjunto activo las restricciones de no negatividad. Este efecto de zig-zag supondrá la necesidad de realizar un mayor número de iteraciones para la obtención del resultado final.

Por último, en el caso de que optemos por trabajar con valores de flujo iguales a cero para alguno de los arcos, esto supone que algunas de las restricciones de no negatividad formarán parte del conjunto activo; entonces tendremos que tener en cuenta que aquellos valores con $x_i = 0$, suponen que no se puede calcular longitud máxima de paso en el caso de que $p_i < 0$. Si estamos utilizando como longitud de paso (10) entonces habrá que tener en cuenta que cuando $p_i > 0$, no se deberán tener en cuenta los valores de x_i / p_i para calcular el valor de α . Esta última advertencia no es necesaria si estamos utilizando (11) como fórmula de obtención de la longitud de paso.

5.2.1.5. Adaptación de la matriz B.

En el algoritmo del conjunto activo la primera iteración es diferente al resto, puesto que no requiere la actualización de la matriz de aproximación a la Hessiana, B, en el punto solución. En las siguientes iteraciones es necesario realizar de forma previa al cálculo de la dirección de búsqueda una actualización de dicha matriz, a partir de los valores de los gradientes y la solución obtenida.

El esquema seguido en este estudio es el establecido por Broyden-Fletcher-Goldfard-Shanno (BFGS) que propone el cálculo de la actualización de la matriz B en la i+1-ésima iteración como:

$$B^{i+1} = B^i - \left(\frac{1}{s^t B^i s} B^i s s^t B^i \right) + \frac{1}{p^t s} p p^t$$

donde $p = g^i - g^{i-1}$, donde g^i indica el gradiente evaluado en x^i , y $s = x^i - x^{i-1}$.

En nuestro estudio, hemos comprobado que en algunas ocasiones la realización de la primera actualización, esto es, el cálculo de B^1 a partir de $B^0 = I_n$,

resulta más eficaz desde el punto de vista de la aceleración de la convergencia si se realiza de acuerdo con la siguiente formulación.

$$B^1 = B^0 - \left(\frac{1}{x_1' B^0 x_1} B^0 x_1 x_1' B^0 \right) + \frac{1}{g_1'} g_1 g_1'$$

El empleo de esta formulación no supone ningún tipo de dificultad a la adaptación de la matriz B, siempre y cuando se mantenga la característica de ser una matriz definida positiva.

5.2.1.6. Restricciones de no negatividad.

Como ya hemos comentado, las restricciones de no negatividad no forman parte del conjunto activo puesto que procuramos que todas las variables de flujo sean positivas. No obstante, puede ocurrir que durante el desarrollo del algoritmo alguna de las variables se anule y, por tanto, habría que incluir la restricción de no negatividad correspondiente a dicha variable dentro del conjunto de trabajo, lo que afectará a la matriz Z, puesto que el espacio nulo se modifica. Ante esta situación es posible adoptar dos tipos medidas:

- a) Eliminar el arco de la red.

Dado que cuando un arco toma un flujo cero, partiendo de un flujo positivo, es indicativo de que dicho arco posiblemente aparezca con flujo cero en la solución final; su eliminación de la red no plantea mayor problema puesto que ambas situaciones son equivalentes.

En caso de optar por esta alternativa, deberemos detener nuestro algoritmo y volver a iniciarlo para la nueva red resultante con el arco eliminado. Esto supone que tengamos que volver a recalcular la función objetivo, y la matriz de restricciones

activas, eliminando la variable cuyo arco ha sido eliminado de la red. Bajo estas circunstancias, cada vez que se obtiene una variable $x_i = 0$ tenemos que volver a iniciar el problema de asignación de flujo desde el principio. Esto implica detener totalmente el algoritmo e inicializar el proceso.

b) Incorporación al Conjunto Activo.

La incorporación de una nueva restricción al conjunto activo supone la necesidad de realizar una adaptación del mismo, así como de la factorización de dicho conjunto y de la matriz del espacio nulo, Z , obtenida para el primer conjunto activo.

Para ello, debemos o bien calcular la nueva factorización de la nueva matriz del conjunto activo, o adaptar la factorización existente:

El proceso de adaptación se basa en calcular una matriz householder H que anule las $t+2$ componentes de L , donde t es el número de restricciones del conjunto activo antes de incorporar la nueva restricción. Así nos queda:

Sea A el conjunto activo anterior y a la nueva restricción:

$$\bar{A} = \begin{pmatrix} A \\ a \end{pmatrix} = \begin{pmatrix} L & 0 \\ \bar{a} & Q \end{pmatrix} Q = (\bar{L} \quad 0) \bar{Q}$$

donde $\bar{Q} = QH$.

La matriz del espacio nulo Z , vendría dada por las $n-m$ últimas columnas de \bar{Q} .

Al añadir restricciones de no negatividad al conjunto activo, puede suceder que $m=n$, y por tanto el espacio nulo sea 0. En esta situación no podemos utilizar el método del conjunto activo. Para evitar esta situación se recomienda que aquellas

restricciones de no negatividad que se cumplan con signo estricto de igualdad se eliminen de la red.

Una forma alternativa de actuar cuando $m=n$ consiste en calcular los valores de los multiplicadores de Lagrange λ , para aquellas restricciones que conforman el conjunto activo, y eliminar del mismo la que corresponda al menor valor de λ , tal que $\lambda < 0$. Si tenemos que eliminar una restricción de no negatividad del conjunto activo, esto es, alguna de las variables de flujo que se ha considerado nulas se le ha asignado algún valor de flujo y, por tanto, no debe permanecer en el conjunto activo, el proceso de adaptación de la matriz Z se vuelve un poco más complejo.

Cuando eliminamos la restricción i -ésima del conjunto activo tenemos que

$$AQ = (M \quad 0)$$

donde M es una matriz cuyas filas de 1 a $i-1$ son de la forma triangular inferior, y el resto tienen un elemento extra en la diagonal. En esta situación debemos realizar una serie de rotaciones planas a la derecha de Q , las cuales no afectarán a las últimas $n-m$ columnas de Q y, por tanto, la nueva matriz de espacio nulo será:

$$\hat{Z} = (Z \quad z)$$

donde Z es la antigua matriz Z y z es una combinación lineal de las m primeras columnas de Q .

5.2.1.7. Criterio de Optimalidad.

El algoritmo deberá finalizar en el momento en el que se haya alcanzado el punto óptimo. En esta situación la solución obtenida deberá cumplir las condiciones de óptimo:

a) Condiciones de 1º orden.

Las condiciones de primer orden para problemas de optimización de funciones no lineales con restricciones lineales son:

Si

$$Z'g(x^*) = 0 \quad (12)$$

donde $g(x^*)$ es el valor del gradiente de la función objetivo en el punto óptimo.

Entonces:

1. Si no existen restricciones activas, el punto x^* es un punto estacionario local.
2. Si existen restricciones activas, se calcula el valor de los multiplicadores de Lagrange en el punto óptimo, y si el resultado es mayor o igual que cero para todos ellos, entonces el punto x^* es un punto estacionario local:

$$\lambda^* \geq 0$$

En nuestro caso, podemos calcular los multiplicadores de Lagrange a través de a fórmula:

$$\lambda = (AA)^{-1} A.g(x)$$

donde A es el conjunto activo.

Si alguno de los valores de λ es negativo, entonces indica que $F(x)$ puede decrecer moviéndonos a lo largo de la dirección de la restricción cuyo multiplicador sea negativo. Esto supone que hemos considerado nula alguna de las variables cuya restricción se introducido en el conjunto activo, y por tanto, podemos mejorar la solución dando un valor a dicha variable.

En nuestro caso, esta situación es poco probable si hemos partido de un punto inicial donde todas las variables tenían asignado un valor de flujo positivo.

b) Condiciones de 2º orden.

La condición de segundo orden para que x^* sea el óptimo es que la matriz Hessiana proyectada sea definida positiva. En nuestro caso, como trabajamos con la estimación quasi-newton de dicha matriz esta condición supone que $Z^t B_k Z$ debe ser una matriz definida positiva. Podemos también calcular los estimadores de segundo orden de los multiplicadores de Lagrange a través de la siguiente fórmula:

$$\eta = (AA^t)^{-1} A(g(x^*) + B_k y_k)$$

En el óptimo, los dos estimadores de los multiplicadores de Lagrange coinciden y son positivos o iguales a cero.

5.2.1.8. Test de Parada.

El supuesto más beneficioso para nuestro algoritmo sería encontrar la solución óptima y detener el proceso iterativo. No obstante, en todo proceso algorítmico la sucesión de iteraciones converge hacia el óptimo, aunque no necesariamente se alcanza en un número relativamente pequeño de iteraciones. En estas circunstancias,

alcanzado determinado valor próximo al objetivo deseado, se procede a parar el algoritmo y se toma la solución alcanzada como óptima.

Dado que nuestro método es iterativo, resulta necesario dotarnos de un test de parada como control del momento en el que debemos detener del algoritmo. El test más habitual consiste en determinar un valor $\varepsilon > 0$ suficientemente pequeño y estudiar la convergencia entre soluciones sucesivas. En general si

$$\frac{\|x^{n+1} - x^n\|}{\|x^n\|} < \varepsilon$$

entonces debemos detener el algoritmo y consideramos que la última solución hallada es el valor óptimo.

Una segunda aproximación considera la variación de las distintas componentes del vector x , como medida del tes de convergencia:

$$\text{Max}_i \left\{ \frac{|x_i^{n+1} - x_i^n|}{x_i^n} \right\} < \varepsilon \quad (13)$$

Otra aproximación alternativa, consiste en controlar el avance de la solución y, si ésta no ha disminuido en una cantidad mayor que el valor de ε , entonces detenemos el algoritmo. Así, tenemos como posible test de convergencia de la solución:

$$\frac{F(x^{n+1}) - F(x^n)}{F(x^n)} < \varepsilon$$

Por último, existen otras situaciones en las cuales el algoritmo acaba puesto que no es posible continuar avanzando:

a) Si $g(x^{n+1}) = g(x^n)$ entonces no es posible calcular la nueva matriz auxiliar B, y por lo tanto no podemos calcular una dirección de búsqueda.

b) Si $p=0$, en este caso la dirección de búsqueda vale cero y, por lo tanto, tenemos dos soluciones sucesivas iguales.

c) Si $\alpha=0$, debemos parar el algoritmo puesto que al igual que en el caso anterior tenemos dos soluciones sucesivas iguales. La obtención de un valor negativo para longitud de paso se produce en el caso de obtener una dirección de búsqueda descendente para una variable cuyo valor sea cero.

En principio, el criterio más habitual considera la diferencia entre soluciones, de forma que a través del test de parada se establece un control de la convergencia de la solución con respecto a un valor que se considerará como óptimo. El control de convergencia de la solución puede presentar algunos problemas puesto que pueden existir efectos de zigzageo en los cuales la diferencia entre soluciones sucesivas sea negativa, indicativo de que el valor no ha disminuido sino que ha aumentado. Para evitarlo se deberá emplear el criterio utilizando el valor absoluto.

En este estudio consideramos que el mejor criterio consiste en controlar la optimalidad de la solución a través del criterio del gradiente proyectado:

$$Z'g(x') \rightarrow 0$$

Cuando el valor de este criterio disminuya por debajo de un $\epsilon > 0$, pequeño, entonces detendremos nuestro algoritmo. En cuanto al valor de la cota de convergencia, deberá tenerse en cuenta que debe ser un número no excesivamente bajo, para evitar un gran número de iteraciones, y al mismo tiempo, debe ser pequeño, para que no se produzca una detección del algoritmo en un punto lejano del valor de convergencia. Normalmente un buen valor de prueba puede ser $\epsilon=0.01$. Aunque en

todo caso, es un criterio que deberá quedar en función de la precisión deseada para el problema en estudio.

5.2.2. Implementación en Mathematica.

El algoritmo propuesto se ha implementado dentro de los procedimientos del programa de cálculo simbólico Mathematica, versión 2.1 (véase [106, 195]), para poder proceder de esta forma a la evaluación de la efectividad de los resultados obtenidos. Para ello, se han implementado una serie de funciones dentro del entorno de Mathematica que permiten obtener los resultados de los diversos pasos del algoritmo para las diferentes iteraciones. Todos estas funciones se han recogido en una paquete bajo el nombre genérico de "cactivo.m".

Las diferentes funciones que componen el paquete, así como su empleo y utilidad, se describen a continuación.

a) **FactorLQ[Matriz]**: Esta función calcula la factorización LQ de matriz, donde Matriz será una matriz $m \times n$; $m > n$. El resultado es la matriz ortogonal Q. La matriz L se puede calcular postmultiplicando la traspuesta de la matriz Q, por Matriz.

En nuestro proceso del algoritmo introduciremos la matriz de restricciones del conjunto activo, para obtener de esta forma la correspondiente matriz Q.

b) **FactorQR[matriz]**: Esta función calcula la factorización QR de matriz, donde matriz será una matriz $m \times n$; $m > n$. El resultado es la matriz ortogonal Q. La matriz R se puede calcular premultiplicando la traspuesta de la matriz Q por Matriz.

- c) Houseanula[Vector, Entero]: Calcula la matriz de Householder que anula los elementos de Vector por debajo de la posición Entero. Entero deberá ser estrictamente menor que el número de componentes de Vector. La matriz de Householder resultante será cuadrada del orden de Vector. Premultiplicando la matriz obtenida por Vector, obtendremos el vector con sus componentes por debajo de Entero con valor cero.
- d) EspacioNulo[Matriz, Entero1,Entero2]: Esta función se utiliza para determinar la matriz del espacio nulo de restricciones, recogido en Matriz, para Entero1= número de restricciones activas del problema, y para Entero2 = número de variables del problema.

En Matriz deberemos introducir la matriz Q resultante de la factorización LQ de la matriz del conjunto activo de restricciones. El resultado será una matriz formada por las n-m últimas columnas de la matriz Q, donde n será el número de variables del problema, y m el número de restricciones activas.

- e) Gradiente[Expresión, Vector]: Mediante esta función se calcula el gradiente de la Expresión, donde Expresión es una función de n variables ($n > 0$), y Vector es el vector de n variables que determinan la función Expresión. El resultado será un vector que contiene el gradiente de Expresión.

En Expresión introduciremos la función objetivo del problema, mientras que Vector será un vector que contiene todas las variables que intervienen en el mismo.

- f) Evaluador[Expresión, Vector1,Vector2]: Sirve para calcular el valor de la expresión Expresión en un punto que se introducirá en Vector2. En Vector1 se introducirá un vector con las variables de las que depende Expresión. Ambos vectores tendrán que ser necesariamente de igual orden para que el funcionamiento del procedimiento sea correcto. El resultado obtenido será un vector que contenga

el valor de la expresión en el punto solicitado. Si Expresión es un vector, el resultado será una matriz de orden $m \times n$, donde n es el número de variables del vector.

- g) BFGS[Matriz, Vector1, Vector2]: Esta función sirve para obtener la actualización de Broyden-Fletcher-Goldfard-Shanno de Matriz.

-Para su correcto funcionamiento se deberá introducir en Matriz, el valor de la Hessiana aproximada en la iteración anterior; en Vector1, la diferencia entre las soluciones obtenidas en las dos iteraciones inmediatamente anteriores a la actual; en Vector 2, la diferencia entre los gradientes evaluados en la dos soluciones introducidas en Vector 1. La función admite que se introduzca las diferencias de forma expresa o bien operativa.

Se puede comprobar que la matriz B actualizada es correcta calculando los autovalores de la misma mediante la orden:

Eigenvalues[Matriz], donde Matriz es la matriz B resultante.

Si todos ellos son positivos, esto indica que la matriz B es definida positiva y por tanto la actualización realizada es correcta.

- h) DirBus[Matriz1, Matriz2, Vector]: Mediante esta función se calcula la dirección de búsqueda de la solución siguiente, utilizando para ello la matriz del espacio nulo Z, la matriz de estimación de la Hessiana, obtenida mediante el método quasi-newton, y el vector gradiente evaluado en la solución anterior. Como resultado obtendremos un vector que representará el valor de la dirección de búsqueda factible.

En Matriz1, se introducirá la matriz del espacio nulo Z , obtenida a través de la función EspacioNulo. Matriz2 recogerá la matriz de estimación de la Hessiana B en la iteración actual, resultante de la ejecución del procedimiento BFGS con los datos anteriores. Por último, en Vector se deberá proporcionar el gradiente evaluado con la solución obtenida en la iteración anterior.

- i) Cota[Vector1, Vector2]: La función cota se utiliza para calcular la longitud máxima del paso a dar, a partir de la solución recogida en Vector2, y siguiendo la dirección de búsqueda establecida en Vector1.

En Vector1, introduciremos la dirección de búsqueda obtenida con DirBus para la presente iteración, y en Vector2, se deberá suministrar a la función la solución obtenida en la iteración anterior.

Esta función no se podrá utilizar en dos casos: bien sea en el caso en el que alguna de las variables tiene un valor cero, o bien si alguno de los componentes del vector de dirección de búsqueda es cero.

- j) IteracionCA[Vector1, Vector2, Vector3, Vector4, Matriz]: Esta función sirve para calcular la solución de la iteración siguiente, a partir de la segunda. El resultado será la nueva solución obtenida mediante el empleo del método del conjunto activo.

- Vector1 será el vector de variables del problema.
- Vector2 recoge el vector gradiente de la función objetivo.
- Vector3 y Vector4 son las soluciones obtenidas en la iteración $i-1$ y la iteración $i-2$.
- Matriz es la matriz de aproximación de la Hessiana obtenida en la iteración anterior, bien de forma directa a través de la formulación BFGS, o bien

utilizando la matriz B, que recoge el valor de dicha aproximación cuando se utiliza el procedimiento IteracionCA para el cálculo de la solución actual.

k) TestCo[Vector1,Vector2]: Realiza el test de parada para los valores de dos soluciones sucesivas que se introducirán en Vector1, el valor de la función objetivo para la solución obtenida en la iteración actual, y en Vector2, el valor de la misma para la solución obtenida en la iteración anterior.

El resultado será un valor que compararemos con el valor de control ϵ con que nos hayamos dotado, para establecer si debemos continuar o no con el algoritmo. Hay que tener en cuenta que el resultado aparece multiplicado por cien.

Este test no se podrá aplicar de forma directa cuando alguna de las componentes del vector solución anterior sea cero.

Todas estas funciones se encuentran disponibles dentro del package "cactivo.m". Para calcular la solución de las diferentes iteraciones del algoritmo deberemos proceder de la siguiente forma:

a) Preliminares.

- Construir un vector con las variables del problema.
- Calcular la función objetivo, a partir de las funciones de costes, y el gradiente de dicha función.
- Establecer la matriz de restricciones del conjunto activo. Obtener la factorización LQ, y la matriz del espacio nulo.

- Construir la primera aproximación de la Hessiana como la matriz Identidad de orden número de variables.
- Determinar una solución inicial.

b) Iteración 1:

Calcular la dirección de búsqueda y la longitud máxima del paso.
Calcular la nueva solución y evaluar la función objetivo en dicha solución.

c) Iteraciones 2 y sucesivas:

En estas iteraciones podemos emplear dos sistemas de obtención de la nueva solución:

1. Calcular el gradiente en la solución anterior. Actualizar el valor de la matriz aproximada Hessiana B, calcular la dirección de búsqueda, la longitud máxima del paso, y el valor de la nueva solución. Evaluar la función objetivo en dicha solución y ejecutar el test de convergencia.

2. Utilizando la función IteraciónCA, obtenemos una nueva solución. A partir de la misma podremos evaluar la función objetivo en base al valor obtenido y ejecutar el test de convergencia.

En el caso de que alguna de las soluciones contenga una componente igual a cero, entonces si mantenemos dicho arco en la red, habrá que realizar una adaptación de la matriz del Espacio Nulo. Esta adaptación se puede calcular a través de la función:

AdaptaEN[Matriz1, Matriz2, Matriz3, Entero1, Entero2]: A través de este procedimiento se obtiene la nueva matriz del espacio nulo al añadir una restricción al conjunto activo:

- Matriz1 es la nueva restricción que deseamos incorporar al conjunto de trabajo.
- Matriz2 es el conjunto de trabajo previo.
- Matriz3 es la matriz Q de la factorización LQ del conjunto de trabajo anterior¹⁰.
- Entero1 es el número de restricciones que contiene el conjunto de trabajo después de añadir la nueva restricción.
- Entero2 es el número de variables.

Para poder calcular la nueva matriz del espacio nulo de la matriz de restricciones activas será obligatorio que $\text{Entero1} < \text{Entero2}$.

La opción de eliminar el arco de la red obliga a replantear el problema sin dicho arco y comenzar el algoritmos desde la fase inicial.

Hay que señalar que si estamos trabajando con vectores solución con componentes igual a cero, entonces la función IteracionCA no se puede emplear como método de cálculo de la solución siguiente. Es necesario utilizar el procedimiento del tipo 1, para obtener dicha solución. Este hecho se debe a que la forma de implementación de la máxima longitud de paso se ha realizado para aquellos casos en los cuales las componentes del vector solución son positivas distintas de cero. Así pues, si alguna componente es negativa es necesario calcular la longitud máxima de paso de forma manual.

Para ello podemos emplear la siguiente orden en Mathematica:

¹⁰ Cuando realizamos más de una adaptación entonces la matriz Q no está expresamente disponible; no obstante, se puede obtener fácilmente porque cuando se ejecuta el procedimiento AdaptaEN se obtiene como resultado no explícito dicha matriz que se almacena bajo el nombre de QA.

Abs[Vector1/Vector2].

Donde Vector1, es el vector que contiene la solución hallada en la iteración actual, y Vector 2 es la dirección de búsqueda utilizada para obtener dicha solución.

El resultado de esta orden es un vector que contiene todos los posibles valores de la longitud de paso. Se escogerá el mínimo de todos ellos mayor que cero, y si éste fuera mayor que 1, se tomaría este último valor como longitud máxima de paso.

Para ver la optimalidad de la solución obtenida, podemos calcular el gradiente reducido y comprobar el valor de los multiplicadores de Lagrange correspondientes a las restricciones que conforman el conjunto activo. Estas serán todas las restricciones de igualdad del problema. Para calcular el gradiente reducido (criterio de optimalidad):

Transpose[Matriz] . Evaluador[Vector1, Vector2, Vector3], donde:

- Matriz, es la matriz del espacio nulo.
- Vector1 es el vector gradiente.
- Vector2 es el vector de variables.
- Vector3 es el vector con la solución optima.

En el óptimo el resultado de esta operación será un vector cuyas componentes sean nulas o tiendan a cero.

Para calcular los multiplicadores de Lagrange:

Inverse[Transpose[Matriz] . Matriz] . Matriz] . Evaluador[Vector1, Vector2, Vector3], donde Matriz es la matriz del conjunto activo y los vectores son los mismos que se establecieron en el caso anterior.

Por último, el diagrama de flujo resultante es el siguiente:

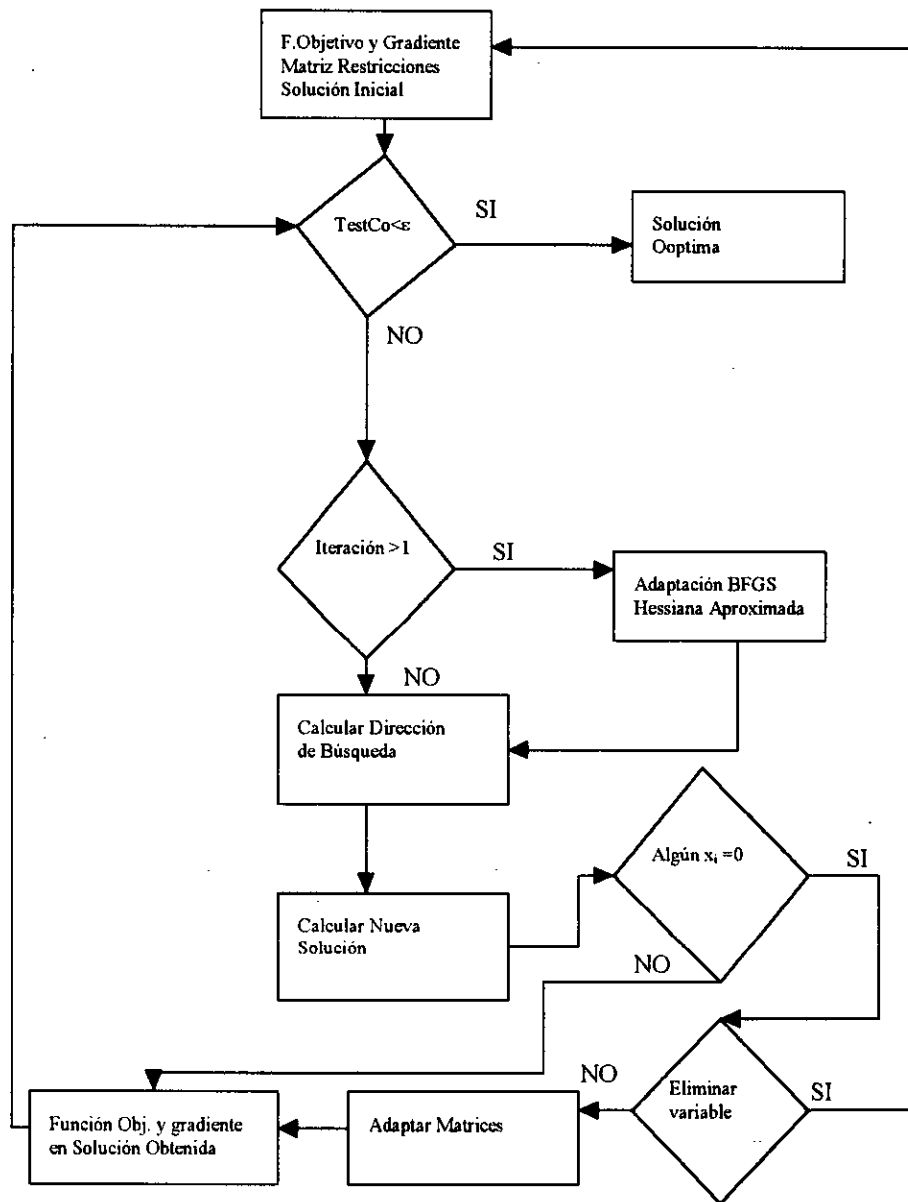


Ilustración 1 : Algoritmo del Conjunto Activo.

5.3. Cálculo del Peaje Optimo.

El peaje se considera en la actualidad como una de las fórmulas más eficaces a la hora de evitar los efectos negativos propiciados por la congestión de las carreteras y vías urbanas. Su implementación en numerosos países ha sido ampliamente estudiada y tratada en la literatura económica sobre el tema. Desde este punto de vista, en este apartado nos centraremos en la obtención de un peaje óptimo asociado a cada arco de la red.

En general, podemos hablar de dos tipos de peajes sobre redes:

- *Peaje de arco:* Este se establecería sobre cada usuario al atravesar un arco de la red, modificando de esta forma el coste asociado a la utilización de dicho arco. Cada ruta tendría asociado un peaje, que vendría dado por la suma de los peajes de los arcos que componen dicha ruta. En esta situación todos los usuarios de un arco serán cargados con el mismo peaje por la utilización del mismo.
- *Peaje de ruta:* Este tipo de peaje se asocia a la utilización de una ruta y afecta por tanto, al conjunto de arcos que configuran la misma. Así pues, los usuarios que utilicen un mismo arco percibirán diferentes importes del peaje según la ruta que sigan.

Desde el punto de vista del conjunto del sistema, el coste global asociado a la utilización de la red se verá modificado puesto que incorporará el conjunto de la suma obtenida por la recaudación de los peajes. En este sentido, los usuarios percibirán la utilización de la red como gravada por un mayor sobrecoste que en el caso de no existir peajes y, por consiguiente, el equilibrio de optimización de usuarios se produce a un nivel de función objetivo superior al análisis sin peajes. Por su parte, el equilibrio de optimización del sistema no varía, puesto que los peajes no deben ser considerados como un mayor sobrecoste del viaje tal y como lo percibe la sociedad en su conjunto,

dado que el peaje no es un dinero perdido, sino que puede ser utilizado de forma beneficiosa por la sociedad. En este sentido, el papel del Estado se vuelve fundamental a la hora de aplicar el peaje recibido de forma que revierta en el beneficio social.

En este estudio nos centraremos en el establecimiento de un peaje asociado a cada arco, de forma tal que el coste total percibido por los usuarios que utilizan la red sea minimizado. Consideraremos que el peaje supone un mayor coste del viaje y que afecta al patrón de distribución de tráfico de la red en su conjunto de modo que el valor final de la función objetivo será superior al valor de la función objetivo en el caso de la no existencia de peajes. Nuestro objetivo será minimizar la diferencia entre ambas situaciones una vez descontado el importe de los peajes.

Dado que en esta aproximación hemos considerado que la demanda de usuarios es fija, el establecimiento del peaje no supondrá una variación de la misma y, por lo tanto, la red tendrá que prestar su servicio al mismo número de usuarios que en el caso de no establecer peajes. Otros autores siguen una aproximación alternativa y consideran una red con funciones de demanda¹¹ de forma tal que el establecimiento de un peaje determina patrones de demanda diferentes pero no afecta al cálculo del equilibrio de la red.

5.3.1. Aproximación Tradicional al Peaje Óptimo.

Supongamos que deseamos calcular el peaje óptimo asociado a la utilización de un arco de la red dada nuestra función de coste de arco tipo BPR. En este caso, el coste total asociado a dicho arco será igual al número de usuarios que lo utilizan por el coste individual que soporta cada uno de ellos:

¹¹ Existen diversos estudios que consideran una función de demanda entre otros podemos citar: Lave [105], McFadden [115], Hensher & Johnson [80], McCarthy [112], Kim [100], McConell [113], Hensher et al. [81], Matas [110].

$$CT = xf(1 + a(x/K)^b) \quad (14)$$

El coste marginal producido al sumarse un nuevo usuario a la red vendrá dado por la primera derivada del coste total, esto es:

$$Cmg = \frac{\partial CT}{\partial x} = c + x \frac{\partial c}{\partial x} \quad (15)$$

Si consideramos la función inversa de demanda de transporte por un arco de la red como $p=D(x)$, donde p es el precio generalizado asociado al arco i estará formado por el Cmg de dicho arco y el peaje π del mismo:

$$p = c + \pi \quad (16)$$

Entonces para calcular el valor del peaje óptimo tendremos que maximizar la diferencia entre el ingreso y el coste asociados a la utilización del arco. El ingreso asociado será igual al área bajo la curva de demanda de dicho arco:

$$I = \int_0^x D(x') dx' \quad (17)$$

Por su parte, el coste total será el establecido en (14), con lo que el peaje óptimo se calculará a través de la maximización de $I-CT$, dado por (17)-(14). Si calculamos la primera derivada resulta:

$$\frac{\partial(I - C)}{\partial x} = D(x) - \frac{\partial CT}{\partial x} \quad (18)$$

Igualando (18) a cero, se obtiene:

$$p = \frac{\partial CT}{\partial x} = c + x \frac{\partial c}{\partial x} \quad (19)$$

Comparando (19) y (16) resulta que el valor del peaje óptimo de congestión es:

$$\pi = x \frac{\partial c}{\partial x} = tfab(x / K)^b \quad (20)$$

Se puede comprobar que el valor de π no es más que la diferencia entre el coste medio y el coste marginal. El peaje óptimo dependerá del grado de congestión del arco, esto es, del parámetro x/K , y supone un sobrecoste que se deberá añadir al coste de atravesar el arco en cuestión. Aunque algunos autores no son partidarios de incorporar el peaje a la hora de establecer la función de coste del arco, puesto que los peajes no suponen coste ya que es dinero que se recauda, podemos definir la función de coste asociada al arco como la suma de ambos componentes.

En este caso, la función de coste de cada arco viene dada por:

$$\int_0^x [c(w) + \pi(w)] dw \quad (21)$$

La incorporación del peaje a una red supondrá que aquellos arcos con menor grado de congestión se verán favorecidos por el incremento del tráfico asignado; mientras que aquéllos que presentan un mayor coste de congestión verán disminuir su ocupación dada la mayor repercusión del peaje. En conjunto, la red alcanzará el equilibrio con un mayor valor de la función objetivo del que se producía sin peajes. Dado que la implantación del peaje supone un mayor valor de la función objetivo y que, por tanto, desde el punto de vista de la optimización se ha empeorado respecto a la situación en la que no existía peaje; no parece deducirse que el conjunto de la sociedad haya mejorado con la incorporación de los peajes. Una solución podría venir a través del uso de un sistema de asignación sin peajes, y el posterior cálculo de los mismos en función de los resultados obtenidos. Otra posibilidad resulta de calcular el

valor de la función objetivo sin peajes aplicando la asignación óptima con peajes, cuanto más se aproximen ambas, menor será el efecto distorsionador de los peajes.

Al estar trabajando con niveles de demanda fijos, no existen usuarios que ante la incorporación de los peajes abandonen la red; esto nos permite afirmar que el patrón de tráfico con peaje será menos óptimo que el patrón obtenido con peajes. El objetivo será, pues, reducir la diferencia entre ambos.

Si consideramos:

$$\int_0^x c(w)dw = \frac{tfx \left(1 + b + a \left(\frac{x}{K} \right)^b \right)}{1 + b} \quad (22)$$

y el peaje como:

$$\int_0^x \pi(w)dw = \frac{ab t f x^{b+1}}{K^b (1 + b)} \quad (23)$$

Se puede observar que el peaje supone incorporar a la función un sobrecoste en cada arco que viene dado por (23). Este sobrecoste será mayor cuanto menor sea el valor de K, esto es, en una red con peaje aquellos arcos con menor capacidad verán reducido su flujo y, por contra, aumentará el de los arcos con mayores niveles de capacidad, respecto al patrón establecido para el caso sin peajes.

Dado que a y b son parámetros iguales para todos los arcos podemos, asimismo, establecer una relación directa entre el peaje y el tiempo con flujo libre; a mayor valor de éste último, mayor es el importe del sobrecoste que el peaje supone para la función objetivo. No obstante, hay que tener en cuenta que el efecto capacidad

es muy superior al efecto del flujo libre, puesto que el primero aparece elevado a una valor b (en general el parámetro b se ha estimado con valor 4).

Al ser todos los factores del peaje positivos, no existe ningún efecto reductor y, por tanto, el peaje tiene una clara función negativa sobre el óptimo suponiendo un mayor valor de la función objetivo, que mide el coste social de la red. Ante esta situación se plantean varias alternativas:

- a) Obtener un nuevo tipo de peaje con un mayor grado de optimalidad.
- b) Establecer una función de demanda de transporte variable.
- c) Trabajar con patrones de asignación sin considerar el peaje.

La segunda posibilidad requiere un análisis más preciso del presentado en este trabajo, puesto que obliga a un estudio previo sobre la función de demanda¹² de transporte y, por tanto, será una posible vía de análisis posteriores. En cuanto a la no consideración del peaje a la hora de establecer el patrón de asignación del tráfico, ya hemos comentado que esta alternativa, ampliamente contemplada en la literatura, no parece una aproximación tan realista al problema como la que se establece mediante la asignación incorporando al coste del arco el peaje establecido.

Consideremos ahora el planteamiento de un sistema de peaje alternativo al peaje óptimo tradicional.

5.3.2. Propuesta de Peaje Alternativo.

El peaje óptimo deberá ser igual a la diferencia entre el coste marginal social y el coste marginal privado. Partiendo de nuestra función de coste, la función BPR,

¹² veáanse referencias [84, 104, 162, 168, 184, 192].

podemos calcular el coste marginal social como la variación producida en el coste social al incorporarse un nuevo usuario al arco de la red.

Si consideramos el coste social asociado a cada arco como la función objetivo de nuestro problema, entonces lo podemos expresar como:

$$CTS = \int_0^x c(w)dw \quad (24)$$

El coste marginal social resultante será la primera derivada respecto a x del CTS y, por tanto, será igual a $c(x)$, esto es, la función de coste asociada a la utilización del arco. Por su parte, el coste marginal privado se puede calcular como la primera derivada con respecto a x de $c(x)$:

$$Cm_{gp} = \frac{\partial c}{\partial x} = \frac{abtf(x/K)^b}{x} \quad (25)$$

El peaje óptimo será igual, por tanto, a la diferencia entre el coste marginal social, igual a la función BPR, y el coste marginal privado, dado por (25):

$$\pi = \frac{tf\left(x - ab\left(\frac{x}{K}\right)^b + ax\left(\frac{x}{K}\right)^b\right)}{x} \quad (26)$$

donde tomando $a=0,15$ y $b=4$ resulta:

$$\pi = tf\left(1 - \frac{0,6x^4}{K^4} + \frac{0,15x^5}{K^4}\right) \quad (27)$$

Se observa que a diferencia del anterior peaje, ahora ya no existe una relación directa entre el valor del mismo y tf ; sino que entran un mayor conjunto de parámetros y matices en juego.

Así pues, el coste que soporta cada usuario por la utilización de un arco de la red vendrá dado por la suma de $c(x)$ y π , esto es:

$$\text{coste} = c + \pi$$

Y, por tanto, el coste global que constituye la función objetivo de nuestro problema de optimización será:

$$\int_0^x \text{coste}(w) dw$$

La función de peaje obtenida será positiva para valores de $K > 1,41861$; y presenta la particularidad de que, a diferencia de la función de peaje óptima que resulta creciente y convexa en todo el dominio de definición, nuestra función alternativa presentará un mínimo y un punto de inflexión.

El análisis de la primera derivada muestra que dicha función es creciente para valores de $x > 3$, y decreciente para valores de $x < 3$, alcanzando un mínimo en 3.

$$\frac{\partial \pi}{\partial x} = \frac{-1,8tfx^2 + 0,6tfx^3}{K^4}$$

Por su parte, la segunda derivada de la función muestra la existencia de un punto de inflexión en $x = 2$ (en dicho punto la función pasaría de ser cóncava a ser convexa):

$$\frac{\partial \pi}{\partial x} = \frac{-3,6tfx + 1,8tfx^2}{K^4}$$

Si comparamos con la función de coste global sin peaje (24), podemos demostrar que la configuración obtenida con esta nueva función tiene un valor óptimo superior a la función de coste global sin peaje, pero inferior a la función de coste con peaje óptimo (21).

