



**AJUSTE DE MATRICES O-D CON ASIGNACIÓN DE  
ALTA CONVERGENCIA**

**Julio 2013**

**TESIS DOCTORAL**

Antonio Reyes Gutiérrez

# TESIS DOCTORAL

## AJUSTE DE MATRICES O-D CON ASIGNACIÓN DE ALTA CONVERGENCIA

**Presentada por:** Antonio Reyes Gutiérrez  
Ingeniero Industrial

**Dirigida por:** Francisco García Benítez  
Dr. Ingeniero Industrial  
Luis M. Romero Pérez  
Dr. Ingeniero Industrial

**Julio 2013**

Departamento de Ingeniería Mecánica y de los Materiales  
Ingeniería e Infraestructura de los Transportes  
Escuela Técnica Superior de Ingenieros  
Universidad de Sevilla

Esta Tesis ha sido realizada gracias al proyecto ENE2008-05552 y al Programa FPI BES-2009-025110, concedido por el *Ministerio de Ciencia e Innovación* a través del programa “*Plan Nacional I+D*”, y el *Ministerio de Fomento* (P63/08).

# CONTENIDO

<b>CAPÍTULO 1</b>	<b>INTRODUCCIÓN</b>	<b>1</b>
1.1	ÁMBITO DE LA TESIS	1
1.2	MOTIVACIÓN	2
1.3	OBJETIVOS	3
1.4	ESTRUCTURA DE LA TESIS	4
<b>CAPÍTULO 2</b>	<b>PROBLEMA DE ASIGNACIÓN DEL TRÁFICO (TAP)</b>	<b>7</b>
2.1	INTRODUCCIÓN	7
2.2	PROPIEDADES DEL PROBLEMA DE ASIGNACIÓN DE TRÁFICO	8
2.2.1	Equilibrio de tráfico	8
2.2.2	Datos relevantes para el modelo de asignación	10
2.2.3	Clasificación de los problemas de asignación de tráfico	11
2.2.4	Asignación mediante equilibrio de usuario determinista	16
2.3	DEFINICIÓN DEL PROBLEMA	17
2.4	ESTADO DEL ARTE DE LOS MÉTODOS DE ASIGNACIÓN DE TRÁFICO	22
2.4.1	Clasificación de los métodos de asignación de tráfico atendiendo a su formulación	23
2.4.2	Clasificación de los métodos de asignación de tráfico atendiendo a su implementación numérica	27
2.5	PRINCIPALES ALGORITMOS DE ASIGNACIÓN	29
2.5.1	Frank & Wolfe	29
2.5.2	Florian et al. (2009) (modificado)	31
2.5.3	Dial (2006)	32
2.5.4	Bar-Gera (2010)	34
2.6	ANÁLISIS COMPUTACIONAL DE LOS ALGORITMOS DE ASIGNACIÓN DE TRÁFICO	36
2.7	IMPLEMENTACIÓN DE UN ALGORITMO DE ASIGNACIÓN DE TRÁFICO	42

2.8	RESULTADOS/APLICACIONES .....	58
2.8.1	REDES DE APLICACIÓN.....	59
2.8.2	RESULTADOS.....	63
<b>CAPÍTULO 3 PROBLEMA DE ESTIMACIÓN DE MATRICES ORIGEN – DESTINO.....</b>		<b>67</b>
3.1	INTRODUCCIÓN .....	67
3.2	PROPIEDADES DEL PROBLEMA DE ESTIMACIÓN DE MATRICES .....	68
3.2.1	Zonificación de la región de estudio .....	68
3.2.2	Periodo de tiempo para el que se define la matriz de viajes.....	69
3.2.3	Recogida de información en la estimación de matrices .....	70
3.2.4	Indeterminación del problema.....	71
3.2.5	La asignación como parte del problema del ajuste.....	74
3.3	DEFINICIÓN DEL PROBLEMA .....	77
3.4	ESTADO DEL ARTE DE LOS MODELOS DE ESTIMACIÓN DE MATRICES .....	
	A PARTIR DE DATOS DE AFOROS .....	81
3.4.1	Modelos Gravitacionales.....	82
3.4.2	Modelos de máxima entropía/mínima información .....	82
3.4.3	Modelos de máxima verosimilitud.....	84
3.4.4	Modelos de inferencia bayesiana .....	85
3.4.5	Modelos de mínimos cuadrados generalizados .....	87
3.5	MÉTODOS DE PROGRAMACIÓN BINIVEL.....	89
3.6	METODOLOGÍA DE RESOLUCIÓN PROPUESTA .....	93
3.7	CÁLCULO DEL JACOBIANO.....	97
3.8	FORMULACIÓN PROPUESTA PARA EL CÁLCULO DEL JACOBIANO .....	102
3.9	ALGORITMOS IMPLEMENTADOS PARA EL CÁLCULO DEL JACOBIANO	123
3.9.1	Algoritmo 1. Asignación basada en caminos. ....	125
3.9.2	Algoritmo 2. Asignación basada en orígenes. ....	129

3.9.3	Algoritmo 3. Asignación basada en orígenes con almacenamiento de segmentos alternativos.	131
3.9.4	Algoritmo 4. Asignación basada en orígenes con equilibrado simultáneo del conjunto de PAS.	134
3.9.5	Análisis de algoritmos implementados.	138
3.10	RESULTADOS/APLICACIONES	140
3.10.1	Ejemplo de aplicación 1: Red sencilla	141
3.10.2	Ejemplo de aplicación 2: Red de Tiergarten	145
3.10.3	Ejemplo de aplicación 3: Red de Chicago Sketch	150
3.10.4	Ejemplo de aplicación 4: Red de Madrid	153
3.10.5	Mejora del proceso de ajuste para la red de Chicago Sketch	156
<b>CAPÍTULO 4</b>	<b>CONCLUSIONES Y LÍNEAS FUTURAS</b>	<b>161</b>
4.1	CONCLUSIONES	161
4.1.1	Estimación de matrices origen – destino	161
4.1.2	Problema de asignación de tráfico	163
4.2	LÍNEAS FUTURAS	164
<b>BIBLIOGRAFÍA</b>		<b>167</b>
<b>APÉNDICE I</b>	<b>MEDIDAS DE DISTANCIA ENTRE MATRICES</b>	<b>177</b>



# ÍNDICE DE FIGURAS

<b>Figura 2-1:</b> Red de Tiergarten. Bush del origen 21.....	35
<b>Figura 2-2:</b> Red de ejemplo. Estructura de datos empleada.....	46
<b>Figura 2-3:</b> Iteraciones para el equilibrio de los caminos del mismo par .....	51
<b>Figura 2-4:</b> Par O-D $\beta$ en equilibrio .....	53
<b>Figura 2-5:</b> Par O-D $\beta$ , nueva situación en equilibrio .....	53
<b>Figura 2-6:</b> Modelo de red de Tiergarten.....	59
<b>Figura 2-7:</b> Modelo de red de Chicago Sketch .....	61
<b>Figura 2-8:</b> Modelo de red de Madrid (sin centroides).....	62
<b>Figura 2-9:</b> Convergencia de los algoritmos de asignación .....	64
<b>Figura 3-1:</b> Red de Tiergarten.....	73
<b>Figura 3-2:</b> Ejemplo 1. Red simulada .....	106
<b>Figura 3-3:</b> Ejemplo 1. Evolución del flujo en el arco $v_5$ .....	109
<b>Figura 3-4:</b> Algoritmo 1. Evolución función objetivo .....	128
<b>Figura 3-5:</b> Algoritmo 1. Matriz desactualizada frente a ajustada .....	128
<b>Figura 3-6:</b> Algoritmo 2. Evolución función objetivo .....	130
<b>Figura 3-7:</b> Algoritmo 2. Matriz desactualizada frente a ajustada.....	131
<b>Figura 3-8:</b> Algoritmo 3. Evolución función objetivo .....	133
<b>Figura 3-9:</b> Algoritmo 3. Matriz desactualizada frente a ajustada .....	134
<b>Figura 3-10:</b> Algoritmo 4. Evolución función objetivo .....	137
<b>Figura 3-11:</b> Algoritmo 4. Matriz desactualizada frente a ajustada .....	137
<b>Figura 3-12:</b> Análisis algoritmos. Función objetivo - iteraciones.....	138
<b>Figura 3-13:</b> Análisis algoritmos. Función objetivo – tiempo de computación.....	139
<b>Figura 3-14:</b> Ajuste Red Tiergarten. Función objetivo – iteraciones.....	146
<b>Figura 3-15:</b> Ajuste Red Tiergarten. Función objetivo – tiempo de computación.....	147
<b>Figura 3-16:</b> Ajuste Red Tiergarten. Distancia entre matriz ajustada y real .....	148

<b>Figura 3-17:</b> Ajuste Red Tiergarten. Distancia entre matriz ajustada y desactualizada.....	149
<b>Figura 3-18:</b> Ajuste Red Tiergarten. Correlación matriz ajustada y desactualizada.....	149
<b>Figura 3-19:</b> Ajuste Red Chicago Sketch. Función objetivo - iteraciones.....	151
<b>Figura 3-20:</b> Ajuste Red Chicago Sketch. Distancia entre matriz ajustada y real .....	152
<b>Figura 3-21:</b> Ajuste Red Chicago Sketch. Distancia entre matriz ajustada y desactualizada ..	152
<b>Figura 3-22:</b> Ajuste Red Madrid. Función objetivo – tiempo de computación .....	153
<b>Figura 3-23:</b> Ajuste Red Madrid. Función objetivo – tiempo de computación .....	155
<b>Figura 3-24:</b> Mejora Ajuste Chicago Sketch. Función objetivo – tiempo de computación.....	157

# ÍNDICE DE TABLAS

<b>Tabla 2-1:</b> Tipos de problemas de asignación de tráfico.....	15
<b>Tabla 2-2:</b> Necesidad de almacenamiento (apartado 2.5) .....	39
<b>Tabla 2-3:</b> Necesidad de almacenamiento.....	39
<b>Tabla 2-4:</b> Vectores A, B y lastnode .....	46
<b>Tabla 2-5:</b> Características red de Tiergarten .....	60
<b>Tabla 2-6:</b> Características red de Chicago Sketch.....	61
<b>Tabla 2-7:</b> Características red de Madrid .....	63
<b>Tabla 2-8:</b> Valor de la función objetivo final.....	65
<b>Tabla 3-1:</b> Expresiones empleadas en el problema de estimación de matrices.....	96
<b>Tabla 3-2:</b> Ejemplo 1. Descripción de los caminos de cada par O-D .....	106
<b>Tabla 3-3:</b> Ejemplo 1. Matriz O-D de partida .....	106
<b>Tabla 3-4:</b> Ejemplo 1. $\partial v_5 / \partial g_i$ .....	107
<b>Tabla 3-5:</b> Ejemplo 1. $\partial Z(g) / \partial g_i$ .....	107
<b>Tabla 3-6:</b> Ejemplo 1. Soluciones inicial y final.....	108
<b>Tabla 3-7:</b> Ejemplo 1. Modificación de la matriz O-D de partida .....	110
<b>Tabla 3-8:</b> Planteamiento inicial del grado de indeterminación del problema.....	116
<b>Tabla 3-9:</b> Indeterminación del problema .....	116
<b>Tabla 3-10:</b> Características de las redes de transporte empleadas.....	125
<b>Tabla 3-11:</b> Algoritmo 1. Tiempos de computación (segundos).....	127
<b>Tabla 3-12:</b> Algoritmo 2. Tiempos de computación (segundos).....	129
<b>Tabla 3-13:</b> Algoritmo 3. Tiempos de computación (segundos).....	133
<b>Tabla 3-14:</b> Algoritmo 4. Tiempos de computación (segundos).....	136
<b>Tabla 3-15:</b> Algoritmos implementados. Resumen del tiempo de ejecución (segundos) .....	139
<b>Tabla 3-16:</b> Ejemplo de aplicación 1. $\partial v_5 / \partial g_i$ .....	142
<b>Tabla 3-17:</b> Ejemplo de aplicación 1. $\partial Z(g) / \partial g_i$ .....	142

<b>Tabla 3-18:</b> Ejemplo de aplicación 1. Solución inicial y final. ....	143
<b>Tabla 3-19:</b> Ejemplo de aplicación 1. Matrices O-D estimadas con los diferentes métodos. ..	144
<b>Tabla 3-20:</b> Ejemplo de aplicación 2. Distorsión de la matriz ajustada respecto de la inicial.	150
<b>Tabla 3-21:</b> Mejora Ajuste Chicago Sketch. Porcentajes respecto del tiempo total de ajuste.	157
<b>Tabla 3-22:</b> Mejora Ajuste Chicago Sketch. Distorsión matriz ajustada. ....	158

# CAPÍTULO 1

---

## INTRODUCCIÓN

### 1.1 ÁMBITO DE LA TESIS

El trabajo desarrollado en esta tesis se centra en el estudio de la modelización del transporte. En las últimas décadas ha habido un creciente interés por parte de las autoridades en mejorar las redes de transporte y su gestión. Este interés se debe a que el constante crecimiento de la movilidad de las personas está derivando en problemas de pérdida de calidad de vida, problemas medioambientales y de integración social.

Además de los medios tecnológicos como las computadoras o internet, el transporte es un elemento importante de la globalización. Gracias al transporte es posible mantener conectada la población facilitando las funciones que desempeñan a diario sus miembros. Es un factor fundamental en la economía de un país e influye en la forma de vida de los habitantes. La movilidad y accesibilidad son factores determinantes del estilo de vida y prosperidad.

A pesar de los aspectos positivos del transporte, éste está teniendo cada vez más importancia por los problemas que está generando y que son difíciles de solventar. El crecimiento económico parece haber introducido una demanda tal que sobrepasa las capacidades de la mayoría de los sistemas de transporte. A su vez, hay que sumar un periodo de limitada inversión, escasa planificación y énfasis en el corto plazo que ha acentuado la gravedad de dichos problemas.

El principal inconveniente al que se enfrentan cada día los centros de transporte de las administraciones es la congestión de vehículos en los centros urbanos. La saturación de vehículos en ciudades de tamaño medio y alto provoca un aumento de la contaminación, suponiendo una amenaza a la calidad de vida y al medio ambiente en dichas zonas. Al mismo tiempo, la congestión implica para los usuarios del transporte un elevado consumo de tiempo y energía. Actualmente la población presenta una gran dependencia del vehículo privado lo que genera una mayor congestión y provoca una menor autosuficiencia de las compañías de transporte público.

La solución de todos estos inconvenientes pasa por llevar a cabo una optimización del sistema de tráfico. Para poder analizar, describir y argumentar diferentes actuaciones en la gestión de la movilidad es necesario emplear la planificación del transporte, y apoyarse en modelos matemáticos que simulen el comportamiento de los diferentes elementos que conforman el sistema de transporte. La modelización es considerada como una herramienta de apoyo para la toma de decisiones.

La planificación del transporte abarca un gran número de actividades encaminadas a resolver distintas facetas de los problemas tráfico y movilidad. Esta tesis se centra en la modelización del transporte que es la parte de la planificación que se encarga de representar de forma simple el sistema de transporte en estudio.

Los modelos de transporte permiten analizar problemas actuales y adelantarse a problemas que podrían aparecer en el futuro. La precisión de los modelos de tráfico depende de la calidad de la información disponible. Una de las metodologías más empleadas es el modelo clásico de cuatro etapas que se divide en: generación de viajes, distribución de viajes, reparto modal y asignación.

La información principal de cada una de estas etapas es el conocimiento de los viajes de usuarios entre las diferentes zonas de la ciudad. Esta información se organiza en las matrices de viajes o matrices origen – destino que proporcionan una estimación del número de personas que se desplazan entre puntos de una determinada red en un periodo de tiempo dado, definiendo el mapa de movilidad. En esta tesis se estudia el problema de actualizar de una forma precisa y en un tiempo razonable este tipo de información para que represente con fidelidad la situación real.

## **1.2 MOTIVACIÓN**

El volumen de información que conforma el estado del arte de la estimación de matrices de viajes es enorme. Un gran número de autores y con muy variadas formulaciones se ha enfrentado a este problema durante los últimos 40 años. Éste es un proceso complejo influido por la congestión e indeterminación que generalmente presentan los diferentes sistemas de transporte. Las motivaciones que llevan al desarrollo de esta tesis son principalmente dos: *i*) los recientes avances metodológicos en la resolución del problema de asignación de tráfico que permiten desarrollar algoritmos de ajuste más eficientes y, *ii*) el avance en la tecnología de las computadoras que ofrece la posibilidad de emplear nuevas formulaciones y metodologías.

El objetivo de la estimación de matrices es el de encontrar la matriz que una vez asignada a la red represente la realidad con mayor fiabilidad. Para ello el proceso de estimación de matrices

incorpora en una de sus etapas el problema de asignación de tráfico. Ésta es una de las principales etapas por su influencia en los tiempos de computación y en la convergencia de la solución.

Desde el trabajo de Bar-Gera (2002) se ha producido una revolución en los métodos que resuelven el problema de asignación de tráfico. Estos nuevos métodos aportan soluciones mucho más convergentes en tiempos de computación más reducidos. La mejora de tiempos de ejecución y de la calidad de la solución plantea la posibilidad de generar nuevos métodos de ajuste de matrices.

Hasta la fecha los procedimientos empleados para definir formalmente y resolver el problema del ajuste de matrices de movilidad se caracterizan por ser simples y heurísticos. Esta simplicidad se debe a la baja capacidad de computación, respecto de las grandes dimensiones que presentan las redes de transporte, existente entre los años 1970 y 2000 donde se publicaron la gran mayoría de referencias bibliográficas. Actualmente la electrónica y la informática han avanzado tanto que posibilitan la implementación de nuevas formulaciones y metodologías que antes no eran posibles.

### **1.3 OBJETIVOS**

Una de las principales preocupaciones en la planificación y gestión del tráfico es la obtención de los datos de movilidad con una alta fiabilidad invirtiendo recursos temporales y económicos razonables. Para lograr estos objetivos es necesario el empleo de una gran cantidad de datos recogidos a través de diferentes fuentes de información. Esto conlleva que sea necesaria la generación de metodologías computacionalmente eficientes capaces de abordar redes de transporte reales (de gran tamaño en la mayoría de los casos).

En esta tesis se pone especial atención en el diseño de formulaciones y metodologías eficientes que ajusten redes de grandes dimensiones. La información que se supone de partida es: *i*) una matriz de viajes desactualizada, ya sea obtenida mediante un proceso de encuesta o rescatada de un proyecto anterior y, *ii*) datos de volúmenes en arcos obtenidos a través de estaciones de aforo.

En este proceso tiene gran importancia el problema de asignación de tráfico. Éste constituye el nivel inferior en los métodos de programación binivel, una de las metodologías más empleadas para la estimación de matrices. En la práctica, la asignación de tráfico supone un porcentaje

elevado del tiempo de ejecución del método de ajuste, aportando variables relevantes al nivel superior. En dicho nivel se resuelve un problema de programación convexa.

Con estas consideraciones previas, se detallan a continuación los objetivos que dirigen el desarrollo de la tesis:

- a. **Desarrollo de un método de estimación de matrices basado en una formulación binivel.** Este método tendrá como entradas la matriz desactualizada y datos de aforos. Deberá poder aplicarse a redes de gran escala.
- b. **Implementación de un algoritmo de estimación de matrices.** Estará basado en la formulación desarrollada e implementará el algoritmo de asignación de tráfico seleccionado.
- c. **Análisis de las diferentes implementaciones.** Se aplicarán los diferentes algoritmos desarrollados en redes de transporte reales de tamaño medio-alto.

Para alcanzar los objetivos enumerados será necesario llevar a cabo el desarrollo e implementación de un algoritmo de asignación de tráfico eficiente. Este algoritmo será diseñado para alcanzar grandes convergencias, deberá ser capaz de asignar redes de gran escala y, como requisito imprescindible, debe poder ser incluido en el algoritmo de programación binivel. Para ello se realizará un análisis del estado del arte y se compararán computacionalmente las diferentes alternativas.

## 1.4 ESTRUCTURA DE LA TESIS

El documento de tesis se divide en 4 capítulos.

En el Capítulo 1 se ha presentado el contexto en el que se encuadra la tesis y se han definido las motivaciones y objetivos que guían el desarrollo de la misma.

En el Capítulo 2 se realiza un estudio del problema de asignación de tráfico. Dicho estudio incluye una descripción general, un análisis del estado del arte y una comparativa de los principales algoritmos de asignación en equilibrio de usuario determinista. Además se lleva a cabo de forma más exhaustiva la descripción de un algoritmo de asignación basado en rutas. Esta implementación es una modificación del algoritmo de Florian et al. (2009) que trata de mejorar el equilibrado de flujo entre los caminos de cada par O-D.

En el Capítulo 3 se desarrolla un método de estimación de matrices origen – destino. Para este fin se realiza una descripción general del problema, se analiza el estado del arte de los modelos

de estimación de matrices que emplean datos de aforos y se propone una formulación y metodología de resolución. Además, según la formulación y metodología propuesta, se lleva a cabo la implementación en Matlab de varios algoritmos de ajuste. Por último se analiza la eficiencia computacional de dichas implementaciones mediante el estudio de varios casos reales.

En el Capítulo 4 se resumen las principales conclusiones acerca del trabajo desarrollado y se sugieren posibles desarrollos futuros.



# CAPÍTULO 2

---

## PROBLEMA DE ASIGNACIÓN DEL TRÁFICO (TAP)

### 2.1 INTRODUCCIÓN

La asignación de tráfico es la parte de la planificación del transporte que predice como se cargará la red de transporte. Conocida la topología de la red y la matriz de viajes, el problema de asignación pronostica cual será la distribución de vehículos sobre los arcos de la red. La solución de dicho problema reporta el flujo y tiempo de viaje en los arcos de la red, que resulta del uso de la misma por parte de todos los viajeros, para ir desde sus orígenes hasta sus destinos.

La cantidad de vehículos que atraviesa una calle o zona de la red depende de la decisión individual de cada uno de los usuarios que realizan un desplazamiento. Esta decisión está influenciada por diferentes factores que miden el nivel de servicio de la red. Los factores más relevantes son: distancia, tiempo de viaje, fiabilidad del tiempo de viaje, hora de salida, presencia de señales, accesibilidad, tipo de maniobras requeridas, tarifas y plazas de aparcamiento en destino, tarifas de acceso, seguridad y otros.

El número de variables que interviene en la decisión de desplazamiento de cada usuario es alto. Esto hace complejo encontrar una expresión que generalice al conjunto de la población, por ello es una práctica frecuente simplificar el número de variables. La aproximación más común es considerar un coste generalizado donde el tiempo de viaje es el principal factor de decisión para los usuarios de la red. Generalmente, cuando un conductor hace uso de la red es porque tiene necesidad de desplazarse para realizar una actividad en destino. El desplazamiento no es la actividad más importante, sino una actividad intermedia necesaria para desarrollar su actividad principal. Es por eso que cada usuario trata de reducir el tiempo de esta actividad intermedia. Para ello, cada usuario seleccionará la ruta cuyos arcos tengan menor tiempo de viaje.

El tiempo de viaje o nivel de congestión alcanzado en un arco es función proporcional a la cantidad de usuarios que deciden atravesarlo para llegar a su destino. Sheffi (1985) detalla la existencia de una interacción entre la congestión en los arcos y la decisión de los conductores de viajar por una determinada ruta. En el problema de asignación se modelan y resuelven simultáneamente todas estas interacciones individuales para encontrar el patrón de flujo sobre la red de transporte.