Proposición 6: Dada la función objetivo (21), con π establecido en (27), el óptimo de dicha función para redes congestionadas, esto es, donde x es relativamente grande respecto a K , será menor que el obtenido para π establecido en (23).

Consideremos una posible aplicación de los dos peajes, comprobaremos si el peaje alternativo resulta en una mayor o menor diferencia con respecto al patrón de flujos sin peaje, frente al peaje óptimo.

Dada la función de coste BPR, el peaje óptimo de un arco vendrá dado por :

$$P_o = tf(0,6(x/K)^4) \quad (28)$$

por su parte, el peaje alternativo será :

$$P_a = tf(1 - 0,6(x^3/K^4) + 0,15(x/K)^4) \quad (29)$$

El coste global al que tiene que hacer frente cada usuario será igual a la suma del coste asociado a atravesar el arco, determinado por la función BPR, y el peaje oportuno. Así pues, las dos funciones de coste de cada arco resultan:

Utilizando el peaje (28):

$$C_1 = tf(1 + 0,15(x/K)^4) + tfabx^4/K^4$$

(30)

a su vez, con el peaje (29):

$$C_2 = 2tf(1 + 0,15(x/K)^4) - tfab x^3/K^4 \quad (31)$$

Así pues, resultan las dos funciones objetivo del problema de asignación óptima :

$$F_1 = \int_0^x C_1(w)dw = tfx + \frac{0,15tfx^5}{K^4} \quad (32)$$

y

$$F_2 = \int_0^x C_2(w)dw = 2tfx - \frac{0,15tfx^4}{K^4} + \frac{0,06tfx^5}{K^4} \quad (33)$$

Si F_2 es menor que F_1 , entonces la función objetivo con el patrón de tráfico utilizando el peaje alternativo será menor que la obtenida con el peaje óptimo. Para ello, podemos comprobar si $F_1 - F_2 > 0$.

A partir de (32) y (33) obtenemos:

$$F_1 - F_2 = \frac{0,09tfx^5}{K^4} + \frac{0,15tfx^4}{K^4} - tfx \quad (34)$$

Por tanto, si (34) es mayor que cero, entonces se cumple:

$$\frac{0,09tfx^5}{K^4} + \frac{0,15tfx^4}{K^4} > tfx \quad (35)$$

Operando en (35) resulta :

$$0,15x^3 + 0,09x^4 > K^4 \quad (36)$$

Por tanto, para aquellos arcos para los cuales se cumpla la condición que hemos establecido en (36), el peaje alternativo derivará en un situación mejor que la establecida con el peaje óptimo.

Así pues, en el caso en que se cumpla:

$$K < \sqrt[4]{0,15x^3 + 0,09x^4} \quad (37)$$

el coste global de la utilización del arco es menor con el peaje alternativo que con el peaje óptimo. Esta situación supone que existirá una mayor tendencia a la congestión de los arcos con menor capacidad a la hora de establecer el peaje. Lógicamente, cuanto mayor sea el grado de congestión de la red, mayor es la diferencia establecida y más eficiente resultará el peaje alternativo.

Si consideramos una aproximación donde el peaje no se incluye a la hora de determinar el patrón de flujos de asignación de la red, tan sólo el coste del arco determinará la cuantía de los flujos, y en función de los mismos se determinarán los peajes pertinentes. En este caso resultan:

$$P_1 = tf(0,6(x / K)^4) \quad (38)$$

$$P_2 = tf(1 - (0,6x^3 - 0,15x^4) / K^4) \quad (39)$$

La diferencia entre P_1 y P_2 determinará cual de los dos peajes establecidos tiene un importe inferior. Así pues, si P_1 es menor que P_2 , el peaje óptimo será menor

que el peaje alternativo, y viceversa. Calculamos la diferencia entre (38) y (39), y comprobamos cuando es mayor que cero. Operando resulta que:

$$K^4 - 0,6x^3 - 0,45x < 0 \quad (40)$$

Por tanto, cuando se cumple (40), el peaje óptimo es mayor que el peaje alternativo. Bajo esta condición, la adopción del peaje alternativo supondría una menor influencia sobre el coste de la red que la del peaje óptimo. En general, cuanto más colapsado esté el arco más se aconseja la utilización del peaje alternativo, puesto que aumentará menos el sobrecoste que tienen que pagar los usuarios por la utilización del arco, respecto a la adopción del criterio del peaje óptimo.

Si consideramos las gráficas de ambos peajes, vemos que el peaje óptimo tiende a ser menor para flujos de tráfico bajos, con respecto a la capacidad del arco, mientras que el peaje alternativo prima aquellos arcos con mayor grado de congestión.

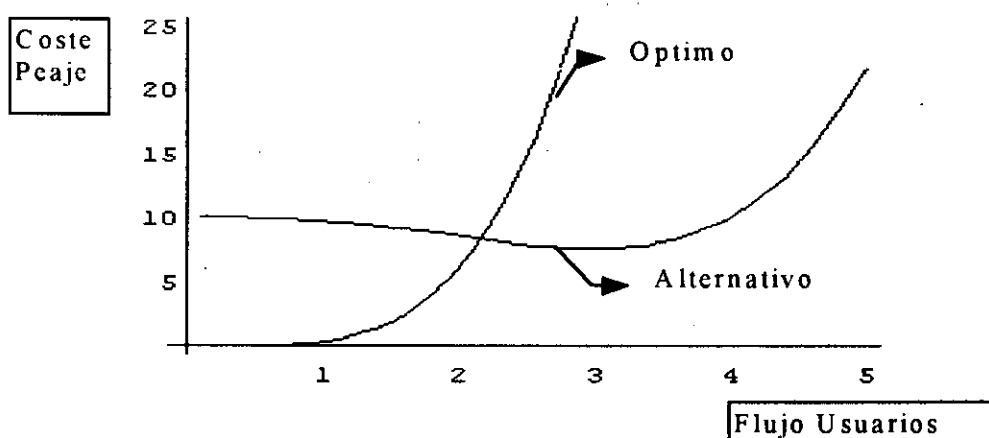


Ilustración 2: Funciones de Peaje para K=2 y $t_f=10$.

Vemos que el peaje óptimo supera al peaje alternativo a partir del punto en el cual $x > 2$.

La relación del peaje con las otras variables también cambia si consideramos el peaje alternativo; así, frente al peaje óptimo que presentaba una relación creciente con respecto al tiempo de flujo libre (tf) y una relación decreciente con respecto a la capacidad del arco (K); el peaje alternativo presentará una relación mucho más compleja. De esta forma la primera derivada con respecto a K resulta:

$$\frac{\partial \pi}{\partial K} = tf \left(\frac{2,4x^3}{K^5} - \frac{0,6x^4}{K^5} \right) \quad (41)$$

Si igualamos (41) a cero y resolvemos, obtendremos que para valores de $x > 6$, la relación del peaje con K pasa a ser decreciente, y es creciente para valores de $x < 6$. Esto supone que el peaje alternativo primará aquellos arcos que presenten un mayor flujo de tráfico en relación a la influencia de la capacidad sobre el peaje.

La derivada del peaje respecto a tf muestra que también la relación entre el peaje y el tiempo de flujo libre es variable, y existirá un valor de x, próximo a K, tal que la relación pasa de ser creciente a ser decreciente.

$$\frac{\partial \pi}{\partial f} = 1 - \frac{0,6x^3 + 0,15x^4}{K^4} \quad (42)$$

La ecuación (42) se iguala a cero para valores de K:

$$K = \sqrt[4]{0,15x^3(4 - x)} \quad (43)$$

por lo tanto, la relación entre el peaje y tf también presentará un punto de inflexión.

Gráficamente podemos ver la evolución de los distintos valores que toma la función de peaje alternativo en la Ilustración 3 y en la Ilustración 4, donde se recoge

las gráficas paramétricas del mismo. En un primer caso para una capacidad de arco que oscila entre $K=2$ y $K=4$, y un flujo entre $x=0$ y $x=5$, dado $t_f=10$. La segunda ilustración considera el valor de $K=2$, y muestra el valor de la función peaje para un flujo de tráfico libre t_f , que varía entre 0 y 10.

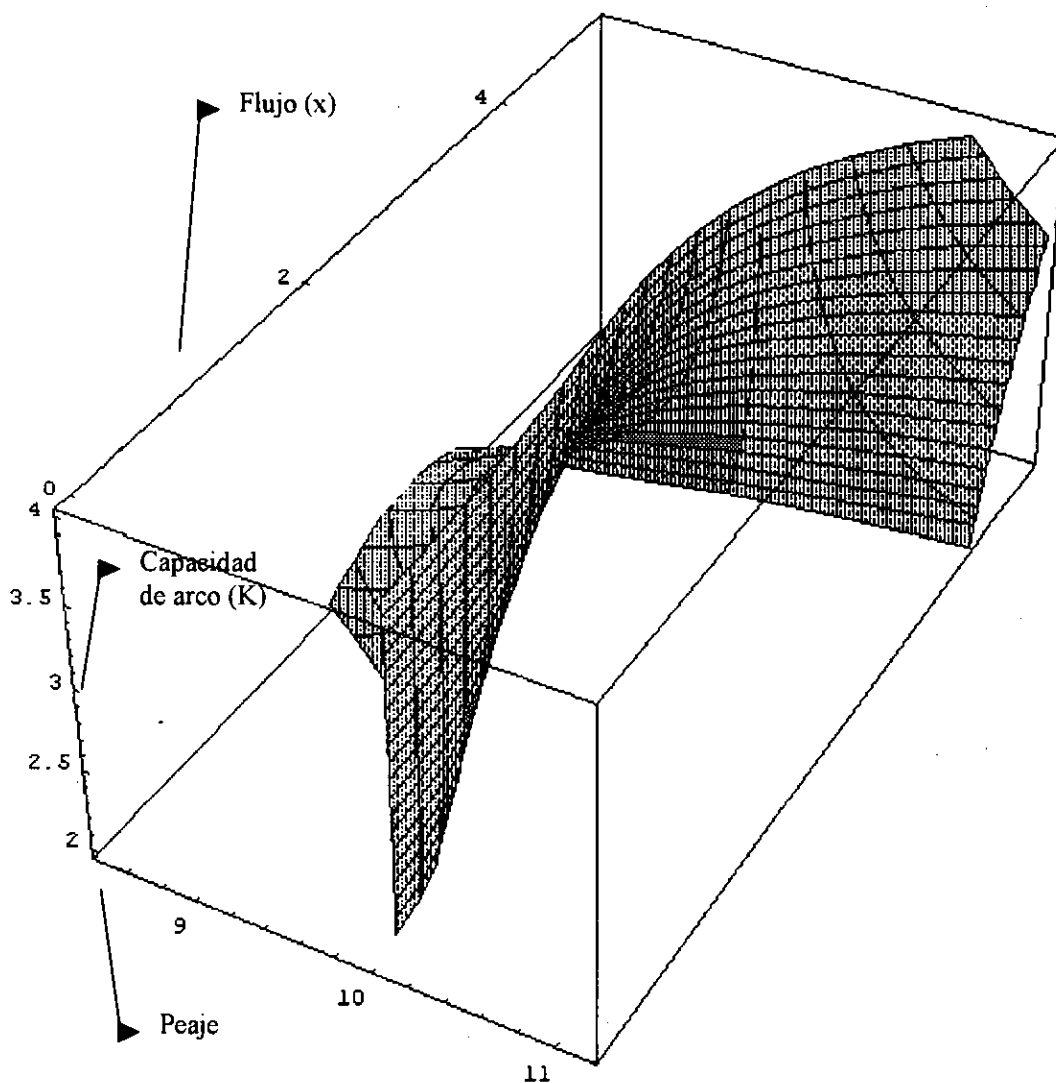


Ilustración 3: Peaje alternativo para $t_f=10$

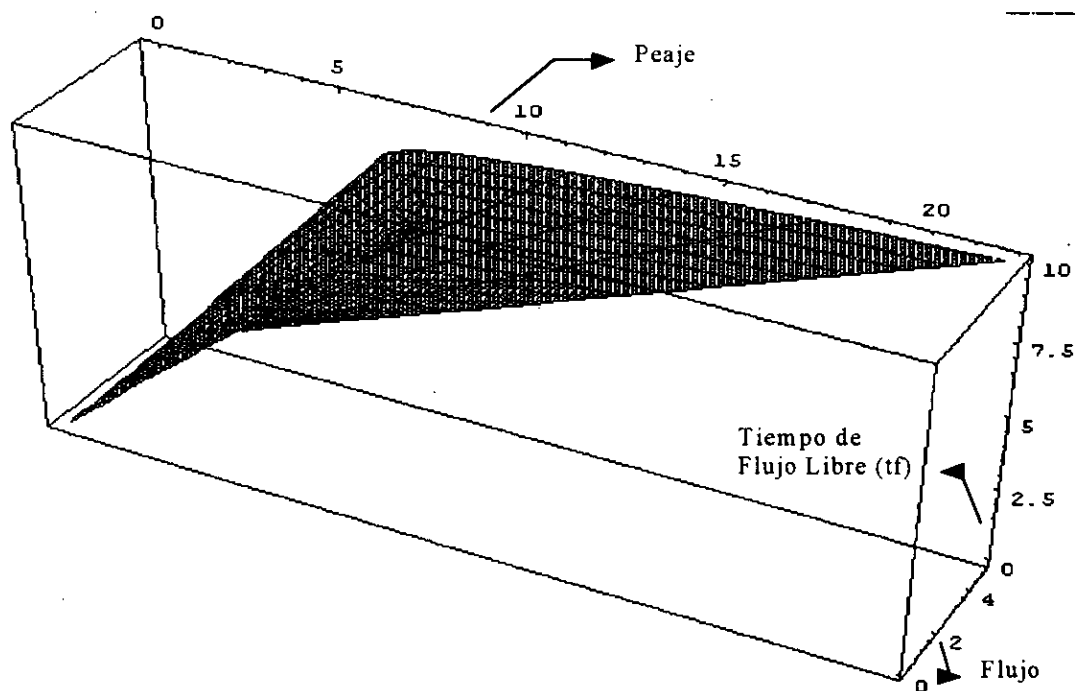


Ilustración 4: Peaje Alternativo para K=2.

5.3.3. Ejemplo de Red con Peajes.

A continuación mostramos un ejemplo¹³ de una red formada por 3 arcos y dos nodos, un nodo origen y un nodo destino. Suponemos que existe una demanda fija de usuarios que desean desplazarse del nodo origen al nodo destino y, para ello, pueden utilizar, alternativamente, cada uno de los tres arcos. La demanda total de usuarios es 10. La implantación del peaje no afectará a la demanda global, aunque si puede suponer una redistribución de la demanda de los diferentes arcos.

Consideramos que cada arco tiene una función de coste del tipo BPR con los valores siguientes de K y tf_i :

$$C_i = tf_i (1 + 0,15(x_i / K_i)^4)$$

¹³ Este caso ha sido tomado del ejemplo 5.1. de [153]

donde los parámetros particulares con respecto al tráfico con flujo libre y a la capacidad de cada arco serán los que se recogen en la siguiente tabla:

| | tf | K |
|--------|----|---|
| ARCO 1 | 10 | 2 |
| ARCO 2 | 20 | 4 |
| ARCO 3 | 25 | 3 |

Cada uno de estos arcos constituye una ruta entre el origen y el destino y, por tanto, hablar de peaje de arco es equivalente a hacerlo de peaje de ruta. Desde el punto de vista práctico, podemos trabajar indiferentemente con ambos conceptos.

Aplicamos cuatro algoritmos para resolver el problema: en el primero consideramos el algoritmo de Frank-Wolfe, tal y como figura en el ejemplo original de Sheffi [153], en el segundo trabajamos con algoritmo del conjunto activo. En ambos casos no consideramos la incorporación del peaje a la función objetivo. En el tercer caso introducimos en la función objetivo el peaje óptimo y aplicamos el método del conjunto activo para obtener la solución; y por último, en el cuarto caso planteamos el mismo problema introduciendo el peaje alternativo en la función objetivo y resolviendo a través del mismo algoritmo. Si consideramos las distintas soluciones obtenidas empleando diversos algoritmos, obtenemos la siguiente tabla resumen:

| | Flujo sin Peajes (Sheffi) | Flujo C.A. sin peaje | Flujo C.A. con peaje óptimo | Peaje Optimo | Flujo C. A. con peaje alternativo | Peaje Alternativo |
|-----------------|---------------------------|----------------------|-----------------------------|--------------|-----------------------------------|-------------------|
| Iteración | 8 | 8 | 8 | -- | 7 | -- |
| x ₁ | 3,58027 | 3,57713 | 2,8354 | 24,2388 | 4,18625 | 11,281 |
| x ₂ | 4,65211 | 4,65413 | 4,31357 | 16,2285 | 5,14869 | 21,8373 |
| x ₃ | 1,76762 | 1,76874 | 2,851 | 12,2347 | 0,6650058 | 25,0635 |
| F. O. | 189,332242 | 189,332563 | 229,303 | --- | 351,082 | --- |
| F.O. sin peajes | --- | ---- | 194,572 | --- | 194,05 | --- |

El uso del peaje alternativo permite reducir el número de iteraciones hasta alcanzar el óptimo en una menos, existiendo fuertes diferencias entre el resultado obtenido con el peaje óptimo y el obtenido con el peaje alternativo.

El patrón de flujos con peaje óptimo muestra una caída del flujo en el arco x_1 y aumento del flujo del arco 3, con respecto a los valores obtenidos en la solución óptima sin peajes. Por el contrario, el peaje alternativo supone que el arco 1 se verá favorecido por un fuerte aumento del flujo y, asimismo, el arco 2 también verá incrementarse el número de usuarios, frente a ellos el arco 3 sufrirá una fuerte reducción de los flujos asignados.

Estos resultados muestran que, mientras el peaje óptimo asigna el mayor valor del mismo a los arcos con mayor grado de congestión y el menor a los que presentan una situación menos congestionada, el peaje alternativo tiende a castigar a aquellos arcos con mayor valor de flujo de tráfico libre, y por contra, favorece a los arcos con menor valor de flujo de tráfico libre. En este sentido, la incorporación del peaje alternativo recoge el hecho de que, ante la aparición de un peaje que suponga un sobrecoste de cruzar el arco, los usuarios se desplazarán hacia aquellos arcos con menor valor del flujo libre.

En lo que respecta a la función objetivo, el valor obtenido con el peaje óptimo es mucho menor que el valor obtenido con el peaje alternativo. Pero, por el contrario, el valor de la función de coste con el patrón de flujos del peaje alternativo es menor que el obtenido empleando el patrón de flujos resultante de la implementación del peaje óptimo. En otras palabras, desde el punto de vista del coste, el peaje alternativo muestra un patrón de flujos más óptimo que el del peaje óptimo. Aunque, lógicamente, ambos son peores que los obtenidos si establecemos la asignación de flujos sin peajes.

Si trabajamos con el patrón de flujos de la asignación sin peaje, podemos calcular los diferentes peajes aplicables a los distintos arcos.

| | Flujos Optimos sin Peajes | Peaje Optimo con Flujo sin peaje | Peaje Alternativo con Flujo sin peaje |
|------------------------|---------------------------|----------------------------------|---------------------------------------|
| Arco 1 | 3,57713 | 61,4003 | 8,18539 |
| Arco 2 | 4,65413 | 21,9935 | 20,7728 |
| Arco 3 | 1,76874 | 1,181243 | 26,44778 |
| F. Objetivo sin Peajes | ----- | 189,333 | 189,333 |
| Total Peajes | ----- | 324,086775 | 172,73876 |

Si consideramos que la red presenta la asignación establecida para la función objetivo sin peaje, vemos que existen fuertes diferencias entre el peaje óptimo y el peaje alternativo por lo que respecta a los arcos 1 y 3, siendo prácticamente el mismo para el arco 2. Este hecho se debe a que el arco 2 presenta la mayor capacidad de los tres. Por contra, el arco 1, el de menor capacidad y menor valor de flujo libre, se ve claramente beneficiado por la implantación del peaje óptimo, frente al arco 3, que presenta una mayor capacidad pero al mismo tiempo el mayor valor de flujo libre y, por lo tanto, el peaje alternativo actúa sobre él de forma claramente negativa respecto al peaje óptimo.

En general, el peaje alternativo supone una mayor congestión de la red y, asimismo, una menor recaudación por peajes, si se usa con criterios de asignación que no consideren el peaje del arco. Como argumentos a su favor tenemos que, posee el menor coste del arco en los casos en los que se utilice procedimientos de asignación incorporando la función peaje a la función objetivo total.

5.4. Evaluación Resultados Obtenidos con el Método del Conjunto Activo.

Para proceder a una evaluación empírica de la utilidad del método del conjunto activo aplicado a redes hemos tomado tres ejemplos de redes sencillas sin peajes y, hemos calculado el óptimo utilizando el método del conjunto activo. Los resultados obtenidos son comparados con los que se obtienen a través del método de Frank-Wolfe, tal y como aparece recogido en [153]. Los parámetros a estudiar de las soluciones obtenidas serán básicamente tres:

- **Precisión de la Solución Óptima Global:** Constituye, si cabe, el parámetro más importante puesto que comprobamos el valor de la función en el óptimo obtenido a través de ambos métodos. Dado que estamos trabajando con problemas de minimización, se considerará que cuanto menor sea el valor de dicha función, mejor es el resultado obtenido a través del método de cálculo de optimización.
- **Número de Iteraciones Necesarias:** Se considerará que un método será mejor que otro cuanto menor sea el número de iteraciones que se deban realizar para encontrar el valor óptimo de la función objetivo. Cuando la diferencia entre iteraciones no resulte excesiva se tendrá que optar por uno u otro método en función de los valores de la solución óptima global.
- **Multiplicadores de Lagrange:** En el óptimo el valor de los multiplicadores de Lagrange es cero. Según este criterio, un método será superior al otro cuanto más próximos a cero estén los multiplicadores en la solución obtenida. Se considerará que un número es prácticamente cero, para valores del mismo del orden de 10^{-4} unidades.

En el estudio de comparación entre métodos se deberán tener en cuenta los resultados obtenidos para los tres parámetros comentados de forma conjunta. En función de estos tres criterios comprobaremos cual de los dos métodos presenta mejores resultados para un total de tres ejemplos.

En el primero de ellos consideramos un red muy sencilla con un sólo par origen-destino y un fuerte grado de congestión, puesto que la demanda conjunta es igual a la superior a la suma de las capacidades de los arcos. El siguiente ejemplo trabaja con una red mucho más compleja donde existen dos pares de nodos origen-destino, lo que supone la existencia de cuatro demandas de tráfico en la red; al igual que en la red anterior, el grado de congestión es muy alto y el tráfico en algunos arcos será necesariamente mayor a su capacidad. El último ejemplo, recoge una red compleja, pero con muy escaso grado de congestión.

Para la realización de todos los cálculos se ha usado un ordenador PC clónico con procesador Pentium de Intel a 50 MHz. Se ha utilizado la versión 2.1 del programa de Mathematica para Windows 3.11. Nos hemos basado en las funciones estándares implementadas en Mathematica y se han realizado dos paquetes con funciones propias: frank.m, para el método de Frank-Wolfe y cactivo.m, para el método del conjunto activo.

5.4.1. Ejemplo 1.

Este ejemplo está tomado de Sheffi [153]. Consideramos una red formada por un nodo origen y un nodo destino conectados por tres arcos. No existirán nodos intermedios por lo que cada arco será equivalente a una ruta entre el origen y el destino. Existe pues, una equivalencia directa entre flujo de arco y flujo de ruta.

Las funciones de coste de cada arco están basadas en la función de flujo-coste BPR, donde el coste de atravesar un arco se considera función directa del coeficiente

flujo/capacidad de dicho arco, y de un parámetro de flujo libre que representa el menor coste temporal posible de circular a través de dicho arco.

Las tres funciones de coste resultantes serán:

a) Arco 1: $c_1 = 10 + 1,5(x_1 / 2)^4$

b) Arco 2: $c_2 = 20 + 3(x_2 / 4)^4$

c) Arco 3: $c_3 = 25 + 3,75(x_3 / 3)^4$

La demanda total de usuarios en la red será de 10. Así pues, sólo existirá una restricción de igualdad en el conjunto activo: $x_1 + x_2 + x_3 = 10$. El flujo por cada arco deberá ser igual o superior a cero.

La red planteada presenta la siguiente ilustración gráfica:

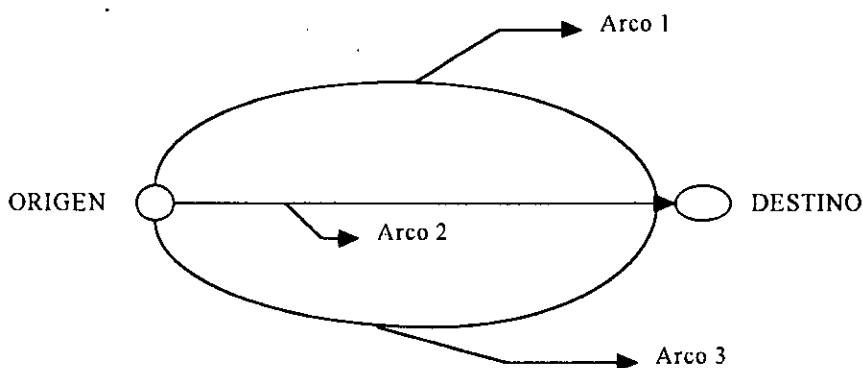


Ilustración 5: Red Ejemplo 1.

La función objetivo será:

$$F = \sum_{i=1}^3 \int_0^{x_i} c_i(w) dw$$

El test de Convergencia entre soluciones se define como:

$$TestC = \text{Max}_i \left| \frac{x_i^{k+1} - x_i^k}{x_i^k} \right| * 100$$

Este test de convergencia se utiliza para comprobar si la diferencia entre dos soluciones sucesivas es significativa y, por tanto, conviene seguir con el proceso iterativo. En caso de que esta diferencia sea no significativa - en general tomaremos para ello que el test de convergencia sea menor que 0,1- , procederemos a detener el algoritmo de búsqueda de la solución.

Los resultados obtenidos mediante el método de Frank-Wolfe son los siguientes:

| Iteración | Fl. Ar. 1 | Fl. Ar. 2 | Fl. Ar. 3 | F. Objetivo | Test Conver. |
|-----------|-----------|-----------|-----------|-------------|--------------|
| Inicial | 10 | 0 | 0 | 1975 | -- |
| 1 | 4,03457 | 5,96543 | 0 | 197,404 | -- |
| 2 | 3,38446 | 5,00419 | 1,61135 | 189,994 | -- |
| 3 | 3,61966 | 4,82628 | 1,55406 | 189,445 | 6,94929 |
| 4 | 3,54581 | 4,72782 | 1,72636 | 189,361 | 11,0873 |
| 5 | 3,59224 | 4,69381 | 1,71395 | 189,34 | 1,30937 |
| 6 | 3,573 | 4,66867 | 1,75832 | 189,334 | 2,58924 |
| 7 | 3,58584 | 4,65935 | 1,75481 | 189,3327169 | 0,359286 |
| 8 | 3,58027 | 4,65211 | 1,76762 | 189,3322426 | 0,729763 |
| 9 | 3,58404 | 4,64938 | 1,76658 | 189,3321016 | 0,10402 |
| 10 | 3,58239 | 4,64723 | 1,77039 | 189,3320596 | 0,215512 |
| 11 | 3,58351 | 4,64641 | 1,77008 | 189,332047 | 0,0314859 |

Los resultados obtenidos utilizando el método del conjunto activo son:

Capítulo 5. Implementación en Redes del Método del Conjunto Activo.

| Iteración | Fl. Ar. 1 | Fl. Ar. 2 | Fl. Ar. 3 | F. Objetivo | Test Conver. |
|-----------|-----------|-----------|-----------|-------------|--------------|
| Inicial | 2 | 4 | 4 | 212,481 | -- |
| 1 | 4 | 4,12764 | 1,87236 | 191,583 | 100 |
| 2 | 2,32143 | 7,56137 | 0,1172 | 236,567 | 93,7405 |
| 3 | 2,38601 | 7,37959 | 0,2344 | 230,057 | 100 |
| 4 | 3,02351 | 6,50769 | 0,468801 | 204,202 | 100 |
| 5 | 3,889 | 5,4607 | 0,650294 | 192,423 | 38,7143 |
| 6 | 3,62727 | 5,11592 | 1,2568 | 190,027 | 93,2668 |
| 7 | 3,56648 | 4,7272 | 1,70632 | 189,353 | 35,7672 |
| 8 | 3,57713 | 4,65413 | 1,76874 | 189,332563 | 3,65816 |
| 9 | 3,58271 | 4,64513 | 1,77216 | 189,3320446 | 0,193443 |
| 10 | 3,58327 | 4,6451 | 1,77164 | 189,3320416 | 0,0295183 |

El análisis de los resultados obtenidos muestra la existencia de varios puntos de divergencia entre ambos métodos empleados. Utilizando el algoritmo de Frank-Wolfe, hemos encontrado la solución óptima en un total de 11 iteraciones, con un test de convergencia final de 0,03%, inferior al 0,1% deseado. El valor de la función objetivo en la última solución es de 189,332047. La distribución de flujo entre los distintos arcos tiende a colapsar el arco 1, 1,5 veces su capacidad, y el arco 2, que está sobrepasado en un 0,25 veces su capacidad máxima; por su parte, el arco 3 registra un nivel de flujo inferior a su capacidad máxima, estando utilizado sobre un 50% de la misma. Este mismo patrón de usos lo encontramos en el método del conjunto activo.

El método del conjunto activo se muestra superior al método de Frank-Wolfe en el número de iteraciones realizadas, 10 frente a 11, en el valor de la función objetivo en la solución óptima final 189,3320416 frente a 189,332047, y en el test de convergencia entre soluciones, 0,029% para el método del conjunto activo. No obstante, un análisis del conjunto de iteraciones muestra que mientras el método de Frank-Wolfe se ha situado en las proximidades de la solución óptima a la altura de la tercera iteración, el método del conjunto activo no alcanza la proximidad del óptimo hasta la séptima iteración. Así pues, el método del conjunto activo se muestra mucho

más lento a la hora de aproximarse al óptimo, pero no pierde velocidad de convergencia en sus proximidades; por su parte, el método de Frank-Wolfe alcanza con gran rapidez las proximidades del óptimo, pero su velocidad de aproximación se reduce de forma muy considerable en el entorno del mismo, en un total de 7 iteraciones alcanza un resultado peor que el algoritmo del conjunto activo en un total de 3 iteraciones.

Este hecho marca una diferencia clara entre ambos métodos, el de Frank-Wolfe es rápido, pero impreciso, mientras que, el método del conjunto activo gana en precisión a costa de una mayor lentitud de convergencia al óptimo. En cuanto a la solución, se observa que ambos algoritmos obtienen prácticamente el mismo resultado, y que el método del conjunto activo obtiene un valor óptimo final de la función objetivo ligeramente inferior al obtenido por Frank-Wolfe.

Por último, el valor de los multiplicadores calculados a través de $Z^1 g(x^k)$, esto es, el gradiente proyectado, es prácticamente igual a cero: $\{0,000023, 0,00028\}$ por lo que se confirma que podemos afirmar que se ha alcanzado la solución óptima. Continuar las iteraciones apenas modificará la solución.

5.4.2. Ejemplo 2.

El siguiente ejemplo está tomado de Hearn & Ribera [79] y considera una red con 18 arcos y 9 nodos, estableciendo dos nodos origen y dos nodos destino, lo que supone un total de 4 demandas fijas de tránsito entre orígenes y destinos. Los arcos presentan funciones de coste del tipo BPR y, a diferencia del problema original planteado por los autores nosotros no hemos considerado la existencia de límites al flujo máximo establecidos por la capacidad del arco; esto es, el flujo por cada arco puede ser superior a su capacidad máxima.

Capítulo 5. Implementación en Redes del Método del Conjunto Activo.

El número de variables del problema es de 18 y viene dado por el número de arcos; el número de restricciones de igualdad establecido por el principio de conservación de flujo es igual al número de nodos, esto supone un total de 9 restricciones de igualdad, asimismo existirán 18 restricciones de no negatividad de las variables.

La demanda total de viajes entre los pares origen-destino es la siguiente:

| Origen / Destino | 3 | 4 |
|------------------|----|----|
| 1 | 10 | 20 |
| 2 | 30 | 40 |

Los parámetros de las funciones de coste de los diferentes arcos son:

| Arco | tf | K |
|------|----|----|
| 1 | 5 | 12 |
| 2 | 6 | 18 |
| 3 | 3 | 35 |
| 4 | 9 | 35 |
| 5 | 9 | 20 |
| 6 | 2 | 11 |
| 7 | 8 | 26 |
| 8 | 4 | 11 |
| 9 | 6 | 33 |
| 10 | 7 | 32 |
| 11 | 3 | 25 |
| 12 | 6 | 24 |
| 13 | 2 | 19 |
| 14 | 8 | 39 |
| 15 | 6 | 43 |
| 16 | 4 | 36 |
| 17 | 4 | 26 |
| 18 | 8 | 30 |

El conjunto de restricciones de igualdad viene representado a través de la siguiente matriz de coeficientes:

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

El vector de constantes del lado derecho será:

$$b = (30 \quad 70 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 40 \quad 60)$$

Gráficamente la red presenta la siguiente configuración:

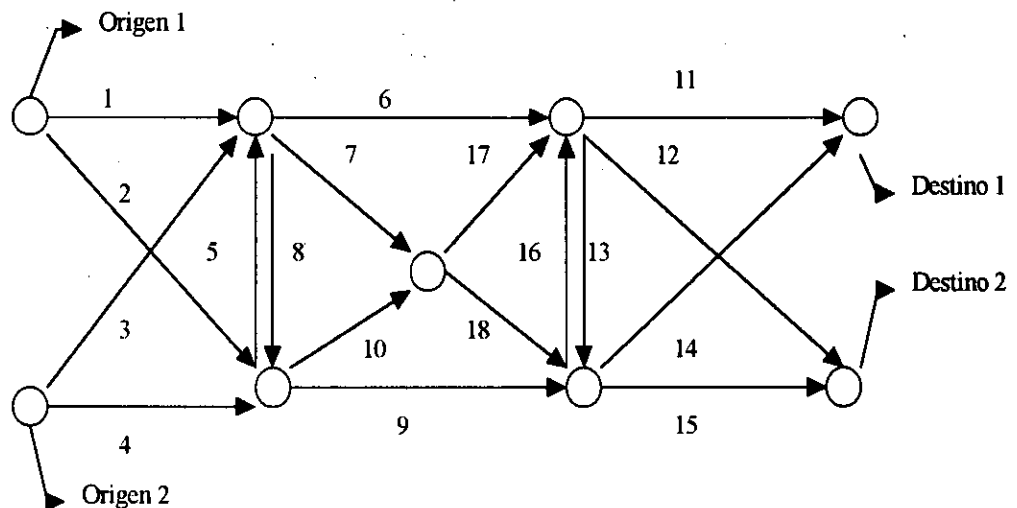


Ilustración 6: Red Ejemplo 2.

La función objetivo del problema es:

$$F = \sum_{i=1}^{18} \int_0^{x_i} c_i(w) dw$$

En este caso, dado que varias variables tienen valores cero, no calcularemos el test de convergencia entre soluciones, puesto que para valores de $x_i = 0$, dicho test no aparece definido y, por lo tanto, no se puede calcular.

Los resultados finales obtenidos a través de ambos métodos son los siguientes:

a) Método de Frank-Wolfe. Resultados después de 14 iteraciones:

| | | | | | | | | | |
|------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| Arco | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| | 11,46 | 18,53 | 48,11 | 21,88 | 0 | 28,37 | 31,21 | 0 | 40,416 |
| Arco | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| | 0 | 27,49 | 29,94 | 3,76 | 12,50 | 30,05 | 1,62 | 31,21 | 0 |

Número de Arcos con Flujo superior a su capacidad: 8.

Número de Arcos con Flujo nulo: 4.

b) Método del Conjunto Activo. Resultado después de 14 iteraciones:

| | | | | | | | | | |
|------|-------|-------|-------|-------|----|--------|-------|-------|-------|
| Arco | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| | 16,38 | 13,61 | 38,15 | 31,84 | 0 | 27,433 | 27,10 | 0 | 45,46 |
| Arco | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| | 0 | 40 | 14,53 | 0 | 0 | 45,46 | 0 | 27,10 | 0 |

Número de Arcos con Flujo superior a su capacidad: 8.

Número de Arcos con Flujo nulo: 7.

Capítulo 5. Implementación en Redes del Método del Conjunto Activo.

En la solución obtenida después de 14 iteraciones, vemos que ambos métodos establecen un flujo superior a la capacidad en 8 arcos, pero a diferencia del método de Frank-Wolfe, donde el número de arcos por debajo de su capacidad es 6; en el método del conjunto activo, tan sólo 3 arcos tienen un flujo inferior a su capacidad, puesto que el número de arcos con flujo nulo es superior en este segundo método.

Este hecho implica que el método del conjunto activo presenta una mayor tendencia a producir congestión en la red, de forma tal, que en la solución óptima, la mayor parte de los arcos de la misma habrán sobrepasado su capacidad.