La visión más general del problema de asignación de tráfico es aquella en la que a partir de una matriz de demanda global se calcula para cada usuario la mejor alternativa de viaje. Cada individuo puede elegir entre diferentes modos de viaje (a pie, autobús, coche, tren,...) o incluso entre una combinación de modos para el mismo viaje. A su vez, dentro de cada modo o combinación, seleccionará la ruta más competitiva según sus prioridades (tiempo, distancia recorrida, número de maniobras,...). Éste es un problema complejo, ya que cada modo de transporte tiene unas características diferentes, además de que presentan valores de congestión distintos, existe interacción entre ellos, algunos tienen exclusividad en carriles y otros accesibilidad restringida a ciertas zonas de la red. Generalmente, para reducir la complejidad de la solución, este tipo de problemas suele enfocarse en asignar a la red únicamente el modo más común, el vehículo privado. Así, se supone que los problemas de congestión se deben principalmente a este modo y que, por tanto, la solución encontrada representa con alta fiabilidad lo que sucede en la red real.

Esta tesis se centra en la asignación de tráfico privado. La matriz de demanda solo representará a los usuarios que tienen la opción de viajar en su propio vehículo. Por tanto, se excluyen los demás modos de transporte y, en consecuencia, todas las combinaciones e interacciones entre ellos.

## **2.2 PROPIEDADES DEL PROBLEMA DE ASIGNACIÓN DE TRÁFICO**

### **2.2.1 Equilibrio de tráfico**

El problema de asignación de tráfico es visto por muchos autores como un problema de equilibrio en redes: “An efficient approach to solving the road network equilibrium traffic assignment problem” (LeBlanc et al, 1975), “Network equilibrium models and algorithms” (Florian y Hearn, 1995),... Esto se debe a que las técnicas utilizadas para resolver el problema de asignación de tráfico están basadas en la noción de equilibrio entre los usuarios del sistema. De esta forma se alcanza la solución resolviendo un problema de programación convexo. Esta

particularidad ofrece la posibilidad de implementar algoritmos que resuelven de forma eficiente problemas con redes de grandes dimensiones.

El patrón de flujo sobre la red de transporte responde a un equilibrio entre la congestión en los arcos y la decisión de viajar de los conductores por una determinada ruta. Este concepto es semejante a la noción de equilibrio del mercado empleada en economía. Para cualquier producto en el mercado siempre existen dos grupos interactuando: los productores y los consumidores. En el caso del problema de asignación, los consumidores son los individuos que tienen una necesidad de viaje o demanda de transporte y los productores, en este caso estarían representados por un único agente, el sistema de transporte formado por las vías, vehículos, terminales y la forma de interactuar entre ellos.

El comportamiento de los productores se rige según la curva de la oferta y el comportamiento de los consumidores según la curva de demanda. Generalmente la curva de la oferta y la demanda son funciones que dependen principalmente del precio. En un mercado perfectamente competitivo, al aumentar el precio, la oferta de productos aumenta mientras que su demanda disminuye, y viceversa.

Existen productos o servicios en los que no solo influye su precio sino que además influyen otros parámetros como, por ejemplo, el tiempo de espera para adquirir dicho producto o servicio. Este caso se presenta en gasolineras (Sheffi, 1985). Si una gasolinera disminuye el precio de sus combustibles, la demanda aumentará de forma considerable. Como cada cliente necesita de un tiempo de repostaje, se formarán colas. En este momento, comienza a influir una nueva variable como es el tiempo de espera en la gasolinera. Ciertos consumidores valoran ese tiempo de espera como negativo y decidirán repostar en otra gasolinera diferente.

En el problema de asignación, el precio del producto se corresponde con el tiempo de flujo libre del arco. Si disminuye este tiempo, el arco tendrá una demanda de vehículos mayor. Como en el caso de la gasolinera, si la demanda de vehículos es mayor a la ofertada, no se satisface instantáneamente formándose colas y aumentando el tiempo de flujo real en el arco.

Como resultado final del problema de asignación se obtienen los flujos en los arcos que se encuentran en equilibrio con el tiempo o coste generalizado de atravesar dichos arcos. Estudiando la estabilidad del problema, se define la solución final como un estado de equilibrio estable. Esto es, cuando el problema se encuentra en dicho estado no existen atracciones actuando sobre él que lo hagan moverse hacia otro estado diferente. Si el problema no está en equilibrio, se generan atracciones que hacen tender la solución hacia un estado en equilibrio estable. Estos factores, que actúan sobre el sistema, son los mecanismos de cambio de rutas de

los usuarios, desde su ruta actual hasta otra más competitiva. En el equilibrio, no existirá ningún incentivo para cambiarse de ruta.

Uno de los objetivos del problema de planificación del transporte es predecir como se van a comportar los vehículos por la red si se modifica algún parámetro del sistema de transporte. Por ejemplo, si se corta la circulación de una vía. Esta acción se modela asignando coste infinito al arco. En este momento la solución se desvía del estado de equilibrio y se activan los mecanismos de cambio de rutas que hacen tender al sistema a un nuevo estado de equilibrio. Después de varias iteraciones el sistema alcanzará un nuevo estado representado por una distribución de vehículos en equilibrio. En la práctica esta transición entre estados de equilibrio puede demorarse desde varios días a varias semanas dependiendo de la magnitud del problema.

## **2.2.2 Datos relevantes para el modelo de asignación**

### *2.2.2.1 Matriz de viajes*

Expresa la demanda de viajes estimada para los diferentes pares origen – destino (O-D). En la práctica, los elementos de la matriz de demanda están en constante variación a lo largo del día. Existe una matriz de viajes diferente para cada instante de tiempo. Como ejemplo, las zonas residenciales son zonas claramente productoras de viajes por lo que generan una gran cantidad de viajes al inicio de la jornada y atraen la misma cantidad de viajes al fin de dicha jornada. Así los pares O-D cuyos orígenes sean este tipo de zonas, presentan una cantidad de viajes alta en el horario de mañana que luego disminuirá. Y los pares O-D cuyos destinos sean este tipo de zonas presentarán un pico de viajes por la tarde.

Dado que la matriz de demanda no es constante, los sistemas de transporte en la práctica nunca están en estado de equilibrio de forma permanente. Por lo general, se supone que el sistema está cerca del equilibrio o tendiendo hacia dicho estado. Solo se considera que se desvía del equilibrio cuando se producen cambios en factores externos como pueden ser los cortes de tráfico, modificaciones del control semafórico, etc.

### *2.2.2.2 Grafo de la red de transporte*

Representa la red de transporte que quiere modelarse. El modelado de dicha red debe ser fiable, para ello es muy importante la definición del nivel de detalle. Cuanto mayor nivel de detalle, mayor semejanza habrá entre el grafo y la red en estudio, y por tanto más cercana a la realidad se encontrará la solución. No obstante, aumentar el nivel de detalle hace necesaria la recopilación de una mayor cantidad de información además de provocar un aumento del costo

computacional en el cálculo de flujos y costes en arcos. El nivel de detalle depende de los datos de salida necesarios. En la práctica, la red suele simplificarse eliminando vías secundarias o de menor importancia y manteniendo únicamente las vías que realmente van a ser importantes atendiendo al flujo de vehículos que las atraviesa.

Un apartado fundamental, dentro del diseño del grafo, es la caracterización funcional de los arcos. Esto conlleva asignar a cada arco una función de demora o curva de oferta de transporte que los caracterice. Las funciones de demora modelizan el comportamiento del arco frente a diferentes condiciones de flujo de vehículos. Los parámetros empleados generalmente en la definición de dichas funciones de demora son dos: la velocidad permitida (o tiempo de flujo libre) y la densidad o capacidad de vehículos que puede soportar el arco.

Es importante señalar que la distribución del flujo de vehículos en la solución final puede ser muy sensible a los parámetros empleados en las funciones de demora. Considerando que la matriz de demanda es estimada de forma muy fiable, el problema de asignación de tráfico proporcionará los flujos observados en los arcos si dichas funciones están correctamente calibradas.

#### 2.2.2.3 *Comportamiento de los usuarios*

Para resolver el problema de asignación de tráfico es necesario suponer que todos los conductores tienen un mismo comportamiento. El equilibrio entre la oferta y demanda de viajes se alcanza si, y solo si, el colectivo de conductores comparten un conocimiento pleno de las posibilidades del sistema de transporte y actúan bajo un mismo criterio. Solo de esta forma se llegará a una situación en la que ningún conductor puede reducir el coste de su desplazamiento mediante el cambio de ruta.

Anteriormente se han enumerado los diferentes factores que influyen en la elección de ruta de los usuarios. La aproximación más común es suponer que su comportamiento está dominado principalmente por un único factor, el coste de viaje (monetario y/o temporal). De ahora en adelante se supondrá que los usuarios actúan con la intención de minimizar únicamente su tiempo de viaje o, en una acepción más amplia, su coste generalizado (combinación de costes unitarios de diversas naturalezas).

### **2.2.3 Clasificación de los problemas de asignación de tráfico**

El problema de asignación de tráfico puede clasificarse en función de las diferentes tipologías de los datos del modelo de asignación descritos en la sección anterior.

### 2.2.3.1 *Atendiendo a la matriz de viajes*

Según se considere variaciones en la matriz de demanda o no, existen dos formulaciones completamente diferentes para el problema de asignación de tráfico:

- **Asignación del tráfico estática.** Se suponen las mismas condiciones de demanda durante un determinado periodo de tiempo (i.e. hora punta de mañana). La solución encontrada responde a una situación de flujo estacionario en equilibrio.
- **Asignación del tráfico dinámica.** Se utiliza la matriz de demanda real que es diferente en cada periodo de tiempo. La solución está cerca del equilibrio o tiende hacia él en todo momento.

La hipótesis formulada en los modelos estáticos es válida para problemas de planificación estratégica o táctica, mientras que para aspectos operacionales es más recomendable el uso de modelos dinámicos.

El uso de modelos dinámicos que sustituyan a los estáticos se debe a que estos últimos son incapaces de explicar cualquier evolución de los flujos de tráfico en cortos periodos de tiempo. Por ejemplo, durante un periodo en hora punta puede ocurrir que el número de vehículos que necesitan utilizar una vía sea mayor que su capacidad. Este exceso de vehículos hace que se formen colas que serán descargadas más tarde.

Los modelos estáticos se centran en varias horas del día utilizando valores medios de capacidad de vía, demanda de viajes y tiempo de flujo libre, encontrando escasos problemas de exceso de vehículos y, por tanto, de colas. Es por esto que muchos autores sostienen que los modelos estáticos subestiman los tiempos totales de viaje (Ben-Akiva y Lerman, 1985).

### 2.2.3.2 *Atendiendo al grafo de la red de transporte*

Relacionadas con las funciones de demora que definen los arcos de la red, existen varias clasificaciones del problema de asignación de tráfico:

- Según la dependencia de la función de demora de cada arco con el flujo de los demás arcos de la red, el problema puede dividirse en **simétrico** o **asimétrico**.
- Según la influencia del flujo de vehículos sobre el coste de viaje en los arcos, se distingue entre problema **con congestión** y **sin congestión**.

En los problemas asimétricos se admite que el coste de viajar por un arco puede verse influenciado por el flujo de cualquier otro arco de la red sin tener que darse la influencia

inversa. Es decir, el coste de atravesar un arco puede verse afectado por el flujo de otro arco pero su flujo no afecta de igual manera al coste de este otro arco.

Los problemas simétricos por otra parte verifican la expresión:

$$\frac{\partial s_a}{\partial v_b} = \frac{\partial s_b}{\partial v_a} \quad (2.1)$$

en los que  $a$  y  $b$  son dos arcos de la red de transporte,  $s_a, s_b$  sus costes y  $v_a, v_b$  los volúmenes en dichos arcos. Entre cada par de arcos, el flujo de uno influye de igual manera en el coste del otro.

En la bibliografía pueden consultarse artículos sobre formulaciones basadas en la asignación asimétrica (Nguyen y Dupuis, 1981; Fisk y Nguyen, 1982; Florian y Spiess, 1982; Smith, 1983) y simétrica (LeBlanc et al., 1975; Larsson y Patriksson, 1992; Jayakrishnan et al., 1994). El problema más extendido es un caso particular de la asignación simétrica, en el que el coste del arco depende únicamente de su propio flujo, esto es:

$$s_a(v) = s_a(v_a) \quad \forall a \in A \quad (2.2)$$

Este caso, también llamado de costes separables, tiene la característica de que las funciones de demora dependen única y exclusivamente de su flujo. Debido a esto, el problema de programación convexa que resuelve la asignación de tráfico se simplifica de forma considerable haciendo posible el desarrollo de algoritmos numéricos eficientes.

Esta tesis se centra en algoritmos que resuelven el problema de asignación estática. En este tipo de asignación, al suavizarse los problemas de congestión, no se forman colas en los arcos y por tanto la hipótesis de costes separables es razonable. Por el contrario en los problemas de asignación dinámica, al considerar la existencia de colas, es necesario considerar las influencias, de forma general, asimétricas entre arcos.

Las redes de transporte pueden dividirse según su escala en: urbanas e interurbanas. Dependiendo del tipo de red analizada, las funciones de demora de los arcos se ven más influenciadas por unas variables que por otras. Entre estas variables destacan la longitud, densidad de vehículos en los arcos, control semafórico, influencia de otros modos de transporte, entre otros.

En redes interurbanas las distancias entre pares O-D son generalmente importantes, la longitud de los arcos que componen este tipo de redes es grande por lo que el coste de atravesar cada

arco se debe principalmente a su longitud, dando lugar a los problemas de asignación de tráfico sin congestión. En el caso de redes urbanas la distancia entre pares O-D se reduce, siendo redes densas con arcos de poca longitud. Es por esto que el coste de atravesar cada arco se encuentra principalmente influenciado por la interacción entre los vehículos que lo atraviesan, dando lugar a los problemas de asignación de tráfico con congestión.

### 2.2.3.3 *Atendiendo al comportamiento de los usuarios*

Considerando el viaje, desde un origen a un destino, como una actividad intermedia necesaria para cumplir la actividad principal en destino, es razonable considerar que cada usuario tendrá un comportamiento egoísta tratando de minimizar su propio coste de viaje para poder desarrollar su actividad principal. Así, cada conductor actuará de forma independiente escogiendo la ruta que menor coste le suponga. Con este comportamiento, desde el punto de vista del colectivo de viajeros, se alcanzará un estado en el que ningún viajero puede mejorar su coste de viaje mediante el cambio de ruta. Éste es un estado en el que se cumple la condición de equilibrio de usuario (*UE: user-equilibrium*).

Si el problema de asignación de tráfico supone que, además de tener información completa acerca de las diferentes rutas posibles, los usuarios perciben el mismo coste de viaje bajo las mismas condiciones, se está resolviendo el problema de **equilibrio de usuario determinista**.

El principal inconveniente del equilibrio de usuario determinista es que se requiere una perfecta, o al menos uniforme, previsión (teóricamente con cero errores de percepción) del coste de viaje por parte de los usuarios. En la realidad, estas percepciones en los costes están sujetas a variaciones. Una generalización del equilibrio de usuario determinista es el problema de **equilibrio de usuario estocástico**. En el equilibrio de usuario estocástico se relaja la hipótesis de información perfecta por lo que no se tiene una percepción idéntica de los costes. En este caso se supone que todos los usuarios conocen la red con igual incertidumbre. Es decir, el coste percibido es considerado como una variable aleatoriamente distribuida entre los conductores. Así, en vez de tener una ruta con menor coste de viaje, existe un conjunto de rutas que pueden seleccionarse como de menor coste.

Una característica a destacar en los algoritmos de asignación es el grado de realismo alcanzado por los modelos. Puede comprobarse que, a través de las asignaciones estocásticas (Sheffi, 1985), asignaciones bicriterio (Dial, 1996; Leurent, 1996) y de los modelos dinámicos, se encuentran soluciones con una representación más realista de la distribución de los vehículos por la red. Además tienen la ventaja, respecto del equilibrio de usuario determinista, que tanto la solución en términos de flujos en arcos como de flujos en rutas es única.

El realismo alcanzado por los modelos no es la única propiedad que los hace útiles, existen otras consideraciones adicionales como son: “sensibilidades adecuadas a las decisiones políticas; capacidad para obtener datos de partida, así como datos de validación y calibración; estabilidad, repetitividad y consistencia; eficiencia computacional; puntos de vista, comprensibilidad y accesibilidad; confianza de los modeladores en los resultados y habilidad para convencer a otros que los resultados proporcionan una sólida base para tomar decisiones; etcétera” (Bar-Gera, 2010).

No existe ningún modelo que cumpla todas estas propiedades ni existe ningún criterio consistente que justifique una elección concreta. En esta tesis se implementa la asignación de equilibrio de usuario determinista por ser la más estudiada y aplicada en la práctica, además de que cumple una gran cantidad de características como indica Bernstein (1990), entre ellas: eficiencia computacional; estabilidad, repetitividad y consistencia.

Un comportamiento diferente al definido hasta ahora se basa en considerar una actuación altruista por parte de los usuarios en la que tratan de minimizar el coste de viaje del colectivo. Este tipo de problemas se denomina asignación óptima del sistema. Existen diferentes problemas en la bibliografía (ver Paradoja de Braess (Braess, 1968)) en los que se demuestra que minimizando el coste de viaje del colectivo se obtendrían mejores costes individuales. Corresponde con el punto de vista del planificador o gestor del sistema, que busca un bien general del mismo. En la práctica no pueden considerarse este tipo de comportamientos por parte de los usuarios aunque estas asignaciones suelen ser empleadas como criterio para diseñar la red de transporte.

#### 2.2.3.4 Resumen de los problemas de asignación de tráfico

En la siguiente tabla se muestran las diferentes clasificaciones del problema de asignación de tráfico descritas en los párrafos anteriores. Los problemas de asignación estarán perfectamente determinados cuando se pueda distinguir su tipología dentro de cada una de las cinco clasificaciones que se muestran en la tabla.

Evolución de la demanda	Comportamiento usuarios		Influencia en las funciones de demora	
	Minimización costes	Percepción costes	Flujo - Distancia	Flujo Entre Arcos
Estático	Equilibrio usuario	Determinista	Con Congestión	Simétrico
Dinámico	Sistema óptimo	Estocástico	Sin Congestión	Asimétrico

Tabla 2-1. Tipos de problemas de asignación de tráfico.

#### **2.2.4 Asignación mediante equilibrio de usuario determinista**

El desarrollo de esta tesis se centra en los modelos más conocidos en la práctica que están siendo empleados más profusamente. Atendiendo a la clasificación descrita en la Tabla 2-1, el modelo de asignación de tráfico que se considera es el estático, supone que los usuarios se comportan según el punto de vista de equilibrio de usuario determinista, considera que puede haber congestión en los arcos y que el coste de dichos arcos es separable, es decir, solo depende de su propio flujo.

Como se ha comentado en apartados anteriores, los modelos estáticos de asignación son apropiados para problemas de planificación táctica o estratégica. En estos modelos se conoce la matriz de demanda, que es constante, y que representa la cantidad de viajes entre cada par O-D durante el periodo de tiempo determinado. Este lapso de tiempo puede ser de minutos, horas, días e incluso semanas dependiendo del objetivo del proyecto. Cuanto mayor sea el periodo de tiempo estudiado, menores serán los problemas de congestión ya que los picos de demanda de tráfico en los arcos se alisan.

El concepto de equilibrio de tráfico aparece por primera vez en 1924, cuando Frank Knight formula la siguiente proposición (Knight, 1924): “Se obtendrá un equilibrio de usuarios, cuando para cada par origen – destino los costos de operación sobre todas las rutas sean iguales”. Según Knight todas las rutas existentes entre cada par O-D deben tener tal cantidad de flujo para que sus costes sean iguales. Esta definición es inconsistente ya que no es necesario utilizar todas las rutas que conectan los pares O-D.

La hipótesis de un comportamiento individualista en la asignación de tráfico es principalmente atribuida a Wardrop (1952), el cual estableció dos principios que formalizan la noción de equilibrio a través del concepto de conducta de los usuarios que minimiza el coste total de viajes.

Según el primer principio de Wardrop: "Ningún usuario puede mejorar su tiempo de viaje o costo generalizado cambiando unilateralmente de ruta". Otra forma de describir el equilibrio de usuario según Wardrop es la siguiente: “El tiempo de viaje en todas las rutas utilizadas es igual, a lo sumo igual o menor, que el tiempo de viaje que experimentaría un usuario adicional en cualquier ruta diferente a las utilizadas”.

El segundo principio de Wardrop establece que: “En el equilibrio el tiempo de viaje medio es mínimo”. Esta afirmación implica que cada usuario se comporta de forma cooperativa en la elección de su propia ruta con el fin de asegurar el uso más eficiente del sistema completo, por

lo que caracteriza el equilibrio para el problema de asignación óptima del sistema. Este tipo de asignación no forma parte de los objetivos de la tesis.

Según la definición del primer principio, las rutas que conectan cada par O-D en el equilibrio se dividen en dos conjuntos. En el primer conjunto se incluyen las rutas que tienen flujo, el coste de todas estas rutas será el mismo. El otro conjunto contiene las rutas que no tienen flujo, el coste de viajar en estas rutas sería mayor al de las rutas del primer conjunto (Sheffi, 1985). Con esta clasificación se evidencia la hipótesis de que todos los usuarios perciben el coste de la misma manera y además, tienen información perfecta de todas las rutas.

El primer modelo matemático para redes de transporte en equilibrio fue formulado por Beckmann et al. (1956). En su artículo demuestran que resolver la “Transformada de Beckmann” es equivalente a resolver un problema de asignación de equilibrio de usuario determinista. Con este modelo se demuestra que la solución de flujos en arcos, bajo ciertas condiciones de monotonía de las funciones de demora, es única, no siendo así para la solución de flujos en caminos.

## 2.3 DEFINICIÓN DEL PROBLEMA

Una vez descritos cualitativamente los diferentes problemas de asignación de tráfico y sus propiedades, se pasa a formular matemáticamente el problema de asignación de equilibrio de usuario determinista. Para ello será necesario introducir la notación empleada.

Se considera que la red de tráfico permite el flujo de un único tipo de modo en sus arcos y que está formada únicamente por arcos direccionales.

El grafo está compuesto por el conjunto de nodos  $i \in \mathcal{N}$ , que representan los orígenes, destinos e intersecciones, y el conjunto de arcos  $a \in \mathcal{A}$ , que representan las vías de la red.

La demanda de viajes entre un origen y un destino se representa indistintamente mediante una matriz de viajes  $T_{ij}$ , donde  $i$  son los orígenes y  $j$  los destinos, o por un pseudovector de flujos  $g_p$ , donde  $p$  son los diferentes pares O-D que componen el sistema.

Se denota por  $K$  al conjunto total de caminos activos que conectan los diferentes pares O-D,  $K_p$  representa los caminos pertenecientes al par O-D  $p$ .  $h_k$  y  $C_k$  son el flujo y el coste de cada camino  $k \in K$ . El flujo de los caminos debe satisfacer las propiedades de conservación del flujo y de no negatividad:

$$\sum_{k \in K_p} h_k = g_p \quad \forall p \in P \quad (2.3)$$

$$h_k \geq 0 \quad \forall k \in K \quad (2.4)$$

La restricción (2.3) puede escribirse de forma matricial como  $g = \Gamma \cdot h$ , donde  $\Gamma$  es la matriz de incidencia pares O-D – caminos y tiene dimensiones  $|P| \times |K|$ , siendo  $|P|$  el número de pares O-D y  $|K|$  el número de rutas.