La evolución del resultado de la función objetivo utilizando ambos métodos se recoge en la siguiente tabla:

| Iteración | F. Objetivo (Frank-Wolfe) | F. Objetivo (Conjunto Activo) |
|-----------|---------------------------|-------------------------------|
| Inicial | 42742,9 | 2300,63 |
| 1 | 2162,03 | 2209,54 |
| 2 | 1941,28 | 2121,1 |
| 3 | 1919,47 | 2117,95 |
| 4 | 1914,11 | 1978,91 |
| 5 | 1904,7 | 1919,29 |
| 6 | 1902,24 | 1886,27 |
| 7 | 1899,73 | 1881,21 |
| 8 | 1897,98 | 1870,15 |
| 9 | 1896,93 | 1860,26 |
| 10 | 1896,07 | 1857,93 |
| 11 | 1895,52 | 1857,55 |
| 12 | 1895,17 | 1856,76 |
| 13 | 1894,94 | 1856,5 |
| 14 | 1894,78 | 1856,498199 |

Consideramos en la siguiente tabla la aproximación del valor de la función objetivo al valor óptimo final en las distintas iteraciones, para ello empleamos la fórmula:

$$TC = \frac{|f^i - f^*|}{f^*} * 100$$

donde f^i representa el valor de la función objetivo en la i -ésima iteración, y f^* es el valor óptimo de la función objetivo.

A través de este test de convergencia podremos medir la velocidad de convergencia al óptimo del algoritmo considerado.

| Iteración | Algoritmo Frank-Wolfe | Conjunto Activo |
|-----------|-----------------------|-----------------|
| Inicial | -- | -- |
| 1 | 16,45 | 19,01 |
| 2 | 4,56 | 14,25 |
| 3 | 3,39 | 14,08 |
| 4 | 3,01 | 6,59 |
| 5 | 2,59 | 3,38 |
| 6 | 2,33 | 1,60 |
| 7 | 2,32 | 1,33 |
| 8 | 2,23 | 0,73 |
| 9 | 2,17 | 0,02 |
| 10 | 2,13 | 0,00 |
| 11 | 2,10 | 0,00 |
| 12 | 2,08 | 0,00 |
| 13 | 2,07 | 0,00 |
| 14 | 2,06 | 0 |

Se observa que la convergencia en las cinco primeras iteraciones es superior en el método de Frank-Wolfe, pero posteriormente este sufre un proceso de ralentización que no ocurre con el método del conjunto activo donde a la altura de la décima iteración, prácticamente se ha alcanzado la solución óptima. Asimismo, debemos señalar que después de 14 iteraciones el método de Frank-Wolfe mantiene una diferencia con el valor óptimo hallado a través del método del conjunto activo ligeramente superior al 2%.

El análisis de las soluciones obtenidas pone de manifiesto dos hechos:

1°. El método del conjunto activo alcanza al cabo de 14 iteraciones una solución óptima superior a la obtenida por el método de Frank-Wolfe. El mínimo alcanzado es un 2% inferior, siendo, por tanto, una diferencia muy significativa que muestra la mayor precisión del método del conjunto activo respecto al método de Frank-Wolfe. De hecho, podemos afirmar que la solución obtenida a través del primer método no es un mínimo.

2° El método del conjunto activo se sigue mostrando más lento a la hora de aproximarse a la solución óptima. El método de Frank-Wolfe muestra resultados superiores hasta la quinta iteración. No obstante, a partir de este momento el método de Frank-Wolfe sufre una ralentización muy fuerte que le impide aproximarse a buen ritmo al mínimo de la función objetivo, siendo ampliamente superado en velocidad por el método del conjunto activo, que en 3 iteraciones se sitúa en el entorno próximo del mínimo. Así pues, el método del conjunto activo se muestra más eficaz para procesos iterativos largos, mientras que para aquéllos en los cuales se alcanza el óptimo en pocas etapas, el método de Frank-Wolfe muestra un mejor grado de convergencia.

Desde el punto de vista operativo hay que significar la mayor eficiencia del método del conjunto activo, puesto que tan sólo requiere una asignación inicial, frente al método de Frank-Wolfe que implica la necesidad de realizar una asignación "todo-o-nada" en cada una de las iteraciones. Este tipo de asignaciones se vuelven muy complicadas cuando existen numerosos arcos que posibilitan la existencia de un número grande de rutas. En nuestro caso, entre cada par origen - destino existirán un total de 24 rutas, lo que para 4 pares, implica la necesidad de trabajar en cada asignación con un total de 96 rutas. Este proceso es realmente complejo y difícil de

implementar en un ordenador¹⁴. En el caso del algoritmo del conjunto activo, este proceso no es necesario.

Por su parte, el algoritmo del conjunto activo obliga a trabajar con matrices de orden 18, lo que supone que se necesitará un cierto espacio de memoria para almacenamiento de las mismas, con el cual deberemos contar en el procesador. Lógicamente, cuanto mayor sea el número de arcos del problema, será superior el orden de las matrices con las cuales trabajamos y se incrementarán de forma importante nuestros requisitos de memoria para el almacenamiento de las mismas.

Por último, el análisis del gradiente proyectado en el método del conjunto activo muestra que hemos encontrado el óptimo puesto que su valor se aproxima de forma significativa a cero.

5.4.3. Ejemplo 3.

Este ejemplo se ha realizado en función del que plantean Nguyen & Dupuis [141]. Consideramos una red de 13 nodos y 19 arcos, y 4 pares origen-destino. Al igual, que en los casos anteriores trabajaremos con funciones de coste del tipo BPR¹⁵, y la demanda entre los diversos pares de nodos origen-destino se considerará fija. Al igual que en el caso anterior, tampoco consideraremos la existencia de límites superiores a la capacidad de los arcos.

¹⁴ Este tipo de asignación conlleva la necesidad de utilizar los algoritmos de cálculo del camino más corto, como medida para la determinación de la ruta de coste mínimo y proceder a la pertinente asignación "todo-o-nada".

¹⁵ Hay que señalar que en el problema original los autores no consideran una función de este tipo, pero dado que nosotros hemos trabajado tan sólo con las funciones BPR, hemos procedido a la adaptación de las mismas.

La principal característica diferencial de este ejemplo es que las capacidades de los arcos son altas en relación al flujo global que circula por la red, de forma tal, que ésta no tiene porque aparecer fuertemente congestionada en la solución óptima, como sucedía en la red del ejemplo 2, donde las capacidades de los arcos apenas llegaban para soportar el flujo circulante entre los diversos orígenes y destinos.

El número total de variables del problema será 19 y, el número de restricciones iniciales del conjunto activo será de 13; asimismo, existirán un total de 19 restricciones de no negatividad que no figurarán en el mismo, pero que se podrán ir incorporando según se vuelvan activas.

La función objetivo del presente ejemplo se calcula, al igual que los anteriores, de acuerdo con la siguiente fórmula:

$$F = \sum_{i=1}^{19} \int_0^{x_i} c_i(w) dw$$

Tal y como ocurría en el caso anterior, dado que diversas variables tienen valores cero, no calcularemos el test de convergencia entre soluciones, puesto que para valores de $x_i = 0$, dicho test no aparece definido y, por lo tanto, no se puede calcular. En el caso del método de Frank-Wolfe, dado que ninguna de las soluciones presenta valores nulos, el test de convergencia sería aplicable.

Mostramos en la siguiente tabla las demandas entre los distintos orígenes (nodos 1 y 4) y los destinos (nodos 2 y 3):

| Origen / Destino | 2 | 3 |
|------------------|-----|-----|
| 1 | 400 | 800 |
| 4 | 600 | 200 |

Los parámetros de las funciones de coste de los diferentes arcos son:

Capítulo 5. Implementación en Redes del Método del Conjunto Activo.

| Arco | tf | K |
|------|----|------|
| 1 | 7 | 600 |
| 2 | 9 | 700 |
| 3 | 9 | 600 |
| 4 | 12 | 400 |
| 5 | 3 | 900 |
| 6 | 9 | 1200 |
| 7 | 5 | 1000 |
| 8 | 13 | 850 |
| 9 | 5 | 1000 |
| 10 | 9 | 750 |
| 11 | 9 | 500 |
| 12 | 10 | 800 |
| 13 | 9 | 1000 |
| 14 | 6 | 700 |
| 15 | 9 | 600 |
| 16 | 8 | 700 |
| 17 | 7 | 900 |
| 18 | 14 | 500 |
| 19 | 11 | 900 |

El conjunto de restricciones de igualdad nos define la siguiente matriz activa:

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

El vector de constantes del lado derecho tiene un total de 13 componentes:

$$b = (1200 \ 800 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1000 \ 0 \ 0 \ 0 \ 0 \ 1000)$$

Gráficamente la red presenta la siguiente configuración:

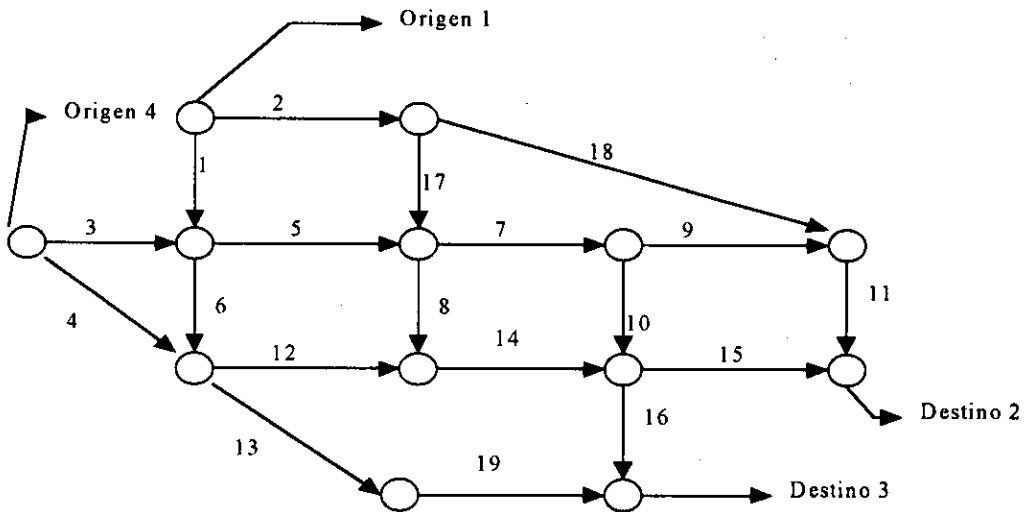


Ilustración 7: Red Ejemplo 3.

La obtención del resultado final ha sido compleja debido a la necesidad de un gran número de iteraciones propiciadas por la complejidad de la red y la dimensión de los flujos existentes; así, en el caso del método del conjunto activo la solución obtenida después de 22 iteraciones presenta el siguiente valor del gradiente proyectado: $\{0,0424; 0,0104; -0,0303; -0,073\}$, lo que supone, que podríamos seguir iterando y mejorando el valor de la función objetivo.

Mediante el uso de algoritmo de Frank-Wolfe la solución no se ve mejorada. Hay que señalar en el caso de este último método que, la necesidad de realizar en cada iteración una asignación del tipo “todo-o-nada” conlleva un coste operacional excesivo¹⁶ lo que limita de manera considerable el posible grado de aplicación de dicho algoritmo frente a otros donde el número de asignaciones del tipo “todo-o-nada” son inferiores.

¹⁶ En cada asignación es necesario calcular el camino más corto para cada uno de los pares origen - destino.

a) Método del Conjunto Activo. Resultado después de 22 iteraciones:

| | | | | | | | | | | |
|----|-------|-------|-------|-------|--------|-------|-------|-------|-------|-------|
| A. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| | 955,4 | 244,5 | 328,2 | 471,8 | 1084,3 | 199,3 | 972,5 | 111,8 | 488,2 | 484,4 |
| A. | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | |
| | 732,7 | 0 | 671,1 | 111,8 | 267,2 | 328,9 | 0 | 244,5 | 671,1 | |

Número de Arcos con Flujo superior a su capacidad: 4.

Número de Arcos con Flujo nulo: 2.

En la solución obtenida después de 22 iteraciones, el método del conjunto activo presenta tan sólo 4 arcos donde el flujo excede a su capacidad y, 2 arcos con flujo nulo. Vemos que a diferencia del ejemplo 2, donde el flujo era relativamente alto en relación a la capacidad de los distintos arcos de la red; en este caso dadas las mayores capacidades de los arcos, el número de ellos que presentan un exceso de flujo es relativamente pequeño.

b) Método de Frank-Wolfe¹⁷. Resultados después de 22 iteraciones:

| | | | | | | | | | | |
|------|-------|-------|-------|-------|--------|-------|---------|-------|-------|--------|
| Arco | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| | 813,2 | 386,7 | 416,1 | 383,9 | 1051,6 | 177,7 | 1057,04 | 13,47 | 333,2 | 723,86 |
| Arco | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | |
| | 701,1 | 190,6 | 370,9 | 204,1 | 298,9 | 629,1 | 18,88 | 367,8 | 370,9 | |

Número de Arcos con Flujo superior a su capacidad: 3.

Número de Arcos con Flujo nulo: 0.

Después de realizadas 22 iteraciones del algoritmo de Frank-Wolfe la solución óptima obtenida es muy diferente a la que muestra el algoritmo del conjunto activo.

¹⁷ Para la realización del método de Frank-Wolfe se ha utilizado el algoritmo de cálculo del camino más corto de una red que figura implementado en el paquete de programación QS (Quantitative System) versión 3.0.

Capítulo 5. Implementación en Redes del Método del Conjunto Activo.

Básicamente, la principal diferencia reside en una mayor utilización del conjunto de arcos de la red - ninguno posee flujo nulo- , y un menor grado de congestión de la misma - la capacidad del arco tan sólo se ve superada en tres de ellos- . Asimismo, el grado de congestión es ligeramente inferior, puesto que existe una mayor distribución de flujo entre los distintos arcos de la red, frente a la tendencia a la concentración, que se deriva de la utilización del método del conjunto activo para asignación de flujos.

La evolución del valor de la función objetivo en las distintas iteraciones es la siguiente:

| Iteración | F. Objetivo (Frank-Wolfe) | F. Objetivo (Conjunto Activo) |
|-----------|---------------------------|-------------------------------|
| Inicial | 76902,3 | 73324,6 |
| 1 | 72057,8 | 73281,9 |
| 2 | 70789,8 | 69752,6 |
| 3 | 69561,9 | 69736,1 |
| 4 | 69005,7 | 68216,3 |
| 5 | 68793,9 | 68209,3 |
| 6 | 68440,9 | 67939,4 |
| 7 | 68315,9 | 67932 |
| 8 | 68247,8 | 67930,1 |
| 9 | 68129,1 | 67925,8 |
| 10 | 68089,4 | 67915,9 |
| 11 | 68066,2 | 67896,8 |
| 12 | 68048,3 | 67871 |
| 13 | 68046,8 | 67850,9 |
| 14 | 68029,1 | 67845 |
| 15 | 68006,6 | 67844,5 |
| 16 | 68000,4 | 67844,2 |
| 17 | 67992,3 | 67843,2 |
| 18 | 67971,9 | 67841,4 |
| 19 | 67965,9 | 67838,9 |
| 20 | 67965,4 | 67837 |
| 21 | 67953,7 | 67836,4 |
| 22 | 67945,8 | 67836,37941 |

En este ejemplo las diferencias entre los valores de la función objetivo obtenidos para los dos algoritmos son bastante importantes. El algoritmo del conjunto activo obtiene, después de 22 iteraciones, una solución sustancialmente menor que la obtenida a través del algoritmo de Frank-Wolfe. Si al igual que en el ejemplo anterior, consideramos el test de convergencia de la función objetivo respecto a su valor óptimo, tomando como tal el obtenido mediante el algoritmo del conjunto activo después de realizadas veintidós iteraciones, los resultados son:

| Iteración | Algoritmo Frank-Wolfe | Conjunto Activo |
|-----------|-----------------------|-----------------|
| Inicial | -- | -- |
| 1 | 6,22 | 8,02 |
| 2 | 4,35 | 2,82 |
| 3 | 2,54 | 2,8 |
| 4 | 1,72 | 0,56 |
| 5 | 1,41 | 0,55 |
| 6 | 0,89 | 0,15 |
| 7 | 0,7 | 0,14 |
| 8 | 0,6 | 0,13 |
| 9 | 0,43 | 0,11 |
| 10 | 0,37 | 0,08 |
| 11 | 0,33 | 0,05 |
| 12 | 0,31 | 0,02 |
| 13 | 0,31 | 0,01 |
| 14 | 0,28 | 0,01 |
| 15 | 0,25 | 0,01 |
| 16 | 0,24 | 0,01 |
| 17 | 0,23 | 0,01 |
| 18 | 0,19 | 0,00 |
| 19 | 0,19 | 0,00 |
| 20 | 0,19 | 0,00 |
| 21 | 0,17 | 0,00 |
| 22 | 0,16 | 0,00 |

Las soluciones finales, después de 22 iteraciones, obtenidas por ambos métodos son fuertemente diferentes; poniendo de relieve dos características:

a) El método del conjunto activo obtiene una solución mucho mejor que la obtenida a través del método de Frank-Wolfe. El valor de la función objetivo en la iteración 22 obtenido mediante la solución del algoritmo del conjunto activo es un 0,2% mejor que el del algoritmo de Frank-Wolfe. Asimismo, el valor obtenido por este segundo algoritmo ya había sido alcanzado en la sexta iteración del algoritmo del conjunto activo.

b) La velocidad de aproximación a la solución final es superior en las primeras etapas mediante el empleo del algoritmo de Frank-Wolfe, en las cuatro primeras iteraciones presenta un valor de la función objetivo inferior al obtenido a través de la solución del algoritmo del conjunto activo. No obstante, según va aumentando el número de etapas el algoritmo de Frank-Wolfe tiende a disminuir su velocidad de convergencia, mientras que la del algoritmo del conjunto activo se mantiene durante un mayor número de etapas, lo que permite una mayor aproximación al óptimo global.

En general, la comparación entre ambos algoritmos favorece al método del conjunto activo, puesto que, en general, obtiene mejores soluciones, y de una forma más rápida. En cuanto al coste operacional de ambos métodos, hay que indicar que mientras el algoritmo de Frank-Wolfe presenta un menor coste de almacenamiento de las soluciones e iterandos, el método del conjunto activo supone la necesidad de un mayor espacio para el almacenamiento de las matrices de trabajo, así como de las soluciones y gradientes anteriores, lo que ralentiza en cierta medida su velocidad de ejecución.

Uno de los parámetros más importantes a la hora de estudiar la complejidad operacional del algoritmo es el número de asignaciones del tipo “todo-o-nada”¹⁸. Aquí el algoritmo del conjunto activo se muestra mucho más eficiente puesto que tan sólo requiere una asignación inicial, que se obtiene asignando una cantidad de flujo a todos los arcos de forma que se cumplan las restricciones del conjunto activo, mientras que el algoritmo de Frank-Wolfe supone la necesidad de realizar una asignación de este tipo en cada una de las iteraciones que se realice. Las asignaciones tipo “todo-o-nada” son relativamente complejas puesto que requieren el cálculo del camino más corto de la red entre cada origen y cada destino para proceder a la asignación de flujo.

¹⁸ Para la implementación de algoritmos de asignación en función del camino más corto se puede consultar [153] o [169]

CAPITULO 6

CONCLUSIONES

6. CONCLUSIONES.

El objetivo principal del presente trabajo ha sido mostrar la aplicación del método de optimización del conjunto activo a un problema económico. La teoría económica tradicional ha abordado los problemas de optimización de funciones no lineales sujetas a restricciones lineales mediante el empleo de la función de Lagrange; este sistema, no obstante, se ha mostrado altamente ineficiente en numerosos casos particulares, lo que ha propiciado la aparición de métodos numéricos de optimización basados en algoritmos iterativos. Todos estos métodos se basan en una idea común consistente en ir aproximándonos mediante etapas sucesivas al óptimo a partir de un punto inicial. El método del conjunto activo responde a este criterio de optimización y ha sido ampliamente desarrollado en la literatura económica y matemática de los últimos años.

El problema de asignación de flujos de tráfico entre los diversos arcos de una red es uno de las cuatro etapas en las que se ha dividido tradicionalmente el proceso de gestión y control del transporte. El método de asignación empleado depende de las condiciones particulares que estemos considerando en nuestro análisis. En el caso concreto de que no exista una demanda de flujo variable y que no se establezcan cotas superiores al flujo asignado a cada arco, uno de los métodos más habituales de asignación se basa en el algoritmo planteado por Frank y Wolfe a mediados de la década de los años 50.

A lo largo de este estudio se ha procedido a la aplicación de un método del conjunto activo al problema de asignación óptima de la red, así como a la comparación de los resultados obtenidos con el método tradicional de Frank-Wolfe.

El algoritmo del conjunto activo converge y, por lo tanto, es un método válido para la realización del procedimiento de asignación de flujo en redes. Dadas las características especiales del problema -existencia de restricciones lineales de igualdad-, el conjunto activo, también denominado conjunto de trabajo, presenta un núcleo de restricciones estables que reduce de forma importante el coste operativo de mantenimiento del mismo. El proceso de actualización del conjunto activo, prácticamente, se reduce a la consideración de las situaciones en las cuales el flujo en un arco se hace nulo. Ante este hecho se pueden establecer dos vías de actuación:

- Replaneamiento del problema desde el punto inicial, eliminando la variable con valor cero de la red.
- Actualización del conjunto activo y matrices asociadas introduciendo la restricción de no negatividad pertinente dentro del conjunto. En este caso no es necesario volver a iniciar el problema y se puede continuar con el proceso iterativo.

La consideración de una línea de búsqueda basada en métodos quasi-newton se ve beneficiada por la no necesidad de cálculo de la Hessiana. La elección de la actualización BFGS pretende beneficiarse de las buenas propiedades de convergencia que presenta dicha metodología. Cuando el problema envuelve la utilización de matrices de gran densidad pueden surgir problemas propiciados por la factorización de matrices; no obstante, en problemas de redes esta actuación no presenta mayores dificultades, puesto que las matrices implicadas son, en general, no densas.

En redes grandes, el método del conjunto activo se muestra más eficiente que el método de Frank-Wolfe, puesto que se aproxima a una mayor velocidad al óptimo. Por otro lado, el mayor coste de almacenamiento que supone el empleo de matrices en el primer método, se compensa ampliamente por la necesidad de tener que realizar solamente una asignación del tipo "todo-o-nada" inicial, frente al método de Frank-Wolfe para el cual es necesaria una asignación de este tipo en cada una de las

iteraciones. La gran ventaja de considerar una asignación inicial con todos los arcos de la red con flujo positivo permite que no sea necesario el cálculo del camino más corto, o del árbol de expansión mínimo, frente a aquellos métodos que utilizan los procedimientos de asignación “todo-o-nada”.

Un análisis de la velocidad de convergencia en varios ejemplos, muestra que en las primeras etapas el método de Frank-Wolfe es muy rápido, pero que posteriormente se va reduciendo su velocidad de forma importante, aún encontrándose relativamente alejado de la solución óptima. En el método del conjunto activo, las primeras etapas presentan una aproximación al óptimo deficiente propiciada por el empleo de una línea de búsqueda quasi-newton, pero a lo largo de las diversas iteraciones mejora substancialmente la velocidad de convergencia y el grado de aproximación al óptimo es mucho mayor que el obtenido por el método anterior.

El método de Frank-Wolfe se mostrará más eficiente en redes sencillas, con pocos arcos; mientras el método del conjunto activo resulta mucho más eficiente para redes complejas con un número de arcos amplio y con elevados niveles de densidad de tráfico.

La implementación de ambos métodos en el programa de cálculo simbólico Mathematica, versión 2.1 para Windows, ha mostrado en varios ejemplos considerados que los resultados obtenidos por el método del conjunto activo son superiores, puesto que se alcanzan valores menores de la función objetivo en un número de iteraciones menor o similar. En general, todos los problemas considerado muestran que el método del conjunto activo proporciona valores óptimos de la función objetivo; en el caso del algoritmo de Frank-Wolfe, se detecta, que en las proximidades del óptimo, la aproximación al mismo se ralentiza de forma importante, con lo que el cumplimiento del criterio de convergencia impediría la continuidad del algoritmo.

Un pequeño problema que puede surgir en el algoritmo del conjunto activo es la aparición de un cierto fenómeno de zig-zag. En estas situaciones el valor de la función objetivo no siempre desciende al efectuar las iteraciones. En algunos casos, este proceso se puede evitar considerando otras posibilidades alternativas de consideración de la longitud de paso; en otras situaciones el fenómeno no es evitable, pero el aumento de iteraciones no afecta de forma negativa al algoritmo cuando el problema incluye un cierto número medio de variables.

La existencia de interacción asimétrica entre los costes de los arcos de la red y la existencia de peajes asociados a la utilización de los arcos han sido otros aspectos relacionados con el procedimiento de asignación de redes que se han considerado en este trabajo.

El establecimiento de peajes sobre la utilización de los arcos de la red se ha revelado como uno de los principales métodos para afrontar la congestión de redes de tráfico urbanas. El peaje actúa con un doble objetivo. Por un lado, regula el acceso al arco y evita que la congestión sea excesiva y afecte de forma considerable al funcionamiento del mismo; por otro lado, su capacidad recaudadora no debe ser menospreciada como fuente de ingresos para las autoridades. Aunque existen autores que se muestran opuestos al establecimiento de los mismos, la mayoría de los expertos considera que la única vía de salida al fuerte grado de congestión que se está produciendo en el tráfico urbano de numerosas ciudades pasa por el establecimiento de algún tipo de peaje.

La implantación práctica de peajes en diversas ciudades muestra unos resultados positivos, puesto que ha contribuido a la reducción de los niveles de congestión en dichas ciudades, al fomentar la utilización racional del vehículo privado y una mayor demanda de los medios de transporte público alternativos. En otras situaciones, donde el objetivo final del peaje era meramente fiscal, los resultados no han sido los esperados.

La teoría económica considera el cálculo del peaje óptimo en función de la diferencia existente entre el coste social y el coste privado asociado a la utilización de un arco de la red. El peaje óptimo se igualaría al coste marginal social de dicho arco. En este trabajo se presenta un peaje alternativo calculado en base a la diferencia entre el coste medio y el coste marginal. El establecimiento de dicho peaje se muestra más eficiente en el caso de arcos con un fuerte nivel de congestión.

Los resultados obtenidos a la hora de realizar los procesos de asignación empleando el método del conjunto activo, muestran que las recaudaciones obtenidas con el peaje alternativo son superiores a las obtenidas utilizando el peaje óptimo. Al mismo tiempo, el coste asociado al patrón de flujos de tráfico, derivado de la consideración de la función de coste y del peaje asociado a cada arco, es menor en el caso de utilización del peaje alternativo frente al peaje óptimo.

La consideración de la asimetría de costes también puede verse beneficiada por la utilización del conjunto activo como método de asignación, frente a los algoritmos de diagonalización tradicionales, especialmente desde el punto de vista de la reducción en el número de iteraciones necesarias para alcanzar el óptimo. En este aspecto nuestros estudios hasta el momento se han centrado en redes sencillas.

Podemos concluir que, la utilización del método del conjunto activo se ha revelado como un algoritmo válido a la hora de calcular el valor óptimo de una función no lineal sujeta a restricciones lineales. El mayor coste asociado al cálculo de la factorización de matrices se compensa ampliamente por el mayor coste de resolución asociado al algoritmo de Lagrange. La solución obtenida en ambos casos cumple las condiciones de optimalidad, y por tanto, ambos métodos resultan válidos para su utilización en la consideración de problemas económicos.

Los objetivos de este estudio no concluyen con el mismo, sino que nos permiten abrir puertas a futuras líneas de investigación que no han sido aquí abordadas, o que simplemente se han señalado pero no profundizado en las mismas.

Nuestras intenciones futuras consideran la relajación de los supuestos establecidos sobre las redes de tráfico consideradas; así pues, consideraríamos la existencia de demandas variables, dependientes del coste que soportan los usuarios; profundización en el análisis de la existencia de interacciones asimétricas entre arcos; establecimiento de cotas al flujo asignado a los distintos arcos de la red; estudio de las posibles implicaciones de la implantación del peaje, tanto sobre la demanda de tráfico, como sobre el establecimiento de patrones de flujo óptimos. En una etapa posterior, podríamos considerar un análisis estocástico así como una aproximación dinámica al análisis de redes. Todas estas vías, algunas de las cuales ya han sido parcialmente exploradas por otros autores, creemos que deben indicar el camino a seguir en futuras investigaciones.

BIBLIOGRAFIA

BIBLIOGRAFIA :

- [1] AGNEW, C. (1977), *The Theory of Congestion Tolls*, en Journal of Regional Science, 17, pp. 381-393.
- [2] AKCELIK, R. (1978), *A New Look at Davidson's Travel Time Function*, en Traffic Engineering and Control, 19, pp. 459-463.
- [3] ANAS, A. (1988), *Statistical Properties of Mathematical Programming Model of Stochastic Network Equilibrium*, en Journal of Regional Science, 28(4), pp. 511-530.
- [4] ARNOTT, R. & KRAUS, M. (1993), *The Ramsey Problem for Congestible Facilities*, en Journal of Public Economics, 50(3), pp. 371-396.
- [5] ARNOTT, R. (1990), *Signalized Intersection Queing Theory and Central Bussines District Auto Congestion*, en Economic Letters, 33(2), pp. 197-201.
- [6] ARNOTT, R.; DE PALMA, A. & LINDSEY, R. (1990), *Economics of a Bottleneck*, en Journal of Urban Economics, 27, pp. 111-130.
- [7] ARNOTT, R.; DE PALMA, A. & LINDSEY, R. (1991), *A Temporal and Spatial Equilibrium Analysis of Commuter Parking*, en Journal of Public Economics, 45(3), pp. 301-335.
- [8] ARNOTT, R.; DE PALMA, A. & LINDSEY, R. (1992), *Route Choice with Heterogeneous drivers and group-specific congestion costs*, en Regional Science and Urban Economics, 22(1), pp. 71-102.
- [9] ARNOTT, R.; DE PALMA, A. & LINDSEY, R. (1993), *A Structural Model of Peak-Period Congestion: A Traffic Bottleneck with Elastic Demand*, en American Economic Review, 83(1), pp. 161-179.
- [10] ASENSIO, J. (1995), *Determinación del Peaje Urbano Optimo y Evaluación de sus Efectos Redistributivos: El caso de Barcelona*, en V Congreso Nacional de Economía, Canarias.
- [11] BAAZARA, S.; HOKHTER, S & SHETTY, C. (1979), Nonlinear Programming. Theory and Algorithms, J. Wiley & Sons, New York.
- [12] BEAVIS, B. & DOBBS, I. (1991), Optimization and Stability Theory for Economic Analysis, Cambridge University Press, Cambridge.
- [13] BECKMAN, M., MCGUIRE, C.B. & WINSTEN, C.B. (1955), Studies in The Economics of Transportation. Yale University Press, New Haven.
- [14] BELTRAMI, E. (1987), Mathematics for Dynamic Modeling. Academic Press, London.
- [15] BELL, M.G.H. (1983), *The Estimation of Origin-Destination Flows and Their Confidence Intervals from Measurements of Link Volumes: A Computer Program*, en Traffic Engineering and Control, 24, pp. 202-205.
- [16] BERGLAS, E.; FRESKO, D. & PINES, D. (1984), *Right of Way and Congestion Toll*, en Journal of Transport Economics & Policy, 18(2), pp. 165-187.

- [17] BERTSIMAS, D. & VAN RYZIN, G. (1991), *A Stochastic and Dynamic Vehicle Routing Problem in the Euclidean Plane*, en Operations Research, 39(4), pp. 601-615.
- [18] BERTSIMAS, D. & VAN RYZIN, G. (1993), *Stochastic and Dynamic Vehicle Routing in the Euclidean Plane with Multiple Capacitated Vehicles*, en Operations Research, 41(1), pp. 60-76.
- [19] BIANCO, L. & TOTH, P. (1996), Advanced Methods in Transportation Analysis, Springer-Verlag, Berlin.
- [20] BIRGE, J. & HO, J. (1993), *Optimal Flows in Stochastic Dynamic Networks with Congestion*, en Operations Research, 41(1), pp. 203-217.
- [21] BLACK, I. (1975), Advanced Urban Transport, Saxon House Studies, Westmead, U.K.
- [22] BOARDMAN, A & LAVE, L. (1977), *Highway Congestion and Congestion Tolls*, en Journal of Urban Economics, 4(3), pp. 340-359.
- [23] BOYCE, D.; LEBLANC, L. & CHON, K. (1988), *Network Equilibrium Models of Urban Location and Travel Choices: A retrospective Survey*, en Journal of Regional Science, 28(2), pp. 159-183.
- [24] BRAEUTIGAM, R. (1979), *Optimal Pricing with Inter-Modal Competition*, en American Economic Review, 69(1), pp. 38-49.
- [25] BRAID, R. (1989), *Uniform versus Peak-Load Pricing Bottleneck with Elastic Demand*, en Journal of Urban Economics, 26(3), pp. 320-327.
- [26] BRANSTON, D. (1976), *Link Capacity Functions. A Review*, en Transportation Research, 10, pp. 223-236.
- [27] BUREAU OF PUBLIC ROADS (BPR) (1964), Traffic Assignment Manual, U.S. Department of Commerce, Washington DC.
- [28] BURNS, M. & FAUROT, D. (1992), *An Econometric Forecasting Model of Revenues from Urban Parking Facilities*, en Journal of Economics and Bussines, 44(2), pp. 143-150.
- [29] BUTTON, K.J. (1993), Transport Economics, Edward-Elgar, Aldershot, U.K.
- [30] CANDIDO CARBAJO, J. & DE RUS, G. (1990), *La Desregulación del Transporte*, en Papeles de Economía Española, 42, pp. 262-291.
- [31] CAREY, M. & SRINIVASAN, A. (1993), *Externalities, Average, and Marginal Costs and Tolls on Congested Networks with Time-Varying Flows*, en Operations Research, 43(1), pp. 217-231.
- [32] CAREY, M. (1987), *Optimal Time-Varying Flows on Congested Networks*, en Operations Research, 35(1), pp. 58-69.
- [33] CATANESE, A. (1972), New Perspectives in Urban Transportation Research, Lexington Books, Massachusetts.
- [34] COLEMAN, R.R. (1961), *A Study of Urban Travel Times in Pennsylvania Cities*, en Highway Research Board Bulletin, 303, pp. 62-75.
- [35] COWAN, P. & FINE, D. (1972), *Sobre el número de enlaces de uns sistema*, en AA.VV. , Modelos de Análisis Territorial, Oikos-Tau, Barcelona.

- [36] CULLINANE, K. (1990), *An Aggregate Dynamic Model of the Parking Compliance Decision*, en International Journal of Transport Econ., 20(1), pp. 27-50
- [37] CHUNG-LUN, L., SIMCHI-LEVI, D. & DESROCHERS, M. (1992), *On the Distance Constrained Vehicle Routing Problem*, en Operations Research, 40(4), pp. 790-799.
- [38] D'OUVILLE, E. & McDONALD, J. (1990), *Optimal Road Capacity with a Suboptimal Congestion Toll*, en Journal of Urban Economics, 28, pp. 34-49.
- [39] DAFERMOS, S. (1971), *An Extended Traffic Assignment Model with Application to Two-Way Traffic*, en Transportation Science, 5, pp. 366-389.
- [40] DAFERMOS, S. (1982), *Relaxation Algorithms for the General Asymmetric Traffic Equilibrium Problem*, en Transportation Science, 16, pp. 231-240.
- [41] DAGANZO, C. & SHEFFI, Y. (1977), *On Stochastic Models of Traffic Assignment*, en Transportation Science, 11, pp. 253-274.
- [42] DAI, J.; NGUYEN, V. & REIMAN, M. (1994), *Special Bottleneck Decomposition: An Approximation Method For Generalized Jackson Networks*, en Operations Research, 42(1), pp. 119-136.
- [43] DANTZIG, G.B. (1963), Linear Programming and Extensions, Princenton University Press, Princenton, N.J.
- [44] DAVIDSON, (1966), *A Flow Travel Time Relationship for Use in Transportation Planning*, en Proceedings of the Australian Road Research Board, 3, pp. 183-194.
- [45] DE PALMA, A. & ARNOTT, R. (1986), *Usage-dependent Peak-Load Pricing*, en Economic Letters, 20, pp. 101-105.
- [46] DE RUS, G. (1987), *Discriminación de Precios y Subvenciones Cruzadas en el Transporte Público*, en Investigaciones Economicas, 11(2), pp. 201-218.
- [47] DE RUS, G. (1989), *El Transporte Público Urbano en España*, en Investigaciones Economicas, 12(2), pp. 207-226.
- [48] DE RUS, G. (1992), *Las empresas Públicas de Transporte en España*, en Papeles De Economía Española, 38, pp. 349-382.
- [49] DESERPA, A. (1971), *A Theory of the Economics of the Time*, en Economic Journal, 81, pp. 828-846.
- [50] DESERPA, A. (1978), *Congestion, Pollution and Impure Public Goods*, en Public Finance, 33(1), pp. 68-81.
- [51] DEWEES, D. (1979), *Estimating the Time Cost of Highway Congestion*, en Econometrica, 47(6), pp. 1499-1512.
- [52] DIMITROU, H. (1992), Urban Transport Planning: A Development Approach, Routledge, London.
- [53] DOWNS, A. (1962), *The Law of Peak-hour Expressway Congestion*, en Traffic Quaterly, 16, pp. 393-409.

- [54] DRINKA, T. & PRESCOTT (1981), *Optimal Roadway Tolls and Operating Cost in Urbanized Regions*, en Regional Science Perspectives, 11(2), pp. 115-128.
- [55] DURAN, M. (1991), *El Tiempo en la Economía Española*, en Boletín Económico Del Ice, 695, pp. 9-48.
- [56] ECMT (1996), Round Table 97: Charging for the use of urban roads, OCDE, Paris.
- [57] EVANS, A. (1992), *Road Congestion Policy: Whe is it a Good Policy?*, en Journal of Transport Economics & Policy, 26(3), pp. 213-243.
- [58] EVANS, A. (1992), *Road Congestion: The Diagrammatic Analysis*, en Journal of Political Economy, 100-4, pp. 211-217.
- [59] EVANS, A. (1993), *Road Congestion Policy: When is it a Good Policy? A Rejoinder*, en Journal of Transport Economics & Policy, 27(1), pp. 99-105.
- [60] EVANS, J. & MINIEKA, E. (1992), Optimization Algorithms for Networks and Graphs, Marcel Dekker, Inc. New York.
- [61] FIELD, B. (1992), *Road Pricing in Practice*, en Transportation Journal, 32(1), pp. 5-14.
- [62] FITZROY, F. & SMITH, I. (1993), *Priority over Pricing: Lesson from Zurich on the Redundancy of Road Pricing*, en Journal of Transport Economics & Policy, 27(2), pp. 209-214.
- [63] FLETCHER, R. (1987), Practical Methods of Optimization, J. Wiley & Sons, New York.
- [64] FLORIAN, M. & SPIESS, H. (1982), *The Convergence of Diagonalization Algorithms for Asymmetric Network Equilibrium Problems*, en Transportation Research, 16B, pp. 477-483.
- [65] GARTNER, N.H. (1980), *Optimal Traffic Assignment with Elastic Demands: A Review; Part II: Algorithmic Approaches*, en Transportation Science, 14, pp. 192-208.
- [66] GARWOOD, F. (1940), *An Application of the Theory of Probability to the Operation of Vehicular-Controlled Traffic Signals*, en Journal of The Royal Statistical Society, VII-1, pp. 65-77.
- [67] GILL, P.E. & MURRAY, W. (1972), *Quasi-Newton methods for unconstrained optimization*, en Journal of Inst. Maths.Applics., 9, pp. 91-108.
- [68] GILL, P. MURRAY, W. & WRIGHT, M. (1981), Practical Optimization, Academic Press, New York.
- [69] GILL, P. MURRAY, W. & WRIGHT, M. (1991), Numerical Linear Algebra and Optimization, Addison-Wesley, New York.
- [70] GILLEN, D. (1977), *Estimation and Specification of the Effects of Parking Cost on Urban Transport Mode Choice*, en Journal of Urban Economics, 4(2), pp. 186-199.
- [71] GLAISTER, S. & CALLINGS, J. (1978), *Maximization of Passanger Miles in Theory and Practice*, en Journal of Transport Economics & Policy, 12(3), pp. 304-321.
- [72] GLAZER, A. & NISKANEN, E. (1992), *Parking fees and Congestion*, en Regional Science and Urban Economics, 22(1), pp. 123-132.

- [73] GLAZER, A. (1981), *Congestion Tolls and Consumer Welfare*, en Public Finance, 36(1), pp. 77-83.
- [74] GOMEZ-IBÁÑEZ, J.A. & MEYER, J.R. (1993), Going Private, The International Experience with Transport Privatization, Brookings Institution, Washington D.C.
- [75] GONZALEZ BLANCH, F. , GASCA ALLUE, C. & IBARZ DEL OLMO, C. (1984), Implantación en España de un sistema de Tarificación por el uso de la Infraestructura viaria, Instituto de Estudios Fiscales, Madrid.
- [76] GRONAU, R. (1994), *Optimal Road Capacity with Suboptimal Congestion Toll*, en Journal of Urban Economics, 36(1), pp. 1-7.
- [77] HASS-KLAU, C. (1990), The Pedestrian and the City, Belhaven, London.
- [78] HAYASHI, Y. (1996), Transport, Land-Use and The Environment, Kluwer Academic Publishers, Dordrecht, Holanda.
- [79] HEARN, D.W. & RIBERA, J. (1980), *Bounded Flow Equilibrium Problems by Penalty Methods*, en Proceedings of the 1980 IEEE International Conference on Circuits and Computers.
- [80] HENSHER, D. & JOHNSON, L. (1981), *Behavioural Response and Form the Representative Component of the Indirect Function in Travel Choice Model*, en Regional Science and Urban Economics, 11(4), pp. 559-572.
- [81] HENSHER, D.; MILTHORPE, F. & SMITH, N. (1990), *The Demand for Vehicle Use in the Urban Household Sector: Theory and Empirical Evidence*, en Journal of Transport Economics & Policy, 24(2), pp. 119-137.
- [82] HERMAN, R. (1991), *Traffic Dynamics through Human Interaction: Reflections on Some Complex Problem*, en Journal of Economic Behaviour and Organ., 15(2), pp. 303-311.
- [83] HEWITT, J. (1964), *The Calculation of Congestion Taxes on Roads*, en Economica, 31(1), pp. 72-81.
- [84] HILL, A. & BENTON, W. (1992), *Modelling Intra-City Time-Dependent Travel Speeds for Vehicle Scheduling Problems*, en Journal of the Operation Research Societ, 43(4), pp. 343-351.
- [85] HILLS, P. (1993), *Road Congestin Pricing: When is it a Good Policy?*, en Journal of Transport Economics & Policy, 27(1), pp. 91-99.
- [86] HOLDEN, D. (1989), *Wardrop's Third Principle: Urban Traffic Congestion and Traffic Policy*, en Journal of Transport Economics & Policy, 23(3), pp. 239-262.
- [87] ICO, Jornadas de Financiación de Infraestructuras de Transporte, Ministerio de Obras Públicas, Transporte y Medio Ambiente, Madrid.
- [88] INMAN, R. (1978), *A Generalized Congestion Function for Highway Travel*, en Journal of Urban Economics, 5(1), pp. 21-34.
- [89] INSTITUTION OF HIGHWAYS AND TRANSPORTATION (IHT) (1992), *Traffic Congestion in Urban Areas*, Paper on Current Policy Issues, IHT, September.

- [90] IRVING, J.M.; OAKLEY, C.F. & RAMSEY, J.B.H. (1986) *The Updating of an O-D Matrix: A Maximun Likelihood Approach*, en *Traffic Engeneering and Control*, 27 (9), pp. 442-446.
- [91] JARA-DIAZ, S. (1982), *The Estimation of Transport Cost Functions*, en *Transport Reviews*, 2, pp. 257-278.
- [92] JARA-DIAZ, S. (1990), *Consumer's Surplus and The Value of Travel Time Savings*, en *Transportation Research*, 24B, pp. 73-77.
- [93] JAYARKRISHNAN, R.; TSAI, W.; PRASHKER, J. & RAJADHAYAKSA, S. (1994) *Faster Path-Based Algorithm for Traffic Assignment*, en *Trasnportation Research Record, Travel Demand Modeling and Network Assignment Models*, National Research Council, Washington, D.C., pp. 75-83.
- [94] JOHNSON, M. (1964), *On the Economics of Road Congestion*, en *Econometrica*, 32, pp. 137-150.
- [95] JONES, I. (1977), *Urban Transport Appraisal*, MacMillan Press, London.
- [96] KAIN, J. (1964), *A Contribution the Urban Transportation Debate: An Econometric Model of Urban Residential and Travel Behavior*, en *The Review of Economics and Statistics*, 46, pp. 55-64.
- [97] KALL, P. (1992), *System Modelling and Optimization*, Springer-Verlag, Berlin.
- [98] KEELER, T. & SMALL, K. (1977), *Optimal Peak-load Pricing, Investment and Service Levels on Urban Expressway*, en *Journal of Political Economy*, 85, pp. 1-25.
- [99] KELLY, F.; ZACHARY, S. & ZIEDINS, I. (1996), *Stochastic Networks: Theory and Applications*, Oxford U. Press, Oxford.
- [100] KIM, T. (1986), *Modeling the Density Variations of Urban Land Uses with Transportation Network Congestion*, en *Journal of Urban Economics*, 19(3), pp. 264-276.
- [101] KNIGHT, F.H. (1924). *Some fallacies in the interpretation of social cost*, en *Quarterly Journal of Economics* 38. pp. 582-606.
- [102] KRAUS, M. (1981). *Scale Economies Analysis for Urban Highway Networks*, en *Journal of Urban Economics*, 9, pp. 1-22.
- [103] KRAUS, M.; MOHRING, H. & PINFOLD, T. (1976), *The Welfare Costs of Nonoptimum Pricing and Investment Policies for Freeway Transportation*, en *American Economic Review*, 66(4), pp. 532-547.
- [104] LABBE, M.; LAPORTE, G. & MERCURE, H. (1991), *Capacitated Vehicle Routing on Trees*, en *Operations Research*, 39(4), pp. 616-622.
- [105] LAVE, C. (1970). *The Demand for Urban Mass Trnsait*, en *The Review of Economics and Statistics*, 52, pp. 320-323.
- [106] MAEDER, R. (1991). *Programming in Mathematica*, Addison-Wesley, Redwood City, Ca.
- [107] MADDEN, P. (1986), *Concavidad y Optimización en Microeconomía*, Alianza Editorial, Madrid.

- [108] MAHER, M.J. (1983), *Inferences on Trip Matrices form Observations on Link Volumes*, en Transportation Research, 17B, pp. 435-447.
- [109] MARSHALL, A. (1890), Principles of Economics, McMillan, London.
- [110] MATAS, A. (1991), *La Demanda de Transporte Urbano, un Análisis de las Elasticidades y Valoraciones del Tiempo*, en Investigaciones Economicas, 25(2), pp. 249-267.
- [111] MATSTOMS, P. (1994), *Sparse QR Factorization in MATLAB*, en ACM Transactions on Mathematical Software, 20, pp. 136-159.
- [112] McCARTHY, P. (1984), *Automobile Captive Choice Behavior - An Application of Nested Logit Analysis*, en Logistics and Transportation Review, 20(2), pp. 127-148.
- [113] McCONNELL, V. (1986), *Automobile Use and Locational Interdependencies*, en Journal of Regional Science, 26(3), pp. 475-498.
- [114] McCORMICK, G. (1983), Nonlinear Programming. Theory, Algorithms and Applications, J. Wiley & Sons, New York.
- [115] McFADDEN, D. (1974), *The Measurement of Urban Travel Demand*, en Journal of Public Economics, 3, pp. 303-328.
- [116] McKEE, M. & WEST, E. (1987), *Further Perspectives on the Theory of Second-Best*, en Public Finance, 42(1), pp. 146-151.
- [117] MERCHANT, D. & NEMHAUSER, G. (1978), *Optimality Conditions for a Dynamic Traffic Assignment Model*, en Transportation Science, 12, pp. 200-207.
- [118] MERCHANT, D. & NEMHAUSER, G. (1978), *A Model and an Algorithm for the Dynamic Traffic Assignment Problem*, en Transportation Science, 12, pp. 183-199.
- [119] MEYER, J.R. & GOMEZ-IBÁÑEZ, J. A. (1981), Autos, Transit and Cities, Harvard University Press, Cambridge, Mass.
- [120] MEYER, J.R., KAIN, J.F. & WOHL, M. (1965). The Urban Transportation Problem. Harvard University Press, Cambridge, Mass.
- [121] MINISTRY OF TRANSPORT (1964), Road Pricing: The economic and technical possibilities (the Smeed Report), HMSO, London.
- [122] MOGRIDGE (1990), Travel in Towns, MacMillan Press, London.
- [123] MOHRING, H. & HARWITZ, M. (1962), Highway Benefits: Ana Analytical Framework. Northwestern University Press, Illinois.
- [124] MOHRING, H. (1962), *Optimization and Scale Economics in Urban Bus Transportation*, en American Economic Review, 62(3), pp. 591-604.
- [125] MOHRING, H. (1970), *The Peak-Load Problem with Increasing Returns and Pricing Constraints*, en American Economic Review, 60(4), pp. 693-705.
- [126] MOHRING, H. (1985), *Profit Maximization, Cost Minimization and Pricing for congestion-prone Facilities*, en Logistics and Transportation Review, 21(1), pp. 27-36.
- [127] MOHRING, H. (1994), The Economics of Transport, Edward-Elgard, Aldershot, U.K.

- [128] MORRISON, S. (1986), *A Survey of Road Pricing*, en Transportation Research, 20A, pp. 89-97.
- [129] MUMY, G. & NISKANEN, E. (1993), *The Impact of Distributional Objectives on the Toll and Capacity of a Congestible Facility*, en Journal of Urban Economics, 34(3), pp. 401-413.
- [130] MUÑOZ, M. (1989), *La Adaptación del Programa Europeo DRIVE a la Situación Española*, en Economía Industrial, 268, pp. 67-74.
- [131] NAGURNEY, A. (1986), *Computational Comparisons of Algorithms for General Asymmetric Traffic Equilibrium Problems with Fixed and Elastic Demand*, en Transportation Research, 20B, pp. 78-84.
- [132] NAGURNEY, A. (1993), Network Economics, Kluwer Academic Publishers, Dordrecht, Holanda.
- [133] NASH, S. & SOFER, A. (1996), Linear and Nonlinear Programming, McGraw-Hill, New York.
- [134] NATIONAL ECONOMIC DEVELOPMENT OFFICE (NEDO) (1991), A Road User Charge? Londoner's View. Informe preparado por Harris Research Centre para NEDO, The London Planning Advisory Committee and the Automobile Association, London.
- [135] NEMHAUSER, RINOOY, & TODD (1989), Optimization, North-Holland, Amsterdam.
- [136] NEWBERY, D. (1988), *Road User Charges in Britain*, en The Economic Journal, 98, pp. 161-176.
- [137] NEWBERY, D. (1990), *Pricing and Congestion: Economic Principles Relevant...*, en Oxford Review of Economic Policy, 6, pp. 22-38.
- [138] NEWBERY, D. (1988), *Road Damage Externalities and Road User Charges*, en Econometrica, 56(2), pp. 295-316.
- [139] NEWELL, G. (1980), Traffic Flow on Transportation Networks, MIT Press, Cambridge, Mass.
- [140] NG, Y. (1987), *Political Distortions and the Relevance of Second and Third-Best Theories*, en Public Finance, 42(1), pp. 137-145.
- [141] NGUYEN, S. & DUPUIS, C. (1984). *An Efficient Method for Computing Traffic Equilibria in Network with Asymmetric Transportation Costs*, Transportation Science, 18, pp. 185-202.
- [142] NOWLAN, D. (1993), *Optimal Pricing of Urban Trips with Budget Restrictions and Distributional Concerns*, en Journal of Transport Economics & Policy, 27(3), pp. 253-276.
- [143] PEDREIRA ANDRADE, L. & SEIJAS MACIAS, J. (1996), *Método del Conjunto Activo: Una primera aproximación a su aplicación al estudio del peaje óptimo*, Documento de Trabajo 1996/2, Facultad de Ciencias Económicas. Universidad de A Coruña.
- [144] PEDREIRA ANDRADE, L. & SEIJAS MACIAS, J. (1997), *Una propuesta de algoritmo del conjunto activo aplicada al análisis de redes con interacción asimétrica*. Trabajo presentado a la "3rd Conference on Operations Research" celebrada en La Habana (Cuba) entre del 10 al 15 de Marzo de 1997.
- [145] PIERRE, D. (1969), Optimization Theory with Applications, Dover, New York.

- [146] PIGOU, A.C. (1920), Wealth and Welfare, McMillan, London.
- [147] POLAK, J. & HEERTJE, A. (1993), European Transport Economics, Basil Blackwell, Oxford.
- [148] RAN, B. ; BOYCE, D. & LEBLANC, L. (1993), *A New Class of Instantaneous Dynamic User-Optimal Traffic Assignment Models*, en Operations Research, 41(1), pp. 192-202.
- [149] RAN, B. & BOYCE, D. (1996), Modeling Dynamic Transportation Networks, Springer-Verlag, Berlin.
- [150] RIBAS, E. & MONTLLOR, J. (1992), *Autopistas de Peaje: El Modelo Español*, en Papeles De Economía Española, 38, pp. 409-432.
- [151] SALTER, R. & HOUNSELL, N. (1996), Highway Traffic Analysis and Design, MacMillan Press, London.
- [152] SHAH, A. (1990), *Optimal Pricing of Traffic Externalities: Theory and Measurement*, en International Journal of Transport Econ., 17(1), pp. 3-19.
- [153] SHEFFI, Y. (1985), Urban Transportation Networks, Prentice-Hall, Englewood Cliffs, N.J.
- [154] SHEPPARD, S. & STOWER, M. (1995), *The Benefits of Transport Improvements in a City with Efficient Development Control*, en Regional Science and Urban Economics, 25(2), pp. 211-222.
- [155] SHERMAN, R. (1971), *Congestion Interdependence and Urban Transit Fares*, en Econometrica, 39, pp. 565-576.
- [156] SMALL, K. (1982), *The Scheduling of Consumer Activities: Work Trips*, en American Economic Review, 72(3), pp. 467-479.
- [157] SMALL, K. (1983), *The Incidence of Congestion Tolls on Urban Highways*, en Journal of Urban Economics, 13(1), pp. 90-111.
- [158] SMALL, K. (1992), Urban Transportation Economics, Harwood Academic, Chur, Switzerland.
- [159] SMALL, K. & WINSTON, C. (1988), *Optimal Highway Durability*, en American Economic Review, 78(3), pp. 560-569.
- [160] SMALL, K.; WINSTON, C. & EVANS, C. (1989), Road Work: A New Highway Pricing and Investment Policy, Brookings Institution, Washington.
- [161] SMEED, R. (1968), *Traffic studies and Urban Congestion*, en Journal of Transport Economics & Policy, 2, pp. 33-70.
- [162] STERN, E.; TZALGOV, J. & HENIK, A. (1983), *Driving Efforts and Urban Route Choice*, en Logistics and Transportation Review, 19(1), pp. 67-79.
- [163] STWEART, G.W. (1973), Introduction to Matrix Computations, Academic Press, London.
- [164] SULLIVAN, A. (1983), *Second-best Policies for Congestion Externalities*, en Journal of Urban Economics, 14, pp. 105-123.
- [165] SUNDARAM, R. (1996), A First Course in Optimization Theory, Cambridge University Press, Cambridge.

- [166] TABUCHI, T. (1993), *Bottleneck Congestion and Modal Split*, en Journal of Urban Economics, 34(3), pp. 414-431.
- [167] TAYLOR, M.; YOUNG, W. & BONSALE, P. (1996), Understanding Traffic Systems, Avebury Press, Aldershot, U.K.
- [168] THOBANI, M. (1984), *A Nested Logit Model of Travel Mode to Work and Auto Ownership*, en Journal of Urban Economics, 15(3), pp. 287-301.
- [169] THOMAS, R. (1991), Traffic Assignment Techniques, Avebury Press, Aldershot, U.K.
- [170] TISATO, P. (1992), *User Cost Minimisation and Transport Subsidy*, en Economic Letters, 39(2), pp. 241-247.
- [171] TOINT, P. & TUYTTENS, D. (1990), *On Large Scale Nonlinear Network Optimization*, en Mathematical Programming, 48, pp. 125-159.
- [172] ULENGIN, F. (1994), *Easing the Traffic in Istanbul: at What Price?*, en Journal of the Operation Research Societ, 45(7), pp. 771-785.
- [173] VAN VLIET, D. (1976), *Road Assignment III: Comparative Tests of Stochastic Methods* en Transportation Research, 10, pp. 151-157.
- [174] VAN ZUYLEN, H.J. & WILLUMSEN, L.G. (1980), *The Most Likely Trip Matrix Estimated form Traffic Counts*, en Transportation Research, 14B, pp. 281-293.
- [175] VERGES, J. (1989), *Medición de la Diferencia en Cuanto a Costes Sociales por Congestión entre el Transporte Individual y T.P.U.*, en Investigaciones Economicas, 23(2), pp. 183-205.
- [176] VICKERMAN, R. (1994), *Transport and Spatial Development in Europe*, en Journal of Common Market Studies, 32(2), pp. 249-256.
- [177] VICKREY, W. (1955), *Some Implications of Marginal Cost Price From Public Utilities*, en American Economic Review, 45(3), pp. 605-620.
- [178] VICKERY, W. (1959), *The Economizing of Curb Parking Space*, en Traffic Engineering, 25, pp. 62-67.
- [179] VICKREY, W. (1963), *Pricing in Urban and Surban Transport*, en American Economic Review, 53(2), pp. 452-465.
- [180] VICKREY, W. (1969), *Congestion Theory and Transport Investment*, en American Economic Review, 59(2), pp. 251-261.
- [181] VICKREY, W. (1971), *Maximum Output or Maximum Welfare? More on the off-Peak Pricing Problem*, en Kyklos, 24(2), pp. 305-330.
- [182] VICKREY, W. (1994), *Reaching an Economic Balance Between Mass Transit and Provision for Individual Automobile Traffic*, en Logistics and Transportation Review, , pp. 3-19.
- [183] WALTERS, A. (1961), *The Theory and Measurement of Private and Social Cost of Highway Congestion*, en Econometrica, 29, pp. 676-699.
- [184] WATERS, C. (1989), *Vehicle-scheduling Problems with Uncertainty and Omitted Customers*, en Journal of the Operation Research Societ, 40-12, pp. 1099-1108.

- [185] WEBSTER, F. (1986), *Transport in Towns: Some of Options*, en Journal of Transport Economics & Policy, 20(2), pp. 129-152.
- [186] WESTIN, R. & GILLEN, D. (1978), *Parking Location and Transist Demand*, en Journal of Econometrics, 8, pp. 75-101.
- [187] WESTLAND, J. (1992), *Congestion and Network Externalities in the Short Run Pricing of Information System Services*, en Management Science, 38(7), pp. 992-1009.
- [188] WHEATON, W. (1978), *Priced-induced Distortions in Urban Highways Investment*, en The Bell Journal of Economics, 9, pp. 622-632.
- [189] WHITELEGG, J. (1992), Traffic Congestion: Is There a Way Out?, Hawes, Leading Edge, U.K.
- [190] WILSON, J. (1983), *Optimal Road Capacity in the Precesenc of Unpriced Congestion*, en Journal of Urban Economics, 13(3), pp. 337-357.
- [191] WILSON, P. (1989), *Scheduling Costs and the Value of Travel Time*, en Urban Studies, 26(3), pp. 356-366.
- [192] WILLIAMS, H. & YAMASHITA, Y. (1992), *Travel Demand Forecasts and The Evaluation of Highway Schemes under Congested Conditions*, en Journal of Transport Economics & Policy, 26(3), pp. 261-282.
- [193] WILLIAMSON, O. (1966), *Peak load Pricing and Optimal Capacity under Indivisibility Constraints*, en American Economic Review, 56(3), pp. 810-827.
- [194] WINSTON, C. (1985), *Conceptual Developments in the Economics of Transportation: An Interpretive Survey*, en Journal of Economic Literature, 23(1), pp. 57-94.
- [195] WOLFRAM, S. (1991), Mathematica, Addison-Wesley, Redwood City, Ca.
- [196] WU, C. (1993), Solving Large-Scale Nonlinear Network Problems with Relaxation and Decomposition Algorithms, Tesis Doctoral, Pennsilvanya State University.
- [197] YANG.H.; YAGAR, S. & IIDA, Y. (1994), *Traffic Assignment in a Congested Discrete-Continuous Transportation System*, en Transportation Research, 28B, pp. 161-174.
- [198] YINGER, J. (1993), *Bumper to Bumper: A New Approach to Congestion in an Urban Model*, en Journal of Urban Economics, 34(2), pp. 249-274.

APENDICES

APENDICE A:

FACTORIZACION DE

MATRICES

APENDICE A: FACTORIZACION DE MATRICES

La factorización de matrices es una técnica aplicable a matrices que se considera especialmente útil en la resolución de sistemas de ecuaciones lineales. Dado un sistema lineal $Ax = b$, donde A es una matriz invertible, y x y b son dos vectores apropiados; la solución de este sistema no se calcula a través del producto $A^{-1}b$; con el objetivo de evitarnos el cálculo de la matriz inversa de A , lo cual puede ser realmente complejo si A es una matriz de un cierto orden, y dado que calcular la matriz A^{-1} es equivalente a resolver n sistemas lineales (donde n es el orden de la matriz A), entonces podemos reemplazar la solución de un sistema lineal por la de n sistemas lineales.

Si además A resulta ser una matriz triangular (ya sea superior o inferior) entonces la solución de un sistema lineal $Ax = b$ es inmediata.

Sea A una matriz triangular de orden $n \times n$, y sea b un vector de orden n y x un vector de orden n . La resolución del sistema $Ax=b$ se puede obtener de como la resolución de n ecuaciones lineales dadas por:

$$\begin{aligned} a_{11}x_1 + \dots + a_{1n}x_n &= b_1 \\ \dots &= \dots \\ a_{n-1,n-1}x_{n-1} + a_{n-1n}x_n &= b_{n-1} \\ a_{nn}x_n &= b_n \end{aligned} \tag{1}$$

El sistema (1) es resoluble en el caso de que el determinante de A sea distinto de 0. Si esto es así entonces calculamos $x_n = b_n / a_{nn}$, despejando en la última ecuación, y

mediante la denominada sustitución hacia atrás podemos hallar el resto de incógnitas del sistema.

Vemos, pues, que si podemos convertir las matrices, en general, a matrices triangular se pueden derivar substanciales beneficios a la hora de resolver los sistemas de ecuaciones lineales. Es necesario, entonces, dotarnos de alguna técnica que nos permita convertir una matriz cualquier A (de cualquier orden $n \times m$) en una matriz triangular¹. Esto es lo que se busca a través de la denominada factorización de matrices.

Pasamos ahora, a ver algunos de los métodos de factorización más habituales.

1. Eliminación Gaussiana.

La eliminación Gaussiana es un método general para la resolución de sistemas lineales $Ax = b$, donde A es una matriz invertible.

El método consta de tres pasos:

1. *Eliminación.* Consiste en determinar una matriz invertible M , tal que MA sea una matriz triangular superior.
2. Cálculo del vector Mb .
3. Resolución del sistema lineal:

$$MAX = Mb \quad (2)$$

mediante el método de resolución hacia atrás mostrado en (1).

¹ Si la matriz no es cuadrada entonces obtenemos una matriz triangular de orden $n \times n$ y una matriz de ceros de orden $n-m \times m$, si $m > n$.

La factorización propiamente dicha de la matriz se produce en la fase de cálculo de la matriz M, lo que es el primer estadio del proceso, conocido como eliminación, y en su posterior multiplicación por la matriz A. En la práctica, no obstante, el proceso de cálculo de la matriz M no se produce de una forma explícita, puesto que se procede mediante la permutación de filas de la matriz A, original, para la obtención de la matriz triangular superior. Analicemos más pormenorizadamente el proceso:

Dado que existe al menos un elemento de la primera columna² de A que es distinto de 0, elegimos este elemento al que denominaremos *pivote*.

A continuación permutamos la fila del pivote y la primera fila de la matriz A. Esto supone premultiplicar la matriz A por un matriz de permutación P. Esta matriz de permutación se construye a partir de la matriz de transposición T, y que para la i-ésima y j-ésima filas resulta:

$$T(i,j) = \begin{pmatrix} & i & j \\ 1 & & \\ & 0 & 1 \\ & & 1 \\ & 1 & 0 \\ & & & j \\ & & & & 1 \end{pmatrix} \quad (3)$$

² Si no existiese A sería singular y por lo tanto, no invertible lo cual supone una contradicción con los supuestos de partida.

Entonces la matriz de permutación P resulta³:

$$P = \begin{cases} I & \text{si } \alpha_{11} \text{ es el pivote} \\ T(1,j) & \text{en otro caso} \end{cases} \quad (4)$$

y la matriz PA = (α_{ij}) satisface que $\alpha_{11} \neq 0$.

Utilizando las combinaciones lineales apropiadas de la primera y del resto de filas de PA, reduciremos a cero todos los elementos por debajo de la diagonal en la primera columna de PA. Esto se puede escribir en notación matricial como premultiplicar la matriz PA por la matriz E, donde:

$$E = \begin{pmatrix} 1 & & & & \\ -\pi^{-1}\alpha_{21} & 1 & & & \\ \pi^{-1}\alpha_{31} & & 1 & & \\ \pi^{-1}\alpha_{n1} & & & 1 & \end{pmatrix} \quad (5)$$

donde $\pi = \alpha_{11}$.

Así pues, la matriz $A^2 = EPA$ resulta de la forma:

$$A^2 = \begin{pmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & \dots & \alpha_{1n} \\ & b_{22} & b_{23} & \dots & b_{2n} \\ & b_{32} & b_{33} & \dots & b_{3n} \\ & \dots & \dots & \dots & \dots \\ & b_{n2} & b_{n3} & \dots & b_{nn} \end{pmatrix} \quad (6)$$

En la matriz A^2 todos los elementos por debajo del primero en la primera columna son cero.

³ I representa la matriz identidad.

Si ahora operamos sobre A^2 podemos obtener las pertinentes matrices E_2 y P_2 que nos transforman en A^3 donde la primera y segunda columna tienen todos sus elementos por debajo de la diagonal nulos.

De forma sucesiva podemos establecer la matriz M como:

$$M = E_{n-1}P_{n-1}\dots E_2P_2E_1P_1 \quad (7)$$

tal que MA es una matriz triangular superior.

Hemos factorizado la matriz A , de forma tal que podemos trabajar con ella de un modo mucho más eficiente

Teorema A.1. Sea A una matriz cuadrada, invertible o no. Existe (al menos) una matriz invertible M tal que la matriz MA es triangular superior.

Demostración:

En el caso de que la matriz sea invertible, el proceso de eliminación gaussiana muestra la existencia de dicha matriz. Para el caso de que la matriz A no sea invertible, entonces dicha matriz será singular si y sólo si los elementos a_{ik}^k , $k \leq i \leq n$, de la matriz A^k son cero para como mínimo un valor de k . Pero entonces, la matriz A^k ya está en la forma de A^{k+1} , y por tanto, es suficiente con establecer $P_k = E_k = I$.

□

Este método puede presentar problemas serios de redondeo de errores si no se adopta una estrategia adecuada a la hora de la elección del pivote. En la práctica existen dos tipos de estrategias para dicha elección:

- a) Pivote parcial. El pivote es cualquier de los elementos a_{ik}^k , $k \leq i \leq n$, tal que satisfaga:

$$|a_{ik}^i| = \max_{k \leq p \leq n} |a_{pk}^k|$$

- b) Pivote completo. El pivote es cualquier de los elementos a_{ij}^k , $k \leq i, j \leq n$, tal que satisfaga:

$$|a_{ij}^i| = \max_{k \leq p, q \leq n} |a_{pk}^k|$$

En esta segunda estrategia si el elemento pivote elegido no pertenece a la k -ésima columna, entonces será necesario llevar a cabo una recolocación de columnas (además de la de filas).

Aunque una estrategia en la elección del pivote es siempre necesaria cuando se trabaja con matrices; existen ciertos caso en los cuales no se concreta ninguna ventaja de este tipo de medidas; esto es se produce en particular en el caso de matrices simétricas y definidas positivas.

Por último comentaremos el número de operaciones simples necesarias para realizar la factorización mediante el método de eliminación gaussiana. No se incluyen las operaciones implicadas en las segunda y tercera etapa del método. Para calcular la matriz A^{k+1} a partir de la matriz A^k , $1 \leq k \leq n-1$, necesitamos realizar un total de $n-k$ divisiones y $(n-k+1)(n-k)$ sumas e igual número de multiplicaciones. El total de operaciones para una matriz de orden n resulta:

$$\text{Total de operaciones} = \begin{cases} \frac{n^3 - n}{3} \text{ sumas} \\ \frac{n^3 - n}{3} \text{ multiplicaciones} \\ \frac{n(n-1)}{2} \text{ divisiones} \end{cases} \quad (8)$$

Por ejemplo, para $n=10$, el total de operaciones simples es de 705, 330 sumas y 330 multiplicaciones y 45 divisiones.

2. Factorización LU.

La factorización LU es muy parecida a la factorización realizada en la eliminación Gaussiana. La principal diferencia es que en esta factorización no se sigue ningún tipo de estrategia con respecto a la eliminación del pivote. Si el elemento diagonal de la primera fila A es distinto de cero, se elige como pivote, y de igual modo el resto.

Las matrices de permutación resultan iguales a la matriz identidad, $P_1 = P_2 = \dots = P_{n-1} = I$. Con lo cual, la matriz $M = E_{n-1} \dots E_2 E_1$. Definimos entonces una matriz triangular inferior:

$$L \stackrel{def}{=} M^{-1} = (E_{n-1} \dots E_2 E_1)^{-1} \quad (9)$$

y, por tanto, la matriz A se puede escribir como el producto $A = LU$, donde U es una matriz triangular superior:

$$U = (E_{n-1} \dots E_2 E_1) A \quad (10)$$

La matriz U se calcula tal y como se ha explicado en el proceso de eliminación gaussiana, $U = A^n$. En cuanto a la matriz L es bastante sencilla de calcular puesto que se obtiene inmediatamente de las matrices E_k , $1 \leq k \leq n$, puesto que dichas matrices son de la forma:

$$E_k = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & -l_{k+1,k} & 1 & \\ & -l_{nk} & & 1 \end{pmatrix}, \text{ con } l_{ik} = \frac{a_{ik}^k}{a_{kk}^k}, k+1 \leq i \leq n \quad (11)$$

por tanto, las matrices inversas serán:

$$(E_k)^{-1} = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & l_{k+1,k} & 1 & \\ & l_{nk} & & 1 \end{pmatrix} \quad (12)$$

y, a partir de (12) resulta la matriz L:

$$L = \begin{pmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \dots & \dots & \dots & \dots & \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{pmatrix} \quad (13)$$

que, evidentemente, es triangular inferior.

Teorema A. 2. Sea $A=(a_{ij})$ una matriz cuadrada de orden n , tal que las submatrices

$$\Delta_k = \begin{pmatrix} a_{11} & \dots & a_{1k} \\ \dots & \dots & \dots \\ a_{k1} & \dots & a_{kk} \end{pmatrix} \quad (14)$$

donde $1 \leq k \leq n$, son invertibles.

Entonces existe una matriz triangular $L=(l_{ij})$, con $l_{ii}=1$, $1 \leq i \leq n$, y una matriz triangular superior U tal que se cumple que $A=LU$. Además dicha factorización es única.

Demostración:

Dado que $a_{11} \neq 0$, entonces es posible elegir $P_1 = I$. supongamos que ha sido posible elegir $P_k = I$, $1 \leq k \leq n-1$, tal que se cumpla la igualdad:

$$(E_{k-1} \dots E_2 E_1)A = A^k$$

que en notación matricial se puede escribir como

$$\begin{pmatrix} 1 & & & & \\ x & 1 & & & \\ \dots & x & 1 & & \\ \dots & \dots & \dots & \dots & \\ \dots & \dots & \dots & \dots & 1 \end{pmatrix} \begin{pmatrix} a_{11} & \dots & a_{1k} & \dots & x \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & a_{k1} & \dots & a_{kk} & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x & \dots & \dots & \dots & \dots & \dots & x \end{pmatrix} = \begin{pmatrix} a_{11} & \dots & a_{1k} & \dots & x \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & a_{kk} & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & x \end{pmatrix}$$

El determinante del bloque diagonal $a_{11} \dots a_{kk}$, Δ_k es igual:

$$\det(\Delta_k) = a_{11}^1 \dots a_{kk}^k$$

y puesto que $\det(\Delta_k) \neq 0$, por hipótesis, entonces el elemento a_{kk}^k no es nulo, y por lo tanto, puede ser elegido como pivote. Lo cual supone que $P_k = I$.

Se demuestra entonces la existencia de la factorización LU definida por:

$$A = (E_{n-1} \dots E_2 E_1)^{-1} (E_{n-1} \dots E_2 E_1 A) \stackrel{def}{=} LU$$

En cuanto a la unicidad de dicha factorización. Supongamos que existen dos factorizaciones de A, tales que $A=L_1 U_1 = L_2 U_2$, entonces se puede escribir

$$L_2^{-1}L_1 = U_2U_1^{-1}$$

pero dado que L es triangular inferior y U es triangular superior, entonces esta igualdad solo es posible si ambos productos de matrices son la matriz identidad, y por tanto, $L_2 = L_1$, y $U_2 = U_1$.

□

3. Factorización de Cholesky.

Supongamos que tenemos una matriz simétrica y positiva definida A, entonces dicha matriz se puede escribir como

$$A = LDL^T \tag{15}$$

donde L es una matriz triangular inferior unitaria⁴, y D es una matriz diagonal, cuyos elementos son estrictamente positivos. A esta factorización se la conoce como factorización LDL^T.

Dado que la diagonal de D es estrictamente positiva entonces podemos escribir la factorización anterior como:

$$A = LD^{\frac{1}{2}}D^{\frac{1}{2}}L^T = BB^T = R^TR$$

⁴ Se dice que una matriz es unitaria si todos sus elementos diagonales son iguales a 1.

donde B es una matriz general triangular inferior, y R es una matriz triangular superior. Esta factorización se conoce como *factorización de Cholesky*, y la matriz B se denomina *factor de Cholesky* o raíz cuadrada de A.

El cálculo de los elementos de las matrices R y B es relativamente sencillo dada la representación matricial. Supongamos el caso de $R^T R$,

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{12} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{1n} & a_{2n} & \dots & a_{nn} \end{pmatrix} = \begin{pmatrix} r_{11} & & & \\ r_{12} & r_{22} & & \\ \dots & \dots & \dots & \\ r_{1n} & r_{2n} & \dots & r_{nn} \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ & r_{22} & \dots & r_{2n} \\ & & \dots & \dots \\ & & & r_{nn} \end{pmatrix} \quad (16)$$

y por lo tanto,

$$a_{11} = r_{11}^2, \quad r_{11} = \sqrt{a_{11}} \quad (17)$$

Considerando (16) y dado (17) el resto de elementos de la primera fila se pueden calcular fácilmente despejando en:

$$a_{12} = r_{11}r_{12}, a_{13} = r_{11}r_{13}, \dots, a_{1n} = r_{11}r_{1n}$$

Para la segunda fila ,

$$a_{22} = r_{12}^2 + r_{22}^2 \quad (18)$$

dado que el primer elemento de (18) es conocido entonces podemos calcular r_{22} , y de forma sucesiva el resto de elementos.

Procederíamos igual para el resto de filas, obteniendo la matriz R.

Teorema A.3. Sea A una matriz simétrica positiva definida, entonces existe (como mínimo) una matriz triangular inferior tal que

$$A=BB^T$$

Si además, todos los elementos de la diagonal de la matriz B son positivos entonces dicha factorización es única.

Demostración.