Pasando al espacio de los arcos,  $v_a$  y  $s_a$  representan el flujo y el coste para cada arco  $a \in A$ . La relación entre los flujos en arcos y en caminos se define como:

$$v_a = \sum_{p \in P} \sum_{k \in K_p} \delta_{ak} h_k \quad \forall a \in A \quad (2.5)$$

donde

$$\delta_{ak} = \begin{cases} 1 & \text{Si el arco } a \text{ pertenece a la ruta } k \\ 0 & \text{e.o.c.} \end{cases}$$

La restricción (2.5) define el flujo de cada arco como la suma de los flujos de los caminos que pasan por él. Esta expresión puede escribirse de forma matricial como:

$$v = \Lambda \cdot h \quad (2.6)$$

donde  $\Lambda$  es la matriz de incidencia arcos – caminos con dimensiones  $|A| \times |K|$ , siendo  $|A|$  el número total de arcos. El valor de un elemento  $ij$  de esta matriz es 1 ó 0, dependiendo de si el arco  $i$  pertenece o no al camino  $j$ .

El coste de cada camino puede calcularse como la suma de los costes (siempre no negativos) de los arcos que lo componen:

$$C_k = \sum_{a \in A} \delta_{ak} s_a(v) = \sum_{a \in A} \delta_{ak} s_a(\Lambda \cdot h) \quad \forall k \in K \quad (2.7)$$

$$C_k \geq 0$$

Para cada par O-D  $p$  se define el parámetro  $U_p$  como el mínimo coste para realizar el viaje. Este coste se calcula como:  $U_p = \min_{k \in K_p} C_k$ , y por tanto  $U_p \geq 0$ .

El concepto de equilibrio de usuario se puede describir en términos matemáticos mediante las siguientes expresiones:

$$\begin{aligned} \text{Si } h_k > 0 &\Rightarrow C_k = U_p \\ \text{Si } h_k = 0 &\Rightarrow C_k \geq U_p \\ C_k, U_p &\geq 0 \end{aligned} \quad \forall \begin{cases} k \in K_p \\ p \in P \end{cases} \quad (2.8)$$

En estas expresiones se distinguen los dos conjuntos de caminos o rutas que se generan aplicando el primer principio de Wardrop, descritos en el apartado anterior.

Por lo general el número de caminos que conectan un par O-D es elevado, por lo que en la práctica solo se consideran o almacenan los que tiene flujo no nulo. Estos se denominan caminos activos y están representados por la primera de las expresiones en (2.8). Una forma equivalente de escribir las expresiones anteriores es la siguiente (Florian y Hearn, 1995):

$$\begin{aligned} (C_k - U_p) h_k &= 0 \\ C_k - U_p &\geq 0 \quad \forall k \in K_p, p \in P \\ C_k, U_p &\geq 0 \end{aligned} \quad (2.9)$$

La formulación de las condiciones de equilibrio mediante un modelo de optimización fue realizada de forma pionera por Beckmann et al. (1956):

$$\begin{aligned} \text{Min} \quad Z &= \sum_{a \in A} \int_0^{v_a} s_a(x) dx \\ \text{s.a.} \quad \sum_{k \in K_p} h_k &= g_p \quad \forall p \in P \\ v_a &= \sum_{k \in K} \delta_{ak} h_k \quad \forall a \in A \\ h_k &\geq 0 \quad \forall k \in K \end{aligned} \quad (2.10)$$

La solución de este problema de optimización cumple las condiciones detalladas en el primer principio de Wardrop. En (2.10) la función objetivo es conocida como *Transformada de Beckmann*. En esta formulación se supone que el coste de cada arco sólo depende de su propio flujo (costes separables). Imponiendo además que este coste es positivo y monótonamente creciente con el flujo, puede comprobarse que la función objetivo  $Z$  es estrictamente convexa (Sheffi, 1985). Dado que cada una de las restricciones en (2.10) es lineal puede concluirse que la solución del problema de equilibrio de usuario es única expresada en términos de flujos en arcos. No ocurre lo mismo cuando se resuelve el problema en términos de flujos en rutas; en

este caso aunque la solución existe, ésta no es única ya que los flujos en arcos pueden obtenerse a través de varias combinaciones de flujos en rutas. Esto se produce porque generalmente en las redes de transporte hay muchos más caminos que arcos.

La función objetivo del problema (2.10), suma de las integrales de la función de demora de los arcos, no tiene interpretación física, económica o conductual; debe ser vista estrictamente como una función matemática que es utilizada para resolver los problemas de equilibrio de forma que satisfaga el primer principio de Wardrop.

### 2.3.1 Formulaciones equivalentes

En este apartado se revisan modelos matemáticos que permiten abordar situaciones más generales que las posibilitadas por la programación convexa diferenciable. Gracias a estos modelos, pueden tratarse tanto problemas de costes simétricos como problemas de costes asimétricos; por ejemplo, el caso en el que el coste de los arcos depende de su flujo y del flujo de los demás arcos de la red. Estos modelos permiten además formular problemas en los que existen varios tipos de usuarios y/o varios modos de transporte.

En la bibliografía se encuentran diferentes formulaciones para resolver los problemas de asignación de tráfico como son los basados en: desigualdades variacionales, problemas de complementariedad y problemas de punto fijo. Todas ellas facilitan el establecimiento de las condiciones de existencia y unicidad de las soluciones y el desarrollo de algoritmos (Florian y Hearn, 1995).

#### a) Desigualdad variacional

Las desigualdades variacionales son las formulaciones más utilizadas para modelar las condiciones de equilibrio de usuario. Fueron introducidas para el problema de asignación de tráfico por Dafermos (1980). Éstas buscan un vector  $x^* \geq 0$  tal que:

$$F(x^*)(x - x^*) \geq 0 \quad \forall x \geq 0 \quad (2.11)$$

donde  $x$  es un vector columna compuesto por el vector de flujo en caminos y el vector de costes mínimos en los pares:

$$x = \begin{Bmatrix} h \\ U \end{Bmatrix} \quad \begin{array}{l} |K| \text{ rutas} \\ |P| \text{ pares OD} \end{array} \quad (2.12)$$

y donde  $F$  es un vector de valores de la función con los siguientes componentes:

$$F(x) = \begin{cases} C_k - U_p \\ \sum_{k \in K_p} h_k - g_p \end{cases}^T \quad \begin{array}{l} |K| \text{ componentes} \\ |P| \text{ componentes} \end{array} \quad (2.13)$$

Resolver el problema de desigualdades variacionales implica resolver las expresiones introducidas anteriormente (2.3) y (2.9):

$$\begin{aligned} h_k \cdot (C_k - U_p) &= 0 & \forall k \in K_p, p \in P \\ (C_k - U_p) &\geq 0 & \forall k \in K_p, p \in P \\ \sum_{k \in K_p} h_k - g_p &= 0 & \forall p \in P \\ h_k &\geq 0, U_p \geq 0 \end{aligned} \quad (2.14)$$

Desarrollando la desigualdad variacional (2.11):

$$\begin{aligned} &\overbrace{\left\{ (C_k^* - U_p^*)^T \left[ \sum_{k \in K_p} h_k^* - g_p \right]^T \right\}}^{F(x^*)} \cdot \overbrace{\left( \left\{ \frac{h_k}{U_p} \right\} - \left\{ \frac{h_k}{U_p} \right\}^* \right)}^{(x-x^*)} = \\ &= \sum_{p \in P} \sum_{k \in K_p} (C_k^* - U_p^*) (h_k - h_k^*) + \sum_{p \in P} \left( \sum_{k \in K_p} h_k^* - g_p \right) (U_p - U_p^*) = \sum_{p \in P} \sum_{k \in K_p} C_k^* (h_k - h_k^*) - \\ &- \sum_{p \in P} \sum_{k \in K_p} U_p^* h_k + \sum_{p \in P} \sum_{k \in K_p} U_p^* h_k^* + \sum_{p \in P} \left( \sum_{k \in K_p} h_k^* - g_p \right) U_p - \sum_{p \in P} \sum_{k \in K_p} h_k^* U_p^* + \sum_{p \in P} g_p U_p^* = \\ &= \sum_{p \in P} \sum_{k \in K_p} C_k^* (h_k - h_k^*) - \sum_{p \in P} \left( \sum_{k \in K_p} h_k - g_p \right) U_p^* + \sum_{p \in P} \left( \sum_{k \in K_p} h_k^* - g_p \right) U_p \geq 0 \end{aligned}$$

Se observa que si los vectores de flujos en caminos  $h_k$  y  $h_k^*$  son factibles, es decir, son positivos y cumplen la condición de conservación del flujo (2.3), el problema de desigualdades variacionales se simplifica a:

$$C(h^*)^T (h - h^*) \geq 0 \quad (2.15)$$

Esta expresión señala que dado un vector de flujos en equilibrio de usuario, cualquier desviación respecto de dicho vector no puede reducir el coste total. En otras palabras, los

usuarios no podrán reducir su coste de viaje al cambiar su ruta ya que no existen rutas alternativas con menores costes (Bell y Iida, 1997). Esta expresión es la que define el principio de Smith (Smith, 1979): “Los usuarios de una red sólo cambiarán de ruta si el costo total de operación sobre la red, calculados en base a los costos observados antes del cambio, disminuye”. Por tanto el principio de Smith permite una nueva formulación matemática del primer principio de Wardrop.

Como el coste en una ruta está definido como la suma de los costes de cada uno de sus arcos (costes aditivos), las condiciones de Wardrop pueden ser descritas en términos de flujos en arcos.

$$s(v^*)^T (v - v^*) \geq 0 \quad (2.16)$$

donde el vector  $v^*$  contiene la solución de flujos en arcos que cumple las condiciones de equilibrio de usuario determinista,  $s(v^*)^T$  es el correspondiente vector de costes en arcos y  $v$  es una solución factible de flujos en arcos que cumple las restricciones de conservación y no negatividad del flujo.

## 2.4 ESTADO DEL ARTE DE LOS MÉTODOS DE ASIGNACIÓN DE TRÁFICO

El principal tema abordado en esta tesis es el ajuste de matrices de viajes. En estos problemas se resuelven, en cada iteración, una gran cantidad de problemas de asignación. En la mayor parte de los algoritmos de ajuste de matrices resueltos en esta tesis, el problema de asignación ha supuesto el 80-90% del tiempo total de computación. Por lo tanto, es muy importante encontrar un algoritmo de asignación que aporte soluciones precisas además de bajo coste en tiempos de ejecución.

Como se describe en los apartados anteriores, existe una multitud de problemas de asignación de tráfico diferentes dependiendo de las características descritas en la Tabla 2-1. Además pueden definirse varios tipos de formulaciones equivalentes para describir cada uno de estos problemas. Debido a ello el número de algoritmos encontrados en la bibliografía es muy extenso y variado. Como ejemplo puede consultarse Patriksson (1994) donde se describen y clasifican más de mil referencias.

De forma general la asignación de viajes a la red puede considerarse como un problema de programación matemática de gran escala. En los inicios de estos algoritmos, tras la publicación

de las condiciones de equilibrio de Wardrop y la transformada de Beckmann, debido a las limitadas capacidades de computación se hicieron importantes ciertos algoritmos como los de Frank & Wolfe (Frank y Wolfe, 1956) y descomposición simplicial (SD) (Holloway, 1974; Hohenbalken, 1977). Hoy en día, las capacidades de computación han aumentado considerablemente permitiendo el uso de algoritmos más precisos y eficientes como los de Bar-Gera (2002, 2010), Dial (2006) y Florian et al. (2009).

Todos los algoritmos nombrados en el párrafo anterior tienen en común que resuelven el mismo problema de asignación de tráfico. Dicho problema se caracteriza por utilizar la transformada de Beckmann para encontrar el equilibrio de usuario en una red congestionada, considerando costes en arcos separables y demanda de viajes constante.

Los algoritmos de asignación estática, que utilizan como función objetivo la transformada de Beckmann, son los más estudiados en la bibliografía. Éstos se formulan como un problema de optimización convexa diferenciable con restricciones lineales (expresión (2.10)). La región factible en este problema viene determinada por las restricciones de conservación y no negatividad del flujo. En estas restricciones no se tiene en cuenta la interacción entre flujos de vehículos que salen de orígenes diferentes. El flujo que sale de un origen únicamente interacciona con el flujo que sale de otro origen por medio de la función objetivo. Esta característica proporciona al problema estructura de producto cartesiano (Larsson y Patriksson, 1992; García, 2001). Es posible construir un amplio rango de algoritmos basados cada uno en una descomposición particular de la estructura del problema.

### **2.4.1 Clasificación de los métodos de asignación de tráfico atendiendo a su formulación**

Florian y Hearn (1995) y Patriksson (1994) coinciden en dividir este tipo de algoritmos en tres categorías. Florian y Hearn (1995) añaden una categoría adicional relativa a los problemas con capacidades restringidas en los arcos que no es objeto de estudio en esta tesis. Así, los algoritmos de asignación de tráfico formulados como problemas convexos con restricciones lineales se dividen en:

#### *2.4.1.1 Métodos de linealización parcial*

Éstos son métodos de dirección factible. Partiendo de una solución factible inicial y resolviendo el subproblema obtenido por la linealización de la función objetivo, genera una nueva dirección de descenso factible. Así, mediante un método de búsqueda unidimensional se alcanza una nueva solución que minimiza la función objetivo.

La relación entre dos estimados sucesivos se puede formular como:

$$x^{k+1} = x^k + \alpha \cdot d^k \quad (2.17)$$

siendo  $d^k$  una dirección factible en  $x^k$  y  $\alpha \geq 0$ . Las soluciones factibles se encuentran definidas en términos de flujos en arcos.

Dentro de esta categoría se incluye el algoritmo de Frank & Wolfe (Frank y Wolfe, 1956). Éste busca el extremo de la región factible que hace que la función objetivo experimente un mayor descenso. La dirección de búsqueda factible se calcula como la diferencia entre el extremo de máximo descenso y la solución actual. Con dicha dirección se realiza una búsqueda unidimensional para obtener la combinación convexa óptima entre el extremo y la solución en flujos actual. En el caso de que la búsqueda lineal resulte computacionalmente costosa, Bell y Iida (1997) recomiendan reemplazar ésta por el método de las medias sucesivas:

$$v' \leftarrow v' \left(1 - \frac{1}{n}\right) + v^* \left(\frac{1}{n}\right) \quad (2.18)$$

donde  $n$  es el número de iteraciones,  $v'$  es la solución en flujos actual y  $v^*$  es el extremo en la iteración  $n$ .

Desde el punto de vista numérico, el cálculo del extremo equivale a obtener las rutas mínimas para cada par O-D, con los costes de la iteración actual, y asignarles toda la demanda (Sheffi, 1985). El flujo en los arcos se calcula a través de la ecuación (2.5), obteniendo el vector de flujos en arcos auxiliar o extremo.

Obsérvese que en esta metodología no es necesario almacenar los caminos mínimos empleados en el cálculo del extremo. Así los requerimientos de memoria son modestos y no aumentan con el número de iteraciones. Para cada iteración sólo es necesario almacenar los vectores actuales de flujos y de costes en los arcos, junto con los datos de la red de transporte.

#### 2.4.1.2 Métodos de generación de columnas

Estos métodos son propuestos para acelerar el algoritmo de Frank & Wolfe. Se centran básicamente en mejorar la búsqueda lineal de los algoritmos de linealización parcial. En el ámbito de la optimización lineal se conocen con el nombre de métodos de optimización lineal de descomposición. Para ello almacenan todos los vectores extremos empleados para la generación de las direcciones de búsqueda. Los vectores extremos en este caso son conocidos

como columnas. En cada iteración se genera y almacena una nueva columna que mejora la solución actual.

La búsqueda lineal que se resuelve en el algoritmo de Frank & Wolfe se transforma en el siguiente problema de optimización convexa conocido como problema maestro restringido:

$$\begin{aligned}
 \text{Min} \quad & Z \left( v = \sum_i v_i^* \cdot \lambda_i \right) \\
 \text{s.a.} \quad & \sum_i \lambda_i = 1 \quad \forall i = 1, \dots, n \\
 & \lambda_i \geq 0 \quad \forall i = 1, \dots, n
 \end{aligned} \tag{2.19}$$

La región factible está delimitada por los vectores extremos (columnas) calculados en las diferentes iteraciones. Cualquier solución puede ser escrita como una combinación convexa de estos vectores. Existe gran variedad de métodos numéricos que resuelven el problema maestro restringido como pueden ser el método de Newton o el de quasi-Newton.

Uno de los métodos de generación de columnas para problemas de optimización no lineal más conocidos es el enfoque de descomposición simplicial (SD) (Holloway, 1974; Hohenbalken, 1977). En este enfoque el problema maestro restringido considera todas las columnas generadas. En la práctica el método alcanza relativamente rápido una solución casi óptima pero pierde eficiencia cerca de la solución óptima. A medida que aumenta el número de puntos extremos necesarios para expresar la solución óptima, aumenta el número de variables en el problema maestro restringido haciendo menos eficiente el algoritmo.

Un segundo método conocido como descomposición simplicial restringida (RSD) fue introducido por Hearn et al. (1985). Estos autores introducen el parámetro  $r$  para limitar el número de columnas almacenadas en el problema maestro restringido. Cuando el número de columnas alcanza el valor  $r$ , se elimina la columna con menos peso en la combinación lineal convexa del problema maestro restringido y se añade una nueva columna. La eficiencia del algoritmo recae en gran medida en que el número de variables que maneja está restringido. En la práctica, un inconveniente de estos métodos es que para encontrar una solución cercana al óptimo se requieren valores elevados del parámetro  $r$  que mantengan una convergencia superlineal.

Larsson y Patriksson (1992) desarrollan el algoritmo de descomposición simplicial desagregada (DSD) haciendo uso de la estructura de producto cartesiano que presenta el problema. Este método se basa en la optimización lineal por descomposición en bloques. Estos autores realizan

una modificación del problema maestro restringido. Sustituyen el cálculo de una combinación convexa en el espacio total por varias combinaciones convexas, tantas como en productos cartesianos se haya dividido la red. Para cada grupo cartesiano se almacena un conjunto de extremos que forman una combinación convexa local. Todos los productos cartesianos constituyen en sí una combinación convexa global. El número de variables que maneja este método es mayor que los dos anteriores pero presenta mejores resultados computacionales.

La formulación de este problema se esquematiza como:

$$\begin{aligned}
 \text{Min} \quad & Z \left( \sum_{j=1}^{m_1} \hat{y}_1^{(j)} \cdot \lambda_1^{(j)}, \sum_{j=1}^{m_2} \hat{y}_2^{(j)} \cdot \lambda_2^{(j)}, \dots, \sum_{j=1}^{m_n} \hat{y}_n^{(j)} \cdot \lambda_n^{(j)} \right) \\
 \text{s.a.} \quad & \sum_{j=1}^{m_i} \lambda_i^{(j)} = 1 \quad \forall i = 1, \dots, n \\
 & \lambda_i^{(j)} \geq 0 \quad \forall j = 1, \dots, m_i; \forall i = 1, \dots, n
 \end{aligned} \tag{2.20}$$

Por último, se hace referencia al método de descomposición simplicial no lineal (NSD) que se presenta como una extensión de los algoritmos RSD. En los algoritmos SD y RSD el cálculo de las columnas se realiza mediante el algoritmo de Frank & Wolfe. El principal inconveniente de estas formulaciones estriba en que las direcciones de búsquedas se deterioran rápidamente mermando la calidad de las columnas. Es por esto que Larsson et al. (1997) modifican el problema de generación de columnas para obtener una mejor aproximación de la función objetivo esperando obtener una mejor aproximación en el problema maestro restringido.

#### 2.4.1.3 Métodos de descomposición (Equilibrio de caminos)

Estos métodos aprovechan la estructura de producto cartesiano que presenta el problema de asignación para transformarlo en una secuencia de problemas de menor dimensión. De igual forma, el algoritmo DSD descompone el problema en otros de menor tamaño. La diferencia de los algoritmos de descomposición respecto del algoritmo DSD estriba en que los primeros son formulados en términos de flujos en caminos.

Estos algoritmos pueden descomponerse por pares O-D o por orígenes. Cada par O-D, o cada origen, dependiendo de la división realizada, constituyen un problema de menor tamaño dentro del problema de asignación. Este enfoque general es equivalente a la descomposición de Gauss-Seidel que divide el problema en productos  $X_i$  de tal forma que representen el sistema global  $X = \prod_{i \in C} X_i$ . La descomposición de Gauss-Seidel es un método iterativo, también llamado de

descomposición cíclica, donde en un ciclo se resuelven uno a uno todos los productos

manteniendo constantes los demás. El problema termina cuando no hay mejoras en la función objetivo durante un ciclo. Cada producto  $X_i$  está asociado a los flujos en la red que salen de un origen o a los flujos entre un par O-D.

La descomposición de Gauss-Seidel resuelve el siguiente problema para cada producto  $p$  en los que se divide:

$$\begin{aligned}
 \text{Min} \quad & \sum_{a \in A} \int_0^{v_a^p + \tilde{v}_a} s_a(x) dx \\
 \text{s.a.} \quad & \sum_{k \in K_p} h_k = g_p \\
 & h_k \geq 0 \quad \forall k \in K_p
 \end{aligned} \tag{2.21}$$

donde  $\tilde{v}_a = \sum_{p' \neq p} \sum_{k \in K_{p'}} \delta_{ak} h_k$  (expresión (2.5)) corresponde al flujo de los demás productos que permanecen constantes, y  $v_a^p = \sum_{k \in K_p} \delta_{ak} h_k$  es el flujo del producto en estudio y las variables del problema.

La convergencia de este método está garantizada siempre que la función objetivo sea estrictamente convexa. El coste computacional es elevado comparado con los algoritmos de linealización parcial o de generación de columnas ya que es necesario almacenar todos los caminos encontrados y sus variables asociadas. Por otro lado, la convergencia alcanzada en estos métodos supera considerablemente la obtenida en los demás. Ejemplos de estos algoritmos son los desarrollados por Jayakrishnan et al. (1994), Bar-Gera (2002, 2010), Dial (2006), Florian (2009), Gentile (2012).

## 2.4.2 Clasificación de los métodos de asignación de tráfico atendiendo a su implementación numérica

A continuación se clasifican los principales algoritmos de asignación de equilibrio de usuario determinista en diferentes grupos con el fin de comentarlos y compararlos computacionalmente de forma sintética en el apartado 2.6. Estos grupos se caracterizan por tener métodos con estructuras internas, requerimientos de computación, requisitos de almacenamiento y resultados similares.

### 2.4.2.1 Grupo 1.- Frank & Wolfe (1956)

Es un algoritmo de linealización parcial. Las soluciones se almacenan en el espacio de los flujos en arcos. En cada iteración es necesario almacenar únicamente un vector de flujos en arcos. El

cálculo de la dirección factible se basa en encontrar un extremo al que trasladar la solución actual para mejorar la función objetivo. Para la obtención de esta dirección es necesario buscar los caminos mínimos desde todos los orígenes hasta todos los destinos. Esta dirección de búsqueda se deteriora cuando se acerca a la solución óptima proporcionando valores de convergencia modestos.

#### 2.4.2.2 Grupo 2.- *SD (1974), RSD (1985), DSD (1992)*

Son métodos de generación de columnas. El algoritmo DSD presenta mejores resultados computacionales y será el que se compare con los demás grupos. Este algoritmo crea para cada origen una copia de la red de transporte donde se representan los flujos que salen hacia los diferentes destinos. Además, almacena para cada red todos los extremos (o columnas) encontrados con el método de Frank & Wolfe.

Las necesidades de almacenamiento aumentan considerablemente en comparación con el anterior grupo. Son necesarios mantener guardados  $nO \cdot nC$  vectores en el espacio de flujos en arcos, donde  $nO$  representa el número de orígenes y  $nC$  es el número de columnas para cada origen. En este caso, al dividir la red en diferentes productos cartesianos, el algoritmo alcanza mejores convergencias que el anterior, aunque sigue presentando el problema de deterioro de las direcciones de búsqueda.