Sea Δ_k la submatriz simétrica con elementos a_{ij} , $1 \leq i, j \leq k$, de la matriz A. Si $w=(w_i)$, $i=1, \dots, k$, es un vector de \mathfrak{R}^k , entonces podemos escribir $w^T \Delta_k w = v^T A v$, con $v \in \mathfrak{R}^n$, $v_i = w_i$, para $1 \leq i \leq k$, y $v_i = 0$ para $k+1 \leq i \leq n$, con lo cual las n submatrices Δ_k son invertibles.

Aplicando el teorema A.2, entonces podemos escribir:

$$A = LU = \begin{pmatrix} 1 & & & & \\ x & 1 & & & \\ \dots & \dots & \dots & & \\ x & x & \dots & 1 & \end{pmatrix} \begin{pmatrix} u_{11} & x & \dots & x \\ & u_{22} & \dots & x \\ & & \dots & \dots \\ & & & u_{nn} \end{pmatrix} \tag{19}$$

donde todos los elementos u_{ii} son positivos.

Si introducimos una matriz diagonal Λ , cuyos elementos sean las raíces cuadradas de los u_{ii} , entonces podemos escribir (19) como:

$$A = (L\Lambda)(\Lambda^{-1}U) = \begin{pmatrix} \sqrt{u_{11}} & & & \\ x & \sqrt{u_{22}} & & \\ \dots & \dots & \dots & \\ x & x & \dots & \sqrt{u_{nn}} \end{pmatrix} \begin{pmatrix} \sqrt{u_{11}} & x & \dots & x \\ & \sqrt{u_{22}} & \dots & x \\ & & \dots & \dots \\ & & & \sqrt{u_{nn}} \end{pmatrix}$$

donde si establecemos $B = L\Lambda$ y $C = \Lambda^{-1}U$, entonces dada la simetría de la matriz A $BC = C^T B^T$, o lo que es lo mismo

$$C(B^T)^{-1} = \begin{pmatrix} 1 & x & \dots & x \\ & 1 & \dots & x \\ & & \dots & \dots \\ & & & 1 \end{pmatrix} = \begin{pmatrix} 1 & & & \\ x & 1 & & \\ \dots & \dots & \dots & \\ x & x & \dots & 1 \end{pmatrix} = B^{-1}C^T \quad (20)$$

Ahora bien, esta igualdad sólo es posible si $C(B^T)^{-1} = B^{-1}C^T = I$, esto es lo mismo que decir que $C = B^T$, lo cual prueba la existencia de al menos una factorización de Cholesky.

Veamos ahora la unicidad. La factorización

$$A = (B\Delta^{-1})(\Delta B^T) \quad (21)$$

donde Δ se define como $\Delta = \text{diag}(b_{ii})$; es simplemente la factorización LU de la matriz A , con $L = B\Delta^{-1}$, $U = \Delta B^T$. Puesto que sabemos que la factorización LU es única, entonces

$$A = B_1 B_1^T = B_2 B_2^T \Rightarrow B_1 \Delta_1^{-1} = B_2 \Delta_2^{-1} \text{ y } \Delta_1 B_1^T = \Delta_2 B_2^T \quad (22)$$

estableciendo $\Delta_\alpha = \text{diag}((B_\alpha)_{ii})$, $\alpha = 1, 2$. La igualdad de los elementos diagonales de las matrices de (22) puede expresarse como

$$(B_1)_{ii}^2 = (B_2)_{ii}^2, \quad 1 \leq i \leq n \quad (23)$$

lo cual prueba que $\Delta_1 = \Delta_2$, puesto que por hipótesis todos los elementos diagonales son positivos. Entonces se sigue que $B_1 = B_2$.

□

Una situación particular interesante surge cuando deseamos actualizar la factorización de Cholesky de una matriz \bar{A} que es una modificación de rango 1 de una matriz A , positivas definidas ambas:

$$\bar{A} = A \pm vv^T \quad (24)$$

Dada una corrección positiva de (15) tenemos:

$$\bar{A} = LDL^T + vv^T = L(D + pp^T)L^T \quad (25)$$

donde p es la solución del sistema triangular $Lp=v$. La naturaleza de la matriz $D+pp^T$ permite que sus factores de Cholesky puedan calcularse directamente (\bar{L} y \bar{D}), y por tanto:

$$\bar{A} = \bar{L}\bar{D}\bar{L}^T = \bar{L}\bar{D}\bar{L}^T \quad (26)$$

donde la matriz \bar{L} es $L\bar{L}$ y \bar{D} es $D\bar{D}$, cuyos elementos se pueden calcular a través de las relaciones:

i) definimos $t_0 = 1$, $v^{(1)}=v$;

ii) para $j=1, 2, \dots, n$ calculamos:

$$\begin{aligned}
 p_j &= v_j^{(j)}, \quad t_j = t_{j-1} + p_j^2 / d_j, \\
 \bar{d}_j &= d_j t_j / t_{j-1}, \quad \beta_j = p_j / (d_j t_j), \\
 \left. \begin{aligned}
 v_r^{(j+1)} &= v_r^{(j)} - p_j l_{rj} \\
 \bar{l}_j &= l_{rj} + \beta_j v_r^{(j+1)}
 \end{aligned} \right\} r = j+1, \dots, n
 \end{aligned} \tag{27}$$

Cuando la corrección es negativa, hay que tener cuidado para prevenir la posible pérdida de la definición positiva de la matriz al calcular los factores de actualización. en este caso las relaciones a aplicar son:

i) resolver las ecuaciones $Lp=v$ y definir $t_{n+1} = 1 - p^T D^{-1} p$; si $t_{n+1} \leq \varepsilon$, entonces hacer $t_{n+1} = \varepsilon$, donde ε es un número suficientemente pequeño.

ii) para $j=n, n-1, \dots, 1$, calcular:

$$\begin{aligned}
 t_j &= t_{j+1} + p_j^2 / d_j, \quad \bar{d}_j = d_j t_{j+1} / t_j, \\
 \beta_j &= -p_j / (d_j t_{j+1}), \quad v_j^{(j)} = p_j \\
 \left. \begin{aligned}
 v_r^{(j)} &= v_r^{(j+1)} - p_j l_{rj} \\
 \bar{l}_j &= l_{rj} + \beta_j v_r^{(j+1)}
 \end{aligned} \right\} r = j+1, \dots, n
 \end{aligned} \tag{28}$$

En cuanto al número de operaciones simples que supone el cálculo de la factorización de Cholesky resulta:

$$\text{Total de operaciones} = \begin{cases} \frac{n^3 - n}{6} \text{ sumas} \\ \frac{n^3 - n}{6} \text{ multiplicaciones} \\ \frac{n(n-1)}{2} \text{ divisiones} \end{cases} \tag{29}$$

Si comparamos esta expresión con (8), vemos que el coste operacional de la factorización de Cholesky es substancialmente menor que el de la eliminación gaussiana. Así en el caso de $n=10$, el número total de operaciones simples es de 375 frente a las 705 del método de eliminación.

4. Factorización QR.

La factorización QR es una de la más conocidas. Trata de factorizar una matriz A a través de una secuencia de matrices ortogonales conocidas como *matrices de Householder*.

Dado un vector v no cero de \mathfrak{R}^n , se dice que $H(v)$ es una matriz Householder si:

$$H(v) = I - 2 \frac{vv^T}{v^T v} \quad (30)$$

Teorema A.4. Sea a un vector de \mathfrak{R}^n tal que la $\sum_{i=2}^n |a_i| > 0$, entonces existen dos matrices Householder H , tales que las $n-1$ componentes del vector Ha son nulas.

Sea $\alpha \in \mathfrak{R}$ tal que $a_1 = e^{i\alpha} |a_1|$. Entonces,

$$H(a + \|a\|_2 e^{i\alpha} e_1)a = -\|a\|_2 e_1 \quad \text{y} \quad H(a - \|a\|_2 e^{i\alpha} e_1)a = \|a\|_2 e_1 \quad (31)$$

donde e_1 representa el primer vector de la base canónica de \mathfrak{R}^n .

En la práctica, dada una matriz A, podemos calcular una sucesión de matrices Householder tal que todos los elementos por debajo de la diagonal se anulen. Podemos construir una sucesión donde si hacemos $A_1 = A$, entonces cada matriz

$$A_k = H_{k-1} \dots H_2 H_1 A, \quad k \geq 1 \quad (32)$$

donde cada A_k presenta la forma:

$$A_k = \begin{pmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x & x \end{pmatrix} \begin{matrix} k - \text{ésima fila} \\ \\ \\ k - \text{ésima columna} \end{matrix} \quad (33)$$

Así pues, si la matriz A es de orden n , entonces la sucesión de premultiplicaciones por las matrices de Householder produce una matriz A_n triangular superior.

Al producto de matrices Householder utilizado en (32) se le denomina matriz Q , y es una matriz ortogonal⁵ de orden $n \times n$, donde n es el número de columnas de la matriz A . Si la matriz A es cuadrada de orden n , entonces podemos escribir:

$$Q A = R$$

donde R es una matriz triangular superior no singular y de orden $n \times n$.

Teorema A.5. Dada una matriz A de orden n , existe una matriz ortogonal Q y una matriz triangular superior R tal que

$$A = QR$$

A esta factorización se la conoce como factorización QR de la matriz A .

⁵ Dado que Q es ortonormal entonces se cumple que $Q^{-1}Q = I$.

Más aun, es posible asegurar que todos los elementos diagonales de R son no negativos, y si además, A es invertible, entonces la correspondiente factorización es única.

Demostración.

Dado que según el teorema A.4. la existencia de matrices Householder ha sido demostrada, entonces ya hemos establecido que toda matriz cuadrada A de orden n puede escribirse como:

$$A = (H_{n-1} \dots H_2 H_1)^{-1} A_n$$

entonces si definimos $Q = (H_{n-1} \dots H_2 H_1)^{-1}$ y $R = A_n$; podemos establecer la existencia de al menos una factorización QR de la matriz A, donde la matriz Q es el producto de n-1 matrices Householder.

Sean $\alpha_j \in \mathfrak{R}$ los números que satisfacen $(R)_{jj} = e^{i\alpha_j} |(R)_{jj}|$, $1 \leq j \leq n$, y sea $D = \text{diag}(e^{i\alpha_j})$. Entonces la matriz $Q' = Q D$ es ortogonal y la matriz $R' = D^{-1} R$ es también triangular superior, y lo que es más, los elementos diagonales son todos no negativos. Así pues, hemos establecido una factorización $A = Q'R'$ con $(R')_{jj} \geq 0$.

Mostremos ahora la unicidad de la factorización, supuesto que A es invertible. Supongamos que existen dos factorizaciones

$$A = Q_1 R_1 = Q_2 R_2$$

se sigue que

$$Q_2^{-1} Q_1 = R_2 R_1^{-1} \stackrel{\text{def}}{=} \Delta \tag{34}$$

En consecuencia,

$$\Delta^{-1} \Delta = Q_1^{-1} Q_2 Q_2^{-1} Q_1 = I \tag{35}$$

es una factorización de Cholesky de la matriz Identidad, puesto que la matriz Δ es triangular superior; y

$$(R_1)_{ii} > 0 \text{ y } (R_2)_{ii} > 0 \Rightarrow (\Delta)_{ii} = \frac{(R_2)_{ii}}{(R_1)_{ii}} > 0$$

y, dada la unicidad de la factorización de Cholesky entonces $\Delta = I$.

□

Si la matriz A es de orden $m \times n$, no es cuadrada, entonces obtendremos dos submatrices, una submatriz triangular superior R de orden $n \times n$ y una submatriz de 0 de orden $(m-n) \times n$. Podemos escribir la factorización de A como:

$$QA = \begin{pmatrix} R \\ 0 \end{pmatrix} \quad (36)$$

Si el rango de A es r, donde $r < n$, entonces es necesario realizar un intercambio de columnas para asegurar que las r columnas linealmente independientes son las que primero se van a procesar. El resultado de estas transformaciones por la izquierda resulta entonces

$$QAP = \begin{pmatrix} T \\ 0 \end{pmatrix} \quad (37)$$

donde P es una matriz de permutación y T es una matriz trapecoidal superior de orden $r \times n$. Si utilizamos una secuencia de matrices Householder aplicadas por la derecha para anular las últimas $n-r$ columnas de T y preservando la estructura triangular de las primeras r columnas, entonces obtenemos

$$QAP\bar{H}_1 \dots \bar{H}_{n-r} = QAV = \begin{pmatrix} R & 0 \\ 0 & 0 \end{pmatrix} \quad (38)$$

donde R es una matriz triangular superior no singular de orden $r \times r$, y V es una matriz ortogonal de orden $n \times n$. A este método se le conoce como *factorización ortogonal completa de A*.

Veamos que sucede cuando tenemos que actualizar la factorización QR de una matriz A . Si añadimos una columna:

$$\bar{A} = (A \ a)$$

donde A es una matriz $m \times n$. y a es un vector de orden m . Entonces tenemos que

$$Q\bar{A} = (QA \ Qa) = \begin{pmatrix} R & v_1 \\ 0 & v_2 \end{pmatrix} \quad (39)$$

donde v_1 y v_2 son las particiones adecuadas del vector $v=Qa$ de orden m . Sea H_{n+1} la matriz Householder que anular los $n+2$ hasta m elementos de v , y deja los primeros n elementos de v sin cambio, tal que

$$H_{n+1}v = \begin{pmatrix} v_1 \\ \gamma \\ 0 \end{pmatrix} \quad (40)$$

donde $|\gamma| = \|v_2\|_2$. Al aplicar H_{n+1} a $Q\bar{A}$ obtenemos

$$H_{n+1}Q\bar{A} = \begin{pmatrix} R & v_1 \\ 0 & \gamma \\ 0 & 0 \end{pmatrix} \quad (41)$$

o lo que es lo mismo,

$$\overline{QA} \equiv \begin{pmatrix} \bar{R} \\ 0 \end{pmatrix} \quad (42)$$

lo que define la factorización QR de \bar{A} .

Observamos, por tanto, que para la actualización de la factorización tan sólo ha sido necesario el cálculo de una matriz de Householder.

La factorización ortogonal de la matriz A proporciona no sólo una matriz triangular, sino que además nos permite tener información sobre el espacio de filas y el de columnas de dicha matriz. Si partimos Q^T en dos submatrices tal que

$$Q^T = (Q_1 \ Q_2),$$

donde Q_1 y Q_2 son dos matrices de orden $m \times r$ y $m \times (m-r)$, respectivamente. Las columnas de Q_1 forman una base ortonormal del espacio de columnas de A , y las columnas de Q_2 forman una base ortonormal para el correspondiente espacio nulo de vectores ortogonales a las columnas de A . Además, en el caso de que $r < n$, las últimas $n - r$ filas de V constituyen una base ortonormal para el conjunto de vectores ortogonales a las filas de A .

A partir de la factorización QR podemos derivar otras dos:

- a) La **factorización LQ**, sólo es admisible para matrices con rango pleno, esto es, cuyas n columnas son independientes.

$$A Q = (L \ 0)$$

Q es una matriz ortogonal y L es una matriz triangular superior.

- b) La factorización de Schur o total.

$$A = QTQ^T$$

donde T es una matriz triangular superior.

Comentemos para finalizar este apéndice que dentro del programa Mathematica están implementadas varias factorizaciones:

- a) QRDecomposition [m] establece la factorización QR de la matriz numérica m. El resultado es una lista {q,r} donde q es una matriz ortogonal y r es una matriz triangular superior.
- b) SchurDecomposition [m] calcula la factorización de Schur para una matriz numérica m. El resultado es una lista {q, r} donde q es una matriz ortogonal y t es una matriz triangular superior.

APENDICE B:
MATHEMATICA

APENDICE B: MATHEMATICA.

Mathematica es un sistema general de software para matemáticas y otras aplicaciones científicas, especialmente todas aquellas donde es necesaria la utilización de métodos cuantitativos. Desarrollado en 1988, año del que data la primera versión, en el presente trabajo hemos utilizado la versión 2.1 para Windows de dicho programa, aunque recientemente ha salido al mercado la versión 3.0 que incorpora numerosas novedades.

Básicamente la utilización del programa Mathematica se dedica a dos frentes:

- a) Calculador simbólico y numérico.

- b) Plataforma software para la construcción de paquetes para aplicaciones específicas.

Asimismo, es aplicable a la creación de programas en lenguaje de alto nivel (tipo C, Fortran, etc.), o documentos mixtos de tipo científico (tipo TeX).

El programa combina la manipulación simbólica (integración simbólica, aritmética de polinomios,...), el cálculo matemático (precisión aritmética arbitraria, funciones de teoría de números,...), interfaz de gráfico (funciones en dos y tres dimensiones,...), y un sistema de lenguaje de programación altamente sofisticado (alto nivel de abstracción a partir de un lenguaje interpretado).

La ventaja principal del programa frente a los lenguajes clásicos de programación es la existencia de 750 funciones matemáticas ya directamente implementadas y, por tanto, lista para su utilización.

Estructura del programa Mathematica.

El interfaz de Mathematica para Windows se realiza a través de unos documentos denominados "notebook"; es en dichos documentos donde nosotros introducimos las órdenes y el programa nos proporciona los resultados de la ejecución de las mismas.

El esquema de la pantalla es el habitual de todos los programas bajo el entorno windows, existen una ventana principal y los diferentes notebooks se agrupan en distintas ventanas. Tenemos un menú con todos los comandos directos disponibles y una barra de herramientas que contiene algunos de dichos comandos.

El notebook inicial es una hoja en blanco, para interactuar con la misma nos situamos con el cursor sobre la dicha hoja y comenzamos a escribir. Al efectuar esta operación hemos creado una celda "cell". Las celdas son el componente básico con el que trabaja el programa, aparecen marcadas con un corchete derecho en el borde del notebook. Se pueden distinguir cuatro tipos básicos:

- a) Celdas de entrada. Etiquetadas con $In[n]$, son aquellas que el usuario crea al escribir las ordenas que desea que el programa ejecute.
- b) Celdas de salida. Etiquetadas con $Out[n]$, las crea automáticamente el programa como respuesta a las celdas del tipo anterior. Cada celda tipo In , genera su equivalente tipo Out , con concordancia del número n . En estas celdas no podremos escribir ni variar la respuesta ejecutada automáticamente por el programa.

- c) Celdas de mensaje. Se caracterizan porque el texto es de color rojo. Se generan automáticamente cuando el programa desea indicarnos algún mensaje.
- d) Celdas de texto. Las crea el usuario y en ellas puede introducir texto no interpretable como funciones por el interfaz del programa. Existen varios tipos de formatos predefinido susceptibles de aplicación al texto escrito.

Cuando queremos ejecutar el contenido de una celda nos situamos con el cursor sobre la misma, para ello pinchamos con el ratón pulsando el botón izquierdo; y una vez que el cursor este parpadeando dentro de la celda, o bien el borde de la misma se halla puesto en vídeo inverso podemos:

- Ejecutar el comando Evaluate Selection del menú Action.
- Pulsar con el ratón sobre el botón de Ejecución.
- Pulsar la combinación de teclas Mayúsc + Intro.

El programa genera una celda de entrada cada vez que vayamos a realizar una evaluación. Una vez ha proporcionado la salida oportuna, queda disponible para la siguiente evaluación. En cada evaluación ejecutara todas las funciones que hayamos escrito dentro de la celda de entrada. Pulsar intro dentro de la celda de entrada supone añadir una nueva línea a la celda actual.

Si deseamos volver a ejecutar una celda de entrada debemos situarnos sobre ella y proceder a la evaluación; automáticamente el programa refresca el contenido de la celda de salida asociada.

Ejemplo: Deseamos obtener el resultado de sumar dos números enteros:

a) Opción cálculo numérico:

In[1]: 2 + 8

Out[1]: 10

b) Opción cálculo simbólico:

In[1]: a + b

Out[1]: a + b

Comandos.

Los comandos se encuentran en la barra de menú de la versión de Mathematica para windows. Sirven para ejecutar ordenes de interfaz entre el programa y el ordenador, o para establecer opciones de ejecución. El menú de Mathematica es accesible de acuerdo con las norma habituales de Windows y su esquema es el habitual a todos los programas diseñados para actuar en este tipo de entornos. Los integrantes del menú son nueve:

- 1) **File**: Este submenú contiene los comandos para la gestión de ficheros (notebooks o packages), imprimir y salir del programa.

Los archivos creados con Mathematica son básicamente de dos tipos:

a) *Notebooks*. Son archivos que contienen procedimientos ejecutados en Mathematica así como su resultados. Cuando se cierran desaparecen los indicativos de las celdas In[] / Out[]. Al recuperarlos será necesario volver a ejecutar todas las celdas de entrada para situar en la memoria del ordenador los resultados del archivo. Los archivos de este tipo se almacenan con la extensión “.ma” y “.mb”. Los archivos que recuperaremos serán los tipo .ma.

b) *Packages*. Son archivos que continen funciones para su posterior utilización en los notebooks. Se almacenan con la extensión “.m”. Pueden ser creados por el usuario o bien adquiridos externamente.

Los comandos disponibles son:

- New. Crea un nuevo archivo.
- Open. Abre un archivo existente.
- Close. Cierra un archivo.
- Save. Guarda un archivo.
- Save As. Guarda un archivo con otro nombre u otro tipo.
- Print. Imprime un archivo abierto (total o parcialmente).
- Exit. Sale del programa.

2) **Edit**: Básicamente los comandos de este menú sirven para seleccionar una parte de un archivo abierto.

Undo: Deshace la última acción de edición realizada.

Cut: Corta una parte del texto seleccionada (En video inverso) o una celda y la envía al portapapeles.

Copy: Copia una parte del texto en el portapapeles.

Paste: Pega el contenido del portapapeles en el punto donde este situado el cursor.

Clear: Borra la parte del texto seleccionada.

Select All Cells: Selecciona todas las celdas del archivo.

3) **Cell**: Se utiliza para trabajar con las celdas.

Formatted.

Group: Agrupa varias celdas en una celda de tamaño mayor.

Ungroup: Desagrupa una celda compuesta.

Open All Subgroups: Sirve para abrir un grupo. Denominamos grupos al conjunto de celdas de entrada y salida. Cuando esta cerrado no es visible la celda de salida.

Close All Subgroups: Cierra un grupo.

4) **Graph**: Comandos para gráficos.

5) **Action:** Comandos de ejecución.

Evaluate: Evalúa la celda seleccionada (Equivalente a Mayús+Intro).

3D View Point Selector: Cambia el ángulo de visión en tres dimensiones.

Color Selector: Cambia los colores asignados.

Interrupt: Interrumpe el proceso de evaluación de una celda.

6) **Style:** Opciones de Estilo de las celdas.

Cell Style: Aplica un estilo predefinido a una celda seleccionada.

Font: Cambia el tipo de letra de la celda seleccionada.

Format: Cambia el formato.

Edit Notebooks & Default Styles: Edita los estilos predefinidos.

7) **Options:** Sirve para cambiar las opciones por defecto del programa.

8) **Window:** Establece el orden de las ventanas: Mosaico (Tile) o Cascada (Cascade).

9) **Help:** Fichero de acceso a la ayuda:

Kernel Help: Ayuda sobre los procedimientos y funciones estándar disponibles al cargar el programa el núcleo inicial (Kernel).

Front End Help: Ayuda sobre los comandos del menú.

Help on Help: Ayuda sobre el manejo y utilización de la ayuda.

Los comandos también se pueden ejecutar pulsando combinaciones de teclas que aparecen en el lado derecho del nombre del comando dentro de los menús desplegables; o bien a través de la barra de botones.

Los botones disponibles son:

- a) Estilo de celda.
- b) Alineación del texto en la celda (Izda., Centrado, Dcha.)
- c) Edición: Cortar, Copiar, Pegar, Clear.
- d) Evaluar Selección.
- e) Animación, Gráficos y Sonido.
- f) 3D View Selector / Color Selector / Interrumpir / Ayuda.

Por último, hay que señalar que existen otros comandos que se utilizan en las celdas del propio notebook:

<< nombre_paquete.

Sirva para que las funciones contenidas en un paquete (package) puedan ser utilizadas en el notebook abierto.

? nombre_función.

Proporciona una ayuda básica sobre la utilización de una función.

?? nombre_función.

Proporciona ayuda completa sobre la utilización de una función.

% permite operar con el resultado de la evaluación inmediatamente anterior.

Ejemplo:

In [1]: 10^6

Out[1]: 60

In[2]: % / 5

Out[2]: 12

In[3]: %+3

Out[3]: 15

Funciones.

Las funciones son los elementos principales del programa puesto que nos permiten ejecutar las operaciones deseadas. Todas las funciones son “llamadas-por-valor”, esto es, no es posible escribir una función que cambie el valor de sus argumentos. Toda función, ya sea creada por nosotros, ya exista previamente definida consta de dos partes:

1. **Cuerpo:** Nombre de la función. Empieza obligatoriamente con una letra mayúscula.
2. **Argumentos:** Reperesentan los valores que pasamos a la función para que proceda a la evaluación. Aparecen encerrados entre corchetes, y separados por comas.

Ejemplo: La función `Log[n]` calcula el logaritmo en base e del número n.

In[1]: `Log[5]`

Out[1]: `Log [5]`

In[2]: `N[Log[5]]`

Out[2]: `1.60944`

En la primera respuesta, dado que Mathematica siempre responde con precisión absoluta el resultado es `Log[5]`, puesto que es un número irracional. Si deseamos una aproximación al resultado podemos emplear la función `N[Función, Número]`, donde función es la función a ejecutar y Número el número de cifras decimales de la aproximación¹.

La definición de funciones por parte de los usuarios es otro de las grandes ventajas de Mathematica. Un ejemplo genérico de función puede ser el siguiente:

¹ Cuando no se especifica ningún número el programa asume de forma automática cinco cifras decimales.

```
NombreFuncion [arg1_Integer?OddQ, Cosa, argn_List]:=
  Block[{z1,z2},
    z1=z2=arg1;
    z2
  ]/; MemberQ[argn, arg1]
```

Esta función puede utilizarse con los siguientes argumentos, por ejemplo:

```
NombreFuncion[3, Cosa, {1,3,5}]
```

Al llamar la función el programa realiza automáticamente el proceso de pattern-matching; este se le indica con el subrayado `_`; así `_Integer?OddQ`, requiere que `arg1` sea un elemento de tipo entero e impar, por su parte `_List`, indica que `argn` debe ser una lista²; por último, la palabra `Cosa`, solo coincide con que escribamos el mismo argumento en la llamada.

Si deseamos que la función se ajuste a determinadas condiciones estas se especificarán al final tras `/;`, aquí la condición `MemberQ` es una función que devuelve `True` si `arg1` pertenece a la lista `argn`. El resultado final de una función será el que se refleja en la celda de salida, y en general es el valor de la última sentencia ejecutada dentro de la función, o el valor de `x` si se incluye la función `Return [x]`. La utilización de `:=`, indica al programa que se trata de una definición y que por tanto no debe producir ningún output en pantalla.

Por ejemplo el output de la llamada anterior es:

3

Las diferentes sentencias del cuerpo de la función se separan con un `;` la colocación de dicho operador al final de una llamada de función inhibe el resultado y hace que el programa devuelva el valor `Null`. El conjunto de sentencias se pueden agrupar usando la estructura `Block [{x1, x2,...}, expresión]`, que permite crear variables locales `x1, x2, ...`, las cuales pueden cambiar su valor durante la ejecución.

² Sucesión de elementos separados entre comas y encerrados entre llaves `{}`.

Los predicados son funciones que devuelven **True** o **False**, llevan nombres que terminan en Q. Así tenemos **MemberQ**, o **OddQ**, la primera comprueba si el argumento pasado en primer lugar aparece dentro de la lista pasada como segundo argumento, y la segunda comprueba si el argumento es un número impar. En caso positivo devuelven **True**, en caso negativo la respuesta es **False**.

Operadores.

Los operadores son estructuras de función con notación infija que sitúan los argumentos a ambos lados del operador. Cada operador sólo puede tener dos argumentos. Los operadores más habituales son los aritméticos³: **+**, **-**, *****, **/**, **^**. Sus equivalentes tipo función son: *Plus*, *Subtract*, *Times*, *Divide*, *Power*. Otros operadores de uso frecuente son:

; para separar dos sentencias o evaluar dos funciones sucesivamente⁴.

/. evalúa una función para los valores de los argumentos que se le proporcionan en una lista.

-> asigna de forma temporal sólo para la evaluación actual un valor a una variable.

³ La multiplicación de enteros se puede indicar con el operador *****, o con un espacio entre los dos argumentos. Así **2*3** es igual a escribir **2 3**. La multiplicación de vectores o matrices se realiza mediante el operador **.** (punto).

⁴ Cuando se evalúan dos funciones concatenadas con **;** el resultado final del output es el de la última función evaluada.

/; equivale a un si condicional, evalua la función situada a la izquierda si al evaluar la función situada a la derecha obtenemos un True.

= realiza una asignación de forma permanente. Si se usa con listas produce una asignación múltiple, así {a,b,c}={1,2,5} es equivalente a a=1; b=2; c=5.

:= realiza una asignación diferida, en este caso se sustituirá con el valor actual de la expresión de la derecha cuando se invoque. Por ejemplo:

```
In[1]: x=3
Out[1]: 3
In[2]: f[x_]:=2*(x+1)
In[3]: f[2]
Out[3]: 6
In[4]: f[x]
Out[4]: 8
In[5]: f[t^2]
Out[5]: 2 (1 + t2)
```

Para eliminar una asignación a una variable se utiliza la función Clear [nombrevariable].

Los pares de signos +=, -=, *= y /= sirven como operadores de asignación para las operaciones aritméticas básicas, después de haber asignado un valor a la variable (así x += dx añade dx a x y devuelve el nuevo valor de x).

Por último, los operadores relaciones de igualdad, desigualdad, mayor o igual, y menor o igual serán ==, !=, >=, <=.

Listas.

La lista es la estructura básica de Mathematica, es una colección de objetos entre llaves separados por comas, la cual permite una gran versatilidad pues sus

miembros no tienen porque ser del mismo tipo. Existen varios modos de acceder a un elemento particular de una lista, como si de una matriz se tratase. El mod más directo es a través del comando `[[.]]`.

Si tenemos una lista $d=\{1,3,5,7\}$, $d[[3]]=5$, el tercer elemento de la lista es 5.

Una lista simple es un vector fila, y como tal puede ser utilizado en todo tipo de operaciones vectoriales. Si deseamos un vector columna tendremos que utilizar una lista anidada; esto es, una lista de listas. En el caso anterior el vector columna pertinente sería $d=\{\{1\},\{3\},\{5\},\{7\}\}$. En este caso 5 es el elemento de la columna 1, fila 3, y se accedra a través de la orden $d[[3,1]]$. Las listas anidadas también pueden servir para representar matrices cuando la continene más de un elemento.

Por ejemplo una matriz 2 x 2 sería $M=\{\{1,3\},\{5,7\}\}$. Donde M sería de la forma:

$$\begin{pmatrix} 1 & 3 \\ 5 & 7 \end{pmatrix}$$

La matriz traspuesta se puede obtener mediante la función: **Transpose** [M].

Si deseamos ver la matriz con forma matricial podemos utilizar la función **MatrixForm** la cual nos permite presentar una matriz con la disposición habitual de filas y columnas. No obstante, esta opción no permite su posterior utilización por el programa puesto que tan sólo afecta a la presentación en pantalla del resultado. En el caso anterior: `MatrixForm[M]` produciría:

$$\begin{array}{cc} 1 & 3 \\ 5 & 7 \end{array}$$

como resultado de salida.

El producto interno de matrices se puede ejecutar mediante el operador punto `.`, o la función **Dot**. Por su parte, el producto externo utiliza el operador `*`.

Las principales funciones disponibles para listas y matrices son:

- a) **Take** [d, n]. Obtiene la lista de los n primeros elementos de la lista d.
- b) **Take** [d, - n]. Obtiene la lista de los n últimos.
- c) **Drop** [d, n]. Obtiene la lista d sin los n primeros elementos.
- d) **First**, **Last**, **Rest**. Obtienen los primero, último y todos menos el primero elementos de una lista dada.
- e) **Join**. Une los elementos de dos lista y forma una nueva.
- f) **Append**. Añade elementos a una lista por detras.
- g) **Prepend**. Añade elementos a una lista por delante.
- h) **Flatten**. Convierte una lista múltiple en una lista sencilla.
- i) **RotateLeft**, **RotateRight**. Rota los elementos de una lista n posiciones a la izquierda y a la derecha.
- j) Operaciones de conjuntos: **Union**, **Intersection** y **Complement**.
- k) **Det**: Calcula el determinante de una matriz.
- l) **Inverse**: Calcula la matriz inversa.
- m) **Eigenvalues**, **Eigenvectors**: Calculan los autovalores y autovectores de una matriz.

Al trabajar con listas hay que indicar que salvo que se asignen a una variable la lista original no cambia⁵. El i-ésimo elemento de una lista se puede cambiar a través de la notación `d[[i]]=3`, pero por razones técnicas, al hacer esto Mathematica reescribe la

⁵ Aparte del comando `=`, hay muy pocas operaciones en Mathematica que posean efectos laterales. Dos excepciones son **AppendTo** y **PrependTo**, equivalentes a **Append** y **Prepend** pero con efectos de cambio de la lista original.

lista totalmente; así pues, modificar un elemento de una lista es, de hecho una operación de tiempo lineal.

Packages.

Los packages sirven para escribir funciones y procedimientos no implementados en el Kernel y que quedan disponibles para ser utilizados posteriormente en otros notebooks. Para crear un package, creamos un notebook nuevo en Mathematica y escribimos un archivo tipo texto⁶ con la siguiente estructura:

```
BeginPackage ["nombre_paquete"]
```

```
Nombre_funcion :: usage = = "Descripcion de la utilización de las funcion  
creada"
```

```
Bengin [" `Private`"]
```

```
Nombre_funcion := .....
```

```
End[ ]
```

```
EndPackage [ ]
```

Una vez escrito se almacena como paquete disponible en Mathematica utilizando el comando *Save As / Export*, y eligiendo en tipo de archivo: Packages. El ordenador automáticamente le asignará la extensión .m al nombre del archivo elegido.

⁶ El package se puede crear en cualquier editor de texto siempre y cuando se almacene como tipo texto (.txt).

Una vez que hemos creado un package podemos:

a) Recuperarlo para modificarlo: Se utiliza el comando *Open / Import*, se escoge el tipo de archivo : Packages; y se selecciona el archiv desado. Una vez modificado si deseamos salvar los cambios lo guardamos utilizando el comando *Save As / Export*.

b) Recupararlo para utilizar las funciones que hayamos incorporado en el package: Se utiliza el comando `<< nombre_paquete` directamente en el notebook donde vayamos a utilizar dichas funciones.

Iteración y Recursividad.

Como lenguaje de programación Mathematica dispone de facilidades para realizar bucles y ejecutar un bloque de código más de una vez. Los procedimientos principales son:

While [test, cuerpo]. Ejecuta cuerpo mientras se cumpla test.

For [inicio, test, incr, cuerpo]. Ejecuta cuerpo desde test= inicio incrementándolo en incr en cada iteración y hasta que no se cumple test.

Do [expr, {imax}]. Ejecuta expresión las veces establecidas en imax.

Table[expr, {imax}]. Constuye una lista cuyo i-ésimo elemento es alguna función dada de i.

Los *iteradores* son listas con un elemento como mínimo y cuatro como máximo, que indican las veces que debe realizarse una operación. Su formato general es $\{i, imin, imax, ipaso\}$, donde *i* es la variable que controla el bucle que irá tomando

valores desde i_{min} hasta i_{max} con saltos de tamaño i_{paso} ; el único elemento obligatorio es i_{max} , siendo el valor estándar del paso y el inicial 1. Así pues $\{i, 100\}$ es equivalente a $\{i, 1, 100, 1\}$. Los iteradores usados en cualquier procedimiento suponen que dicha función se ejecute varias veces. Ejemplos de utilización de los iteradores son:

Table [i^3 , {i,5}]

{1, 8, 27, 64, 125}

Sum [$i^2 x^i$ {i, 2, 8, 2}]

$4 x^2 + 16 x^4 + 36 x^6 + 64 x^8$

A menudo es necesario realizar una función sobre todos y cada uno de los elementos de una lista, o bien, calcular una función de toda la lista. Para estos procesos iterativos Mathematica dispone de dos funciones:

Map (en notación infija se puede utilizar el operador $/@$). Para una función f y una lista l devuelve una lista cuyo i -ésimo elemento es $f[l[[i]]]$.

Map [f , {a,b,c}]

{f[a], f[b], f[c]}

Apply (como operadore se puede usar $@@$). invoca f una sola vez utilizando $l[[i]]$ como i -ésimo argumento de f .

Apply [f , {a,b,c}]

$f[a,b,c]$

Dado que **Map** y **Apply** llevan como argumentos funciones, para una utilización completa de ambas es necesario un mecanismo para declarar funciones cortas. Tales funciones se denominan *puras*, o anónimas dado que no llevan nombre

asociado, y se distinguen por terminar en `&`; utilizan el símbolo `#` para denotar el argumento, tratándolo como cualquier otro parámetro. Por ejemplo:

```
(Apply[Plus, 2*#]&)[{5,2,3}]
20
```

que se puede escribir de forma equivalente como:

```
Function[n, Apply[Plus, 2*n]][{5,2,3}]
20
```

En matrices el uso de estas funciones se vuelve muy frecuente puesto que las entradas están a dos niveles (esto es, hay dos listas). Si deseamos aplicar una función a cada una de las entradas se puede utilizar `Map[f, matriz, {2}]` (si no se ponen las llaves en `{2}` se aplicaría la función a todos los niveles a partir del segundo).

```
Map[#^(1/3)&, {{27,64},{8,125}},{2}]
{{3,4},{2,5}}
```

Para realizar iteraciones puras existe la función `Nest` y `NestList`. Ambas funciones realizan anidamiento de otras funciones. Ejemplo:

```
Nest[f, x, 3]
f[f[f[x]]]
NestList[f,x,3]
{x, f[x], f[f[x]], f[f[f[x]]]}
```

Otras de las posibilidades de programación disponible en Mathematica es la *recursividad*. La recursividad es una de las técnicas más naturales para la implementación de muchos algoritmos. La idea básica es dividir el problema en problemas más pequeños del mismo tipo. Si obtenemos un algoritmo que resuelva el problema pequeño también podrá resolver el problema mayor. En Mathematica la recursividad permite, en la mayoría de los casos sustituir a la iteración, con la ventaja

añadida de que un programa recursivo será usualmente más rápido que el programa iterativo equivalente.

Los programas recursivos plantean, en general, un caso básico y luego una regla de aplicación general para casos superiores. Un ejemplo típico es el cálculo del i -ésimo elemento de la sucesión de Fibonacci. Tomamos dos casos básicos el elemento 0 es 0 y el elemento 1 es 1, la regla general es que el elemento i es la suma de los dos anteriores, esto es, el elemento $i-2$ y el elemento $i-1$.

En Mathematica podemos implementar la sucesión de Fibonacci como:

```
Fib [0] = 0;  
Fib [1] = 1;  
Fib [n_Integer]= Fib[n-1] + Fib [n-2]
```

Lógicamente, en este tipo de programación recursiva no es muy conveniente profundizar de forma excesiva en el grado de recursión puesto que podría agotar los recursos del ordenador . En el ejemplo anterior, prácticamente resultaría imposible que con este programa obtuviésemos el elemento 100 de la sucesión de Fibonacci.

Si deseamos elegir entre diversas alternativas de implentar un procedimiento podemos comprobar la eficacia de los mismos mediante la utilización de la función **Timing**. Esta función muestra el tiempo de CPU empleado en un cálculo.

Por ejemplo podemos plantearnos la forma más rápida de de obtener la suma de los 1000 primeros números naturales. Las implementaciones tradicionales mediante el empleo de bucles arrojan los siguientes resultados⁷:

⁷ Estos resultados se han obtenido empleando un ordenador con procesador 486 DX4 con coprocesador matemático a 75 Mhz. y 8 Mb de RAM.

a) Utilizando una estructura Do:

```
Timing[suma=0; Do[suma+=i, {i,1000}]; suma]
{0.49 Second, 500500}
```

b) Utilizando un bucle For:

```
Timing[For[suma=0; i=1, i<=1000, i++,suma+=i]; suma]
{0.66 Second, 500500}
```

c) Utilizando la estructura While:

```
Timing[suma=0; i=1; While[i<=1000, suma+={i++}]; suma]
{0.88 Second, {500500}}
```

d) Utilizando listas y la estructura Nest.

```
[[s_,x_]:=s+x,x+1];
Timing[First[Nest[f,{0,1},1000]]]
{0.66 Second, 500500}
```

e) Mediante la utilización de la función Apply.

```
Timing[Apply[Plus, Range[1000]]]
{0.06 Second, 500500}
Timing[Plus @@ Range[1000]]
{0.06 Second, 500500}
```

En este caso la estrategia recursiva no resulta válida puesto que se supera el límite de recursión y el programa falla.

Se puede observar que la utilización de Apply es mucho más eficaz que las otras estructuras iterativas tradicionales; de entre estas la que arroja un mejor tiempo es la estructura Do, seguida de For y While.

Programación en Mathematica.

Como ya hemos comentado podemos utilizar Mathematica para construir nuestras propias funciones y generar nuestros propios programas de cálculo simbólico. Existen tres tipos de programación disponibles:

- Procedural: utilizan estructuras de bloque, condicionales, iteraciones y recursividad. Es similar a la programación en lenguajes de alto nivel: Pascal, Fortran, C,...
- Funcional: utiliza funciones puras y operadores funcionales. Se aproxima al estilo de los lenguajes funcionales tipo Lisp, Caml,...
- Basada en Reglas: utiliza pattern-matching y orientada a objetos. Se asemeja a los lenguajes lógicos tipo Prolog.

El programa básico es la función, esta, como ya hemos comentado, tiene la siguiente estructura genérica:

Nom_función [arg1_, arg2_,...] := cuerpo.

donde Nom_función es el nombre de la función, arg1_, arg2_, son los patrones para los distintos argumentos de la función, := es el operador de asignación diferida; por último, el cuerpo puede ser, o bien una expresión, o bien varias expresiones separadas por ; e incluidas entre paréntesis.

Los argumentos se pueden llamar para todos los tipos de variables posibles, o bien para uno concreto. Si no se especifica un tipo concreto se utiliza el identificador `_` (barra de subrayado). Los tipos básicos existentes en Mathematica son⁸:

1. Integer: Representa un número entero.
2. Real: Representa un número real.
3. Rational: Números racionales.
4. Complex: Números Complejos.
5. Symbol: Símbolos matemáticos (ej: Pi, e, etc).
6. String: Cadenas de caracteres, van entre “ ”.
7. List: Tipo listas.

Para asignar un tipo concreto a un argumento simplemente se escribe al lado del mismo separado por una barra de subrayado. Al llamar a la función el programa devolverá la misma llamada si el argumento que se le pasa no coincide con el argumento declarado. Por ejemplo: `f[a_Integer]:= a * 5`, devolverá la propia función si se llama como `f[3/4]`, puesto que $\frac{3}{4}$ no es un número entero. A este tipo de concordancias se les denomina *pattern-matching*, y son una de las grandes ventajas del Mathematica, puesto que proporcionan una gran flexibilidad para especificar los argumentos de las funciones. Aparte de la especificación de tipos concreta, también es posible utilizar predicados para concretar aun más el tipo de un argumento. Existen dos métodos para especificar los tipos mediante predicados:

a) Utilizando el operador `?`. Unido al tipo en el *pattern-matching* permite optar por un grupo concreto dentro del tipo asignado.

⁸ Si deseamos conocer el tipo de una expresión atómica se puede usar la función **Head**.

Por ejemplo, si deseamos establecer una función sólo para los números enteros pares, podemos añadir un predicado que devuelva True si el número es par.

```
f[x_Integer?EvenQ]:= x/2
```

b) Utilizando el operador condicional /;. Este se escribe después del cuerpo de la función y permite modificar el tipo de los argumentos.

En el ejemplo anterior podemos exigir que la función sólo sea aplicable a enteros pares, y mayores que 1 o menores que -1. Para ello escribiríamos:

```
f[x_Integer?EvenQ]:= x/2 /; x < - 1 || 1 < x
```

El operador || es el operador lógico OR.

También podemos añadir reglas a símbolos definidos por el usuario mediante la notación *símbolo* /: *regla*. Para ver las reglas asociadas a una función se puede usar el operador ?. Si la función ha sido definida por el usuario entonces es necesario documentar la función utilizando la notación:

Nombre_función ::usage = “ función [arg1, arg2, ...] realiza ... “

Es importante tener cuidado de escribir todos los caracteres de ::usage sin espacios intermedios.

Por ejemplo, podemos añadir a la función f creada anteriormente una regla:

```
f /: f[x_] + f[y_] := f[x+y]
```

```
f[8]+ f[10]
```

```
9
```

En ciertos caso puede ser necesario la utilización de variables locales, las cuales pueden tener el mismo nombre que las variables globales. Para la declaración de este tipo de variables se utilizan dos funciones: **Block** y **Module**. Ambas son similares pero sus efectos son diferentes.

La diferencia básica es que mientras **Module** realiza una asignación léxica, **Block** realiza la denominada asignación dinámica. La primera es la asignación habitual en la mayoría de los lenguajes de alto nivel, mientras que la segunda es específica a ciertos lenguajes simbólicos, tal y como Lisp.

En la asignación léxica, las variables se tratan como locales a una sección particular del código dentro del programa. En la asignación dinámica, los valores de las variables son locales a una parte de la *historia de la ejecución* del programa.

Module [*variables*, *cuerpo*] trata la forma de la expresión *cuerpo* en el momento en cual se ejecuta el módulo como código de un programa de Mathematica. Entonces cuando cualquier variable aparece explícitamente en este código se considera como local.

Block [*variables*, *cuerpo*] no considera forma de la expresión *cuerpo*, en vez de ello, a la hora de la evaluación de *cuerpo*, considera los valores locales de las variables *variables*.

Veamos un ejemplo:

Definimos una variable global: $m = i^2$

A continuación establecemos la función **Block** y vemos el resultado obtenido:

Block[{i=a}, i + m]

a + a²

Si utilizamos la función **Module** el resultado es distinto.

Module [{i=a}, i + m]

a + i²

Vemos que en el primer caso, utilizando **Block**, la asignación {i = a} afecta a los valores de la variable en todo el proceso de ejecución. Por contra, la utilización de **Module** sólo afecta a los valores de la variable de forma local.

Mathematica también permite la utilización de estructuras condicionales dentro de los programas lo que permite la consideración de diversas ramas a la hora de proceder a una ejecución. La función condicional clásica es: **If** [condición, t, f], si condición es True entonces ejecuta la rama t, por contra si condición es False, ejecuta la rama f⁹. También es posible una estructura similar a la del **Case**, esto es, un múltiple con la función: **Switch** [expr, forma1, valor1, forma2, valor2, ...].

Para construir los predicados de *condición*, Mathematica dispone de un conjunto de funciones cuyos nombre habitualmente terminan en Q, y que devuelven True o False. Los más habituales son: **Positive**, **Negative**, **EvenQ**, **OddQ**, **MatrixQ**, **NumberQ**, **IntegerQ**, **VectorQ**, **MemberQ**, **PolynomialQ**. Asimismo tenemos disponible un conjunto de operadores y funciones relacionales tales como <, >, <=, >=, =, !=, (o como funciones: **Less**, **Greater**, **LessEqual**, **GreaterEqual**, **Eequal**, **Unequal**). Así como las funciones lógicas **And**, **Or**, **Not** y **Xor**, (los tres primeros tienen sus respectivos operadores lógicos: &&, ||, !).

También existen una serie de funciones que pueden realizar acciones similares a las de un **If**, pero sin necesidad de recurrir al uso explícito del mismo. Es el caso de **Position** que identifica los elementos de una lista que cumplen una condición, devolviendo una lista de posiciones que puede ser utilizada con la función **MapAt**; y

⁹ En el caso de que condición no sea ni True ni False se puede añadir una tercera rama de ejecución **If**[condición, t, f, u].

Select que devuelve una lista con los elementos que cumplen un criterio; por último, Count calcula el número de veces que aparece un elemento particular en una lista.

Veamos un ejemplo:

```
Position [Range[10, 15], _?EvenQ]
{{1}, {3}, {5}}
MapAt [(-1*#)&, Range[10,15], %]
{-10, 11, -12, 13, -14, 15}
Select [Range[ 10, 15], OddQ]
{11, 13, 15}
Count [Table[ {i+1,i}, {i,3}], {_?EvenQ, _?OddQ}]
2
```

Veamos por último un ejemplo de cada uno de los distintos tipos de programación disponible:

a) Programación procedural:

Programa que calcula el primer número primo después de un entero dado n.

```
NextPrime [n_Integer]:=
Module[{k=0}, While[!PrimeQ[k], k++]; Return[k]]
```

b) Programación funcional:

Programa que genera la matriz de Hilbert de orden n x n.

```
Hilbert [n_]:= Table [ 1/(i+j-1), {i,n}, {j,n}]
```

c) Programación de pattern-matching:

Programa que calcula la transformada de Laplace.

Laplace [c_, t_, s_] := c / s /; FreeQ[c,t]

Laplace [a_ + b_, t_, s_] := Laplace [a, t, s] + Laplace [b, t, s]

Laplace [c_ a_, t_, s_] := c Laplace [a, t, s] /; FreeQ[c,t]

Laplace [t_ ^ n_, t_, s_] := n! / s^(n+1) /; (FreeQ[n,t] && n>0)

Laplace [a_ . Exp[b_ + c_ . t_], t_, s_] := Laplace [a Exp[b], t, s - c] /; FreeQ[{b,c}, t]

Bibliografía sobre Mathematica.

Citamos a continuación una pequeña bibliografía sobre el programa Mathematica.

ABELL, M. & BRESELTON, J. (1992), Mathematica by Example. Academic Press, New York.

BLANCHMAN, N. (1993), Mathematica. Un enfoque práctico. Ariel, Barcelona.

CRANDALL, R. (1990), Mathematica for the Sciences. Addison-Wesley, Redwood City, Ca.

MAEDER, R. (1991), Programming in Mathematica. Addison-Wesley, Redwood City, Ca.

SKIENA, S. (1990), Implementing Discrete Mathematics. Combinatorics and Graph Theory with Mathematica. Addison-Wesley, Redwood City, Ca.

WOLFRAM, S. (1991), Mathematica. Addison-Wesley, Redwood City, Ca.

WOLFRAM, S. (1996), The Mathematica Book. Cambridge University Press, Cambridge. (Recoge la actualización al la version 3.0)

PACKAGE: Cativo.m

Este package contiene las funciones implementadas y desarrolladas en el presente estudio para la utilización del método del conjunto activo en los procedimientos de asignación de redes de tráfico.

`BeginPackage["cactivo`"]`

`FactorLQ::usage="FactorLQ[v] calcula la matriz Q de la factorización LQ, donde L es una matriz triangular inferior y Q es ortogonal, de una matriz v."`

`FactorQR::usage="FactorQR[v] calcula la matriz Q, ortogonal, que nos factoriza la matriz v, de forma tal, que $Q \cdot v = R$, donde R es una matriz triangular superior."`

`Houseanula::usage="Houseanula[v,i] calcula la matriz de Householder que anula los elementos del vector v, por debajo de la posición i-ésima."`

`Gradiente::usage="Gradiente[expr,var] calcula la función gradiente de la función expr respecto a la lista de variables var."`

`BFGS::usage="BFGS[Matriz,vector_v,vector_t] calcula la adaptación BFGS de la matriz Matriz, utilizando los vectores vector_v,vector_t, donde v, recoge la variación en la solución, y t, la variación en el gradiente"`

`DirBus::usage="DirBus[Matriz,Matriz,vector] calcula la dirección de búsqueda usando la matriz Z, y la estimación quasinewton de la hessiana, y el vector gradiente, en el método del conjunto activo"`

EspacioNulo::usage="EspacioNulo[Matriz,var1,var2] calcula la matriz del espacio nulo de Matriz, para la variable var1=n° de restricciones del problema y var2=n° de variables del problema"

Evaluador::usage="Evaluador[expr, vector1,vector2] calcula el valor de la expresión expr, sustituyendo las variables del vector1 con los valores de vector2"

Cota::usage="cota[vector1,vector2] calcula la longitud de paso máxima, para vector1, dirección de búsqueda y para vector2, solución anterior"

TestCo::usage="TestCo[vector1,vector2] calcula el test de convergencia entre vector1, nueva solución y vector2, solución anterior"

IteracionCA::usage="IteraciónCA[a_List, b_List,c_List,e_List, f_List] calcula una iteración mediante el método del conjunto activo, superior a la primera, donde a es el vector de variables del problema, b es el vector gradiente, c es la solución anterior, e es la solución con dos retardos, y f es la matriz auxiliar de la iteración anterior."

AdaptaEN::usage="AdaptaEN[a_List, A_List, A1_List, n,,l] calcula la nueva matriz del espacio nulo cuando añadimos una restricción, a es la restricción añadida, A es la matriz anterior del conjunto activo, A1 es la matriz Q de la factorización LQ de A, y n es el número de restricciones y l es el de variables. $r < l$."

Begin["Private"]

Normavector [a_List]:=N[Sqrt[Dot[a,a]],10]

**Householder [b_List]:= IdentityMatrix[Length[b]] -
(2/Normavector[b]^2)* Outer[Times,b,b]**

```

Houseanula [v_List,n_Integer]:=
Module[{I=IdentityMatrix[Length[v]],
H=Householder[
With[
{t=v[[Range[n,Length[v]],Range[n,n]]},
E=IdentityMatrix[Length[v]-(n-1)]
},
Normavector[Map#[[1]]&,t]]*
Map#[[1]]&,E]+Map#[[1]]&,t]],
If[n>1,k=n-1;Do[Do[H[[i]]=Prepend[H[[i]],0],
{i,Length[v]-(n-1)}],{n-1}];Do[H=Join[{H[[j]]},H],{j,k,1,-1}];H]

```

```

FactorQR[w_List]:=
Module[{T=w,Q=IdentityMatrix[Length[w]]},
Do[H=Houseanula[T,i];T=H.T;Q=Q.H,
{i,1,Min[Dimensions[w]]}];
Q]

```

```

FactorLQ[v_List]:=
Module[{Q=FactorQR[Transpose[v]]},
Q]

```

```

Gradiente[expr_,var_List]:=
Module[{g={D[expr,#]& /@ var}},
Transpose[g]]

```

```

Hessiana[expr_,var_List]:=
Module[{H=D[D[expr,#]& /@ var,#] & /@ var},
Transpose[H]]

```

```

BFGS[v_List,w_List,t_List]:=
v-(First[1/Flatten[Transpose[w].v.w]]*(v.w.Transpose[w].v)+
(First[1/Flatten[Transpose[t].w]]*(t.Transpose[t]))

```

```

DirBus[Z_List,B_List,g_List]:=
Z.(Inverse[Transpose[Z]].B.Z).(Transpose[Z].g)

```

```
EspacioNulo[A_List,nv_Integer,L_Integer]:=
A[Range[1,L],Range[nv+1,L]]
```

```
Evaluardor[f_,b_List,c_List]:=
Module[{e={},l=Length[c]},
Do[e=Append[e,b[[i]]->c[[i]],{i,1,l}];
Flatten[f/.{e},2]]
```

```
Cota[a_List,b_List]:=
Min[Min[Abs[(b/a)],1]
```

```
TestCo[a_List,b_List]:=Max[Abs[(a-b)/b]*100]
```

```
IteracionCA[a_List,b_List,c_List,e_List,f_List]:=
Module[{gl=Evaluardor[b,a,c],d=Evaluardor[b,a,c]},B=BFGS[f,c-e,gl-d];
p=DirBus[Z,B,gl];alfa=Cota[p,c];Sol=c+alfa*p]
```

```
AdaptaEN[a_List,A_List,A1_List,n,l]:=
Module[{L=A.A1,Q=a.A1},
W=Join[L,Q];H=Houseanula[Transpose[W],n];
QA=A1.H;EspacioNulo[QA,n,l]]
```

```
End []
```

```
EndPackage[]
```

APENDICE C:
RESULTADOS

APENDICE C: RESULTADOS

El presente apéndice contiene los resultados generados por el programa Mathematica al aplicarse a distintos ejemplos de redes comentados en este estudio.

En algunos casos los ejemplos aparecen con las celdas cerradas, como forma de ahorrar espacio, y dado que los resultados tienen carácter intermedio y no se han considerado relevantes.

1. Ejemplo de Red 5.1 (Tomado de Sheffi, 1985)

Consideramos una red formada por un nodo origen y un nodo destino conectados por 3 arcos. Las funciones de coste de cada arco están basadas en la función flujo-coste BPR(1964). Las funciones de coste tienen una parte independiente del flujo y la capacidad, dado por la constante C_0 , y un coste que se relaciona directamente con el grado de congestión del arco.

La fórmula general de la función de coste a corto plazo para cada usuario es:

$C = CF(1 + a(x/K)^b)$, donde CF es coste con flujo libre, y K es la capacidad máxima del arco.

El usuario se enfrenta a un coste total $= C$, sin peajes.

Método del Conjunto Activo

<<cactivo.m

Funciones de Coste del Usuario:

$$C1 := 10 + 1.5 * (x1/2)^4$$

$$C2 := 20 + 3 * (x2/4)^4$$

$$C3 := 25 + 3.75 * (x3/3)^4$$

arcos:=3

restricciones:=1

$x := \{x1, x2, x3\}$

$c := \{C1, C2, C3\}$

Función Objetivo:

$$F = \text{Integrate}[C1, \{x1, 0, x1\}] + \text{Integrate}[C2, \{x2, 0, x2\}] + \text{Integrate}[C3, \{x3, 0, x3\}]$$

$$10 x1 + 0.01875 x1^5 + 20 x2 + 30/1280 x2^5 + 25 x3 + 0.00925926 x3^5$$

RESTRICCIONES:

La única restricción del problema viene dada por la tasa de usuarios entre el nodo origen y el nodo destino, dado que estamos ante una aproximación con demanda fija, se reduce a que la tasa total de usuarios es 10.

Así pues, el flujo por los tres arcos deberá ser igual a 10.

$$x1 + x2 + x3 = 10$$

Solución Inicial

Sol0={{2},{4},{4}}

{{2}, {4}, {4}}

Evaluador[F,x,Sol0]

{212.481}

Gradiente función Objetivo

g=Gradiente[F,x]

{{10 + 0.09375 x1⁴}, {20 + 3/256 x2⁴}, {25 + 0.0462963 x3⁴}}

A:={{1,1,1}}

A1:=FactorLQ[A]

Z=EspacioNulo[A1,restricciones,arcos]

{{-0.57735, -0.57735}, {0.788675, -0.211325}, {-0.211325, 0.788675}}

g0=Evaluador[g,x,Sol0]

{{11.5}, {23}, {36.8519}}

B0=IdentityMatrix[arcos]

{{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

1ª Iteración

p0=DirBus[Z,B0,g0]

{{12.284}, {0.783951}, {-13.0679}}

alfa=Cota[p0,Sol0]

0.162814

Sol1=Sol0+alfa*p0

{{4.}, {4:12764}, {1.87236}}

Evaluador[F,x,Sol1]

{191.583}

TestCo[Sol1,Sol0]

100.

2ª Iteración

Sol2=IteracionCA[x,g,Sol1,Sol0,B0]

{{2.32143}, {7.56137}, {0.1172}}

Evaluador[F,x,Sol2]

{236.567}

TestCo[Sol2,Sol1]

93.7405

3ª Iteración

Sol3=IteracionCA[x,g,Sol2,Sol1,B]

{{2.38601}, {7.37959}, {0.2344}}

Evaluador[F,x,Sol3]

{230.057}

TestCo[Sol3,Sol2]

100.

4ª Iteración

Sol4=IteracionCA[x,g,Sol3,Sol2,B]

{{3.02351}, {6.50769}, {0.468801}}

Evaluador[F,x,Sol4]

{204.202}

TestCo[Sol4,Sol3]

100.

5ª Iteración

Sol5=IteracionCA[x,g,Sol4,Sol3,B]

{{3.889}, {5.4607}, {0.650294}}

Evaluador[F,x,Sol5]

{192.423}

TestCo[Sol5,Sol4]

38.7143

6ª Iteración

Sol6=IteracionCA[x,g,Sol5,Sol4,B]

{{3.62727}, {5.11592}, {1.2568}}

Evaluador[F,x,Sol6]

{190.027}

TestCo[Sol6,Sol5]

93.2668

7ª Iteración

Sol7=IteracionCA[x,g,Sol6,Sol5,B]

{{3.56648}, {4.7272}, {1.70632}}

Evaluador[F,x,Sol7]

{189.353}

TestCo[Sol7,Sol6]

35.7672

8ª Iteración

Sol8=IteracionCA[x,g,Sol7,Sol6,B]

{{3.57713}, {4.65413}, {1.76874}}

N[Evaluador[F,x,Sol8],10]

{189.332563}

TestCo[Sol8,Sol7]

3.65816

9ª Iteración

Sol9=IteracionCA[x,g,Sol8,Sol7,B]

{{3.58271}, {4.64513}, {1.77216}}

N[Evaluador[F,x,Sol9],10]

{189.3320446}

TestCo[Sol9,Sol8]

0.193443

10ª Iteración

Sol10=IteracionCA[x,g,Sol9,Sol8,B]

{{3.58327}, {4.6451}, {1.77164}}

N[Evaluador[F,x,Sol10],10]

{189.3320416}

TestCo[Sol10,Sol9]

0.0295183

2. Ejemplo de Red (Tomado de Hearn & Ribera , 1980)

Este ejemplo se configura como una red con 18 arcos y 9 nodos, estableciéndose un total de 2 pares de nodos origen (1 y 2), y 2 pares de nodos destino (3 y 4), existiendo una demanda fija entre ambos pares de nodos. El número de restricciones existentes es de 9, y el número de variables es de 18, lo que garantiza la aplicabilidad del método del conjunto activo.

Las demandas entre nodos son:

De 1 a 3: 10

De 1 a 4: 20

De 2 a 3: 30

De 2 a 4: 40

<<cactivo.m

Definición de Número de Restricciones y Número de Variables

Nº de variables=Nº de arcos en la red.

Nº de restricciones= Ecuaciones de Flujo

arcos:=18

restricciones:=9

x:={x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,x13,x14,
x15,x16,x17,x18}

Función de coste.

c1:=5(1+0.15(x1/12)^4);c2:=6(1+0.15(x2/18)^4);
c3:=3(1+0.15(x3/35)^4);c4:=9(1+0.15(x4/35)^4);
c5:=9(1+0.15(x5/20)^4);c6:=2(1+0.15(x6/11)^4);
c7:=8(1+0.15(x7/26)^4);c8:=4(1+0.15(x8/11)^4);
c9:=6(1+0.15(x9/33)^4);c10:=7(1+0.15(x10/32)^4);
c11:=3(1+0.15(x11/25)^4);c12:=6(1+0.15(x12/24)^4);
c13:=2(1+0.15(x13/19)^4);c14:=8(1+0.15(x14/39)^4);
c15:=6(1+0.15(x15/43)^4);c16:=4(1+0.15(x16/36)^4);
c17:=4(1+0.15(x17/26)^4);c18:=8(1+0.15(x18/30)^4)

Función Objetivo.

F=Integrate[c1,{x1,0,x1}]+Integrate[c2,{x2,0,x2}]+
Integrate[c3,{x3,0,x3}]+Integrate[c4,{x4,0,x4}]+
Integrate[c5,{x5,0,x5}]+Integrate[c6,{x6,0,x6}]+
Integrate[c7,{x7,0,x7}]+Integrate[c8,{x8,0,x8}]+
Integrate[c9,{x9,0,x9}]+Integrate[c10,{x10,0,x10}]+
Integrate[c11,{x11,0,x11}]+Integrate[c12,{x12,0,x12}]+
Integrate[c13,{x13,0,x13}]+Integrate[c14,{x14,0,x14}]+
Integrate[c15,{x15,0,x15}]+Integrate[c16,{x16,0,x16}]+
Integrate[c17,{x17,0,x17}]+Integrate[c18,{x18,0,x18}]

Determinamos el punto inicial, lo que determina un valor de la función objetivo

Sol0:={{12},{18},{35},{35},{12},{13},{24},{2},{32},{31},
{25},{17},{2},{15},{43},{5},{26},{29}}

Evaluar[F,x,Sol0]

{2300.63}

Gradiente de la función objetivo

g=Gradiente[F,{x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,
x11,x12,x13,x14,x15,x16,x17,x18}]

Matriz de Restricciones (Columna: variables)

A={{1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{1,0,1,0,-1,-1,-1,1,0,0,0,0,0,0,0,0,0,0}}

```

{0,1,0,1,1,0,0,-1,-1,-1,0,0,0,0,0,0,0},
{0,0,0,0,0,0,1,0,0,1,0,0,0,0,0,-1,-1},
{0,0,0,0,0,1,0,0,0,0,-1,-1,-1,0,0,1,1,0},
{0,0,0,0,0,0,0,1,0,0,0,1,-1,-1,-1,0,1},
{0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0}}

Vector constantes lado derecho
c:={{30},{70},{0},{0},{0},{0},{0},{40},{60}}
Factorización LQ de la matriz de Restricciones.
A1=FactorLQ[A]
Z=EspacioNulo[A1,restricciones,arcos]
Matriz Auxiliar
B=IdentityMatrix[arcos]
Iteración Inicial:
B0:=B
g0=Evaluador[g,x,Sol0]
{{5.75}, {6.9}, {3.45}, {10.35}, {9.17496}, {2.58523}, {8.87123}, {4.00066}, {6.79577},
{7.92478}, {3.45}, {6.22657}, {2.00004}, {8.02626}, {6.9}, {4.00022}, {4.6}, {9.04782}}
p0=DirBus[Z,B0,g0]
{{2.74026}, {-2.74026}, {0.28646}, {-0.28646}, {-5.14129}, {6.5741}, {-2.60814}, {-4.20205},
{0.19265}, {-4.15861}, {2.61733}, {0.31938}, {-1.72104}, {-2.61733}, {-0.31938}, {-4.35863}, {-
0.999795}, {-5.76695}}
alfa=Cota[p0,Sol0]
0.475958
Sol1=Sol0+alfa*p0
{{13.3042}, {16.6958}, {35.1363}, {34.8637}, {9.55296}, {16.129}, {22.7586}, {0.}, {32.0917},
{29.0207}, {26.2457}, {17.152}, {1.18086},{13.7543}, {42.848}, {2.92547}, {25.5241},
{26.2552}}
Evaluador[F,x,Sol1]
{2209.54}
a={{0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0}}
Z=AdaptaEN[a,A,A1,10,18]
A=Join[A,a]
2ª Iteración
g1=Evaluador[g,x,Sol1]
{{6.13318}, {6.66616}, {3.45705}, {10.3291}, {9.07027}, {3.38669}, {8.70449}, {4.}, {6.80493},
{7.71026}, {3.54662}, {6.23478}, {2.}, {8.01856}, {6.88734}, {4.00003}, {4.55727}, {8.70397}}

```

B1=BFGS[B0,Sol1-Sol0,g1-g0]

p1=DirBus[Z,B1,g1]

{{3.43225}, {-3.43225}, {-0.314846}, {0.314846}, {-16.1822}, {25.5294}, {-6.22982}, {5.67278
10⁻¹⁵}, {-1.08884}, {-18.2107}, {10.3109}, {1.24314},{-7.53133}, {-10.3109}, {-1.24314}, {-
18.0119}, {-3.49473}, {-20.9458}}

Abs[Sol1/p1]

{{3.87625}, {4.86438}, {111.599}, {110.733}, {0.590339}, {0.631782}, {3.65318}, {0.},
{29.4732}, {1.5936}, {2.54543}, {13.7973}, {0.156792}, {1.33395}, {34.4675}, {0.162419},
{7.3036}, {1.25348}}

alfa=0.156792

0.156792

Sol2=Sol1+alfa*p1

{{13.8424}, {16.1576}, {35.087}, {34.913}, {7.01572},{20.1318}, {21.7819}, {8.89447 10⁻¹⁶},
{31.921}, {26.1654}, {27.8624}, {17.3469}, {2.93627 10⁻⁶}, {12.1376}, {42.6531}, {0.101345},
{24.9762}, {22.971}}

Evaluar[F,x,Sol2]

{2121.1}

a={{0,0}}

Z=AdaptaEN[a,A,QA,11,18]

A=Join[A,a]

3ª Iteración

g2=Evaluar[g,x,Sol2]

{{6.32795}, {6.58433}, {3.45449}, {10.3366}, {9.02044}, {5.36574}, {8.59111}, {4.}, {6.78794},
{7.46935}, {3.69427}, {6.24563}, {2.}, {8.01126}, {6.8713}, {4.}, {4.51093}, {8.41249}}

B2=BFGS[B1,Sol2-Sol1,g2-g1]

p2=DirBus[Z,B2,g2]

{{3.09721}, {-3.09721}, {-1.10317}, {1.10317}, {-18.5725}, {25.106}, {-4.53944}, {6.02798 10⁻¹⁵
}, {-0.671035}, {-19.8955}, {9.70962}, {-1.20872}, {-2.08357 10⁻¹⁶}, {-9.70962}, {1.20872}, {-
17.1966}, {0.591522}, {-25.0265}}

Abs[Sol2/p2]

{{4.46931}, {5.21682}, {31.8056}, {31.6479}, {0.377748}, {0.801872}, {4.79835}, {0.147553},
{47.5698}, {1.31514}, {2.86957}, {14.3515}, {1.40925 10¹⁰}, {1.25006}, {35.2878},
{0.00589331}, {42.2236}, {0.917869}}

alfa=0.00589331

0.00589331

Sol3=Sol2+alfa*p2

{{13.8607}, {16.1393}, {35.0805}, {34.9195}, {6.90627}, {20.2798}, {21.7551}, {9.24972 10⁻¹⁶},

{31.917}, {26.0481}, {27.9196}, {17.3398}, {2.93627 10⁻⁶}, {12.0804}, {42.6602}, {7.43345 10⁻⁸}, {24.9797}, {22.8235}}

Evaluador[F,x,Sol3]

{2117.95}

a={{0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0}}

Z=AdaptaEN[a,A,QA,12,18]

A=Join[A,a]

4ª Iteración

g3=Evaluador[g,x,Sol3]

{{6.33497}, {6.5817}, {3.45415}, {10.3376}, {9.01919}, {5.46579}, {8.58821}, {4.}, {6.78755}, {7.46099}, {3.69999}, {6.24523}, {2.}, {8.01105}, {6.87189}, {4.}, {4.51122}, {8.402}}

B3=BFGS[B2,Sol3-Sol2,g3-g2]

p3=DirBus[Z,B3,g3]

{{1.27732},{-1.27732}, {-1.79254}, {1.79254}, {-10.7697}, {11.3269},{-1.07241}, {3.09002 10⁻¹⁵}, {2.99532}, {-13.2498}, {8.87183}, {1.54103},{1.62476 10⁻¹⁵},{-8.87183}, {-1.54103}, {-2.75982 10⁻¹⁵}, {-0.914017}, {-13.4082}}

Abs[Sol3/p3]

{{10.8513}, {12.6353}, {19.5703}, {19.4805}, {0.64127}, {1.79041}, {20.2861}, {0.299341}, {10.6556}, {1.96593}, {3.147}, {11.2521}, {1.80721 10⁹}, {1.36165}, {27.683}, {2.69346 10⁷}, {27.3295}, {1.70221}}

alfa=0.64127

0.64127

Sol4=Sol3+alfa*p3

{{14.6798}, {15.3202}, {33.931}, {36.069}, {-2.95495 10⁻⁶}, {27.5433}, {21.0674}, {2.90651 10⁻¹⁵}, {33.8378}, {17.5514}, {33.6089}, {18.328}, {2.93627 10⁻⁶}, {6.39113}, {41.672}, {7.43345 10⁻⁸}, {24.3935}, {14.2253}}

Evaluador[F,x,Sol4]

{1978.91}

a={{0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0}}

Z=AdaptaEN[a,A,QA,13,18]

A=Join[A,a]

5ª Iteración

g4=Evaluador[g,x,Sol4]

{{6.67962}, {6.4723}, {3.39749}, {10.5226}, {9.}, {13.7928}, {8.51729}, {4.}, {6.99494}, {7.09503}, {4.46983}, {6.3061}, {2.}, {8.00087}, {6.79386}, {4.}, {4.4649}, {8.06067}}

B4=BFGS[B3,Sol4-Sol3,g4-g3]

p4=DirBus[Z,B4,g4]

Z=AdaptaEN[a,A,QA,14,18]

A=Join[A,a]

7ª Iteración

g6=Evaluador[g,x,Sol6]

{{6.89831}, {6.41851}, {3.48635}, {10.2472}, {9.}, {19.8776}, {8.4424}, {4.}, {8.19869},
{7.00394}, {5.94912}, {6.2879}, {2.}, {8.}, {6.81532}, {4.}, {4.74946}, {8.}}

B6=BFGS[B5,Sol6-Sol5,g6-g5]

p6=DirBus[Z,B6,g6]

{{0.00752017}, {-0.00752017}, {0.188811}, {-0.188811}, {-1.92095 10⁻¹⁶}, {-1.13273},
{1.32906}, {9.28417 10⁻¹⁶}, {7.82747}, {-8.02381}, {2.40225 10⁻¹⁶}, {1.17057}, {1.79339 10⁻¹⁶},
{-1.22124 10⁻¹⁷}, {-1.17057}, {-2.20929 10⁻¹⁵}, {2.3033}, {-8.99805}}

Abs[Sol6/p6]

{{2012.7}, {1976.57}, {189.006}, {181.735}, {1.53828 10¹⁰}, {26.9813}, {15.2436}, {5.48768},
{5.27075}, {0.987196}, {1.66511 10¹⁷}, {15.4192}, {1.63727 10¹⁰}, {1.56831 10¹¹}, {35.8378},
{3.36463 10⁷}, {11.9336}, {0.0771294}}

alfa=0.0771294

0.0771294

Sol7=Sol6+alfa*p6

{{15.1365}, {14.8635}, {35.7009}, {34.2991}, {-2.95495 10⁻⁶}, {30.4752}, {20.3622}, {5.16646
10⁻¹⁵}, {41.8604}, {7.3022}, {40.}, {18.1396}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {41.8604},
{7.43345 10⁻⁸}, {27.6644}, {3.17166 10⁻⁷}}

Evaluador[F,x,Sol7]

{1881.21}

a={{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1}}

Z=AdaptaEN[a,A,QA,15,18]

A=Join[A,a]

8ª Iteración

g7=Evaluador[g,x,Sol7]

{{6.8986}, {6.41845}, {3.48714}, {10.2451}, {9.}, {19.674}, {8.45143}, {4.}, {8.33024},
{7.00285}, {5.94912}, {6.2937}, {2.}, {8.}, {6.80832}, {4.}, {4.76903}, {8.}}

B7=BFGS[B6,Sol7-Sol6,g7-g6]

p7=DirBus[Z,B7,g7]

{{0.0843182}, {-0.0843182}, {0.334514}, {-0.334514}, {1.96655 10⁻¹⁷}, {-1.45963}, {1.87846},
{4.52759 10⁻¹⁶}, {0.937982}, {-1.35681}, {5.89601 10⁻¹⁷}, {-0.937982}, {-3.09137 10⁻¹⁶},
{9.89125 10⁻¹⁸}, {0.937982}, {4.89013 10⁻¹⁷}, {0.521648}, {-2.04138 10⁻¹⁶}}

Abs[Sol7/p7]