#### 2.4.2.3 Grupo 3.- *Jayakrishnan et al. (1994), Florian et al. (2009)*

Son métodos de descomposición o de equilibrio de caminos. En este caso se almacenan todos los caminos activos entre cada par O-D. Las diferencias entre cada algoritmo se centran en el cálculo del flujo desplazado entre los caminos de cada par O-D para que sus costes se igualen (primer principio de Wardrop).

#### 2.4.2.4 Grupo 4.- *Bar-Gera (2002, 2010), Dial (2006), Gentile (2012)*

Son métodos de descomposición o de equilibrio de caminos. En este caso se generan tantas subredes o arbustos acíclicos “*bushes*” como orígenes contiene la red principal de transporte. Son subredes porque solo incluyen los arcos que pertenecen a los caminos mínimos entre un origen y todos los destinos. En estos métodos no es necesaria la enumeración de caminos por lo que mejoran el costo computacional respecto de los métodos del grupo 3. El equilibrio de usuario se alcanza estudiando para cada origen todos los nodos intermedios de la subred. Desde cada nodo se desplaza el flujo entre los segmentos que salen de dichos nodos y llegan hasta los diferentes destinos.

## 2.5 PRINCIPALES ALGORITMOS DE ASIGNACIÓN

Para el desarrollo de esta tesis se ha llevado a cabo la implementación de los principales algoritmos que resuelven el problema de asignación del transporte con el fin de evaluar la eficiencia y oportunidades que cada uno ofrece en el problema de ajuste. Según se ha ido desarrollando la investigación en el ajuste de matrices, se ha hecho necesario un avance o cambio en el algoritmo de asignación. Finalmente los algoritmos implementados en MATLAB han sido: Frank & Wolfe, una modificación de Florian et al. (2009), Dial (2006) y Bar-Gera (2010). Estos algoritmos son mencionados en todo el documento de tesis por lo que se describirá su metodología destacando sus ventajas e inconvenientes. Una descripción detallada de estos algoritmos puede encontrarse en las publicaciones originales de sus autores. Sin embargo, como se han llevado a cabo modificaciones importantes en la metodología de Florian et al. (2009), la implementación de este algoritmo se detalla en el apartado 2.7.

### 2.5.1 Frank & Wolfe

Éste es el algoritmo más implementado y utilizado en la práctica por su simplicidad en el diseño, bajas necesidades de almacenamiento y obtención de soluciones con convergencias aceptables.

El algoritmo puede resumirse en los siguientes pasos:

**Paso 0.** *Inicialización.* Cálculo de una solución factible. Con los costes de flujo libre en los arcos se realiza la búsqueda de caminos mínimos desde cada origen a todos los destinos. Una vez encontrado un camino para cada par O-D, se le asigna toda la demanda,  $h^0 = g$ . La solución factible se obtiene sumando para cada arco el flujo de los caminos que pasan por él.

$$v_a^0 = \sum_{k \in K} \delta_{ak} h_k^0$$

**Paso 1.** *Actualización de costes de los arcos a partir de la solución de flujos en arcos actual.*

$$s^n = s(v^n)$$

**Paso 2.** *Búsqueda de un nuevo extremo de máximo descenso.* Con los costes en arcos actuales se realiza una nueva búsqueda de caminos mínimos desde cada origen

a todos los destinos. Una vez encontrado un camino para cada par O-D, se le asigna toda la demanda,  $h^n = g$ . El extremo se obtiene sumando para cada arco el flujo de los caminos que pasan por él.

$$e_a^n = \sum_{k \in K} \delta_{ak} h_k^n$$

**Paso 3.** *Cálculo de la dirección de búsqueda factible.* Ésta será la diferencia entre el extremo de máximo descenso y la solución actual.

$$d^n = e^n - v^n$$

**Paso 4.** *Actualización del flujo en arcos.* Se realiza una búsqueda unidimensional para obtener la combinación convexa óptima entre el extremo y la solución actual. La solución en arcos encontrada minimizará la función objetivo (función de Beckmann).

$$v^{n+1} = v^n + \lambda^n \cdot d^n$$

**Paso 5.** *Criterio de parada.* Si se alcanza el nivel de convergencia buscado, el algoritmo finaliza. En otro caso, el algoritmo establece  $n = n+1$  y continúa por el paso 1.

En todo momento la única variable que se almacena es la solución actual, es decir, el vector de flujos en arcos actual. La solución tiene dimensiones  $|A| \times |I|$ .

Los módulos implementados son: el cálculo de los caminos mínimos y la optimización unidimensional. Se comprueba la baja necesidad de almacenamiento y la simplicidad en el diseño.

Los inconvenientes encontrados en este método son principalmente el nivel de convergencia alcanzado y la falta de variables de salida que aporten información valiosa para el problema de estimación de matrices O-D.

La dirección de búsqueda de Frank & Wolfe, al estar basada en un extremo de la región factible, pierde efectividad en cuanto la solución se acerca al óptimo. Esto hace que una vez alcanzado el nivel de convergencia máximo aun se den diferencias de flujo en arcos considerables entre dos iteraciones consecutivas.

La principal característica que se destaca del algoritmo de Frank & Wolfe es la baja necesidad de almacenamiento. Esta característica se traduce en un inconveniente, para el problema de estimación de matrices, ya que la única información que aporta este método es la solución de flujos en arcos. El problema de asignación de tráfico puede aportar más información como es la relación de caminos que conecta los diferentes pares O-D y su flujo. Para ello se pasa a estudiar otros métodos que consideren estas variables.

### 2.5.2 Florian et al. (2009) (modificado)

La modificación de Florian et al. (2009) implementada se describe de forma exhaustiva en el apartado 2.7. Este método almacena los caminos de cada par O-D. El equilibrio de usuario se alcanza cuando no se encuentran caminos nuevos mínimos y los caminos que pertenecen a cada par O-D presentan igual coste. Así, sin necesidad de evaluar la transformada de Beckmann, alcanza el equilibrio de usuario cumpliendo el primer principio de Wardrop.

El diseño del algoritmo es simple, se buscan los caminos mínimos para cada par O-D, se comparan con los existentes y si son diferentes se almacenan. En el siguiente paso, para cada par O-D se calculan los costes de los caminos y si no coinciden se traspasa flujo entre dichos caminos hasta alcanzar el equilibrio en costes.

El equilibrado de los caminos de un par O-D tiene influencia sobre los costes de los demás caminos de la red ya que modifica el flujo en los arcos. Por tanto, al equilibrar un par O-D se están desequilibrando otros pares O-D. Para llegar a la situación de equilibrio de usuario se implementa un algoritmo iterativo que recorre los pares O-D con más de un camino activo.

Las variables principales son la matriz de caminos que tiene dimensiones  $|A| \times |K|$  y el vector de flujos en caminos con dimensiones  $|K| \times |1|$ . En la optimización del algoritmo se emplean variables internas adicionales que ayudan a agilizarlo.

Los módulos principales del algoritmo son: la búsqueda de caminos mínimos para cada par O-D, la gestión de caminos y el equilibrado de los caminos.

Gracias a la capacidad de los procesadores actuales, este algoritmo presenta varias ventajas respecto del algoritmo de Frank & Wolfe. Mejora de forma considerable los niveles de convergencia y lo hace en tiempos incluso inferiores a los de Frank & Wolfe ya que sus direcciones de búsqueda finales son eficientes durante todo el proceso.

La información desagregada en caminos que ofrece este algoritmo permite diseñar un método numérico para el cálculo del gradiente en el problema de estimación de matrices. Este método numérico obtiene muy buenos resultados aunque solo puede emplearse en redes de pequeño tamaño y no en redes de tamaño medio-alto. Por eso se pasa a estudiar otros métodos que aportan la misma información pero almacenada en variables de menor tamaño.

### 2.5.3 Dial (2006)

El algoritmo de Dial (2006) ha sido una revolución en los problemas de asignación de tráfico. Mejora los tiempos de asignación de los métodos anteriores, reduce la necesidad de almacenamiento y alcanza el mismo nivel de convergencia que los algoritmos basados en caminos.

La principal característica de este método es la no enumeración de caminos. Al no almacenar ningún camino se reduce en gran medida las necesidades de almacenamiento.

Este algoritmo necesita almacenar para cada origen un vector de flujos en arcos. Estos vectores de flujos en arcos deben cumplir ciertas propiedades para evitar la enumeración de los caminos y alcanzar el equilibrio de usuario. Dial los define como *bushes* acíclicos. Estos *bushes* no contienen todos los arcos de la red, solo contienen los arcos que salen desde el origen y llegan hasta todos los destinos, además no existe la posibilidad de que los arcos incluidos generen un ciclo.

Durante la ejecución del algoritmo los *bushes* acíclicos van siendo equilibrados y mejorados con la adición de nuevos arcos. El equilibrio de usuario se alcanza cuando no es necesario agregar más arcos y los diferentes *bushes* están equilibrados.

El algoritmo puede resumirse en los siguientes pasos:

**Paso 0.** *Inicialización.* Para cada origen se genera un *bush* acíclico como sigue. Con los costes de flujo libre en los arcos se realiza la búsqueda de caminos mínimos desde cada origen a todos los destinos. Una vez encontrado un camino para cada par O-D, se le asigna toda la demanda,  $h^0 = g$ . Cada *bush* acíclico está compuesto por los arcos con flujo estrictamente positivo. En esta iteración, cada *bush* coincide con el minimum spanning tree del origen al que pertenecen.

**Paso 1.** *Mejora de los bushes acíclicos.* Para cada *bush* se llevan a cabo los siguientes pasos: *i)* considerando los arcos del *bush*, se resuelve el algoritmo de Dijkstra

obteniendo el coste de alcanzar desde el origen  $i$  cada destino  $j$  ( $d_{ij}^{anterior}$ ), *ii*) a continuación, con todos los arcos de la red, se calcula el minimum spanning tree desde dicho origen hasta los destinos obteniendo el coste mínimo de alcanzarlos,  $d_{ij}^{actual}$ , *iii*) por último, si en algún destino se cumple que  $d_{ij}^{actual} < d_{ij}^{anterior}$ , significa que hay nuevos arcos o caminos que reducen el coste de alcanzar los destinos por lo que será necesario mejorar el *bush* agregando nuevos arcos. Antes de agregar cada arco debe comprobarse que no formará ciclos.

**Paso 2.** *Equilibrado de los bushes acíclicos.* Según Ahuja et al. (1993) es posible encontrar los caminos de coste mínimo y máximo entre el origen y los diferentes destinos siguiendo el orden topológico del *bush* en un único recorrido. Una vez calculados los caminos mínimos y máximos, se recorre cada destino dentro del *bush* comprobando si coincide el coste de los caminos de mínimo y máximo coste y, si no es el caso, se equilibran.

**Paso 3.** *Criterio de parada.* Si se alcanza el nivel de convergencia buscado, el algoritmo finaliza. En otro caso, el algoritmo establece  $n = n+1$  y continúa por el paso 1.

Las variables empleadas por el algoritmo son los *bushes* acíclicos. Es necesario un *bush* por cada origen que tiene dimensión de número de arcos. La necesidad de almacenamiento es de  $nOrigen \cdot |A| \times |1|$ , superando al algoritmo de Frank & Wolfe en  $nOrigen$  veces; sin embargo es muy inferior al tamaño de las variables utilizadas en los algoritmos basados en caminos.

Los módulos que componen el algoritmo son: cálculo del minimum spanning tree para cada origen, gestión de los arcos del *bush*, cálculo de los caminos de coste mínimo y máximo de cada par O-D y equilibrado de dichos caminos.

La enumeración de caminos es un proceso costoso que este algoritmo evita. Al no necesitar enumeración, este método a su vez obvia la gestión de dichos caminos que engloba la adición, almacenamiento y eliminación de éstos. Para agregar un nuevo camino es necesario compararlo arco a arco con los caminos ya existentes del par O-D. En los algoritmos de asignación basados en caminos, este proceso de comparación se realiza para todos los pares O-D en cada una de las iteraciones.

Una vez completado el proceso de asignación, la variable de salida es el flujo en arcos desagregado por orígenes. Ésta es una información más valiosa que la aportada por el método de

Frank & Wolfe aunque dista mucho de la aportada por los algoritmos basados en caminos ya que estos últimos desagregan el flujo en arcos por par O-D. Sin embargo, mediante la implementación de una nueva estructura algorítmica que considere el flujo en arcos desagregados por orígenes, es posible alcanzar métodos de ajuste de matrices que aporten soluciones semejantes a los basados en las variables salientes de los algoritmos de asignación basados en caminos.

#### **2.5.4 Bar-Gera (2010)**

En Bar-Gera (2010) se presenta un nuevo algoritmo, TAPAS. El propio autor reconoce que la idea básica del algoritmo, donde se traspasa flujo entre el segmento de máximo y el de mínimo coste, es muy parecida a la propuesta por Dial (2006). Al equilibrar dos caminos que conectan un par O-D, es posible eliminar los arcos que coinciden en dichos caminos y equilibrar únicamente la pareja de segmentos restante (PAS, paired alternative segments).

El algoritmo de Bar-Gera (2010) es muy similar al de Dial (2006), la principal característica que los diferencia es el concepto de parejas de segmentos alternativos, su identificación, gestión y almacenamiento, ya que los PAS se conservan de iteración en iteración. Puede parecer que con este algoritmo se vuelve a una enumeración similar a la de los algoritmos basados en caminos, pero no es así. Existe enumeración de segmentos pero el número de parejas de segmentos es incomparablemente inferior al número de caminos. Bar-Gera (2010) detalla en su artículo que: “afortunadamente no es necesario considerar todos los PAS posibles en una red”, “es suficiente elegir un subconjunto básico de PASs”.

Para un *bush* concreto, donde un origen se conecta con los diferentes destinos, es muy probable que existan pares O-D con el mismo PAS. Equilibrando un único PAS se alcanza el equilibrio en varios pares O-D. A su vez, en los demás *bushes* del problema existirán pares O-D que compartan de igual forma dicho PAS. Este hecho es común en las redes de transporte debido a su topología. A continuación se presenta como ejemplo la red de Tiergarten. Las características de esta red se describen en el apartado 2.8.1.1.

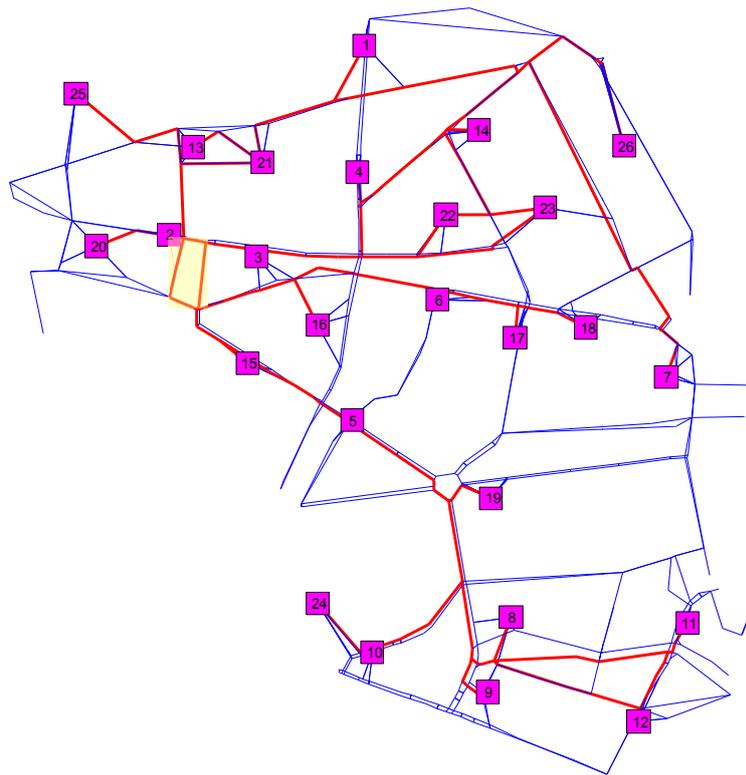


Figura 2-1. Red de Tiergarten. *Bush* del origen 21.

En la Figura 2-1 se muestra el *bush* acíclico perteneciente al origen 21 en la segunda iteración. Se dibujan de color rojo los arcos que pertenecen al *bush* y sombreado en amarillo se muestra un PAS que va a ser estudiado. Puede observarse que este PAS está contenido en los caminos correspondientes a los siguientes pares O-D: 21-5, 21-8, 21-9, 21-10, 21-11, 21-12, 21-15, 21-19 y 21-24. Por tanto, equilibrando dicho PAS, se estarán equilibrando todos los pares O-D enumerados.

En la figura anterior se observa que en los *bushes* acíclicos pertenecientes a los orígenes 2, 13 y 25 aparecerá el mismo PAS para unir dichos orígenes con los destinos 5, 8, 9, 10, 11, 12, 15, 19 y 24. Por tanto, se comprueba que el problema de asignación de tráfico puede simplificarse equilibrando un pequeño número de PAS.

El esquema algorítmico de Bar-Gera (2010) es similar al de Dial (2006). La diferencia se encuentra en que cuando Dial (2006) equilibra una pareja de caminos, los de máximo y mínimo coste, Bar-Gera (2010) además almacena el PAS que forma dicha pareja de caminos. Una vez que han sido estudiados todos los orígenes en cada iteración, el algoritmo de Bar-Gera (2010) recorre adicionalmente un bucle en el que todos los PAS almacenados son equilibrados, y eliminados en el caso de que el segmento de mayor coste se quede con flujo cero. Una vez se

equilibra un PAS almacenado, se redistribuye el flujo de forma proporcional entre todos los orígenes que lo contienen.

Las variables de salida de este algoritmo son los flujos en arcos desagregados por orígenes y el conjunto de segmentos PAS. El número de segmentos PAS es muy inferior al de pares O-D, además presenta la característica de que equilibrando todos los segmentos PAS se llegará a una situación de equilibrio de usuario del problema. Esta relación de segmentos PAS es una información muy importante en el ajuste de matrices de viajes que se implementa en esta tesis.

## **2.6 ANÁLISIS COMPUTACIONAL DE LOS ALGORITMOS DE ASIGNACIÓN DE TRÁFICO**

Para poder realizar una buena estimación de matrices O-D es necesario conocer como es el modelado de los flujos en la red para así seleccionar de forma coherente las variables que discriminen qué pares O-D serán ajustados. Los programas comerciales no ofrecen todos los detalles del proceso de asignación, además no describen los procesos internos que se llevan a cabo, ni ponen a disposición del usuario todas las variables que pueden ser de utilidad para desarrollar un algoritmo de ajuste sobre dicho software.

Es por ello que en esta investigación se decidió implementar algoritmos para resolver el problema de asignación de viajes, necesarios en el ajuste de matrices O-D, con el objetivo de alcanzar alta convergencia en tiempos razonables de computación aunque no alcancen la eficiencia de los existentes en la bibliografía. Además, se estudiará la posibilidad de aplicar estrategias de Warm Start con las que, partiendo de una solución anterior, se acelere el proceso de asignación. De esta forma, se consiguen disminuir los tiempos de computación de la asignación y, por tanto, el peso de la asignación en el tiempo global del problema del ajuste.

En la práctica los tiempos de computación marcan en gran medida la calidad de los resultados ya que los tiempos de proyectos están limitados cronológica y presupuestariamente. Esto hace que muchos modelos de redes de transporte no se encuentren lo suficientemente calibrados como para llevar a cabo estudios de previsión fiables debido a estos inconvenientes. Los costes de viaje en arcos calculados en la asignación son utilizados en los modelos de distribución, elección del modo de transporte, ajuste de la matriz de viajes, etc. Obtener soluciones con un alto nivel de convergencia es imprescindible para mejorar la consistencia de estos modelos. Si la convergencia alcanzada no es la adecuada, los flujos o costes en arcos fluctuarán en todas las iteraciones (Rabinowicz, 2008). Dependiendo de la cantidad de vehículos que atraviesen los arcos, las fluctuaciones entre dos iteraciones consecutivas pueden ser importantes sin dar

garantías de convergencia, obteniéndose patrones de flujo diferentes al cambiar mínimamente los parámetros de la asignación.

El algoritmo de asignación de tráfico más ampliamente implementado, con diferencia, para resolver el problema de equilibrio de usuario determinista es el de Frank & Wolfe. Tiene la particularidad de que su implementación es muy simple y el requerimiento de memoria modesto. Actualmente existen algoritmos que superan con creces las características que presenta este método en cuanto a tiempos de computación y niveles de convergencia alcanzados. Es posible que en la práctica aún haya que esperar varios años para que los modeladores del transporte se actualicen a uno de estos nuevos algoritmos ya que, aunque la mejora de los resultados es enorme (se alcanzan equilibrios más estables), la necesidad de calibración o adaptación de proyectos de gran escala a nuevas tecnologías supone un gran obstáculo para su implementación.

Los rápidos avances en la computación en cuanto a potencia de procesado, capacidad de memoria RAM y espacio de almacenamiento hace que se tengan en cuenta métodos que ahora son implementables cuando antes no lo fueron. Estos nuevos métodos, que actualmente están incorporándose en los programas comerciales, son los que se analizan en comparación con el algoritmo de Frank & Wolfe.

En los siguientes subapartados se comparan los diferentes grupos presentados, en la sección anterior, en cuanto a eficiencia de las direcciones de búsqueda, necesidad de almacenamiento, niveles de convergencia y tiempos de computación, entre otros parámetros.

### **2.6.1 Direcciones de búsqueda**

De los cuatro grupos enumerados en el apartado 2.4.2, los dos primeros utilizan el método de Frank & Wolfe para calcular las direcciones de búsqueda. Los grupos 3 y 4 utilizan otros métodos con mejores propiedades como son: el gradiente proyectado (Jayakrishnan et al., 1994; Chen et al., 2002), Rosen (Florian et al., 2009), Newton (Dial, 2006), Cuasi-Newton (Bar-Gera, 2002) y linealización de costes (Gentile, 2009). Las direcciones de búsqueda de los grupos 1 y 2 inicialmente presentan un buen comportamiento y tienden con rapidez al óptimo, pero a medida que la solución se acerca a dicho óptimo se deterioran de tal forma que pueden alcanzarse direcciones de búsqueda incluso perpendiculares entre iteraciones consecutivas. En la bibliografía existen intentos para mejorar este inconveniente ya que el algoritmo tiene muy buen comportamiento en las primeras iteraciones. Las principales modificaciones son: emplear el método de las medias sucesivas, en lugar de calcular la combinación lineal convexa de las

soluciones, PARTAN (Florian et al., 1987) y métodos de direcciones conjugadas (Mitradjieva y Lindberg, 2012). Estas modificaciones no presentan mejoras considerables en cuanto al tiempo de computación y convergencia, en comparación con los nuevos métodos, salvo el método de direcciones bi-conjugadas, propuesto por Mitradjieva y Lindberg (2012). Este método ha sido comparado directamente con los propuestos por Bar-Gera (2002), Dial (2006) y Florian et al. (2009) hasta convergencias de  $10^{-4}$ , mejorando incluso sus resultados si se utiliza la computación en paralelo (Caliper Corporation, 2010; Zhou y Martimo, 2010).

## 2.6.2 Necesidad de almacenamiento

Hasta hace pocos años, el espacio de almacenamiento requerido por los algoritmos ha sido la variable que ha determinado si éstos podían ser implementados o no, ya que cualquier programa de asignación debe ser capaz de resolver problemas de asignación de tráfico con redes de gran escala. Actualmente el espacio de almacenamiento sigue siendo una variable importante aunque es menos restrictiva gracias a los avances en la tecnología de almacenamiento.

Históricamente el método de Frank & Wolfe ha sido el algoritmo por excelencia ya que cumplía estos requerimientos de memoria almacenando únicamente un vector de flujos en arcos. Clasificando los demás algoritmos por orden de menor a mayor necesidad de almacenamiento, en primer lugar se sitúan los algoritmos del grupo 4 y en segundo lugar, en un mismo escalón, los del grupo 2 y 3. El orden de necesidad de almacenamiento de los métodos del grupo 4 es aproximadamente tantos vectores de flujos en arcos como orígenes tenga la red. Es muy superior a Frank & Wolfe pero en la actualidad puede ser fácilmente implementado y está demostrado que aporta mejoras a la obtención de la solución.

El algoritmo DSD y los métodos del grupo 3 pueden llegar a necesitar niveles altos de almacenamiento. Estos niveles actualmente pueden ser abordados por los equipos de computación aunque con grandes repercusiones en los tiempos de cálculo y procesado. El algoritmo DSD almacena para cada origen tantos extremos (vectores de flujo en arcos) como sean necesarios hasta alcanzar la convergencia. Los métodos del grupo 3 necesitan enumerar todos los caminos activos entre cada pareja O-D.

Algoritmos	Necesidad almacenamiento
Frank & Wolfe	$ A  \times  1 $
Dial (2006)	$ A  \times  Origenes $
Bar-Gera (2010)	$ A  \times  Origenes  +  A  \times  Segmentos $
DSD	$Origenes \cdot  A  \times  Extremos $
Florian et al. (2009)	$ A  \times  K  +  K  \times  1 $

Tabla 2-2. Necesidad de almacenamiento (apartado 2.5).

A continuación (Tabla 2-3) se muestra la necesidad de almacenamiento (Tabla 2-2) de cada uno de los algoritmos descritos cuando son aplicados sobre diferentes redes de transporte. Las redes empleadas se describen en el apartado 2.8.1.

Algoritmos	Tiergarten	Chicago Sketch	Madrid
Frank & Wolfe	2,9 KB	11,5 KB	68,5 KB
Dial (2006)	75,9 KB	4,4 MB	79,0 MB
Bar-Gera (2010)	81,7 KB	4,5 MB	83,2 MB
DSD	379,3 KB	21,8 MB	394,9 MB
Basado en caminos	74,2 KB	63,7 MB	3,4 GB

Tabla 2-3. Necesidad de almacenamiento.

Para el cálculo de los elementos totales en el algoritmo DSD se ha supuesto que se almacenan 5 extremos para cada origen. En el cálculo de los elementos totales en el algoritmo basado en caminos se ha supuesto que un 20% de los pares O-D tienen dos caminos y el resto un único camino. Para los demás algoritmos se han obtenido los datos resolviendo los problemas de asignación. Los tipos de datos considerados para calcular los valores de la Tabla 2-3 han sido *float* y *boolean*.

En la Tabla 2-3 se observa una diferencia apreciable entre los métodos que almacenan caminos respecto del resto. Para disminuir dicha necesidad de almacenamiento, en los métodos basados en caminos se emplean variables de tipo rala (*sparse*) ya que la matriz arcos – caminos tiene muchas casillas nulas. Aun así el tamaño de la información almacenada es muy alto comparado con los demás algoritmos.

### 2.6.3 Niveles de convergencia

Otra característica exigida a los algoritmos de asignación es alcanzar grandes niveles de convergencias (Boyce et al., 2004). Es importante que el algoritmo se caracterice por ser repetitivo, estable y consistente. A medida que disminuye el intervalo de tiempo en el que se centra el estudio o se disminuye el número de carriles, los efectos de saturación se agravan. Esto provoca que las alternativas de viajes de cada usuario también se multipliquen haciendo más difícil alcanzar la convergencia. Los algoritmos deben estar diseñados para solventar este tipo de problemas con altos grados de saturación.

Las convergencias que pueden alcanzar los algoritmos de asignación dependen principalmente de las direcciones de búsquedas y la existencia o no de ciclos. En los grupos 3 y 4 por construcción no pueden generarse ciclos. Además, en esta tipología de algoritmos las direcciones de búsqueda mejoran la convergencia en todas las iteraciones. Los grupos 1 y 2, y de una forma más acentuada Frank & Wolfe, tienen problemas de eliminación de ciclos y de deterioro de la dirección de búsqueda proporcionando valores de convergencia más modestos. Para un nivel alto de convergencia el número de iteraciones que necesita el método de Frank & Wolfe es desconocido pero muy alto. Janson y Zozaya-Gorostiza (1987) sugieren que su baja convergencia es debida a los ciclos.

### 2.6.4 Tiempos de computación

Como se ha puntualizado previamente, en la práctica, los tiempos de computación marcan en gran medida la calidad de los resultados ya que los tiempos de ejecución de los proyectos de aplicación profesional se encuentran limitados. Durante la calibración o validación de la red y durante el ajuste de la matriz de viajes es necesario llevar a cabo una gran cantidad de asignaciones. Para no hacer tediosos o incluso impracticables los pasos anteriores, el tiempo de cálculo no puede ser excesivo.

Los tiempos de computación de un algoritmo son debidos, en gran parte, a la potencia de procesado y capacidad de RAM de los ordenadores empleados. Si todos los algoritmos se implementan en una misma computadora, las diferencias de tiempos entre algoritmos se deben principalmente a la convergencia exigida, efectividad de las direcciones de búsqueda, cantidad de datos a mover o almacenar, existencia de ciclos, y a las pautas seguidas para el equilibrado de los orígenes o pares O-D.

Actualmente, los algoritmos del grupo 4 son los que necesitan un menor tiempo de computación para altos grados de convergencias (Slavin et al., 2006). Si la exigencia de convergencia es

modesta, los tiempos de computación de Frank & Wolfe (usando computación paralela) son menores a los del grupo 4 (Mitradjieva y Lindberg, 2012).

Para altos grados de convergencia, los únicos algoritmos viables son los pertenecientes a los grupos 3 y 4. En estos algoritmos no se forman ciclos y además tienen la capacidad de centrarse únicamente en ajustar los flujos de los caminos que aún no están en equilibrio. En los grupos 1 y 2, en cada momento se ajustan todos los caminos tanto si están como si no en equilibrio. Éstos, además, son incapaces de eliminar los caminos que durante el proceso, debido a sus altos tiempos de viaje respecto de los demás, se quedan con flujo cero.

La desventaja de los algoritmos del grupo 3 respecto del 4 está en la gran cantidad de información que almacenan. La no enumeración de los caminos activos, en el grupo 4, acelera de forma considerable la ejecución de los algoritmos. Comparando los tiempos de computación de los algoritmos dentro de un mismo grupo, la principal diferencia se encuentra en la eficiencia de las direcciones de búsqueda. Por ejemplo, Gentile (2012) presenta un algoritmo muy similar al que presenta Dial (2006) aunque teóricamente genera direcciones de búsqueda más efectivas. En este caso, según Nie et al. (2012) el método de Gentile (2012) irónicamente obtiene peores resultados. El hecho de emplear direcciones más efectivas acelera la solución de los subproblemas que componen la asignación, aunque podría alterar el suave comportamiento que muestra la convergencia en los métodos de Bar-Gera (2002) y Dial (2006).

En la sección 2.7 se lleva a cabo la descripción de un algoritmo de asignación que aporta de forma eficiente los datos necesarios para el ajuste de las matrices O-D. En este caso se diseña un algoritmo que pertenece al grupo 3 en el que, comparándolo con los ya existentes dentro de este grupo, mejora los tiempos de computación por medio de direcciones de búsquedas más efectivas.

### **2.6.5 Otras características importantes**

Por último, una característica que incluyen algunos programas comerciales para los problemas de asignación es la opción de Warm Start. Con esta opción la asignación no comienza desde el inicio sino que parte de una solución anterior. Dicha opción es válida para problemas en los que se resuelve una gran cantidad de asignaciones donde las entradas de una iteración a la siguiente se modifican levemente (es posible modificar cualquier dato de entrada siempre que el problema resultante sea muy similar al anterior). Estos datos son: la matriz de viajes, las funciones de demora e incluso la red (direcciones, eliminación y creación de arcos,...). Según Slavin et al.

(2006), la estructura del patrón de flujos en la red no cambia incluso si se modifica ligeramente la matriz de viajes.

Utilizando la opción de Warm Start los tiempos de computación disminuyen en gran medida ya que parten de una solución próxima al óptimo. Esta opción es fácilmente implementada en la mayoría de los algoritmos analizados hasta el momento. Según Slavin et al. (2006), la opción de Warm Start para el algoritmo de Frank & Wolfe tiene el mismo coste computacional que si empezara desde el principio.

La posibilidad de implementar la opción de Warm Start hace que los tiempos de computación de los algoritmos del grupo 3 se equiparen a los de grupo 4 a partir de la segunda iteración. Los algoritmos del grupo 4 son computacionalmente más rápidos que los del grupo 3, aunque a partir de la segunda iteración la principal desventaja de los algoritmos del grupo 3, que es el coste de almacenamiento, se reduce considerablemente.

## **2.7 IMPLEMENTACIÓN DE UN ALGORITMO DE ASIGNACIÓN DE TRÁFICO**

En este apartado se diseña un método que resuelve el problema de asignación. Dicho método se incluye dentro del grupo 3. Las ventajas que principalmente destacan en este tipo de algoritmos son las siguientes:

- Simplicidad de diseño. Básicamente se trata de encontrar los caminos mínimos para cada par O-D, almacenarlos y repartir el flujo de modo que se igualen sus costes. Si algún camino tiene flujo cero y coste mayor a los demás se elimina. Realizando estas acciones de forma reiterada se llega al equilibrio de usuario.
- Está basado en términos de flujos en caminos. La información que almacena es mucho más detallada que los basados en términos de flujos en arcos. A partir de la información en caminos siempre pueden calcularse datos agregados a nivel de arcos, siendo muy difícil el proceso contrario.
- La principal característica de los algoritmos del grupo 4 es que no enumeran los caminos. Pero, la no enumeración es un inconveniente en el ajuste de matrices ya que los datos son almacenados de forma agregada por orígenes o pares O-D. En este sentido los métodos del grupo 3 dan un nivel de detalle mayor. Trabajar con flujos en caminos proporciona una información más detallada en el análisis de la distribución de flujo por la red: es posible conocer los caminos (o pares O-D) que pasan por un cierto arco,

cuántos y cuáles arcos comparten dos caminos (o pares O-D), relaciones entre caminos y entre pares O-D.

- Los algoritmos del grupo 3 tienen un tiempo de computación superior a los del grupo 4 para una asignación que no parte de una solución anterior. Pero, cuando se emplea la opción de Warm Start dichos tiempos se aproximan, siendo una opción válida para los ajustes de matrices donde se modifican ciertos elementos de la matriz de viajes en cada iteración.

Es conocido que para una misma matriz de viajes no existe la unicidad de soluciones en caminos aunque cualquiera de las posibles soluciones, de forma agregada, representan una única y óptima solución en arcos. Si se modifica el orden de elección de los pares O-D para el establecimiento del equilibrio de usuarios, las soluciones en caminos encontradas serán diferentes aunque siempre se obtendrá la misma solución en arcos. La discrepancia entre las soluciones en caminos se debe a diferencias en el flujo de estos caminos pero no a su enumeración. Los caminos enumerados en las diferentes soluciones son siempre los mismos. A partir de la solución óptima en arcos pueden extraerse los caminos que conectan los pares O-D, éstos tendrán igual coste y será el mínimo que conecte dicho par O-D. Al tener una única solución en arcos, la enumeración de caminos también será única. Sin embargo, el flujo que pasa por los caminos puede tomar diferentes configuraciones, siempre guardando relación con los flujos de los caminos con los que comparten arcos.

La no unicidad de soluciones en caminos para una misma distribución de flujos en arcos no representa un inconveniente en los cálculos que realiza el algoritmo implementado para la obtención de las direcciones de búsqueda. Bajo ciertas hipótesis, descritas por Tobin y Friesz (1988), se demuestra que es posible utilizar cualquier solución en términos de flujo en caminos, que se encuentre en equilibrio de usuarios, para el cálculo de derivadas. Estos autores demuestran que al realizar la transformación de esas derivadas a términos de flujo en arcos coinciden con las del problema original.

El algoritmo puede resumirse en los siguientes pasos:

- Paso 0.** *Inicialización.* Con los costes de flujo libre en los arcos se realiza la búsqueda de caminos mínimos desde cada origen a todos los destinos. Una vez encontrado un camino para cada par O-D, se le asigna toda la demanda,  $h^0 = g$ ;  $n = 1$ .

**Paso 1.** *Actualización de costes en arcos a partir de los flujos en caminos*

$$v_a^{n-1} = \sum_{k \in K} \delta_{ak} h_k^{n-1}; s^{n-1} = s(v^{n-1}).$$

**Paso 2.** *Adición de caminos alternativos.* Con los costes en arcos actualizados se realiza la búsqueda de caminos mínimos desde cada origen a todos los destinos. Para cada par O-D, se comparan los caminos encontrados con los existentes y si no coinciden se agregan a la solución actual.

**Paso 3.** *Cálculo de la dirección  $\Delta h_k^n$*  que reparte la demanda de los pares O-D entre sus caminos. El cálculo de esta dirección se describe en los siguientes apartados.

**Paso 4.** *Actualización del flujo en caminos.*

$$h_k^n = h_k^{n-1} + \Delta h_k^n \quad \forall k \in K_p, \forall p \in P$$

**Paso 5.** *Actualización de la Red.* Eliminación de los caminos que han pasado a tener flujo cero por presentar un coste superior al coste mínimo del par O-D.

**Paso 6.** *Criterio de parada.* Si se alcanza el nivel de convergencia buscado, el algoritmo finaliza. En otro caso, el algoritmo establece  $n = n+1$  y continúa por el paso 2.

Para llevar a cabo los pasos descritos, el algoritmo tiene implementado de forma modular una serie de elementos que se detallan a continuación.

### 2.7.1 Almacenamiento de la red

El algoritmo se diseña para su ejecución en Matlab. Éste es un programa muy extendido por la capacidad de manejo de matrices con grandes dimensiones y por la gran cantidad de toolboxes específicas para diferentes áreas científicas que incorpora. Además presenta una gran cantidad de ayuda online por parte de los administradores, a través de su web, y por parte de usuarios particulares, a través de los foros. Matlab aún no incorpora ninguna toolbox especial para trabajar con redes que pueda ser útil en el problema de asignación de tráfico. Únicamente implementa un módulo de presentación de grafos por pantalla que se utilizará para mostrar los resultados.

Las redes de transporte pueden encontrarse en diferentes formatos como son: ESRI shapefile, MapInfo TAB file, EMME, TransCAD, SATURN, etc. Independientemente del formato empleado, todas las redes están compuestas por arcos y nodos. Cada arco consta de un nodo de

inicio y final, y una serie de características que definen su comportamiento funcional. Se ha utilizado una estructura de datos muy común en las aplicaciones informáticas que implementan algoritmos relacionados con la teoría de grafos, la representación *forward star*. Cualquier red puede transformarse en el siguiente conjunto de vectores:

- a) Dos vectores columna  $A$  y  $B$  con dimensión igual al número de nodos de la red. Hay que apuntar que será necesario realizar una nueva enumeración de nodos que comience en 1 para el primer nodo y no permita huecos, obteniendo una numeración continua hasta el último nodo. También es necesario enumerar los arcos comenzando por el nodo 1 y recorriéndolos de forma correlativa, enumerando de forma continua los arcos salientes de cada nodo. Después de esto, se indica en el vector  $A$ , para cada nodo, la enumeración más baja de entre sus arcos salientes, y en el vector  $B$  la enumeración más alta. De esta forma con los vectores  $A$  y  $B$  puede identificarse los arcos salientes de un nodo cualquiera. Al coincidir la numeración de los nodos con el índice de los vectores  $A$  y  $B$ , el algoritmo es muy eficiente obteniendo los *id* de los arcos salientes.
- b) Si el algoritmo se sitúa en un arco cualquiera y necesita conocer hacia qué nodo se dirige, debe consultar el vector columna llamado *lastnode*. Este vector indica el nodo destino hacia el que apunta cada arco. Tiene como dimensión el número de arcos. Si la enumeración de los arcos se inicia a partir del valor 1 y es continua, los índices del vector *lastnode* coincidirán con el *id* de los arcos, en otro caso es necesario emplear un vector intermedio que dirija el algoritmo hacia los arcos.

Esta forma de almacenar la red no permite que existan arcos de doble sentido. Este tipo de arcos se descompone en dos de un único sentido. Para completar la red se añaden tantos vectores columna como sean necesarios para almacenar las características de arcos y nodos. Los vectores más importantes para los arcos son aquellos que describen las funciones de demora:  $t=t(t_0, \alpha, \beta, capacidad)$ . A continuación se representa una pequeña red de ejemplo mediante dos formas distintas: a través de un grafo (Figura 2-2) y mediante los vectores (Tabla 2-4) descritos en este apartado.

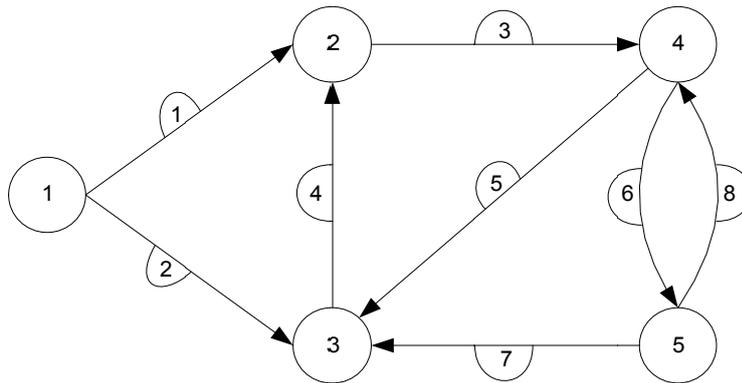


Figura 2-2. Red de ejemplo para mostrar la estructura de datos empleada.

ID Arco	<i>lastnode</i>
1	2
2	3
3	4
4	2
5	3
6	5
7	2
8	4

ID Nodo	<i>A</i>	<i>B</i>
1	1	2
2	3	3
3	4	4
4	5	6
5	7	8

Tabla 2-4. Vectores *A*, *B* y *lastnode*.

*lastnode* indica el nodo al que apunta cada arco. Los vectores *A* y *B* almacenan los arcos que salen de cada nodo. Éstos indican, respectivamente, los índices inicial y final de los arcos salientes de cada nodo, todos los valores enteros comprendidos entre estos dos números también corresponderán a IDs de arcos salientes del nodo.

### 2.7.2 Almacenamiento y tratamiento de la solución

Para almacenar la solución es necesario conocer todos los caminos activos además de su flujo y coste. Estos últimos son valores asociados a cada camino activo y se almacenan en dos vectores columnas. Cada camino es identificado por los arcos que contiene. Esta información se almacena en la matriz de incidencia arcos – caminos ( $\Lambda$ ) con tamaño igual a  $|A| \times |K|$ . Es una matriz dispersa de unos y ceros, el elemento  $ij$  tiene valor 1 si el arco  $i$  pertenece al camino  $j$ .

En las diferentes iteraciones del problema el número de caminos no es constante. El algoritmo encuentra nuevos caminos con menor coste y elimina caminos con flujo cero y coste alto. De

esta forma la matriz de caminos se ve sometida a un constante ordenamiento y redimensionamiento. Estos procesos tienen un gran coste computacional debido a las grandes cantidades de datos con las que se trabaja. Para evitar este sobre coste computacional se emplea una matriz estática de grandes dimensiones que trabaja como una lista enlazada. Para recorrer la matriz de incidencia arcos – caminos es necesario recorrer la “matriz lista” que indica, para cada par O-D, el lugar donde se encuentran sus caminos. Además, es necesaria la creación de un vector que indique los espacios libres en la matriz de incidencia arcos – caminos por la eliminación de los caminos no activos. En estos espacios se insertan los nuevos caminos.

La “matriz lista” se compone de tantas filas como pares O-D tiene la red. Para cada par O-D, esta matriz dirige el algoritmo hacia las filas de la matriz de incidencia arcos – caminos donde se encuentran sus caminos. Los caminos se van insertando en dicha matriz por orden de aparición. No es necesario ordenarlos por pares O-D gracias a la matriz lista. Cuando se elimina algún camino se borran los valores de la fila de la matriz arcos – caminos y se indica al vector de filas vacías la posición donde puede insertarse un nuevo camino.

### 2.7.3 Gestión de caminos

Este bloque de código busca, para cada par O-D, caminos mínimos alternativos a los actualmente activos. Las entradas de este bloque son: la red, los costes en arcos y los caminos actualmente activos. Los procesos que tiene implementados son el método de Dijkstra para la búsqueda de caminos mínimos y un código de comparación de caminos para evitar repeticiones. Las salidas son, si existen, los caminos alternativos con menor o igual coste que los activos.

### 2.7.4 Cálculo de la dirección $d^n$

La dirección  $d^n$  reparte el flujo entre los caminos de cada par O-D con el fin de alcanzar una solución en equilibrio de usuario. La función de esta dirección es la de equilibrar los costes de los caminos de cada par O-D, para ello reduce el flujo de los caminos más costosos y aumenta el flujo de los menos costosos.

El cálculo de la dirección implementado en el algoritmo es resultado del estudio realizado en el CAPÍTULO 3 de estimación de matrices. Dicho estudio analiza el comportamiento de los usuarios de la red cuando aumentan o disminuyen los viajes en un par O-D concreto. En él se comprueba que existen ciertas influencias entre los diferentes pares O-D. La variación de viajes en un par O-D cualquiera hace que se reconfigure, además del flujo de sus caminos, el flujo de los demás caminos de la red. Estas influencias se deben a la saturación en los arcos que hace

que se establezcan relaciones entre los diferentes caminos que los atraviesan y, por tanto, entre los diferentes pares O-D.

En los problemas de asignación de tráfico, la saturación en los arcos hace complejo el cálculo del trasvase de flujo entre los caminos de un mismo par O-D con el fin de igualar sus costes. Por este motivo todos los algoritmos de asignación de tráfico son iterativos y calculan en cada iteración una nueva dirección de forma rápida y aproximada. Con esta dirección y aplicando un método de búsqueda lineal minimizan la Transformada de Beckmann. A continuación se describen las direcciones de búsqueda que emplean los principales algoritmos de asignación.

#### 2.7.4.1 Direcciones de búsqueda de los principales algoritmos de asignación

Jaykrishnan et al. (1994) realizan el cálculo de la dirección a través de un algoritmo basado en el método de proyección del gradiente de Goldstein-Levitin-Polyak formulado por Bertsekas (1976). En este algoritmo no se realiza una búsqueda lineal, sino que los flujos de los caminos se obtienen seleccionando un valor de  $\alpha(n)$  en las ecuaciones (2.22) y (2.23):

$$f_k^p(n+1) = \left[ f_k^p(n) - \frac{\alpha(n)}{s_k^p(n)} (d_k^p(n) - d_{\bar{k}_p(n)}^p) \right]^+ \quad \forall \begin{cases} k \in K_p, k \neq \bar{k}_p \\ p \in P \end{cases} \quad (2.22)$$

$$f_{\bar{k}_p(n)}^p(n+1) = q_p - \sum_{k \in K_p, k \neq \bar{k}_p(n)} f_k^p(n+1) \quad \forall p \in P \quad (2.23)$$

donde  $n$  es el número de iteración,  $\alpha(n)$  el tamaño del paso,  $s_k^p(n)$  es una escala diagonal definida positiva basada en la segunda derivada del Hessiano,  $d_k^p(n)$  y  $d_{\bar{k}_p(n)}^p$  son las primeras derivadas de la función objetivo a lo largo del camino  $k$  y el camino mínimo  $\bar{k}_p$ , respectivamente, y  $[ ]^+$  representa la proyección ortogonal positiva.