{{179.516}, {176.279}, {106.725}, {102.534}, {1.50261 10¹¹}, {20.8787}, {10.8398}, {11.4111},
 {44.6282}, {5.38187}, {6.78425 10¹⁷}, {19.3389}, {9.49831 10⁹}, {1.93634 10¹¹}, {44.6282},
 {1.52009 10⁹}, {53.0327}, {1.55369 10⁹}}

alfa=1

1

Sol8=Sol7+alfa*p7

{{15.2208}, {14.7792}, {36.0354}, {33.9646}, {-2.95495 10⁻⁶}, {29.0156}, {22.2407}, {5.61922
 10⁻¹⁵}, {42.7984}, {5.94538}, {40.}, {17.2016}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {42.7984},
 {7.43345 10⁻⁸}, {28.186}, {3.17166 10⁻⁷}}

Evaluador[F,x,Sol8]

{1870.15}

9ª Iteración.

g8=Evaluador[g,x,Sol8]

{{6.94126}, {6.40903}, {3.50566}, {10.1972}, {9.}, {16.5236}, {8.64251}, {4.}, {8.54623},
 {7.00125}, {5.94912}, {6.2375}, {2.}, {8.}, {6.88324}, {4.}, {4.82869}, {8.}}

B8=BFGS[B7,Sol8-Sol7,g8-g7]

p8=DirBus[Z,B8,g8]

{{0.490867}, {-0.490867}, {1.10856}, {-1.10856}, {-9.59106 10⁻¹⁷}, {-2.19877}, {3.7982},
 {1.25484 10⁻¹⁵}, {1.90318}, {-3.5026}, {2.63466 10⁻¹⁶}, {-1.90318}, {-7.64305 10⁻¹⁶}, {-2.66124
 10⁻¹⁷}, {1.90318}, {3.03861 10⁻¹⁸}, {0.295592}, {-4.62383 10⁻¹⁶}}

Abs[Sol8/p8]

{{31.0079}, {30.1084}, {32.5066}, {30.6385}, {3.08094 10¹⁰}, {13.1963}, {5.85558}, {4.47804},
 {22.4878}, {1.69742}, {1.51822 10¹⁷}, {9.03834}, {3.84176 10⁹}, {7.19695 10¹⁰},
 {22.4878}, {2.44633 10¹⁰}, {95.3546}, {6.85939 10⁸}}

alfa=1

1

Sol9=Sol8+alfa*p8

{{15.7116}, {14.2884}, {37.144}, {32.856}, {-2.95495 10⁻⁶}, {26.8168}, {26.0389}, {6.87406 10⁻¹⁵},
 {44.7016}, {2.44278}, {40.}, {15.2984}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {44.7016},
 {7.43345 10⁻⁸}, {28.4816}, {3.17166 10⁻⁷}}

Evaluador[F,x,Sol9]

{1860.26}

10ª Iteración.

g9=Evaluador[g,x,Sol9]

{{7.20406}, {6.35734}, {3.57081}, {10.0484}, {9.}, {12.5969}, {9.20719}, {4.}, {9.03025},
 {7.00004}, {5.94912}, {6.14859}, {2.}, {8.}, {7.05114}, {4.}, {4.86401}, {8.}}

B9=BFGS[B8,Sol9-Sol8,g9-g8]

p9=DirBus[Z,B9,g9]

{{0.43794}, {-0.43794}, {0.784334}, {-0.784334}, {-1.36364 10⁻¹⁶}, {-0.451891}, {1.67416},
{7.88724 10⁻¹⁶}, {0.885638}, {-2.10791}, {2.17553 10⁻¹⁶}, {-0.885638}, {-4.28934 10⁻¹⁶}, {-4.43675
10⁻¹⁷}, {0.885638}, {-6.67856 10⁻¹⁷}, {-0.433747}, {-2.46107 10⁻¹⁶}}

Abs[Sol9/p9]

{{35.8763}, {32.6263}, {47.3574}, {41.8903}, {2.16696 10¹⁰}, {59.3435}, {15.5533}, {8.71543},
{50.4739}, {1.15886}, {1.83863 10¹⁷}, {17.2739}, {6.84551 10⁹}, {4.31686 10¹⁰}, {50.4739},
{1.11303 10⁹}, {65.6641}, {1.28873 10⁹}}

alfa=1

1

Sol10=Sol9+alfa*p9

{{16.1496}, {13.8504}, {37.9283}, {32.0717}, {-2.95495 10⁻⁶}, {26.3649}, {27.713}, {7.66279 10⁻¹⁵},
{45.5872}, {0.334867}, {40.}, {14.4128}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {45.5872}, {7.43345
10⁻⁸}, {28.0479}, {3.17166 10⁻⁷}}

Evaluar[F,x,Sol10]

{1857.93}

11ª Iteración.

g10=Evaluar[g,x,Sol10]

{{7.46026}, {6.3155}, {3.62058}, {9.95181}, {9.}, {11.9004}, {9.5489}, {4.}, {9.27763}, {7.},
{5.94912}, {6.11705}, {2.}, {8.}, {7.13695}, {4.}, {4.81257}, {8.}}

B10=BFGS[B9,Sol10-Sol9,g10-g9]

p10=DirBus[Z,B10,g10]

{{0.787642}, {-0.787642}, {1.28938}, {-1.28938}, {-2.74654 10⁻¹⁶}, {0.0472443}, {2.02977},
{1.23036 10⁻¹⁵}, {1.18223}, {-3.25925}, {3.8339 10⁻¹⁶}, {-1.18223}, {-6.21196 10⁻¹⁶}, {-9.21144
10⁻¹⁷}, {1.18223}, {-1.62261 10⁻¹⁶}, {-1.22948}, {-3.57653 10⁻¹⁶}}

Abs[Sol10/p10]

{{20.5037}, {17.5847}, {29.416}, {24.8738}, {1.07588 10¹⁰}, {558.055}, {13.6533}, {6.22807},
{38.5603}, {0.102744}, {1.04332 10¹⁷}, {12.1912}, {4.72681 10⁹}, {2.07924 10¹⁰}, {38.5603},
{4.58117 10⁸}, {22.8129}, {8.86797 10⁸}}

alfa=0.102744

0.102744

Sol11=Sol10+alfa*p10

{{16.2305}, {13.7695}, {38.0608}, {31.9392}, {-2.95495 10⁻⁶}, {26.3697}, {27.9216}, {7.7892 10⁻¹⁵},
{45.7087}, {-1.27685 10⁻⁶}, {40.}, {14.2913}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {45.7087},
{7.43345 10⁻⁸}, {27.9216}, {3.17166 10⁻⁷}}

Evaluar[F,x,Sol11]

{1857.55}

$a = \{ \{ 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0 \} \}$

$Z = \text{AdaptaEN}[a, A, QA, 16, 18]$

$A = \text{Join}[A, a]$

12ª Iteración.

$g11 = \text{Evaluador}[g, x, \text{Sol11}]$

$\{ \{ 7.50995 \}, \{ 6.30819 \}, \{ 3.62929 \}, \{ 9.93618 \}, \{ 9. \}, \{ 11.9077 \}, \{ 9.59605 \}, \{ 4. \}, \{ 9.3127 \}, \{ 7. \}, \{ 5.94912 \}, \{ 6.11316 \}, \{ 2. \}, \{ 8. \}, \{ 7.14912 \}, \{ 4. \}, \{ 4.79803 \}, \{ 8. \} \}$

$B11 = \text{BFGS}[B10, \text{Sol11} - \text{Sol10}, g11 - g10]$

$p11 = \text{DirBus}[Z, B11, g11]$

$\{ \{ 0.0815159 \}, \{ -0.0815159 \}, \{ 0.0586104 \}, \{ -0.0586104 \}, \{ -4.66571 \cdot 10^{-17} \}, \{ 0.54002 \}, \{ -0.399894 \}, \{ 1.02081 \cdot 10^{-17} \}, \{ -0.140126 \}, \{ -2.36411 \cdot 10^{-17} \}, \{ 3.44934 \cdot 10^{-17} \}, \{ 0.140126 \}, \{ 2.83796 \cdot 10^{-17} \}, \{ -1.71222 \cdot 10^{-17} \}, \{ -0.140126 \}, \{ -4.26759 \cdot 10^{-17} \}, \{ -0.399894 \}, \{ 1.71757 \cdot 10^{-17} \} \}$

$\text{Abs}[\text{Sol11}/p11]$

$\{ \{ 199.109 \}, \{ 168.918 \}, \{ 649.386 \}, \{ 544.941 \}, \{ 6.33333 \cdot 10^{10} \}, \{ 48.831 \}, \{ 69.8224 \}, \{ 763.038 \}, \{ 326.196 \}, \{ 5.40098 \cdot 10^{10} \}, \{ 1.15964 \cdot 10^{18} \}, \{ 101.989 \}, \{ 1.03464 \cdot 10^{11} \}, \{ 1.11859 \cdot 10^{11} \}, \{ 326.196 \}, \{ 1.74184 \cdot 10^9 \}, \{ 69.8224 \}, \{ 1.8466 \cdot 10^{10} \} \}$

$\text{alfa} = 1$

1

$\text{Sol12} = \text{Sol11} + \text{alfa} \cdot p11$

$\{ \{ 16.312 \}, \{ 13.688 \}, \{ 38.1194 \}, \{ 31.8806 \}, \{ -2.95495 \cdot 10^{-6} \}, \{ 26.9098 \}, \{ 27.5217 \}, \{ 7.79941 \cdot 10^{15} \}, \{ 45.5686 \}, \{ -1.27685 \cdot 10^{-6} \}, \{ 40. \}, \{ 14.4314 \}, \{ 2.93627 \cdot 10^{-6} \}, \{ -1.91528 \cdot 10^{-6} \}, \{ 45.5686 \}, \{ 7.43345 \cdot 10^{-8} \}, \{ 27.5217 \}, \{ 3.17166 \cdot 10^{-7} \} \}$

$\text{Evaluador}[F, x, \text{Sol12}]$

$\{ 1856.76 \}$

13ª Iteración.

$g12 = \text{Evaluador}[g, x, \text{Sol12}]$

$\{ \{ 7.56075 \}, \{ 6.30096 \}, \{ 3.63318 \}, \{ 9.92932 \}, \{ 9. \}, \{ 12.7446 \}, \{ 9.50656 \}, \{ 4. \}, \{ 9.27226 \}, \{ 7. \}, \{ 5.94912 \}, \{ 6.11766 \}, \{ 2. \}, \{ 8. \}, \{ 7.13509 \}, \{ 4. \}, \{ 4.75328 \}, \{ 8. \} \}$

$B12 = \text{BFGS}[B11, \text{Sol12} - \text{Sol11}, g12 - g11]$

$p12 = \text{DirBus}[Z, B12, g12]$

$\{ \{ 0.0777264 \}, \{ -0.0777264 \}, \{ 0.0490798 \}, \{ -0.0490798 \}, \{ -4.58463 \cdot 10^{-17} \}, \{ 0.548627 \}, \{ -0.421821 \}, \{ 2.47163 \cdot 10^{-18} \}, \{ -0.126806 \}, \{ -2.74258 \cdot 10^{-17} \}, \{ 3.34984 \cdot 10^{-17} \}, \{ 0.126806 \}, \{ 3.56602 \cdot 10^{-17} \}, \{ -1.70602 \cdot 10^{-17} \}, \{ -0.126806 \}, \{ -4.347 \cdot 10^{-17} \}, \{ -0.421821 \}, \{ 1.53919 \cdot 10^{-17} \} \}$

$\text{Abs}[\text{Sol12}/p12]$

$\{ \{ 209.865 \}, \{ 176.104 \}, \{ 776.683 \}, \{ 649.567 \}, \{ 6.44534 \cdot 10^{10} \}, \{ 49.0493 \}, \{ 65.2449 \}, \{ 3155.57 \}, \{ 359.356 \}, \{ 4.65564 \cdot 10^{10} \}, \{ 1.19409 \cdot 10^{18} \}, \{ 113.807 \}, \{ 8.23405 \cdot 10^{10} \}, \{ 1.12266 \cdot 10^{11} \}, \{ 359.356 \}, \{ 1.71002 \cdot 10^9 \}, \{ 65.2449 \}, \{ 2.0606 \cdot 10^{10} \} \}$

alfa=1

1

Sol13=Sol12+alfa*p12

{{16.3898}, {13.6102}, {38.1685}, {31.8315}, {-2.95495 10⁻⁶}, {27.4584}, {27.0998}, {7.80188 10⁻¹⁵}, {45.4418}, {-1.27685 10⁻⁶}, {40.}, {14.5582}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {45.4418}, {7.43345 10⁻⁸}, {27.0998}, {3.17166 10⁻⁷}}

Evaluardor[F,x,Sol13]

{1856.5}

14ª Iteración.

g13=Evaluardor[g,x,Sol13]

{{7.60991}, {6.29418}, {3.63644}, {9.92361}, {9.}, {13.648}, {9.4163}, {4.}, {9.23599}, {7.}, {5.94912}, {6.12185}, {2.}, {8.}, {7.12251}, {4.}, {4.70815}, {8.}}

B13=BFGS[B12,Sol13-Sol12,g13-g12]

p13=DirBus[Z,B13,g13]

{{-0.00842712}, {0.00842712}, {-0.0121972}, {0.0121972}, {3.59856 10⁻¹⁸}, {-0.0254251}, {0.00480081}, {-7.60453 10⁻¹⁸}, {0.0206243}, {-1.96044 10⁻¹⁸}, {-3.01713 10⁻¹⁸}, {-0.0206243}, {4.82193 10⁻¹⁸}, {1.10822 10⁻¹⁸}, {0.0206243}, {1.90657 10⁻¹⁸}, {0.00480081}, {-2.66423 10⁻¹⁸}}

Abs[Sol13/p13]

{{1944.88}, {1615.05}, {3129.29}, {2609.75}, {8.21147 10¹¹}, {1079.97}, {5644.84}, {1025.95}, {2203.31}, {6.51307 10¹¹}, {1.32576 10¹⁹}, {705.878}, {6.08942 10¹¹}, {1.72825 10¹²}, {2203.31}, {3.89887 10¹⁰}, {5644.84}, {1.19046 10¹¹}}

alfa=1

1

Sol14=Sol13+alfa*p13

{{16.3813}, {13.6187}, {38.1563}, {31.8437}, {-2.95495 10⁻⁶}, {27.433}, {27.1046}, {7.79428 10⁻¹⁵}, {45.4624}, {-1.27685 10⁻⁶}, {40.}, {14.5376}, {2.93627 10⁻⁶}, {-1.91528 10⁻⁶}, {45.4624}, {7.43345 10⁻⁸}, {27.1046}, {3.17166 10⁻⁷}}

N[Evaluardor[F,x,Sol14],10]

{1856.498199}

3. Ejemplo de Red (Tomado de Nguyen & Dupuis, 1984)

Este ejemplo se configura como una red con 19 arcos y 13 nodos, estableciendo un total de 2 pares de nodos origen (1 y 4), y 2 pares de nodos destino (2 y 3), existiendo una demanda fija entre ambos pares de nodos.

El número de restricciones existentes es de 13, y el número de variables es de 19, lo que garantiza la aplicabilidad del método del conjunto activo.

Las demandas entre nodos son:

De 1 a 2: 400

De 1 a 3: 800

De 4 a 2: 600

De 4 a 3: 200

<<cactivo.m

Definición de Número de Restricciones y Número de Variables

Nº de variables=Nº de arcos en la red.

Nº de restricciones= Ecuaciones de Flujo

arcos:=19

restricciones:=13

$x:=\{x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,x13,x14, x15,x16,x17,x18,x19\}$

Función de coste.

$c1:=7(1+0.15(x1/600)^4);c2:=9(1+0.15(x2/700)^4);$

$c3:=9(1+0.15(x3/600)^4);c4:=12(1+0.15(x4/400)^4);$

$c5:=3(1+0.15(x5/900)^4);c6:=9(1+0.15(x6/1200)^4);$

$c7:=5(1+0.15(x7/1000)^4);c8:=13(1+0.15(x8/850)^4);$

$c9:=5(1+0.15(x9/1000)^4);c10:=9(1+0.15(x10/750)^4);$

$c11:=9(1+0.15(x11/500)^4);c12:=10(1+0.15(x12/800)^4);$

$c13:=9(1+0.15(x13/1000)^4);c14:=6(1+0.15(x14/700)^4);$

$c15:=9(1+0.15(x15/600)^4);c16:=8(1+0.15(x16/700)^4);$

$c17:=7(1+0.15(x17/900)^4);c18:=14(1+0.15(x18/500)^4);$

$c19:=11(1+0.15(x19/900)^4)$

Función Objetivo.

$F=\text{Integrate}[c1,\{x1,0,x1\}]+\text{Integrate}[c2,\{x2,0,x2\}]+\text{Integrate}[c3,\{x3,0,x3\}]+\text{Integrate}[c4,\{x4,0,x4\}]$
 $+\text{Integrate}[c5,\{x5,0,x5\}]+\text{Integrate}[c6,\{x6,0,x6\}]+\text{Integrate}[c7,\{x7,0,x7\}]+\text{Integrate}[c8,\{x8,0,x8\}]$
 $+\text{Integrate}[c9,\{x9,0,x9\}]+\text{Integrate}[c10,\{x10,0,x10\}]+\text{Integrate}[c11,\{x11,0,x11\}]+\text{Integrate}[c12,\{x12,$
 $0,x12\}]+\text{Integrate}[c13,\{x13,0,x13\}]+\text{Integrate}[c14,\{x14,0,x14\}]+\text{Integrate}[c15,\{x15,0,x15\}]+\text{Integ}$
 $rate}[c16,\{x16,0,x16\}]+\text{Integrate}[c17,\{x17,0,x17\}]+\text{Integrate}[c18,\{x18,0,x18\}]+\text{Integrate}[c19,\{x19,0,$
 $,x19\}]$

Determinamos el punto inicial, lo que determina un valor de la función objetivo

Sol0:={{600},{600},{400},{400},{500},{500},{400},{400},{200},{200},{500},{450},{450},{850},
{500},{550},{300},{300},{450}}

Evaluador[F,x,Sol0]

{73324.6}

Gradiente de la función objetivo

g=Gradiente[F,x]

Matriz de Restricciones (Columna: variables)

A={{1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{1,0,1,0,-1,-1,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{0,0,0,1,0,1,0,0,0,0,0,-1,-1,0,0,0,0,0,0,0},
{0,0,0,0,1,0,-1,-1,0,0,0,0,0,0,0,0,1,0,0,0},
{0,0,0,0,0,0,1,0,-1,-1,0,0,0,0,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,1,0,-1,0,0,0,0,0,0,1,0,0},
{0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,0,0},
{0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-1,-1,0,0},
{0,0,0,0,0,0,0,1,0,0,0,1,0,-1,0,0,0,0,0,0},
{0,0,0,0,0,0,0,0,0,1,0,0,0,1,-1,-1,0,0,0,0},
{0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,-1},
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1}}

Vector constantes lado derecho

b={{1200},{800},{0},{0},{0},{0},{0},{1000},{0},{0},{0},{0},{1000}}

Factorización LQ de la matriz de Restricciones.

A1=FactorLQ[A]

Z=EspacioNulo[A1,restricciones,arcos]

Matriz Auxiliar

B=IdentityMatrix[arcos]

Iteración Inicial:

B0:=B

g0=Evaluador[g,x,Sol0]

{{8.05}, {9.7287}, {9.26667}, {13.8}, {3.04287}, {9.04069}, {5.0192}, {13.0956}, {5.0012},
{9.00683}, {10.35}, {10.1502}, {9.05536}, {7.95671}, {9.65104}, {8.45734}, {7.01296},
{14.2722}, {11.1031}}

p0=DirBus[Z,B0,g0]

B2=BFGS[B1,Sol2-Sol1,g2-g1]

p2=DirBus[Z,B2,g2]

{{0.81372}, {-0.81372}, {-0.253309}, {0.253309}, {2.04267}, {-1.48226}, {2.29942}, {-0.256752}, {1.70542}, {0.593998}, {0.891704}, {-2.32876}, {1.09981}, {-2.58551}, {-0.891704}, {-1.09981}, {1.55704 10⁻¹⁶}, {-0.81372}, {1.09981}}

alfa=Sol2/p2

{{942.357}, {-532.353}, {-1190.94}, {1967.25}, {356.72}, {-229.265}, {211.332}, {-945.341}, {158.491}, {363.044}, {788.918}, {-131.633}, {483.366}, {-212.437}, {-332.531}, {-425.88}, {0.}, {-532.353}, {483.366}}

alfa=1

Sol3=Sol2+alfa*p2

{{767.628}, {432.372}, {301.423}, {498.577}, {730.703}, {338.348}, {488.242}, {242.462}, {272.}, {216.241}, {704.372}, {304.213}, {532.712}, {546.674}, {295.628}, {467.288}, {1.55704 10⁻¹⁶}, {432.372}, {532.712}}

Evaluar[F,x,Sol3]

{69736.1}

4ª Iteración

g3=Evaluar[g,x,Sol3]

{{9.81311}, {9.1965}, {9.08599}, {16.3447}, {3.19553}, {9.00853}, {5.04262}, {13.0129}, {5.00411}, {9.00933}, {14.317}, {10.0314}, {9.10872}, {6.33478}, {9.07956}, {8.2383}, {7.}, {15.1743}, {11.2025}}

B3=BFGS[B2,Sol3-Sol2,g3-g2]

p3=DirBus[Z,B3,g3]

{{176.92}, {-176.92}, {-53.9718}, {53.9718}, {445.147}, {-322.198}, {500.073}, {-54.9262}, {369.192}, {130.882}, {192.271}, {-507.471}, {239.244}, {-562.397}, {-192.271}, {-239.244}, {3.3788 10⁻¹⁴}, {-176.92}, {239.244}}

alfa=Sol3/p3

{{4.33883}, {-2.44388}, {-5.58483}, {9.23773}, {1.64149}, {-1.05012}, {0.97634}, {-4.41431}, {0.736746}, {1.65219}, {3.66343}, {-0.599469}, {2.22665}, {-0.972044}, {-1.53756}, {-1.95319}, {0.00460826}, {-2.44388}, {2.22665}}

alfa=0.599469

0.599469

Sol4=Sol3+alfa*p3

{{873.686}, {326.314}, {269.069}, {530.931}, {997.555}, {145.2}, {788.02}, {209.535}, {493.319}, {294.701}, {819.633}, {-0.0000138872}, {676.131}, {209.535}, {180.367}, {323.869}, {2.04106 10⁻¹⁴}, {326.314}, {676.131}}

Evaluar[F,x,Sol4]

{68216.3}

a={{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0}}

Z=AdaptaEN[a,A,QA,15,19]

A=Join[a,A]

5ª Iteración

g4=Evaluador[g,x,Sol4]

{{11.7207}, {9.06375}, {9.0546}, {17.5871}, {3.67919}, {9.00029}, {5.28921}, {13.0072},
{5.04442}, {9.03218}, {18.7484}, {10.}, {9.28214}, {6.00723}, {9.01102}, {8.05499}, {7.},
{14.381}, {11.5256}}

B4=BFGS[B3,Sol4-Sol3,g4-g3]

p4=DirBus[Z,B4,g4]

{{0.3021}, {-0.3021}, {0.50911}, {-0.50911}, {0.346478}, {0.464732}, {0.397905}, {-
0.0514277},{-1.00939}, {1.40729}, {-1.31149}, {1.94199 10⁻¹⁶}, {-0.0443779}, {-0.0514277},
{1.31149}, {0.0443779}, {-1.15381 10⁻¹⁶}, {-0.3021}, {-0.0443779}}

alfa=Sol4/p4

{{2892.05}, {-1080.15}, {528.509}, {-1042.86}, {2879.13}, {312.438}, {1980.42}, {-4074.36}, {-
488.732}, {209.41}, {-624.965}, {-7.151 10¹⁰}, {-15235.8}, {-4074.36}, {137.529}, {7297.98},
{-176.897}, {-1080.15}, {-15235.8}}

alfa=1

1

Sol5=Sol4+alfa*p4

{{873.988}, {326.012}, {269.578}, {530.422}, {997.901}, {145.665}, {788.418}, {209.484},
{492.31}, {296.108}, {818.322}, {-0.0000138872}, {676.087}, {209.484}, {181.678}, {323.913},
{2.02952 10⁻¹⁴}, {326.012}, {676.087}}

Evaluador[F,x,Sol5]

{68209.3}

6ª Iteración

g5=Evaluador[g,x,Sol5]

{{11.7272}, {9.06351}, {9.05501}, {17.5657}, {3.68013}, {9.00029}, {5.28979}, {13.0072},
{5.04406}, {9.0328}, {18.6861}, {10.}, {9.28206},{6.00722}, {9.01135}, {8.05502}, {7.},
{14.3796}, {11.5254}}

B5=BFGS[B4,Sol5-Sol4,g5-g4]

p5=DirBus[Z,B5,g5]

{{21.9399}, {-21.9399}, {36.1509}, {-36.1509}, {25.0832}, {33.0076}, {29.3937}, {-4.31053}, {-
71.1354}, {100.529}, {-93.0753}, {1.36972 10⁻¹⁴}, {-3.14326}, {-4.31053}, {93.0753}, {3.14326},
{-8.23235 10⁻¹⁵}, {-21.9399}, {-3.14326}}

Sol5/p5

{{39.8355}, {-14.8593}, {7.45703}, {-14.6724}, {39.7837}, {4.41307}, {26.8227}, {-48.5981}, {-6.92074}, {2.9455}, {-8.79204}, {-1.01387 10⁹}, {-215.091}, {-48.5981}, {1.95195}, {103.05}, {-2.4653}, {-14.8593}, {-215.091}}

alfa=1

1

Sol6=Sol5+alfa*p5

{{895.928}, {304.072}, {305.729}, {494.271}, {1022.98}, {178.672}, {817.812}, {205.173}, {421.174}, {396.637}, {725.246}, {-0.0000138872}, {672.943}, {205.173}, {274.754}, {327.057}, {1.20628 10⁻¹⁴}, {304.072}, {672.943}}

Evaluar[F,x,Sol6]

{67939.4}

7ª Iteración

g6=Evaluar[g,x,Sol6]

{{12.2201}, {9.04807}, {9.09101}, {16.1966}, {3.75114}, {9.00066}, {5.33549}, {13.0066}, {5.0236}, {9.1056}, {14.9758}, {10.}, {9.27685}, {6.00664}, {9.05936}, {8.05718}, {7.}, {14.2872}, {11.5157}}

B6=BFGS[B5,Sol6-Sol5,g6-g5]

p6=DirBus[Z,B6,g6]

{{3.92391}, {-3.92391}, {5.68599}, {-5.68599}, {4.40336}, {5.20654}, {5.70068}, {-1.29732}, {-10.7163}, {16.417}, {-14.6402}, {2.07555 10⁻¹⁵}, {-0.479453}, {-1.29732}, {14.6402}, {0.479453}, {-1.33454 10⁻¹⁵}, {-3.92391}, {-0.479453}}

Sol6/p6

{{228.326}, {-77.4922}, {53.7688}, {-86.9278}, {232.319}, {34.3169}, {143.459}, {-158.151}, {-39.3023}, {24.1602}, {-49.538}, {-6.69083 10⁹}, {-1403.57}, {-158.151}, {18.7671}, {682.146}, {-9.03893}, {-77.4922}, {-1403.57}}

alfa=1

1

Sol7=Sol6+alfa*p6

{{899.852}, {300.148}, {311.415}, {488.585}, {1027.39}, {183.879}, {823.512}, {203.876}, {410.458}, {413.054}, {710.606}, {-0.0000138872}, {672.464}, {203.876}, {289.394}, {327.536}, {1.07283 10⁻¹⁴}, {300.148}, {672.464}}

Evaluar[F,x,Sol7]

{67932.}

8ª Iteración

g7=Evaluar[g,x,Sol7]

{12.3121}, {9.04563}, {9.09797}, {16.0068}, {3.76415}, {9.00074}, {5.34494}, {13.0065},
{5.02129}, {9.1242}, {14.5077}, {10.}, {9.27606}, {6.00648}, {9.07306}, {8.05752}, {7.},
{14.2727}, {11.5143}}

B7=BFGS[B6,Sol7-Sol6,g7-g6]

p7=DirBus[Z,B7,g7]

{1.74881}, {-1.74881}, {1.77778}, {-1.77778}, {1.8809}, {1.6457}, {2.96933}, {-1.08843}, {-
2.83269}, {5.80202}, {-4.5815}, {5.62577 10⁻¹⁶}, {-0.132081}, {-1.08843}, {4.5815}, {0.132081},
{-4.61227 10⁻¹⁶}, {-1.74881}, {-0.132081}}

Sol7/p7

{514.55}, {-171.629}, {175.171}, {-274.829}, {546.223}, {111.733}, {277.339}, {-187.311}, {-
144.9}, {71.1915}, {-155.103}, {-2.46849 10¹⁰}, {-5091.31}, {-187.311}, {63.1657}, {2479.82},
{-23.2603}, {-171.629}, {-5091.31}}

alfa=1

1

Sol8=Sol7+alfa*p7

{901.601}, {298.399}, {313.193}, {486.807}, {1029.27}, {185.525}, {826.482}, {202.787},
{407.625}, {418.856}, {706.025}, {-0.0000138872}, {672.332}, {202.787}, {293.975}, {327.668},
{1.02671 10⁻¹⁴}, {298.399}, {672.332}}

Evaluar[F.x,Sol8]

{67930.1}

9ª Iteración

g8=Evaluar[g.x,Sol8]

{12.3535}, {9.04458}, {9.10022}, {15.9488}, {3.76976}, {9.00077}, {5.34994}, {13.0063},
{5.02071}, {9.13133}, {14.367}, {10.}, {9.27585}, {6.00634}, {9.0778}, {8.05761}, {7.},
{14.2664}, {11.5139}}

B8=BFGS[B7,Sol8-Sol7,g8-g7]

p8=DirBus[Z,B8,g8]

{3.754}, {-3.754}, {2.49802}, {-2.49802}, {3.89566}, {2.35636}, {7.12798}, {-3.23233}, {-
2.67951}, {9.80749}, {-6.43351}, {5.72654 10⁻¹⁶}, {-0.141659}, {-3.23233}, {6.43351},
{0.141659}, {-7.56961 10⁻¹⁶}, {-3.754}, {-0.141659}}

Sol8/p8

{240.171}, {-79.4884}, {125.377}, {-194.877}, {264.209}, {78.7336}, {115.949}, {-62.7372}, {-
152.127}, {42.7078}, {-109.742}, {-2.42505 10¹⁰}, {-4746.11}, {-62.7372}, {45.6944}, {2313.07},
{-13.5635}, {-79.4884}, {-4746.11}}

alfa=1

1

Sol9=Sol8+alfa*p8

{905.355}, {294.645}, {315.691}, {484.309}, {1033.16}, {187.881}, {833.61}, {199.555},
{404.946}, {428.664}, {699.591}, {-0.0000138872}, {672.19}, {199.555}, {300.409}, {327.81},
{9.5101 10⁻¹⁵}, {294.645}, {672.19}}

Evaluador[F,x,Sol9]

{67925.8}

10ª Iteración

g9=Evaluador[g,x,Sol9]

{12.4433}, {9.04238}, {9.10346}, {15.8683}, {3.78148}, {9.00081}, {5.36217}, {13.0059},
{5.02017}, {9.14406}, {14.174}, {10.}, {9.27561}, {6.00594}, {9.08484}, {8.05771}, {7.},
{14.2532}, {11.5134}}

B9=BFGS[B8,Sol9-Sol8,g9-g8]

p9=DirBus[Z,B9,g9]

{8.09328}, {-8.09328}, {3.15013}, {-3.15013}, {8.15919}, {3.08422}, {16.664}, {-8.50484},
{0.0276552}, {16.6364}, {-8.06563}, {1.49368 10⁻¹⁶}, {-0.0659131}, {-8.50484}, {8.06563},
{0.0659131}, {-1.23571 10⁻¹⁵}, {-8.09328}, {-0.0659131}}

Sol9/p9

{111.865}, {-36.4061}, {100.215}, {-153.743}, {126.626}, {60.917}, {50.0245}, {-23.4637},
{14642.7}, {25.7667}, {-86.7373}, {-9.29729 10¹⁰}, {-10198.1}, {-23.4637}, {37.2456},
{4973.36}, {-7.69606}, {-36.4061}, {-10198.1}}

alfa=1

1

Sol10=Sol9+alfa*p9

{913.448}, {286.552}, {318.841}, {481.159}, {1041.32}, {190.965}, {850.274}, {191.05},
{404.973}, {445.3}, {691.525}, {-0.0000138872}, {672.124}, {191.05}, {308.475}, {327.876},
{8.27439 10⁻¹⁵}, {286.552}, {672.124}}

Evaluador[F,x,Sol10]

{67915.9}

11ª Iteración

g10=Evaluador[g,x,Sol10]

{12.6405}, {9.03791}, {9.10765}, {15.7687}, {3.80647}, {9.00087}, {5.39201}, {13.005},
{5.02017}, {9.16776}, {13.9395}, {10.}, {9.27551}, {6.00499}, {9.09432}, {8.05776}, {7.},
{14.2265}, {11.5132}}

B10=BFGS[B9,Sol10-Sol9,g10-g9]

p10=DirBus[Z,B10,g10]

{15.2982}, {-15.2982}, {2.47053}, {-2.47053}, {15.052}, {2.71674}, {33.563}, {-18.511},
{9.18638}, {24.3766}, {-6.11183}, {-1.42914 10⁻¹⁵}, {0.246218}, {-18.511}, {6.11183}, {-
0.246218}, {-1.71605 10⁻¹⁵}, {-15.2982}, {0.246218}}

Sol10/p10

{{59.7095}, {-18.7311}, {129.058}, {-194.76}, {69.1818}, {70.292}, {25.3337}, {-10.3209},
{44.0841}, {18.2675}, {-113.145}, {9.71712 10⁹}, {2729.79}, {-10.3209}, {50.4717}, {-1331.65},
{-4.82176}, {-18.7311}, {2729.79}}

alfa=1

1

Sol11=Sol10+alfa*p10

{{928.746}, {271.254}, {321.311}, {478.689}, {1056.38}, {193.682}, {883.837}, {172.539},
{414.16}, {469.677}, {685.414}, {-0.0000138872}, {672.371}, {172.539}, {314.586}, {327.629},
{6.55834 10⁻¹⁵}, {271.254}, {672.371}}

Evaluador[F,x,Sol11]

{67896.8}

12ª Iteración

g11=Evaluador[g,x,Sol11]

{{13.028}, {9.03044}, {9.11103}, {15.6919}, {3.85411}, {9.00092}, {5.45767}, {13.0033},
{5.02207}, {9.20763}, {13.7672}, {10.}, {9.27591}, {6.00332}, {9.10202}, {8.05759}, {7.},
{14.1819}, {11.514}}

B11=BFGS[B10,Sol11-Sol10,g11-g10]

p11=DirBus[Z,B11,g11]

{{20.2633}, {-20.2633}, {-1.48377}, {1.48377}, {19.4357}, {-0.656175}, {47.3631}, {-27.9273},
{24.8194}, {22.5437}, {4.55605}, {-4.27126 10⁻¹⁵}, {0.827592}, {-27.9273}, {-4.55605}, {-
0.827592}, {-1.42246 10⁻¹⁵}, {-20.2633}, {0.827592}}

Sol11/p11

{{45.8339}, {-13.3864}, {-216.551}, {322.617}, {54.3523}, {-295.168}, {18.6609}, {-6.17814},
{16.687}, {20.834}, {150.44}, {3.2513 10⁹}, {812.442}, {-6.17814}, {-69.0481}, {-395.883}, {-
4.61055}, {-13.3864}, {812.442}}

alfa=1

1

Sol12=Sol11+alfa*p11

{{949.01}, {250.99}, {319.828}, {480.172}, {1075.81}, {193.026}, {931.2}, {144.612},
{438.979}, {492.221}, {689.97}, {-0.0000138872}, {673.198}, {144.612}, {310.03}, {326.802},
{5.13588 10⁻¹⁵}, {250.99}, {673.198}}

Evaluador[F,x,Sol12]

{67871.}

13ª Iteración

g12=Evaluador[g,x,Sol12]

{13.5715}, {9.02231}, {9.10899}, {15.7378}, {3.91873}, {9.0009}, {5.56394}, {13.0016},
 {5.02785}, {9.25045}, {13.8952}, {10.}, {9.27727}, {6.00164}, {9.09624}, {8.05701}, {7.},
 {14.1333}, {11.5165}}

B12=BFGS[B11,Sol12-Sol11,g12-g11]

p12=DirBus[Z,B12,g12]

{13.259}, {-13.259}, {-6.58006}, {6.58006}, {12.1342}, {-5.45521}, {34.5756}, {-
 22.4414}, {31.4785}, {3.09709}, {18.2195}, {-5.67271 10^{-15} }, {1.12485}, {-22.4414}, {-18.2195},
 {-1.12485}, {8.02059 10^{-17} }, {-13.259}, {1.12485}}

Sol12/p12

{71.5746}, {-18.9298}, {-48.6056}, {72.9739}, {88.6596}, {-35.3838}, {26.9323}, {-6.44397},
 {13.9454}, {158.93}, {37.8699}, {2.44806 10^9 }, {598.478}, {-6.44397}, {-17.0164}, {-290.529},
 {64.0337}, {-18.9298}, {598.478}}

alfa=1

1

Sol13=Sol12+alfa*p12

{962.269}, {237.731}, {313.248}, {486.752}, {1087.95}, {187.571}, {965.775}, {122.17},
 {470.458}, {495.318}, {708.189}, {-0.0000138872}, {674.323}, {122.17}, {291.811}, {325.677},
 {5.21608 10^{-15} }, {237.731}, {674.323}}

Evaluar[F,x,Sol13]

{67850.9}

14ª Iteración

g13=Evaluar[g,x,Sol13]

{13.9466}, {9.01796}, {9.1003}, {15.947}, {3.96088}, {9.00081}, {5.65248}, {13.0008},
 {5.03674}, {9.25682}, {14.4331}, {10.}, {9.27913}, {6.00084}, {9.07553}, {8.05623}, {7.},
 {14.1073}, {11.52}}

B13=BFGS[B12,Sol13-Sol12,g13-g12]

p13=DirBus[Z,B13,g13]