Florian et al. (2009) desarrolla un algoritmo basado en la descomposición secuencial por pares O-D. Cada subproblema es resuelto mediante una adaptación del método de proyección del gradiente de Rosen (Rosen, 1960). Tiene la ventaja de que todos los caminos con flujos son ajustados simultáneamente. El proceso que sigue el algoritmo puede resumirse como un movimiento de flujo desde los caminos con coste superior a la media hasta los caminos con coste inferior a la media. La expresión de cada componente de la dirección es la siguiente:

$$d_k = (s_k - \bar{s}) \quad k \in K_i^+ \quad (2.24)$$

donde  $K_i^+$  son los caminos del par O-D  $i$  que tienen flujo,  $\bar{s}$  es el coste medio de todos los caminos y  $s_k$  es el coste del camino  $k$ .

En este algoritmo es necesario realizar la siguiente búsqueda lineal para encontrar el tamaño óptimo del paso:

$$\min \sum_{a \in A} \int_0^{v_a + \bar{v}_a + \lambda y_a^i} s_a(x) dx \quad 0 \leq \lambda \leq \min \left( \frac{-h_k}{d_k} \mid d_k < 0 \right) \quad (2.25)$$

donde  $y_a^i = \sum_{k \in K_i^+} \delta_{ak} h_k$  y  $\bar{v}_a = \sum_{(i' \neq i) \in I} \sum_{k \in K_{i'}^+} \delta_{ak} h_k$  es el flujo constante de los demás pares O-D de la red.

La actualización del flujo de cada camino es la que sigue:

$$h_k = h_k + \lambda^* d_k \quad k \in K_i^+, \quad v_a^i = v_a^i + \lambda^* y_a^i, \quad a \in A \quad (2.26)$$

Bar-Gera (2002) presenta un algoritmo similar al propuesto por Bertsekas y Gallager (1992) para el enrutamiento de las redes de telecomunicación. Emplea un método de dirección de búsqueda quasi-Newtoniano para la actualización del flujo de los arcos basado en orígenes. Sugiere que la cantidad de flujo a trasvasar entre dos rutas corresponderá a la diferencia entre el coste de dichas rutas dividido por la suma de la derivada de sus costes. El numerador se corresponde con la primera derivada de la función objetivo y el denominador con la segunda derivada aunque, en este caso, se usa una aproximación.

$$\frac{\partial T / \partial \alpha_a^{NB}}{\partial^2 T / \partial \alpha_a^{NB^2}} \approx \frac{1}{g_j} \cdot \frac{\mu_a - \mu_b}{v_a + v_b - 2 \cdot \rho_{lcn_j}} \quad (2.27)$$

donde  $a$  y  $b$  son los segmentos entre los que se trasvasa flujo para equilibrar sus costes,  $\mu_a$  y  $\mu_b$  el coste de cada segmento,  $j$  el nodo donde se encuentran los segmentos  $a$  y  $b$ ,  $g_j$  el flujo total que pasa por el nodo  $j$  y  $v_a, v_b, \rho_{lcn_j}$  son parámetros que dan una aproximación razonable de la segunda derivada de la función objetivo.

El flujo traspasado desde  $a$  hacia  $b$  es:

$$z_{a \rightarrow b}(\alpha, t, t') = \frac{\mu_a(\alpha, t) - \mu_b(\alpha, t)}{\max(\varepsilon_v, \nu_a(\alpha, t') + \nu_b(\alpha, t') - 2 \cdot \rho_{lcn_j}(\alpha, t'))} \quad (2.28)$$

donde  $\varepsilon_v$  es una constante positiva de pequeño valor.

Dial (2006) emplea en su algoritmo el método de Newton (Bertsekas, 1976) para trasvasar el flujo desde los caminos más costosos hacia los menos costosos. Para dos caminos, o segmentos de caminos  $\bar{p}$  y  $\underline{p}$ , en los cuales quiere equilibrarse su coste, se procede calculando la cantidad de flujo trasvasada  $\Delta x_o$  con las expresiones (2.29) y (2.30).

- Linealización de los costes de los caminos

$$\bar{c}(z - \Delta x_o) - \underline{c}(z + \Delta x_o) \approx \bar{c}(z) - \bar{c}'(z) \Delta x_o - \underline{c}(z) - \underline{c}'(z) \Delta x_o \quad (2.29)$$

- Haciendo tender la parte derecha de la ecuación a cero:

$$\Delta x_o = \frac{\bar{c}(z) - \underline{c}(z)}{\bar{c}'(z) + \underline{c}'(z)} \quad (2.30)$$

donde  $\bar{c}$ ,  $\underline{c}$  son los costes, y  $\bar{c}'$ ,  $\underline{c}'$  los costes marginales (primera derivada de la función de costes respecto de su flujo) de los caminos  $\bar{p}$ ,  $\underline{p}$ , respectivamente.

En este algoritmo, si las derivadas presentan un mal comportamiento, se emplea el método de la sección dorada.

Gentile (2012) calcula la dirección de descenso basándose en la linealización de los costes en arcos al igual que Bar-Gera (2002) y Dial (2006) aunque sin emplear el método de Newton. El cálculo de la dirección se describe de forma resumida:

- Se emplea una formulación basada en nodos. Se definen los nuevos flujos que salen del nodo  $i$  al nodo  $j$  y se dirigen hasta el destino  $d$  como  $e_{ij}^d$  y los de la solución anterior como  $f_{ij}^d$ . Debe cumplirse que la suma de los flujos traspasados  $(e_{ij}^d - f_{ij}^d)$  entre las diferentes alternativas para alcanzar  $d$  es nula. Además se define  $x_j = e_{ij}^d / e_i^d$ , de igual forma, la suma de estas proporciones deben sumar 1.
- Se define la variable  $v = V_i^d$  como el coste en equilibrio local para alcanzar el destino  $d$ .

- De las ecuaciones que definen el equilibrio de tráfico, debe satisfacerse una de las siguientes ecuaciones: *i)*  $x_j = 0$ , y por tanto la solución no se mueve, *ii)*  $a_j + b_j \cdot x_j = v$ , donde  $a_j$  y  $b_j$  se describen en Gentile (2012).
- Así se despeja  $v$  como sigue:

$$\sum_{j \in H} \frac{(v - a_j)}{b_j} = 1 \tag{2.31}$$

$$v = \frac{1 + \sum_{j \in H} \frac{a_j}{b_j}}{\sum_{j \in H} \frac{1}{b_j}} \tag{2.32}$$

- Conociendo el valor de  $v$ , se calculan los nuevos flujos  $e_{ij}^d$ .

2.7.4.2 Crítica de las direcciones de búsqueda descritas

El cálculo de la dirección de búsqueda que se implementa en los algoritmos analizados se centra en encontrar la intersección de las tangentes de las funciones de demoras de los caminos. A continuación se muestra que aunque tiene una solución rápida y eficiente, el cálculo de la dirección de búsqueda puede ser mejorado.

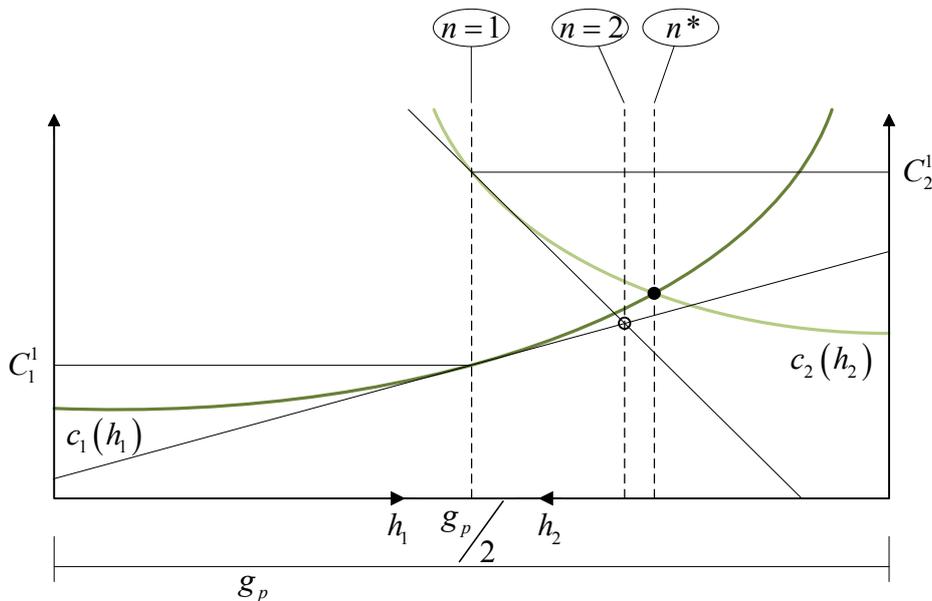


Figura 2-3. Iteraciones para el equilibrio de los caminos del mismo par.

En la Figura 2-3 se muestra un ejemplo de dos caminos ( $c_1, c_2$ ) que pertenecen a un mismo par  $p$ . Inicialmente, iteración  $n = 1$ , los caminos se reparten la demanda de viajes al 50%

(  $h_1 = h_2 = \frac{g_p}{2}$  ) resultando el coste del camino  $c_2$  superior al coste del camino  $c_1$  ( $C_2^1 > C_1^1$ ). Si en esta iteración se trazan las tangentes a las funciones de demora de los caminos en el punto actual, su intersección indica los nuevos valores de flujo y coste. Se observa en la iteración  $n = 2$  que los costes se acercan al equilibrio.

Estos métodos alcanzan rápidamente el equilibrio de usuario para los caminos de un mismo par O-D cuando las funciones de demora de dichos caminos dependen únicamente de su propio flujo. En los problemas de transporte reales, la función de coste de cada camino generalmente depende del flujo de los todos los caminos de la red repercutiendo en la efectividad de las direcciones de búsqueda descritas.

Las funciones de demora de los caminos son la resultante de la suma de las funciones de coste de los arcos que los forman (ecuación (2.7)).

$$C_i = \sum_a \delta_{ia} \cdot s_a(v_a) \quad \forall a \in A$$

El coste de un arco depende generalmente de su propio flujo y está formado por la suma de los flujos de los caminos que pasan por dicho arco.

$$C_i = \sum_a \delta_{ia} \cdot s_a \left( \sum_k \delta_{ka} \cdot h_k \right) \quad \forall a \in A, k \in K$$

Por tanto, cuando se modifica el flujo de un camino cualquiera, éste repercute directamente en el coste de todos los caminos que comparten arco con él, modificando su curva de demora.

A continuación se muestra un ejemplo con dos pares O-D  $\alpha$  y  $\beta$  en los que dos de sus caminos comparten arcos. Para el par O-D  $\alpha$  se reutiliza la Figura 2-3 en la que se observa que aún no se encuentra en equilibrio y que se compone de los caminos  $c_1$  y  $c_2$ . Utilizando la dirección de búsqueda de cualquiera de los métodos anteriores se aprecia que en la siguiente iteración el camino  $c_1$  aumenta su flujo (y coste) y el  $c_2$  disminuye su flujo (y coste).

El par O-D  $\beta$  compuesto por los caminos  $c_3$  y  $c_4$  (Figura 2-4) se encuentra inicialmente en equilibrio ( $C_3^1 = C_4^1$ ).

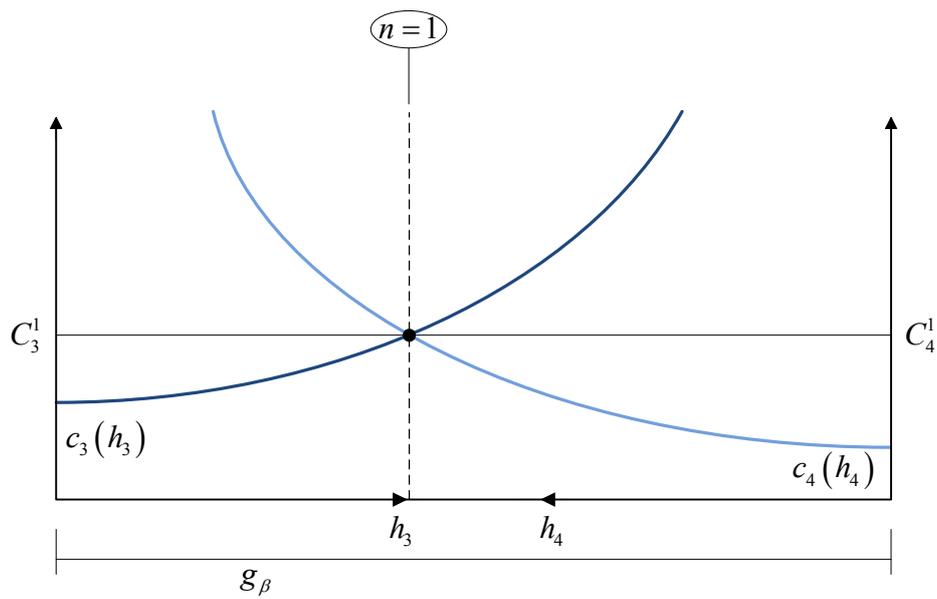


Figura 2-4. Par O-D  $\beta$  en equilibrio.

Los caminos que comparten arcos son  $c_2$  y  $c_3$ . Al tratar de alcanzar el equilibrio en el par O-D  $\alpha$ , el flujo y coste de  $c_2$  ha disminuido. Esto hace que el coste de  $c_3$  disminuya y se abandone el equilibrio en el par O-D  $\beta$  (Figura 2-5).

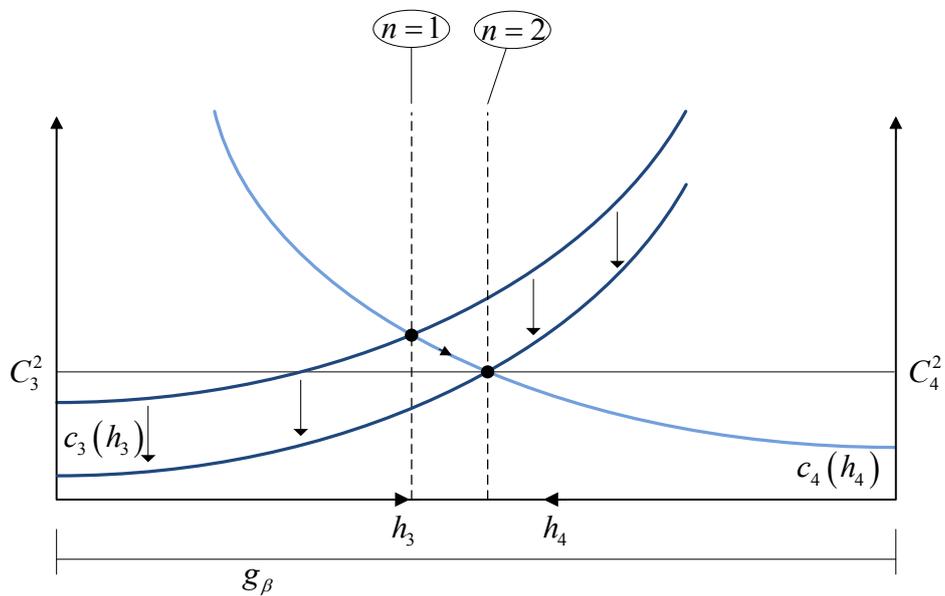


Figura 2-5. Par O-D  $\beta$ , nueva situación en equilibrio.

Al encontrar un desequilibrio en el par O-D  $\beta$  el algoritmo pasa a equilibrarlo modificando el coste de los caminos del par  $(c_3, c_4)$  y, consecuentemente, modificando el coste de los caminos que comparten arcos con dichos caminos ( $c_2$  que pertenece al par O-D  $\alpha$ ). Estas influencias entre caminos que comparten arcos podrían denominarse efectos indirectos entre pares O-D. A

medida que el algoritmo va iterando, estos efectos se atenúan hasta llegar a un estado de equilibrio de usuario en todos los pares O-D. Para una red compleja, es decir de gran escala y con congestión, los efectos indirectos hacen que sea necesaria una gran cantidad de iteraciones para alcanzar el equilibrio. En esta tesis se implementa un módulo de cálculo de la dirección de búsqueda que a través de una aproximación lineal de los costes de los caminos, tiene en cuenta las dependencias o influencias entre los caminos del mismo par.

#### 2.7.4.3 Cálculo de la dirección de búsqueda propuesto

El cálculo de la dirección de búsqueda implementado en el algoritmo es consecuencia del estudio de ajuste de matrices. En dicho estudio se describen con más detalle las influencias entre caminos (o pares O-D). A continuación se introducen ciertas variables que serán de ayuda para la descripción del módulo de cálculo.

Realizando una linealización de los costes de los caminos se obtiene:

$$C = C^\phi + \sum_k \frac{\partial C}{\partial h_k} (h_k - h_k^\phi) + O(2) \quad (2.33)$$

Escrito de forma matricial:

$$C(h) = A + B \cdot \Delta h \quad (2.34)$$

donde  $A$  son los costes fijos y  $B$  representa la relación lineal entre el coste y el flujo de los caminos en la solución actual.

Los métodos analizados anteriormente (apartado 2.7.4.1) aproximan la matriz  $B$  a una matriz diagonal para simplificar los cálculos. Así el incremento de flujo de un camino afecta únicamente al coste del mismo. A través de los resultados obtenidos en estos métodos, se comprueba que es una aproximación válida que genera soluciones convergentes aunque tiene margen de mejora.

En caso de no realizar dicha simplificación a una matriz diagonal, el coste de un camino puede representarse como función del flujo de todos los caminos de la red. De esta forma el algoritmo tendría en cuenta las influencias mencionadas, obteniendo una convergencia más rápida en cada iteración. Aun así el algoritmo debe ser iterativo ya que las funciones de demora son monótonamente crecientes y la matriz  $B$  es calculada mediante una aproximación lineal de dichas funciones en la solución actual.

$$B_{kl} = \frac{\partial C_k}{\partial h_l} = \sum_a \delta_{ak} \cdot \delta_{al} \cdot \frac{\partial s_a}{\partial v_a} \quad (2.35)$$

En (2.35) se muestra la expresión de los elementos de la matriz  $B$ . Se observa que la variación del coste del camino  $k$  respecto al flujo del camino  $l$  está formada por la suma de la derivada de la función de demora de los arcos comunes a dichos caminos respecto a su flujo.

La matriz  $B$  es cuadrada de tamaño igual al número de caminos. Generalmente no es posible calcular su inversa ya que, en los problemas de transporte, el número de caminos suele ser mayor al de arcos existiendo relaciones de dependencia entre caminos. Esta característica hace que sea complejo su tratamiento para la generación de expresiones que relacionen los costes de todos los caminos con sus flujos. El uso de esta matriz por parte de Tobin y Friesz (1988) en su trabajo de análisis de sensibilidad para redes con flujos en equilibrio ha generado muchas críticas (Bell y Iida, 1997; Patriksson, 2004; Lundgren y Peterson, 2008). En esta tesis se resuelve una expresión que evita el cálculo de la inversa de la matriz  $B$  a través de un algoritmo iterativo.

A continuación, se desarrolla la formulación asociada al cálculo de la dirección  $d^n$  que se implementa. Para cada par O-D  $p$ , se resuelve el siguiente sistema de ecuaciones.

$$\sum_k \Delta h_k = 0 \quad \forall k \in K_p \quad (2.36)$$

$$C_k^n = C_k^{n-1} + \Delta C_k = U_p^n \quad \forall k \in K_p \quad (2.37)$$

Con la expresión (2.36) se traspa flujo entre los caminos del par  $p$  sin modificar el número de viajes recogido en la matriz de demanda. Las variables del problema serán los incrementos de flujo de los caminos del par  $p$  junto con el coste mínimo del par  $U_p^n$ , conservándose constantes el flujo de los caminos de los restantes pares O-D.

Los parámetros  $C_k^{n-1}$  representan el valor del coste de los caminos del par  $p$  en la iteración anterior y a priori ni coinciden entre ellos ni son mínimos. Estos valores se consideran constantes en la iteración actual y para equilibrarlos se aplica un incremento de coste  $\Delta C_k$ . Al traspasar flujo entre los diferentes caminos del par  $\Delta h_k$  se modificará el valor de  $\Delta C_k$ , así los costes de los caminos que conectan el par O-D  $p$  en la iteración actual  $C_k^n$  estarán equilibrados y coincidirán con el coste mínimo del par  $U_p^n$ .

Los incrementos de costes  $\Delta C_k$  se pueden descomponer en dos términos, uno que depende del flujo del propio par y otro que depende de los flujos de los demás pares:

$$\Delta C_k = \sum_i B_{ki} \cdot \Delta h_i + \sum_j B_{kj} \cdot \Delta h_j \quad \forall i, k \in K_p, j \in K, j \notin K_p \quad (2.38)$$

En la expresión (2.38) se calcula  $\Delta C_k = B_k \cdot \Delta h$ , donde  $B_k$  es la fila de la matriz  $B$  correspondiente al camino  $k$ . En el vector  $\Delta h$  se distingue entre los caminos que pertenecen al par O-D  $p$ , las variables del problema, y los caminos que pertenecen al resto de pares O-D, que se mantienen constantes. A través de la matriz  $B$ : *i*) el término  $\Delta h_i$  tiene en cuenta el incremento de coste en el camino  $k$  debido al aumento de flujo del propio camino y el incremento debido a los demás caminos del par que comparten arcos con él, *ii*) y el término  $\Delta h_j$  tiene en cuenta la influencia que realizan sobre el coste del camino  $k$  los demás pares O-D.

Para el cálculo de la dirección de búsqueda  $\Delta h$  de cada par O-D, se empleará el método que se describe en el apartado 3.8.4.2.2, donde se construye y resuelve un sistema con las ecuaciones (2.36) y (2.37). El algoritmo de asignación finaliza cuando, después de realizar varias pasadas sobre todos los pares O-D con más de un camino activo, se compruebe que los costes han sido equilibrados. Para evitar la aparición de flujos negativos que llevan a soluciones no factibles, los valores negativos de  $\Delta h_k$  no pueden superar el valor de  $h_k^{n-1}$ .

Como se indica en el esquema del algoritmo descrito en la introducción del apartado 2.7, en el paso 4 del algoritmo los flujos de los caminos se actualizan a través de la siguiente expresión.

$$h_k^n = h_k^{n-1} + \Delta h_k \quad \forall k \in K_p \quad (2.39)$$

Esta formulación presenta una serie de características que la diferencia de las demás y que le aportan una mayor eficiencia para equilibrar los costes y alcanzar las condiciones de equilibrio de usuario de Wardrop:

- En cada iteración se equilibran a la vez todos los caminos de un mismo par O-D. No todos los métodos tienen esta propiedad, existen métodos que equilibran caminos o segmentos dos a dos.
- Tiene en cuenta las influencias del flujo del resto de caminos que no pertenecen al par O-D. Los algoritmos que existen en la bibliografía técnica no tienen en cuenta estas influencias para calcular la dirección. Generalmente, estas influencias las consideran una vez realizan la búsqueda lineal y actualizan los costos en arcos y caminos. Esta forma de actuar tiene un mayor coste computacional.

- Tiene en cuenta las influencias del flujo entre caminos del mismo par O-D. Cuando el número de caminos que conectan un par O-D es tres o superior, éstos pueden compartir conjuntos de arcos diferentes dos a dos. Debido a esto el aumento del flujo de un camino afecta de forma diferente sobre cada camino del par. Estas influencias no son posibles computarlas si únicamente se consideran los costes marginales de cada camino (elementos de la diagonal de  $B$ ).
- La dirección  $d^n = \Delta h^n$  no necesita la implementación de una búsqueda lineal posterior que minimice el valor de la función objetivo, en este caso el valor de la transformada de Beckmann. El método implementado incluye las condiciones de equilibrio de Wardrop al incluir en las expresiones el coste mínimo de los caminos del par  $U_p^n$ .

### 2.7.5 Criterios de convergencia

Al implementar algoritmos iterativos para la solución del problema de asignación de tráfico se requiere un criterio de parada. Existen varias formas de detectar cuando un algoritmo ha alcanzado una cierta convergencia. A continuación se detallan las más empleadas (Sheffi, 1985; Rose et al., 1988):

- Similitud de los sucesivos costes de viaje en los pares O-D

$$\sum_p \frac{|u_p^n - u_p^{n-1}|}{u_p^n} \tag{2.40}$$

donde  $u_p^n$  es el mínimo coste de viaje del par O-D  $p$  en la iteración  $n$ .

- Cambio en los flujos en arcos

$$\frac{\sqrt{\sum_{a \in A} (x_a^{n+1} - x_a^n)^2}}{\sum_{a \in A} x_a^n} \tag{2.41}$$

- Máximo cambio de flujo en arcos

$$\max_{a \in A} \left( \frac{|x_a^n - x_a^{n-1}|}{x_a^n} \right) \tag{2.42}$$

- Cambio en la función objetivo

$$\left( \frac{z^n - z^{n-1}}{z^n} \right) \cdot 100\% \quad (2.43)$$

- *Relative gap*

$$\frac{\sum_{p \in P} \sum_{k \in K_p} h_k C_k - \sum_{p \in P} g_p U_p}{\sum_{p \in P} \sum_{k \in K_p} h_k C_k} \quad (2.44)$$

donde  $\sum_{p \in P} \sum_{k \in K_p} h_k C_k$  es el coste total del sistema con la configuración actual de flujos en caminos y  $\sum_{p \in P} g_p U_p$  es el coste total del sistema si la demanda de todos los pares O-D se distribuye por los caminos de menor coste.

- Excedente de coste promedio

$$\frac{\sum_{p \in P} \sum_{k \in K_p} h_k C_k - \sum_{p \in P} g_p U_p}{\sum_{p \in P} g_p} \quad (2.45)$$

En este caso se implementa como criterio de parada el *relative gap* (2.44). Este método de medida de convergencia tiene una gran sensibilidad y es apto para presentar la evolución de la solución cuando los algoritmos alcanzan altas convergencias como se indica en Slavin et al. (2006). Además, tiene sentido físico ya que compara la solución actual donde aún los caminos no están en equilibrio, con una solución donde toda la demanda se distribuye por el camino de menor coste.

## 2.8 RESULTADOS/APLICACIONES

Una vez descrita toda la teoría relevante sobre el problema de asignación de tráfico, en cuanto a su formulación e implementación algorítmica, se muestran los resultados obtenidos con los métodos implementados. Para ello se emplean varias redes de transporte que se detallan a continuación.

## 2.8.1 REDES DE APLICACIÓN

En este apartado se realiza una breve descripción de las redes de transporte empleadas en las diferentes investigaciones que componen este documento de tesis. Todas las redes descritas son casos reales ampliamente estudiados. La principal diferencia entre cada una de ellas es su tamaño; se ha empleado una red de pequeña escala, una de escala media y otra de gran escala.

### 2.8.1.1 Red de Tiergarten

Es una red de pequeña escala, pertenece a una zona del centro de Berlín llamada Tiergarten. La información topológica para el diseño de la red (Figura 2-6) se ha obtenido de la web de Bar-Gera (2012); sus características se encuentran recogidas en la Tabla 2-5.

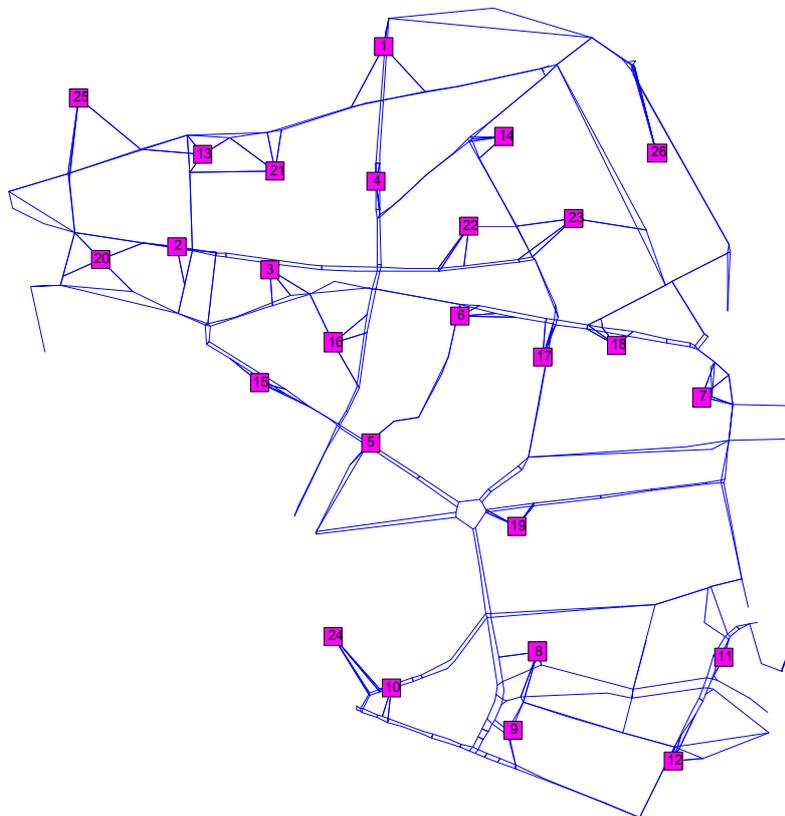


Figura 2-6. Modelo de red de Tiergarten.

<b>Arcos</b>	747
<b>Arcos aforados</b>	120
<b>Porcentaje Arcos aforados</b>	16%
<b>Nodos</b>	344
<b>Centroides</b>	26
<b>Porcentaje Centroides</b>	7,6%
<b>Pares O-D</b>	650
<b>Pares O-D con flujo no nulo</b>	650
<b>Demanda</b>	26.658,54
<b>Ratio Arcos aforados – Pares O-D</b>	0,185

**Tabla 2-5. Características red de Tiergarten.**

El porcentaje de arcos aforados respecto del total de arcos de la red es del 16%. Este valor en muchos proyectos puede ser un límite inalcanzable aunque para el problema de ajuste es un porcentaje muy insuficiente.

El porcentaje de centroides respecto del total de nodos da una idea del tamaño de las zonas y de lo expandidas que están sobre la red. Cuanto mayor sea este valor, menor dimensión tienen las zonas que representan los centroides y más cercanas están dentro de la red. En este caso el ratio de centroides es bajo, además puede apreciarse en la Figura 2-6 que están bien distribuidos por toda la red.

Un dato importante en el ajuste de matrices es el grado de indeterminación. En este problema cuanto menor sea el ratio arcos aforados – pares O-D mayor será la indeterminación. Por un lado los arcos aforados representan las restricciones del problema y por otro, los pares O-D las variables. Lo ideal es que este valor esté próximo a 1 aunque en la práctica, en redes de tamaño medio o superiores, no alcanzan el 0,1.

#### 2.8.1.2 Red de Chicago Sketch

Es una red de tamaño medio, es una simplificación de la red de la ciudad de Chicago en la que solo se recogen las calles principales (Figura 2-7). La información topológica para el diseño de la red también se ha obtenido de la web de Bar-Gera (2012), sus características se muestran en la Tabla 2-6.

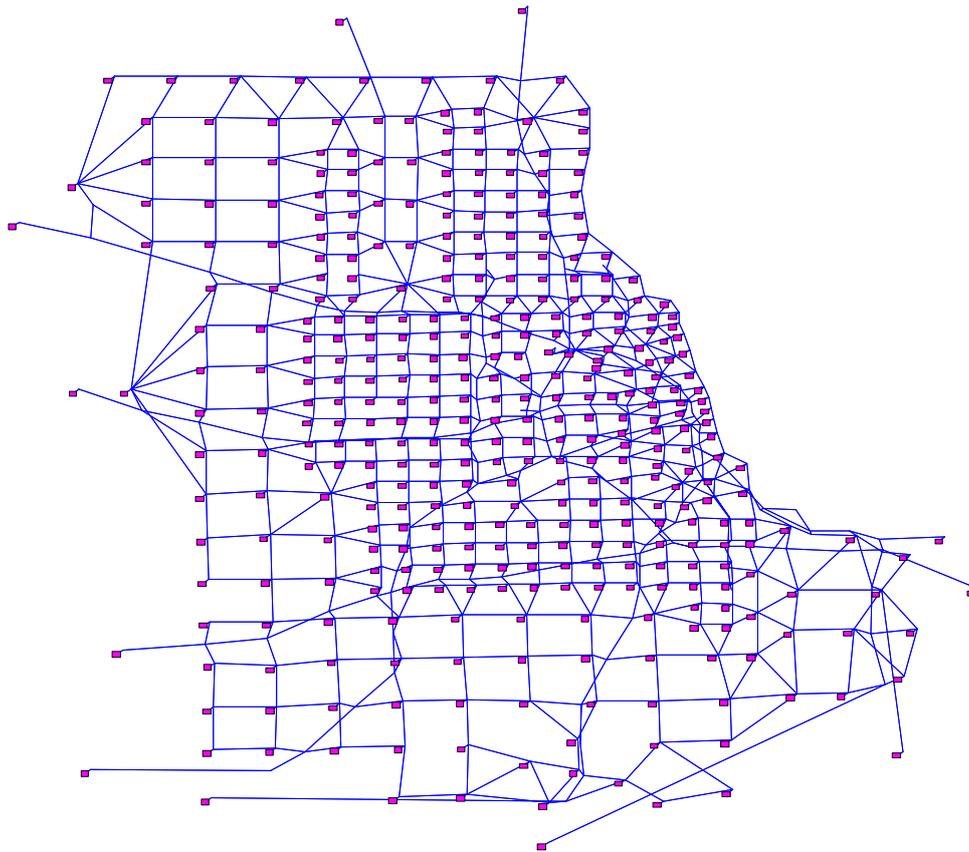


Figura 2-7. Modelo de red de Chicago Sketch.

<b>Arcos</b>	2.950
<b>Arcos aforados</b>	500
<b>Porcentaje Arcos aforados</b>	17%
<b>Nodos</b>	933
<b>Centroides</b>	387
<b>Porcentaje Centroides</b>	41,5%
<b>Pares O-D</b>	149.382
<b>Pares O-D con flujo no nulo</b>	93.135
<b>Demanda</b>	1.137.493,44
<b>Ratio Arcos aforados – Pares O-D</b>	0,00335

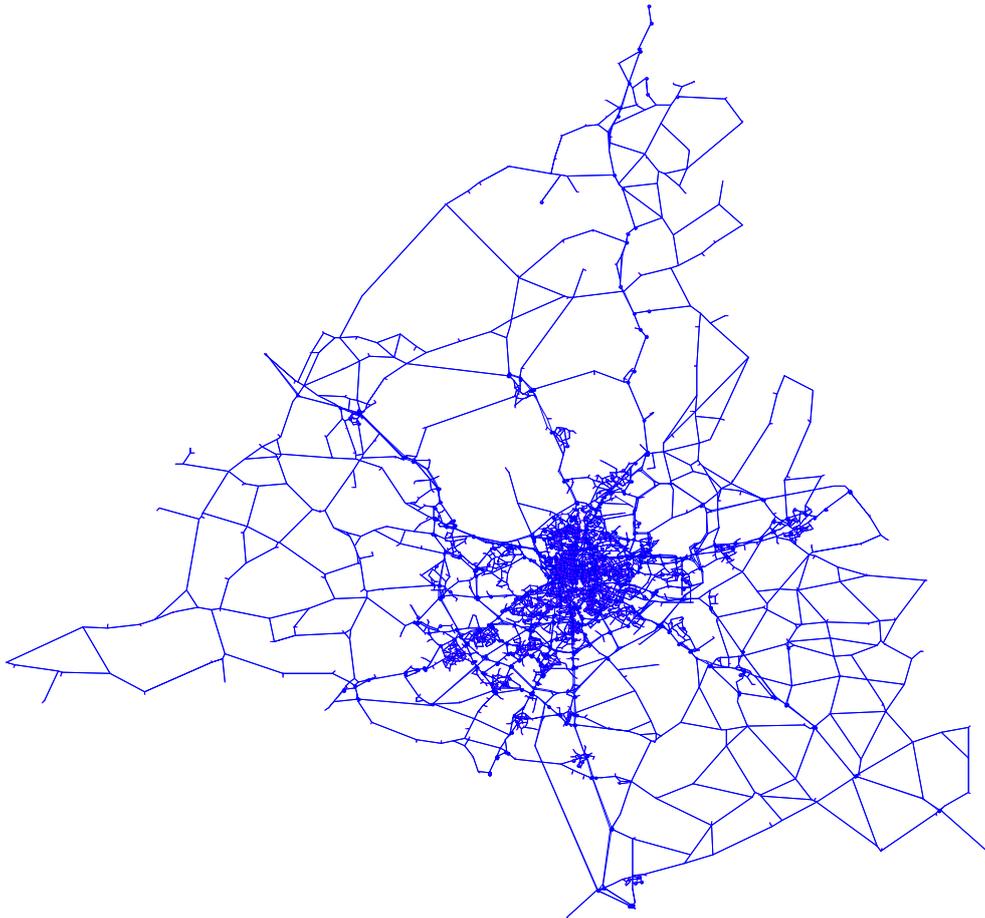
Tabla 2-6. Características red de Chicago Sketch.

El porcentaje de arcos aforados es semejante al empleado en la red de Tiergarten. El principal inconveniente de esta red es la cantidad de nodos centroides, representan el 41,5% del total. En la Figura 2-7 se observa que están uniformemente distribuidos por la red pero el tamaño de cada zona es muy pequeño. La conexión entre las zonas limítrofes es a través de un único arco. Para

llevar a cabo un proceso de estimación de viajes en cualquier red es importante interceptar flujo de cada pareja O-D, en este caso no existe otra opción que aforar todos los arcos. A su vez, la indeterminación de este problema es muy elevada como muestra el ratio arcos aforados – pares O-D.

### 2.8.1.3 Red de Madrid

Es una red de gran escala, incluye la mayor parte de carreteras de la Comunidad Autónoma de Madrid y del callejero de la ciudad de Madrid. Las características de esta red (Figura 2-8) se muestran en la Tabla 2-7.



**Figura 2-8. Modelo de red de Madrid (sin centroides).**

<b>Arcos</b>	17.544
<b>Arcos aforados</b>	491
<b>Porcentaje Arcos aforados</b>	3%
<b>Nodos</b>	7.966
<b>Centroides</b>	1.180
<b>Porcentaje Centroides</b>	14,8%
<b>Pares O-D</b>	1.391.220
<b>Pares O-D con flujo no nulo</b>	304.678
<b>Demanda</b>	451.564,24
<b>Ratio Arcos aforados – Pares O-D</b>	0,000353

**Tabla 2-7. Características red de Madrid.**

Analizando los ratios que se muestran en la Tabla 2-7 puede concluirse que el número de arcos aforados en esta red es insuficiente y el número de centroides es elevado, lo que lleva a una alta indeterminación en el problema de estimación de matrices.

## 2.8.2 RESULTADOS

En esta sección se analizan numéricamente los algoritmos implementados para el desarrollo de esta tesis. En total se han realizado tres implementaciones en el software Matlab: el algoritmo basado en caminos propuesto en el apartado 2.7 y dos algoritmos basados en Dial (2006) y Barger (2010) (apartados 2.5.3 y 2.5.4).

A su vez se ejecutan varios algoritmos incorporados en el programa TransCAD 5.0. Este programa está diseñado para abordar todos los modelos necesarios en planificación de redes de transporte y ésta siendo utilizado por numerosas consultorías a nivel mundial. Los algoritmos que se ejecutarán serán el de Frank & Wolfe (apartado 2.5.1), Orígenes UE (apartado 2.5.3) y path based.

El algoritmo de orígenes UE en TransCAD 5.0 coincide a grandes rasgos con el algoritmo implementado en Matlab basado en Dial (2006). La comparación de los algoritmos diseñados con código Matlab y los implementados en TransCAD es sesgada ya que este último está exclusivamente diseñado para este fin y Matlab es un programa de propósito general que abarca muchas materias. Aun así, la comparación da una idea de la eficiencia de los algoritmos generados mediante el código Matlab.

A continuación se muestran los resultados obtenidos en los diferentes algoritmos de asignación. Todos son obtenidos empleando la red de Chicago Sketch ya que supone un esfuerzo computacional superior al empleado en la red de Tiergarten y pueden llevarse a cabo comparaciones tanto en convergencias como en tiempos de ejecución.

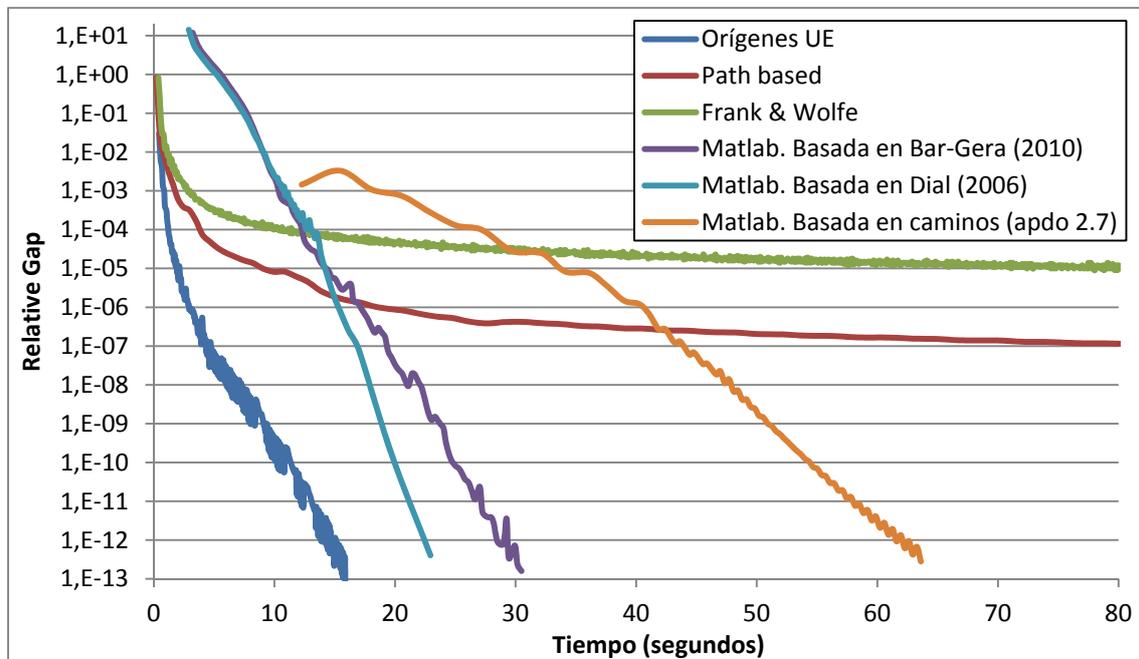


Figura 2-9. Convergencia de los algoritmos de asignación.

Se observa que el inicio de los algoritmos de TransCAD está muy optimizado, realizan las primeras iteraciones de forma casi instantánea. Los modelos en Matlab necesitan de un tiempo superior para comenzar a realizar iteraciones.

Pueden comprobarse en la Figura 2-9 las limitaciones e inconvenientes descritos en los apartados 2.5.1 y 2.6 para el método de Frank & Wolfe. Estos inconvenientes impiden alcanzar valores de *relative gap* inferiores a  $10^{-5}$ .

Sorprenden los resultados aportados por el método *path based* implementado en TransCAD. Este método se estanca en un *relative gap* de  $10^{-8}$  a los 9 minutos de ejecución. Estos métodos son computacionalmente más costosos debido al coste de almacenamiento aunque gracias a la enumeración de caminos pueden alcanzar soluciones convergentes.

De las tres implementaciones diseñadas en Matlab, el algoritmo basado en caminos emplea mayores tiempos de ejecución, consume el doble de tiempo en superar el *relative gap* de  $10^{-12}$  respecto de las otras dos implementaciones. El coste de inicialización es excesivo, se debe a que

en la iteración inicial se almacena todo el conjunto de caminos. En las siguientes iteraciones solo es necesario sustituir los caminos que no pertenecen a la solución.

Origin UE es el algoritmo de asignación determinista emblemático en TransCAD. Los resultados son muy buenos. Alcanza niveles de convergencia muy elevados en cortos instantes de tiempo. El algoritmo equivalente diseñado en Matlab alcanza niveles de convergencia similares con un tiempo de ejecución ligeramente superior.

Con los resultados mostrados en la Figura 2-9 podría concluirse que la formulación de Dial (2006) es la más eficiente dentro de los algoritmos de asignación. Esto es así porque no se ha encontrado ningún software comercial que implemente la formulación de Bar-Gera (2010). Esta formulación se presenta como una mejora de Dial (2006). Su implementación en Matlab no ha conseguido mostrarse como una mejora pero hay que especificar que Bar-Gera (2010) incluye un módulo que debido a su complejidad no ha sido implementado en Matlab. Éste se encarga de seleccionar en cada iteración una solución de flujo en ruta estable a partir de una condición de proporcionalidad equivalente a la maximización de entropía.

En la Tabla 2-8 se muestra el valor final de la función objetivo, transformada de Beckmann, alcanzado por los algoritmos. El primer objetivo de un problema de asignación debe ser conseguir el mínimo valor posible de la transformada de Beckmann, esto significará que el equilibrio de usuario ha sido alcanzado. Minimizar este valor es más importante que alcanzar un valor de *relative gap* muy convergente. Un valor de *relative gap* excesivamente bajo implica que el coste de los caminos que componen un par O-D es muy semejante pero no implica que sea el mínimo que conecta dicho par O-D. El coste mínimo de los caminos del par O-D solo se asegura consiguiendo el mínimo valor de la función objetivo.

Algoritmo asignación	Función Objetivo
Origin UE	16.750.713,75
Path based	16.750.713,75
Frank & Wolfe	16.750.784,12
Basado en Bar-Gera (2010)	16.750.713,73
Basado en Dial (2006)	16.750.713,73
Basado en Caminos (apdo 2.7)	16.750.713,75

Tabla 2-8. Valor de la función objetivo final.

Puede apreciarse que la gran mayoría de algoritmos alcanzan un valor semejante de la función objetivo. La ejecución del método de Frank & Wolfe ha sido detenida después de 1000

iteraciones y 109 segundos pero el valor de la función objetivo aún no se ha estabilizado. Con seguridad empleando un número de iteraciones considerablemente superior se alcanzará el mismo valor de función objetivo que los demás métodos.

Los métodos que han alcanzado un mayor equilibrio de usuario son los basados en Bar-Gera (2010) y Dial (2006), implementados en Matlab. Los demás métodos obtienen funciones objetivos similares, seguramente con iguales valores de flujo casi en el 100% de los arcos pero con una solución de flujos en caminos sensiblemente diferente. Esta solución en caminos nunca alcanzará el nivel de equilibrio de estos métodos.

# CAPÍTULO 3

---

## PROBLEMA DE ESTIMACIÓN DE MATRICES ORIGEN – DESTINO

### 3.1 INTRODUCCIÓN

En la actualidad, los sistemas de transporte de las áreas urbanas se caracterizan por presentar una gran complejidad. Cada usuario tiene la posibilidad de alcanzar su destino a través de una gran variedad de rutas con diferentes modos de transporte y generalmente en un tiempo razonable.

Para llevar a cabo la planificación del transporte es necesario considerar todas las combinaciones de viaje existentes entre los diferentes orígenes y destinos repartidos por la red. Además, para cada pareja origen – destino (O-D) se debe tener en cuenta todos los modos de transporte que la relacionan. Éste es un problema combinatorio en el número de centroides y en el número de medios disponibles.