{-0.627165}, {0.627165}, {-5.35176}, {5.35176}, {-1.1499}, {-4.82903}, {2.22716}, {-3.37706},
 {14.393}, {-12.1658}, {15.0202}, {-2.75149 10^{-15} }, {0.522731}, {-3.37706}, {-15.0202}, {-
 0.522731}, {1.02924 10^{-15} }, {0.627165}, {0.522731}}

alfa=1

1

Sol14=Sol13+alfa*p13

{961.642}, {238.358}, {307.896}, {492.104}, {1086.8}, {182.742}, {968.003}, {118.793},
 {484.851}, {483.152}, {723.209}, {-0.0000138872}, {674.846}, {118.793}, {276.791}, {325.154},
 {6.24532 10^{-15} }, {238.358}, {674.846}}

Evaluar[F,x,Sol14]

{67845.}

15ª Iteración

g14=Evaluador[g,x,Sol14]

{{13.9285}, {9.01815}, {9.09361}, {16.1234}, {3.95683}, {9.00073}, {5.65852}, {13.0007},
{5.04145}, {9.2325}, {14.9089}, {10.}, {9.28}, {6.00075}, {9.06114}, {8.05587}, {7.}, {14.1085},
{11.5216}}

B14=BFGS[B13,Sol14-Sol13,g14-g13]

p14=DirBus[Z,B14,g14]

{{-2.61146}, {2.61146}, {-0.614764}, {0.614764}, {-2.56718}, {-0.659042}, {-5.18408},
{2.61689}, {-0.1836}, {-5.00048}, {2.42786}, {-5.27749 10⁻¹⁷}, {-0.0442789}, {2.61689}, {-
2.42786}, {0.0442789}, {3.49084 10⁻¹⁶}, {2.61146}, {-0.0442789}}

alfa=1

Sol15=Sol14+alfa*p14

{{959.03}, {240.97}, {307.281}, {492.719}, {1084.23}, {182.083}, {962.818}, {121.41},
{484.667}, {478.151}, {725.637}, {-0.0000138872}, {674.801}, {121.41}, {274.363}, {325.199},
{6.5944 10⁻¹⁵}, {240.97}, {674.801}}

Evaluador[F,x,Sol15]

{67844.5}

16ª Iteración

g15=Evaluador[g,x,Sol15]

{{13.8535}, {9.01896}, {9.09287}, {16.1441}, {3.94782}, {9.00072}, {5.64452},
{13.0008}, {5.04138}, {9.22302}, {14.9887}, {10.}, {9.27992}, {6.00081}, {9.05902}, {8.0559},
{7.}, {14.1133}, {11.5215}}

B15=BFGS[B14,Sol15-Sol14,g15-g14]

p15=DirBus[Z,B15,g15]

{{-1.44428}, {1.44428}, {0.187453}, {-0.187453}, {-1.34198}, {0.0851572}, {-2.76065},
{1.41867}, {-0.630125}, {-2.13053}, {0.814155}, {3.35173 10⁻¹⁷}, {-0.102296}, {1.41867}, {-
0.814155}, {0.102296}, {1.20421 10⁻¹⁶}, {1.44428}, {-0.102296}}

alfa=1

1

Sol16=Sol15+alfa*p15

{{957.586}, {242.414}, {307.469}, {492.531}, {1082.89}, {182.168}, {960.058}, {122.829},
{484.037}, {476.021}, {726.451}, {-0.0000138872}, {674.699}, {122.829}, {273.549}, {325.301},
{6.71483 10⁻¹⁵}, {242.414}, {674.699}}

Evaluador[F,x,Sol16]

{67844.2}

17ª Iteración

g16=Evaluador[g,x,Sol16]

{{13.8123}, {9.01942}, {9.0931}, {16.1378}, {3.94314}, {9.00072}, {5.63716}, {13.0009},
{5.04117}, {9.21907}, {15.0156}, {10.}, {9.27975},{6.00085}, {9.05833}, {8.05597}, {7.},
{14.116}, {11.5211}}

B16=BFGS[B15,Sol16-Sol15,g16-g15]

p16=DirBus[Z,B16,g16]

{{-2.9889}, {2.9889}, {1.3872}, {-1.3872}, {-2.60166}, {0.999963}, {-4.93503}, {2.33337}, {-
1.12903}, {-3.806}, {1.85987}, {-8.25068 10⁻¹⁷},{-0.387242}, {2.33337}, {-1.85987}, {0.387242},
{1.40665 10⁻¹⁶}, {2.9889}, {-0.387242}}

alfa=1

1

Sol17=Sol16+alfa*p16

{{954.597}, {245.403}, {308.856}, {491.144}, {1080.28}, {183.168}, {955.123}, {125.162},
{482.908}, {472.215}, {728.311}, {-0.0000138872}, {674.312}, {125.162}, {271.689}, {325.688},
{6.85549 10⁻¹⁵}, {245.403}, {674.312}}

Evaluador[F,x,Sol17]

{67843.2}

18ª Iteración

g17=Evaluador[g,x,Sol17]

{{13.7277}, {9.02039}, {9.09479}, {16.0914}, {3.93411}, {9.00073}, {5.62416},
{13.0009},{5.04079}, {9.21215}, {15.0775}, {10.}, {9.27911}, {6.00092}, {9.05676}, {8.05623},
{7.}, {14.1219}, {11.5199}}

B17=BFGS[B16,Sol17-Sol16,g17-g16]

p17=DirBus[Z,B17,g17]

{{-3.15953}, {3.15953}, {3.57194}, {-3.57194}, {-2.38046}, {2.79287}, {-3.49912}, {1.11866}, {-
0.663229}, {-2.83589}, {2.4963}, {-4.44842 10⁻¹⁶}, {-0.779072}, {1.11866}, {-2.4963},
{0.779072}, {-7.62428 10⁻¹⁷}, {3.15953}, {-0.779072}}

alfa=1

Sol18=Sol17+alfa*p17

{{951.437}, {248.563}, {312.428}, {487.572}, {1077.9}, {185.961}, {951.624}, {126.281},
{482.245}, {469.379}, {730.807}, {-0.0000138872}, {673.533}, {126.281}, {269.193}, {326.467},
{6.77925 10⁻¹⁵}, {248.563}, {673.533}}

Evaluador[F,x,Sol18]

{67841.4}

19ª Iteración

g18=Evaluador[g,x,Sol18]

{{13.639}, {9.02146}, {9.09925}, {15.9736}, {3.9259}, {9.00078}, {5.61507}, {13.0009},
{5.04056}, {9.2071}, {15.1612}, {10.}, {9.27782}, {6.00095}, {9.0547}, {8.05677}, {7.},
{14.1283}, {11.5175}}

B18=BFGS[B17,Sol18-Sol17,g18-g17]

p18=DirBus[Z,B18,g18]

{{-1.85861}, {1.85861}, {6.89289}, {-6.89289}, {-0.568046}, {5.60233}, {1.84679}, {-2.41484},
{0.8135}, {1.03329}, {2.67211}, {-1.07588 10⁻¹⁵}, {-1.29056}, {-2.41484}, {-2.67211}, {1.29056},
{-5.57322 10⁻¹⁶}, {1.85861}, {-1.29056}}

alfa=1

1

Sol19=Sol18+alfa*p18

{{949.579}, {250.421}, {319.321}, {480.679}, {1077.34}, {191.563}, {953.47}, {123.866},
{483.058}, {470.412}, {733.48}, {-0.0000138872}, {672.242}, {123.866}, {266.52}, {327.758},
{6.22193 10⁻¹⁵}, {250.421}, {672.242}}

Evaluador[F,x,Sol19]

{67838.9}

20ª Iteración

g19=Evaluador[g,x,Sol19]

{{13.5873}, {9.02211}, {9.1083}, {15.7537}, {3.92395}, {9.00088}, {5.61986}, {13.0009},
{5.04084}, {9.20893}, {15.2518}, {10.}, {9.2757}, {6.00088}, {9.05256}, {8.05768}, {7.},
{14.1321}, {11.5136}}

B19=BFGS[B18,Sol19-Sol18,g19-g18]

p19=DirBus[Z,B19,g19]

{{1.75022}, {-1.75022}, {6.77145}, {-6.77145}, {2.82197}, {5.6997}, {9.0515}, {-6.22953},
{2.53635}, {6.51514}, {0.786133}, {-1.23553 10⁻¹⁵}, {-1.07175}, {-6.22953}, {-0.786133},
{1.07175}, {-8.95175 10⁻¹⁶}, {-1.75022}, {-1.07175}}

alfa=1

1

Sol20=Sol19+alfa*p19

{{951.329}, {248.671}, {326.092}, {473.908}, {1080.16}, {197.263}, {962.522}, {117.637},
{485.595}, {476.927}, {734.266}, {-0.0000138872}, {671.171}, {117.637}, {265.734}, {328.829},
{5.32675 10⁻¹⁵}, {248.671}, {671.171}}

Evaluador[F,x,Sol20]

{67837.}

21ª Iteración

g20=Evaluador[g,x,Sol20]

{13.636}, {9.0215}, {9.11778}, {15.5466}, {3.93367}, {9.00099}, {5.64373}, {13.0007},
{5.0417}, {9.22075}, {15.2787}, {10.}, {9.27395}, {6.00072}, {9.05194}, {8.05843}, {7.},
{14.1285}, {11.5103}}

B20=BFGS[B19,Sol20-Sol19,g20-g19]

p20=DirBus[Z,B20,g20]

{2.98601}, {-2.98601}, {2.39252}, {-2.39252}, {3.21152}, {2.16701}, {7.98849}, {-4.77697},
{2.05566}, {5.93283}, {-0.930355}, {-5.59654 10^{-16} }, {-0.22551}, {-4.77697}, {0.930355},
{0.22551}, {-5.47527 10^{-16} }, {-2.98601}, {-0.22551}}

alfa=1

1

Sol21=Sol20+alfa*p20

{954.315}, {245.685}, {328.485}, {471.515}, {1083.37}, {199.43}, {970.51}, {112.86},
{487.65}, {482.86}, {733.335}, {-0.0000138872}, {670.945}, {112.86}, {266.665}, {329.055},
{4.77922 10^{-15} }, {245.685}, {670.945}}

Evaluador[F,x,Sol21]

{67836.4}

22ª Iteración

g21=Evaluador[g,x,Sol21]

{13.7197}, {9.02049}, {9.12128}, {15.4755}, {3.94482}, {9.00103}, {5.66537}, {13.0006},
{5.04241}, {9.23194}, {15.2469}, {10.}, {9.27358}, {6.00061}, {9.05267}, {8.0586}, {7.},
{14.1224}, {11.5096}}

B21=BFGS[B20,Sol21-Sol20,g21-g20]

p21=DirBus[Z,B21,g21]

{1.11159}, {-1.11159}, {-0.278455}, {0.278455}, {0.962974}, {-0.129839}, {2.03193}, {-
1.06895}, {0.509003}, {1.52292}, {-0.602588}, {-2.17906 10^{-17} }, {0.148616}, {-1.06895},
{0.602588}, {-0.148616}, {-7.95246 10^{-17} }, {-1.11159}, {0.148616}}

alfa=1

1

Sol22=Sol21+alfa*p21

{955.427}, {244.573}, {328.206}, {471.794}, {1084.33}, {199.3}, {972.542}, {111.791},
{488.159}, {484.383}, {732.733}, {-0.0000138872}, {671.094}, {111.791}, {267.267}, {328.906},
{4.6997 10^{-15} }, {244.573}, {671.094}}

N[Evaluador[F,x,Sol22],10]

{67836.37941}

g22=Evaluador[g,x,Sol22]

{{13.7511}, {9.02012}, {9.12087}, {15.4837}, {3.94818}, {9.00103}, {5.67096}, {13.0006},
{5.04259}, {9.23488}, {15.2264}, {10.}, {9.27382}, {6.00059}, {9.05315}, {8.05849}, {7.},
{14.1202}, {11.5101}}

Transpose[Z].g22

{{0.0424948}, {0.0104299}, {-0.0303003}, {-0.0731967}}

4. Ejemplo de Redes con Peaje.

Este ejemplo de redes con peaje se basa en el ejemplo 5.1 de Sheffi (1985). El primer ejemplo resuelve utilizando el método del conjunto activo la red con el peaje tradicional incorporado a la función de coste; mientras que el segundo ejemplo considera el peaje alternativo propuesto en el presente estudio.

4.1 Peaje Tradicional.

Consideramos una red formada por un nodo origen y un nodo destino conectados por 3 arcos. Las funciones de coste de cada arco están basadas en la función flujo-coste BPR(1964).

Las funciones de coste tienen una parte independiente del flujo y la capacidad, dado por la constante C_0 , y un coste que se relaciona directamente con el grado de congestión del arco.

La fórmula general de la función de coste a corto plazo para cada usuario es:

$C = CF(1 + a(x/K)^b)$, donde CF es coste con flujo libre, y K es la capacidad máxima del arco.

A este coste, se debe añadir el importe del peaje que el usuario tendrá que pagar y que viene dado por:

$$\text{Peaje} = a b CF (x/K)^b$$

El usuario se enfrenta a un coste total = $C + \text{Peaje}$.

METODO DEL CONJUNTO ACTIVO

<<cactivo.m

Arco 1:

$$t1:=10*(1+0.15*(x1/2)^4)+6*(x1/2)^4$$

$$c1:=10*(1+0.15*(x1/2)^4)$$

Arco 2:

$$t2:=20*(1+0.15*(x2/4)^4)+12*(x2/4)^4$$

$$c2:=20*(1+0.15*(x2/4)^4)$$

Arco 3:

$$t3:=25*(1+0.15*(x3/3)^4)+15*(x3/3)^4$$

$$c3:=25*(1+0.15*(x3/3)^4)$$

Función Objetivo:

$$F=\text{Integrate}[t1,\{x1,0,x1\}]+\text{Integrate}[t2,\{x2,0,x2\}]+\text{Integrate}[t3,\{x3,0,x3\}]$$

$$10 x1 + 0.09375 x1^5 + 20 x2 + 0.0117188 x2^5 + 25 x3 + 0.0462963 x3^5$$

$$M=\text{Integrate}[c1,\{x1,0,x1\}]+\text{Integrate}[c2,\{x2,0,x2\}]+\text{Integrate}[c3,\{x3,0,x3\}]$$

$$10 x1 + 0.01875 x1^5 + 20 x2 + 0.00234375 x2^5 + 25 x3 + 0.00925926 x3^5$$

Restricciones:

$$r:=x1+x2+x3==10$$

$$x:=\{x1,x2,x3\}$$

$$\text{arcos}:=3$$

$$\text{restricciones}:=1$$

Determinamos el punto inicial, igualando la primera variable a 2, la segunda a 4 y la tercera a 4, lo que nos determina un valor inicial de la función objetivo

$$\text{Sol0}=\{\{2\},\{4\},\{4\}\}$$

$$\{\{2\},\{4\},\{4\}\}$$

$$\text{Evaluador}[F,x,\text{Sol0}]$$

$$\{262.407\}$$

Gradiente de la función objetivo

$$g=\text{Gradiente}[F,x]$$

$$\{\{10 + 0.46875 x1^4\}, \{20 + 0.0585937 x2^4\}, \{25 + 0.231481 x3^4\}\}$$

Factorización LQ de la matriz A

La matriz A, formada por la única restricción activa en el punto inicial, se factoriza, y sirve de base para la formación de la matriz Z, (formada por las n-t columnas de la matriz Q).

$$A=\{\{1,1,1\}\}$$

```
{{1, 1, 1}}
A1=FactorLQ[A]
{{-0.57735, -0.57735, -0.57735}, {-0.57735, 0.788675, -0.211325}, {-0.57735, -0.211325,
0.788675}}
Z=EspacioNulo[A1, restricciones, arcos]
{{-0.57735, -0.57735}, {0.788675, -0.211325}, {-0.211325, 0.788675}}
g0=Evaluador[g,x,Sol0]
{{17.5}, {35.}, {84.2593}}
Matriz Auxiliar (Método Quasi-Newton)
B0:=IdentityMatrix[arcos]
1ª Iteración
p0=DirBus[Z,B0,g0]
{{28.0864}, {10.5864}, {-38.6728}}
alfa=Cota[p0,Sol0]
0.0712088
Sol1=Sol0+alfa*p0
{{4.}, {4.75385}, {1.24615}}
Evaluador[F,x,Sol1]
{290.822}
Adaptación de la matriz B.
g1=Evaluador[g,x,Sol1]
{{130.}, {49.9248}, {25.5582}}
B1=BFGS[B0,Sol1,g1]
{{22.016, 7.75038, 4.08598}, {7.75038, 3.59547, 1.46931}, {4.08598, 1.46931, 1.78904}}
2ª Iteración
p1=DirBus[Z,B1,g1]
{{-7.08177}, {1.57328}, {5.50848}}
alfa=Cota[p1,Sol1]
Sol2=Sol1+alfa*p1
{{2.39793}, {5.10976}, {2.49231}}
Evaluador[F,x,Sol2]
{241.188}
Adaptación de la matriz B.
g2=Evaluador[g,x,Sol2]
{{25.4984}, {59.9442}, {33.9315}}
B2=BFGS[B1,Sol2-Sol1,g2-g1]
{{61.2613, -5.13091, -3.63561}, {-5.13091, 1.7359, 0.948094}, {-3.63561, 0.948094, 1.77451}}
```

3ª Iteración

$p2 = \text{DirBus}[Z, B2, g2]$

$\{-0.0334978\}, \{-16.1286\}, \{16.1621\}$

$\text{alfa} = \text{Cota}[p2, \text{Sol2}]$

0.154207

$\text{Sol3} = \text{Sol2} + \text{alfa} * p2$

$\{2.39277\}, \{2.62262\}, \{4.98462\}$

$\text{Evaluador}[F, x, \text{Sol3}]$

$\{352.266\}$

Adaptación de la matriz B.

$g3 = \text{Evaluador}[g, x, \text{Sol3}]$

$\{25.3653\}, \{22.772\}, \{167.904\}$

$B3 = \text{BFGS}[B2, \text{Sol3} - \text{Sol2}, g3 - g2]$

$\{60.1129, -4.46496, -4.38452\}, \{-4.46496, 4.60399, -10.3296\}, \{-4.38452, -10.3296, 43.4369\}$

4ª Iteración

$p3 = \text{DirBus}[Z, B3, g3]$

$\{0.4139\}, \{1.78909\}, \{-2.20299\}$

$\text{alfa} = \text{Cota}[p3, \text{Sol3}]$

$\text{Sol4} = \text{Sol3} + \text{alfa} * p3$

$\{2.80667\}, \{4.41171\}, \{2.78162\}$

$\text{Evaluador}[F, x, \text{Sol4}]$

$\{229.464\}$

Adaptación de la matriz B.

$g4 = \text{Evaluador}[g, x, \text{Sol4}]$

$\{39.0873\}, \{42.1963\}, \{38.8583\}$

$B4 = \text{BFGS}[B3, \text{Sol4} - \text{Sol3}, g4 - g3]$

$\{58.4804, -6.07262, -0.173156\}, \{-6.07262, 3.10023, -7.44039\}, \{-0.173156, -7.44039, 52.5022\}$

5ª Iteración

$p4 = \text{DirBus}[Z, B4, g4]$

$\{0.0333427\}, \{-0.072924\}, \{0.0395813\}$

$\text{alfa} = \text{Cota}[p4, \text{Sol4}]$

$\text{Sol5} = \text{Sol4} + \text{alfa} * p4$

$\{2.84001\}, \{4.33879\}, \{2.8212\}$

$\text{Evaluador}[F, x, \text{Sol5}]$

$\{229.32\}$

Adaptación de la matriz B.

$g5 = \text{Evaluador}[g, x, \text{Sol5}]$

{{40.4944}, {40.7647}, {39.6641}}

B5=BFGS[B4,Sol5-Sol4,g5-g4]

{{45.1438, -9.75073, -20.445}, {-9.75073, 12.0695, -5.71789}, {-20.445, -5.71789, 27.0458}}

6ª Iteración

p5=DirBus[Z,B5,g5]

{{0.00153032}, {-0.0230875}, {0.0215572}}

alfa=Cota[p5,Sol5]

Sol6=Sol5+alfa*p5

{{2.84154}, {4.3157}, {2.84276}}

Evaluador[F,x,Sol6]

{229.305}

Adaptación de la matriz B.

g6=Evaluador[g,x,Sol6]

{{40.5601}, {40.3262}, {40.1174}}

B6=BFGS[B5,Sol6-Sol5,g6-g5]

{{44.4707, -13.7234, -14.8031}, {-13.7234, 14.4853, -3.8514}, {-14.8031, -3.8514, 17.9568}}

7ª Iteración

p6=DirBus[Z,B6,g6]

{{-0.00409796}, {-0.00286416}, {0.00696211}}

alfa=Cota[p6,Sol6]

Sol7=Sol6+alfa*p6

{{2.83744}, {4.31284}, {2.84972}}

Evaluador[F,x,Sol7]

{229.304}

Adaptación de la matriz B.

g7=Evaluador[g,x,Sol7]

{{40.3842}, {40.2723}, {40.2661}}

B7=BFGS[B6,Sol7-Sol6,g7-g6]

{{35.5842, -9.98936, -8.43111}, {-9.98936, 15.9462, -7.06231}, {-8.43111, -7.06231, 13.4817}}

8ª Iteración

p7=DirBus[Z,B7,g7]

{{-0.00200236}, {0.000729831}, {0.00127253}}

alfa=Cota[p7,Sol7]

Sol8=Sol7+alfa*p7

{{2.83544}, {4.31357}, {2.851}}

N[Evaluador[F,x,Sol8],10]

{229.303818}

Evaluador[M,x,Sol8]
{194.581}

4.2 Peaje Alternativo.

Consideramos una red formada por un nodo origen y un nodo destino conectados por 3 arcos. Las funciones de coste de cada arco están basadas en la función flujo-coste BPR(1964).

Las funciones de coste tienen una parte independiente del flujo y la capacidad, dado por la constante C_0 , y un coste que se relaciona directamente con el grado de congestión del arco.

La fórmula general de la función de coste a corto plazo para cada usuario es:

$C = CF(1 + a(x/K)^b)$, donde CF es coste con flujo libre, y K es la capacidad máxima del arco.

A este coste, se debe añadir el importe del peaje que el usuario tendrá que pagar y que viene dado por:

$$\text{Peaje} = CF(1 - abx^{(b-1)}/K^b + a(x/K)^b)$$

El usuario se enfrenta a un coste total = $C + \text{Peaje}$.

METODO DEL CONJUNTO ACTIVO

<<cactivo.m

Arco 1:

$$t1 := 20 * (1 + 0.15 * (x1/2)^4) - 6(x1^{3/2})^4$$

$$c1 := 10 * (1 + 0.15 * (x1/2)^4)$$

Arco 2:

$$t2 := 40 * (1 + 0.15 * (x2/4)^4) - 12(x2^{3/4})^4$$

$$c2 := 20 * (1 + 0.15 * (x2/4)^4)$$

Arco 3:

$$t3 := 50 * (1 + 0.15 * (x3/3)^4) + 15(x3^{3/3})^4$$

$$c3 := 25 * (1 + 0.15 * (x3/3)^4)$$

Función Objetivo:

F=Integrate[t1,{x1,0,x1}]+Integrate[t2,{x2,0,x2}]+Integrate[t3,{x3,0,x3}]
 $20x_1 + 3x_1^4 / 32 + 0.0375 x_1^5 + 40 x_2 - 3x_2^4 / 256 + 0.0046875x_2^5 + 50 x_3 + 5x_3^4 / 108 + 0.0185185x_3^5$

M=Integrate[c1,{x1,0,x1}]+Integrate[c2,{x2,0,x2}]+Integrate[c3,{x3,0,x3}]

$10 x_1 + 0.01875 x_1^5 + 20 x_2 + 0.00234375 x_2^5 + 25 x_3 + 0.00925926 x_3^5$

Restricciones:

r:=x1+x2+x3==10

x:={x1,x2,x3}

arcos:=3

restricciones:=1

Determinamos el punto inicial, igualando la primera variable a 2, la segunda a 4 y la tercera a 4, lo que nos determina un valor inicial de la función objetivo

Sol0={{2},{4},{4}}

{{2}, {4}, {4}}

Evaluador[F,x,Sol0]

{432.315}

Gradiente de la función objetivo

g=Gradiente[F,x]

{{20 - 3x1³/8 + 0.1875 x1⁴}, {40 - 3x2³/64 + 0.0234375 x2⁴}, {50 + 5x3³/27 + 0.0925926 x3⁴}}

Factorización LQ de la matriz A

La matriz A, formada por la única restricción activa en el punto inicial, se factoriza, y sirve de base para la formación de la matriz Z, (formada por las n-t columnas de la matriz Q).

A={{1,1,1}}

{{1, 1, 1}}

A1=FactorLQ[A]

{{-0.57735, -0.57735, -0.57735}, {-0.57735, 0.788675, -0.211325}, {-0.57735, -0.211325, 0.788675}}

Z=EspacioNulo[A1,restricciones,arcos]

{{-0.57735, -0.57735}, {0.788675, -0.211325}, {-0.211325, 0.788675}}

g0=Evaluador[g,x,Sol0]

{{20.}, {43.}, {85.5556}}

Matriz Auxiliar (Método Quasi-Newton)

B0:=IdentityMatrix[arcos]

1ª Iteración

p0=DirBus[Z,B0,g0]

{{29.5185}, {6.51852}, {-36.037}}

Abs[Sol0/p0]

{{0.0677541}, {0.613636}, {0.110997}}

alfa=0.107084

0.107084

alfa=Cota[p0,Sol0]

0.0677541

Sol1=Sol0+alfa*p0

{{4.}, {4.44166}, {1.55834}}

Evaluador[F,x,Sol1]

{353.969}

Adaptación de la matriz B.

g1=Evaluador[g,x,Sol1]

{{44.}, {45.0145}, {51.2468}}

B1=BFGS[B0,Sol1-Sol0,g1-g0]

{{4.94811, 0.277489, -5.72613}, {0.277489, 1.01139, -0.414829}, {-5.72613, -0.414829, 9.28601}}

2ª Iteración

p1=DirBus[Z,B1,g1]

{{-0.439953}, {1.18104}, {-0.741091}}

alfa=Cota[p1,Sol1]

1

Sol2=Sol1+alfa*p1

{{3.56005}, {5.6227}, {0.817253}}

Evaluador[F,x,Sol2]

{358.014}

Adaptación de la matriz B.

g2=Evaluador[g,x,Sol2]

{{33.1979}, {55.0931}, {50.1424}}

B2=BFGS[B1,Sol2-Sol1,g2-g1]

{{10.2516, -6.74466, -2.25871}, {-6.74466, 6.36808, 0.552906}, {-2.25871, 0.552906, 3.71236}}

3ª Iteración

p2=DirBus[Z,B2,g2]

{{0.886651}, {-0.420582}, {-0.466068}}

Abs[Sol2/p2]

{{4.01516}, {13.3688}, {1.75351}}

alfa=1

1

Sol3=Sol2+alfa*p2

{{4.4467}, {5.20212}, {0.351185}}

Evaluador[F,x,Sol3]

{352.397}

Adaptación de la matriz B.

g3=Evaluador[g,x,Sol3]

{{60.3361}, {50.5655}, {50.0094}}

B3=BFGS[B2,Sol3-Sol2,g3-g2]

{{28.6993, -4.7003, 0.611403}, {-4.7003, 2.5083, -1.49087}, {0.611403, -1.49087, 2.79379}}

4ª Iteración

p3=DirBus[Z,B3,g3]

{{-0.344878}, {-0.109883}, {0.454761}}

Abs[Sol3/p3]

{{12.8935}, {47.3424}, {0.772241}}

alfa=0.117483

Sol4=Sol3+alfa*p3

{{4.40618}, {5.18921}, {0.404612}}

Evaluador[F,x,Sol4]

{352.007}

Adaptación de la matriz B.

g4=Evaluador[g,x,Sol4]

{{58.5937}, {50.4447}, {50.0147}}

B4=BFGS[B3,Sol4-Sol3,g4-g3]

{{47.7348, -0.118958, 3.55796}, {-0.118958, 2.58657, -1.72515}, {3.55796, -1.72515, 2.38097}}

5ª Iteración

p4=DirBus[Z,B4,g4]

{{-0.199102}, {-0.0409287}, {0.240031}}

alfa=Cota[p4,Sol4]

Sol5=Sol4+alfa*p4

{{4.20708}, {5.14828}, {0.644643}}

Evaluador[F,x,Sol5]

{351.09}

Adaptación de la matriz B.

g5=Evaluador[g,x,Sol5]

{{50.8149}, {50.0686}, {50.0656}}

B5=BFGS[B4,Sol5-Sol4,g5-g4]

{{42.8099, -0.749264, 2.97496}, {-0.749264, 2.53358, -1.75634}, {2.97496, -1.75634, 2.38006}}

6ª Iteración

```

p5=DirBus[Z,B5,g5]
{{-0.019101}, {0.000573339}, {0.0185277}}
alfa:=Cota[p5,Sol5]
Sol6=Sol5+alfa*p5
{{4.18798}, {5.14885}, {0.66317}}
Evaluador[F,x,Sol6]
{351.082}
Adaptación de la matriz B.
g6=Evaluador[g,x,Sol6]
{{50.134}, {50.0738}, {50.0719}}
B6=BFGS[B5,Sol6-Sol5,g6-g5]
{{37.4464, -1.91345, 1.91475}, {-1.91345, 2.51597, -1.76994}, {1.91475, -1.76994, 2.36992}}
7ª Iteración
p6=DirBus[Z,B6,g6]
{{-0.00172371}, {-0.000164081}, {0.00188779}}
alfa:=Cota[p6,Sol6]
Sol7=Sol6+alfa*p6
{{4.18625}, {5.14869}, {0.665058}}
Evaluador[F,x,Sol7]
{351.082}
Evaluador[M,x,Sol7]
{194.05}

```

5. Ejemplo de Red con Costes con Interacción Asimétrica.

El siguiente ejemplo considera una red, formada por dos arcos, donde existen costes asimétricos. Para su resolución se ha empleado el método del conjunto activo modificado comentado en el capítulo 3. qrfactor.m es el package que contiene las funciones del conjunto activo modificadas para poder ser utilizadas en la resolución de redes con interacción de costes asimétrica.

METODO DEL CONJUNTO ACTIVO

```
<<qrfactor.m
```

```
Arco 1:
```

```
c1:= 2*(1+0.15*(x1/12)^4)+0.30*(x2/8)^2
```

```
Arco 2:
```

```
c2:= 3*(1+0.15*(x2/8)^4)+0.20*(x1/12)^2
```

```
costes:={c2,c1}
```

```
x:={x1,x2}
```

```
c:={c1,c2}
```

```
Restricciones:
```

```
r:=x1+x2==20
```

```
arcos:=2
```

```
restricciones:=1
```

```
Inicialización:
```

```
Factorización LQ de la matriz A
```

La matriz A, formada por la única restricción activa en el punto inicial, se factoriza, y sirve de base para la formación de la matriz Z, (formada por las n-t columnas de la matriz Q).

```
A={{1,1}}
```

```
{{1, 1}}
```

```
A1=FactorLQ[A]
```

```
{{-0.707107, -0.707107}, {-0.707107, 0.707107}}
```

```
Z=EspacioNulo[A1,restricciones,arcos]
```

```
{{-0.707107}, {0.707107}}
```

```
Matriz Auxiliar (Método Quasi-Newton)
```

```
B0=IdentityMatrix[2]
```

```
{{1, 0}, {0, 1}}
```

```
Solución Inicial
```

```
Sol0={{10},{10}}
```

```
{{10}, {10}}
```

```
costes0=Evaluador[c,x,Sol0]
```

```
{2.61343, 4.23752}
```

```
Paso 1:
```

```
Función Objetivo Inicial:
```

```
cw:=VectorCostes[x,costes,Sol0]
```

```
F0:=FObjetivo[cw,x]
```

```
Gradiente de la función objetivo
```

```
g:=Gradiente[F0,x]
```

```
g0:=Evaluador[g,x,Sol0]
```

1ª Iteración

$p0 = \text{DirBus}[Z, B0, g0]$

$\{\{0.812048\}, \{-0.812048\}\}$

Longitud de Paso

$\text{alfa} = \text{Cota}[p0, \text{Sol0}]$

$\text{Sol1} = \text{Sol0} + \text{alfa} * p0$

$\{\{10.812\}, \{9.18795\}\}$

$\text{costes1} = \text{Evaluador}[c, x, \text{Sol1}]$

$\{2.59342, 3.9453\}$

2ª Iteración

$\text{Sol2} = \text{Iteracion}[x, \text{costes}, \text{Sol1}, g0, \text{alfa} * p0, B0]$

$\{\{14.8448\}, \{5.15518\}\}$

$\text{TestCo}[\text{Sol1}, \text{Sol2}]$

78.2275

$\text{TestCo}[\text{Sol}, \text{Sol2}]$

54.8825

$\text{costes2} = \text{Evaluador}[c, x, \text{Sol2}]$

$\{2.82715, 3.38366\}$

3ª Iteración

$\text{Sol3} = \text{Iteracion}[x, \text{costes}, \text{Sol2}, g1, \text{alfa} * p, B]$

$\{\{17.6665\}, \{2.33352\}\}$

$\text{TestCo}[\text{Sol3}, \text{Sol2}]$

54.7344

$\text{TestCo}[\text{Sol}, \text{Sol3}]$

0.327184

$\text{costes3} = \text{Evaluador}[c, x, \text{Sol3}]$

$\{3.4348, 3.43674\}$

4ª Iteración

$\text{Sol4} = \text{Iteracion}[x, \text{costes}, \text{Sol3}, g1, \text{alfa} * p, B]$

$\{\{17.6763\}, \{2.32368\}\}$

$\text{TestCo}[\text{Sol4}, \text{Sol3}]$

0.421992

$\text{TestCo}[\text{Sol}, \text{Sol4}]$

0.0952097

$\text{costes4} = \text{Evaluador}[c, x, \text{Sol4}]$

$\{3.43773, 3.43716\}$

5ª Iteración

Sol5=Iteracion[x,costes,Sol4,gl,alfa*p,B]

{{17.6741}, {2.3259}}

TestCo[Sol5,Sol4]

0.0958432

TestCo[Sol,Sol5]

0.000632939

costes5=Evaluador[c,x,Sol5]

{3.43707, 3.43707}

6ª Iteración

Sol6=Iteracion[x,costes,Sol5,gl,alfa*p,B]

{{17.6741}, {2.3259}}

TestCo[Sol6,Sol5]

0.0000770365

TestCo[Sol,Sol6]

0.000555903

costes6=Evaluador[c,x,Sol6]

{3.43707, 3.43707}

La misma Red empleando la propuesta de algoritmo de Sheffi (1985) basada en una diagonalización del método de Frank-Wolfe arroja el siguiente resultado:

METODO DE DIAGONALIZACIÓN DE SHEFFI

<<frank.m

Funciones de Coste:

c1:= 2*(1+0.15*(x1/12)^4)+0.30*(x2/8)^2

c2:= 3*(1+0.15*(x2/8)^4)+0.20*(x1/12)^2

Restricción:

x:={x1,x2}

c:={c1,c2}

costes:={c2,c1}

r:=x1+x2==20

Solución Inicial

Consideramos que la solución inicial es x1=20 y x2=0.

La tasa de convergencia será 0.01.

Sol0:={{20},{0}}

costes1=Evaludador[c,x,Sol0]
{4.31481, 3.55556}
Valor de los flujos auxiliares: y1->0, y2->20

Iteración 1:

Aux1:={{0},{20}}
Sol1=Iteracion[x,costes,Sol0,c,Aux1]
{{18.1121}, {1.88786}}
TestCo[Sol1,Sol0]
Infinity

Clear[alfa]

costes2=Evaludador[c,x,Sol1]
{3.57366, 3.45702}

Iteración 2:

Aux2:={{0},{20}}
Sol2=Iteracion[x,costes,Sol1,c,Aux2]
{{17.7671}, {2.23293}}
TestCo[Sol2,Sol1]
18.2781

costes3=Evaludador[c,x,Sol2]
{3.46502, 3.44116}

Iteración 3:

Aux3:={{0},{20}}
Sol3=Iteracion[x,costes,Sol2,c,Aux3]
{{17.6943}, {2.30574}}
TestCo[Sol3,Sol2]
3.26073

costes4=Evaludador[c,x,Sol3]
{3.44308, 3.43795}

Iteración 4:

Aux4:={{0},{20}}
Sol4=Iteracion[x,costes,Sol3,c,Aux4]
{{17.6785}, {2.32151}}
TestCo[Sol4,Sol3]
0.683975

costes5=Evaludador[c,x,Sol4]
{3.43838, 3.43726}

Iteración 5:

Aux5={{0},{20}}
Sol5=Iteracion[x,costes,Sol4,c,Aux5]
{{17.6751}, {2.32494}}
TestCo[Sol5,Sol4]
0.147959
costes6=Evaluador[c,x,Sol5]
{3.43735, 3.43711}
Iteración 6:
Aux6={{0},{20}}
Sol6=Iteracion[x,costes,Sol5,c,Aux6]
{{17.6743}, {2.32569}}
TestCo[Sol6,Sol5]
0.0322167
costes7=Evaluador[c,x,Sol6]
{3.43713, 3.43708}
Iteración 7:
Aux7={{0},{20}}
Sol7=Iteracion[x,costes,Sol6,c,Aux7]
{{17.6741}, {2.32586}}
TestCo[Sol7,Sol6]
0.00702484
costes8=Evaluador[c,x,Sol7]
{3.43708, 3.43707}
Iteración 8:
Aux8={{0},{20}}
Sol8=Iteracion[x,costes,Sol7,c,Aux8]
{{17.6741}, {2.32589}}
TestCo[Sol8,Sol7]
0.00153224
costes9=Evaluador[c,x,Sol8]
{3.43707, 3.43707}

UNIVERSIDADE DA CORUÑA
Servicio de Bibliotecas



1700759621