El planificador necesita del uso de herramientas analíticas que le permitan conocer y comprender el sistema para así poder realizar predicciones. Estas herramientas o modelos matemáticos permiten representar los fenómenos del transporte de una forma más simple.

Una de las metodologías más extendidas en la modelización del transporte es el modelo clásico de cuatro etapas que se compone de: generación, distribución, reparto modal y asignación. Todas estas etapas necesitan de una recopilación previa de información a fin de caracterizar el sistema de transporte. Esta tarea tiene un gran coste por la cantidad de datos que se requiere para representar la realidad a través de los modelos. Dentro de dicho proceso, además de la caracterización de la red y la recogida de información socioeconómica, se encuentra la estimación de la matriz de viajes que será abordada en esta tesis.

La generación de la matriz O-D es muy importante en los proyectos de planificación. La calidad del proyecto dependerá de la calidad de los datos de esta matriz, ya que todas las etapas en la modelización del transporte dependen de dichos datos. La evaluación de situaciones hipotéticas futuras, prognosis, es un proceso que puede tener una fiabilidad limitada. Debido a ello es preciso una monitorización muy exhaustiva de la metodología de modelización a fin de que los modelos generados representen con fidelidad el sistema de transporte actual y la matriz de viajes generada tenga un alto nivel de confianza.

Cuanto más preciso sea el conocimiento sobre las elecciones de los usuarios individuales del sistema de transporte, más exacta será modelización. Pero también, cuanto mayor sea el volumen de información recogido, más elevado será el coste del proceso. La mayoría de los proyectos de planificación son inviables por este motivo. Es imprescindible, por tanto, reducir estos costes sin que afecte de forma apreciable a la calidad en los modelos desarrollados.

Un inconveniente importante en las matrices de viajes es su obsolescencia: suelen quedarse desactualizadas relativamente pronto, especialmente en áreas urbanas. Cambios en población, actividades económicas y usos del suelo pueden dejar la matriz de demanda obsoleta para las circunstancias actuales. Volver a restablecer dicha matriz no debería suponer un problema para el planificador. Por eso resulta de la mayor importancia disponer de un proceso de estimación o ajuste rápido y de bajo coste.

## **3.2 PROPIEDADES DEL PROBLEMA DE ESTIMACIÓN DE MATRICES**

### **3.2.1 Zonificación de la región de estudio**

Cada viaje sobre la red de transporte es fruto de una decisión individual por parte de los usuarios, los cuales tienen la necesidad de realizar alguna actividad en destino. Cada usuario, dependiendo de sus características personales y del sistema de transporte, decidirá sobre la ruta y modo de transporte con el que viajar. Conocer y modelar las decisiones de todos los usuarios de forma individual es un proceso inviable por lo que todo el proceso se basa en la captura de una muestra de datos (bien agregados o desagregados). La modelización agregada se lleva a cabo realizando agrupaciones de usuarios por zonas. Las matrices O-D contienen los viajes agregados de todos los usuarios que deciden de forma individual viajar desde una zona origen hacia otra zona destino. Desde otro punto de vista, las matrices O-D representan el número de viajeros que se originan y atraen entre las diversas zonas de la red de transporte.

Dividir el sistema de transporte en zonas permite abordar los diferentes métodos que componen el modelo clásico de cuatro etapas. Actualmente el estudio y aplicación de las etapas de generación, distribución, reparto modal y asignación están muy desarrollados. A partir de la matriz O-D puede modelarse los flujos de vehículos por la red de una forma muy próxima a la realidad. El principal inconveniente sigue siendo el cálculo de la matriz de viajes que exige un esfuerzo presupuestario para conocer las decisiones de viaje de cada usuario. Aunque los modelos se generan desde un punto de vista agregado, los viajes de una zona a otra dependen de la decisión individual de cada usuario.

Al realizar la zonificación de la red, se establece la hipótesis de que las características de cada zona se concentran en un único punto llamado centroide y se agregan sus características socioeconómicas. La división en zonas se realiza manteniendo el esqueleto principal de la red. Los arcos contenidos dentro de cada división son simplificados o sustituidos por conectores. Estos últimos son los que enlazan los centroides con la red, sus propiedades son diferentes a los demás arcos.

### **3.2.2 Periodo de tiempo para el que se define la matriz de viajes**

La oferta de servicio de la red permanece constante en todo momento mientras no aparezcan problemas que disminuyan su capacidad como pueden ser accidentes, estacionamiento en doble fila, etc. En cambio la demanda de vehículos está en constante variación a lo largo del día. Los usuarios acceden a la red desde diferentes situaciones y en diferentes instantes de tiempo. En cada momento del día se refleja una matriz de viajes distinta sobre la red. Puede afirmarse por tanto que los sistemas de transporte en la práctica no están en estado de equilibrio de forma permanente, aunque generalmente se supone que están cerca del equilibrio o tendiendo hacia dicho estado.

Al igual que las agregaciones de viajes por zonas, es necesario agregar dicha información por periodos de tiempo con el fin de simplificar y reducir la complejidad del problema. Según la longitud del periodo de tiempo considerado el proyecto puede afrontarse desde un punto de vista estático o dinámico.

Si el periodo de tiempo considerado en el proyecto es amplio (p. ej. hora punta de mañana), el problema se analiza desde un punto de vista estático. En este caso se suponen las mismas condiciones de demanda durante dicho periodo de tiempo y estarán representadas por una única matriz de viajes.

Los modelos estáticos se centran en varias horas del día y utilizan valores medios de capacidad de la vía, demanda de viajes y tiempo de flujo libre en arcos, encontrando escasos problemas de exceso de vehículos y, por tanto, de colas. Esta hipótesis se supone válida para problemas de planificación estratégica o táctica.

En los problemas planteados desde una perspectiva dinámica, se consideran cortos periodos de tiempo (p. ej. minutos, cuartos de hora,...). En estos problemas se utiliza la matriz de demanda real que es diferente para cada periodo de tiempo. Este tipo de proyectos es más recomendable para analizar aspectos operacionales.

El uso de modelos dinámicos que sustituyan a los estáticos se debe a que estos últimos son incapaces de explicar cualquier evolución de los flujos de tráfico en cortos periodos de tiempo. Por ejemplo, durante un periodo de tiempo en hora punta, puede ocurrir que el número de vehículos que necesitan utilizar una vía sea mayor que lo que puede soportar. Este exceso de vehículos hace que se formen colas que serán descargadas más tarde. Cada elemento de la matriz de viajes O-D muestra valores de viajes originados y destinados diferentes en todo momento. Estos valores pueden llegar a ser incluso opuestos para una misma zona en diferentes instantes de tiempo. Por ejemplo, si se observa una zona que es completamente residencial, al inicio de la jornada ésta es claramente origen de viajes siendo relativamente pequeños el valor de viajes destinados. Pero al finalizar dicha jornada los valores se invierten originando muy pocos viajes y presentando un pico en el destino. Si el periodo de tiempo del proyecto se define de tal forma que recoja tanto el inicio como el fin de la jornada, la matriz de viajes va a tener una estructura aproximadamente simétrica.

### **3.2.3 Recogida de información en la estimación de matrices**

Generalmente pueden realizarse diferentes acciones para reducir el coste y tiempo de estimación de la matriz de viajes. A continuación se exponen las más profusamente empleadas. Todas estas acciones pueden combinarse para obtener una mayor calidad de los datos de entrada.

#### *3.2.3.1 Matriz de viajes base*

Si está disponible, es muy recomendable partir de una matriz O-D base o de partida, generada para el sistema de transportes en un proyecto anterior al que actualmente se acomete. La matriz O-D estará evidentemente desactualizada, pero puede contener información muy valiosa en cuanto a su estructura. Antes de utilizar una matriz O-D desactualizada es necesario comprobar la fuente de procedencia, antigüedad y esfuerzo económico que supuso para poder así evaluar su fiabilidad o “confianza” respecto del resto de información disponible.

### 3.2.3.2 *Encuestas de viajes*

La realización de encuestas sobre una muestra estadísticamente representativa de la población es una técnica muy empleada. La información que se obtiene es de mucha calidad ya que puede conocerse la hora y motivo del viaje, ruta seleccionada, etc. para cada familia encuestada. La principal desventaja de este método es que el hecho de responder a una encuesta consume tiempo y puede constituir una intromisión a la privacidad, por lo que además del coste de recogida de la información, es necesario luego realizar una corrección y validación de los datos.

### 3.2.3.3 *Modelos de distribución*

Los modelos de distribución zonal simulan el comportamiento de los usuarios por la red. Partiendo de la matriz de viajes base, se realizan las mínimas modificaciones posibles de forma que la matriz resultado sea consistente con los datos de generación y atracción observados o predichos. Estos modelos requieren como entrada de datos una matriz de demanda observada cuya obtención es bastante cara. Además, dependen en gran medida de la fiabilidad de la matriz de viajes relativa al año base. Esta metodología tiene más éxito y por eso es empleada con mayor frecuencia para pronosticar matrices futuras.

### 3.2.3.4 *Recogida y estimación mediante datos agregados*

La obtención de información de forma agregada es una tarea rápida, simple y menos costosa que la realización de encuestas. Las principales formas de recogida de datos agregados son el conteo de vehículos y el escaneo de matrículas. Estos métodos no presentan ninguna influencia sobre los viajeros. La primera opción obtiene el número de vehículos que atraviesan una vía. Esta opción es ampliamente utilizada, muchas administraciones recogen esta información de forma automática. Los métodos de ajuste de matrices O-D intentarán reproducir de la forma más fidedigna posible el volumen medido en estas vías. La segunda opción es menos empleada, es necesario procesar las matrículas para distinguir la ruta de cada vehículo y puede constituir una intromisión a la privacidad. Mediante el escaneo de matrículas se adquieren datos de mayor calidad ya que se recoge, además del conteo de vehículos, información de las rutas seguidas por los usuarios.

## **3.2.4 Indeterminación del problema**

En apartados anteriores se ha comentado la no viabilidad del proyecto cuando se pretende conocer la elección de cada usuario de forma individual. Además, para abordar el problema de transporte se ha indicado la necesidad de dividir el sistema en zonas y establecer como objetivo

principal conocer los movimientos de grupos de usuarios entre estas zonas. Con esta división se reduce la complejidad del sistema, aunque la estimación de la matriz de viajes sigue siendo un problema indeterminado como se detalla a continuación.

La división de la red en zonas constituye una parte importante de la modelización, debe ser un proceso trabajado y exhaustivo para no llegar a una distorsión de la realidad. Cuando se crean las diferentes zonas, se están simplificando los arcos de la red y se está suponiendo que todos los individuos de la zona siguen el mismo patrón de desplazamiento. Es decir, todos los individuos salen desde el mismo punto (centroide) y acceden a la red por las mismas salidas (conectores).

Para llegar a una solución que represente fielmente el desplazamiento de los viajeros es necesario que existan centroides distribuidos por toda la red y que el tamaño de la zona que representa no sea muy elevado para no simplificar en demasía el problema. Si las zonas de transporte en las que se divide el problema son muy grandes, pueden englobar arcos importantes que no serán considerados ya que a las zonas se accede a través de los conectores.

Para llevar a cabo esta división zonal se suelen utilizar las áreas censales o divisiones de tamaño similar para la zonificación de los centros urbanos, y distritos o municipios para la zonificación en las áreas interurbanas.

Las zonas se distribuyen por toda la red y cada individuo puede acceder desde su centroide origen hasta cualquier centroide destino diferente del de salida (no se tienen en cuenta los viajes internos). El número de combinaciones entre centroides, eliminando los viajes intra-zonales, es igual a  $N^2 - N$ . Es decir, el número de rutas es al menos  $N-1$  veces el número de zonas. Este valor indica el número de elementos de la matriz de viajes que conforman las variables del problema y que, generalmente, resultan varios órdenes superiores al número de arcos de la red. Sabiendo que cada arco aforado aporta una restricción lineal al problema del ajuste, puede demostrarse que, aun aforando el conjunto íntegro de arcos, el sistema es indeterminado.

La medición del volumen de vehículos en todos los arcos de la red es inviable, en la práctica se suele aforar aproximadamente un 10% del total. Además, el aforado de todos los arcos es innecesario, puede no aportar más información al problema debido a que existe dependencia lineal entre ellos. A través de las leyes de Kirchhoff o de conservación de flujo en los nodos, existen tantas relaciones entre arcos como nodos internos tenga el grafo.

A esta indeterminación, hay que añadirle un problema que aporta mayor complejidad a la estimación de matrices. El comportamiento de los usuarios es individualista ya que su objetivo a

la hora de elegir ruta no es el de disminuir el tiempo total del sistema sino el suyo propio. Cada usuario selecciona las calles, carreteras o autopistas principales evitando las que le consuman un mayor tiempo de viaje. Esto hace que los usuarios de un gran número de pares O-D utilicen las mismas vías, las principales. Con este comportamiento no todos los arcos de la red son empleados. Grandes cantidades de vehículos se acumulan sobre las vías principales y hacen complicada la discriminación de los porcentajes de viajeros de cada par O-D en los arcos.

Para ilustrar lo anterior se estudia la indeterminación de la red de Tiergarten (Figura 3-1). Esta red se presenta en el apartado 2.8.1.1; ha sido seleccionada por tener una baja indeterminación respecto de la mayoría de casos que se dan en la práctica.

La red se compone de 747 arcos y 344 nodos, de los cuales 26 son centroides. La matriz de viajes por tanto consta de 650 elementos (existen más arcos que pares O-D). Si se afora un 10% de los arcos se tendría 75 datos de volumen de tráfico para estimar el valor de 650 elementos de la matriz. Los arcos aforados se marcan en rojo (trazado grueso).

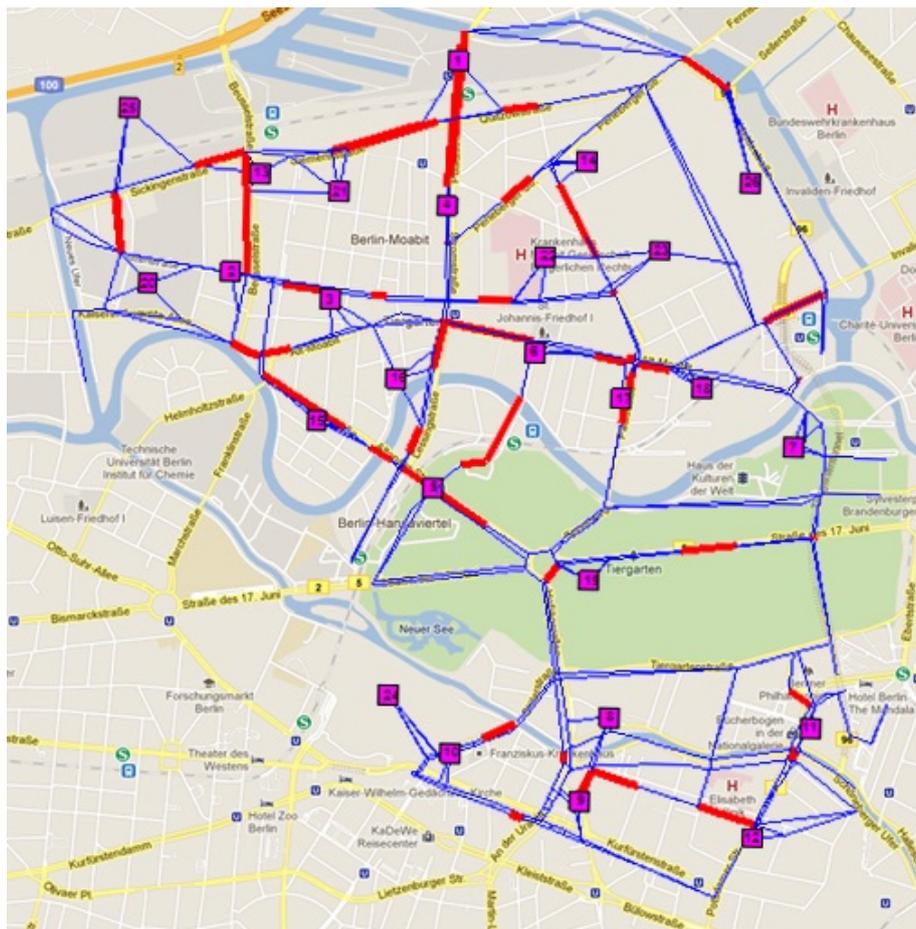


Figura 3-1. Red de Tiergarten.

Con el fin de acentuar los problemas que genera la congestión, se realiza una modificación respecto del problema original aumentando los viajes entre zonas. Cuando se asigna dicha matriz de demanda a la red, el número de arcos con flujo es 521. Si fuera posible la opción de aforar todos estos arcos, se tendrían 521 datos de aforo frente a 650 pares O-D, con lo que incluso en ese caso el problema sigue estando indeterminado. Además, aplicando las leyes de Kirchhoff de conservación de flujo en los nodos del grafo se deduce que entre los 521 arcos con flujo existen relaciones de dependencia, siendo menor el número de arcos linealmente independientes.

Con este pequeño ejemplo puede comprobarse la fuerte indeterminación que existe en la estimación de matrices. Aumentar el nivel de detalle de la red, agregando más zonas al problema, con el fin de encontrar una solución de mayor calidad puede elevar de forma considerable la indeterminación. Por ejemplo, al emplear 30 zonas se trabaja con una dimensión de 870 pares O-D para el mismo número de arcos y aforos. De forma contraria, dividiendo la red en un número inferior de zonas el número de incógnitas se reduce de forma considerable (22 zonas generan un total 462 pares O-D).

Otros datos interesantes de esta red son los siguientes:

- Total de 761 caminos activos para este nivel de congestión. Existen 5 pares O-D unidos por 4 caminos, 5 pares O-D por 3 caminos, 86 pares O-D por 2 caminos y el resto con sólo un camino.
- Existen arcos con hasta 139 pares O-D diferentes pasando por ellos. El algoritmo de ajuste debe discriminar que porcentaje ocupa cada par O-D sobre el volumen total del arco.

### 3.2.5 La asignación como parte del problema del ajuste

Dada una matriz de viajes  $g$ , cuyo elemento  $g_i$  contiene el número de viajes del par O-D  $i$  durante el intervalo temporal considerado, el procedimiento de asignación reparte los viajes en los distintos arcos de la red de transporte de acuerdo a un determinado principio de equilibrio. Como resultado, si  $A$  es el conjunto de arcos de la red de transporte, el volumen de vehículos  $v_a$  en cada arco viene dado por:

$$v_a = \sum_{i \in P} p_{i,a} \cdot g_i \quad \forall a \in A \quad (3.1)$$

donde  $P$  representa el conjunto de pares O-D y  $p_{i,a}$  representa la fracción de viajes del par O-D  $i$  que utilizan el arco  $a$ . El tratamiento de la matriz de viajes como pseudovector hace más simple y comprensible las expresiones. En apartados sucesivos se emplearán indistintamente las variables  $T_{ij}$  y  $g_i$  aunque se continúan nombrando como matriz de viajes.

La ecuación (3.1) representa el proceso de asignación de viajes de una red de transporte. Para una matriz de viajes determinada y un comportamiento de usuarios predefinido, este procedimiento obtiene una única solución en arcos que está en equilibrio. La estimación de matrices de viajes representa el proceso contrario, dados unos valores de aforos medidos en los arcos de la red se estima la matriz que los representa. Esta técnica en cambio no tiene solución única debido a su indeterminación.

A diferencia de la asignación, el ajuste de matrices O-D es un proceso complejo debido a que los valores de flujo observados en los aforos tienen la limitación de que son captados de forma agregada. Los métodos de ajustes más conocidos necesitan resolver múltiples problemas de asignación de tráfico en cada iteración para ir corrigiendo la diferencia entre los flujos modelados y los observados. Desagregar el flujo observado entre sus diferentes orígenes – destinos es una tarea ardua y por esto el peso computacional de la asignación dentro de un problema de ajuste es muy elevado.

Conocida una cierta matriz O-D es necesario resolver un problema de asignación para conocer la distribución de sus flujos sobre los arcos de la red. Sabiendo que la solución en arcos es única, el problema de asignación permite analizar si la matriz O-D obtenida representa o se ajusta a la información disponible sobre la red. Una vez realizado el proceso de ajuste será necesario, por tanto, ejecutar un algoritmo de asignación que permita analizar la bondad del mismo. Los métodos de ajuste iterativos asignan la matriz O-D obtenida en cada iteración ya que, a través de las soluciones intermedias es posible decidir sobre qué elementos de la matriz actuar para mantener un control del proceso.

Para simplificar el problema y centrarlo exclusivamente en el ajuste de matrices se realizan las siguientes hipótesis: *i)* el algoritmo de asignación se supone que representa fielmente el comportamiento de los usuarios, *ii)* la red se considera correctamente calibrada con funciones de demora que modelan de forma exacta la relación tiempos – volúmenes en las vías. La calibración de los parámetros de la red y el análisis del comportamiento de los usuarios son procesos importantes y complejos que no se abordan en esta tesis.

A continuación se analizan las expresiones empleadas en el problema de asignación ya que dependiendo de los valores de  $p_{i,a}$  se distinguen diferentes tipologías de ajuste. La ecuación (3.1)

está formada por la unión de la ecuación de conservación del flujo (2.3) y la ecuación de transformación del flujo en caminos a flujo en arcos (2.5):

$$\begin{aligned} \sum_{k \in K_i} h_k &= g_i & \forall i \in P \\ v_a &= \sum_{i \in P} \sum_{k \in K_i} \delta_{ak} h_k & \forall a \in A \end{aligned}$$

donde

$$\delta_{ak} = \begin{cases} 1 & \text{Si el arco } a \text{ pertenece a la ruta } k \\ 0 & \text{e.o.c.} \end{cases}$$

De las expresiones anteriores se deduce que

$$p_{i,a} = \frac{\sum_{k \in K_i} \delta_{ak} h_k}{\sum_{k \in K_i} h_k} = \frac{\sum_{k \in K_i} \delta_{ak} h_k}{g_i} \quad \forall a \in A \quad (3.2)$$

Se observa que los valores de  $p_{i,a}$  están comprendidos entre 0 y 1.  $p_{i,a}$  tiene valor nulo cuando ningún camino del par  $i$  pasa por el arco  $a$ , tiene valor igual a 1 si todos los caminos del par  $i$  lo atraviesan y un valor intermedio ( $0 \leq p_{i,a} \leq 1$ ) si solo una fracción de los caminos pasa por dicho arco.

Según la influencia del flujo de vehículos sobre el coste de viaje en los arcos, dentro del algoritmo de asignación se distingue entre problema con congestión y sin congestión. El problema sin congestión sólo encuentra una ruta competitiva para cada pareja O-D y ésta será siempre la mejor opción independientemente de su flujo. En este caso no existen valores intermedios entre 0 y 1 de  $p_{i,a}$ . El problema con congestión, ya sea resuelto mediante equilibrio de usuario determinista o estocástico, puede contener valores intermedios de  $p_{i,a}$  y además las rutas pueden variar para matrices O-D diferentes.

Con respecto a la variable  $p_{i,a}$  calculada en la asignación, los métodos de ajuste pueden clasificarse en dos, dependiendo de si se considera constante o no la proporción en el arco  $p_{i,a}$  durante las diferentes iteraciones: métodos de ajuste con asignación proporcional y métodos de ajuste con asignación no proporcional.

Los métodos de ajuste con asignación proporcional suponen que las proporciones de viaje en el arco se mantienen aproximadamente constantes durante todo el proceso. Para no violar esta

hipótesis, los caminos entre pares O-D deben mantener además la relación en flujos respecto de los demás caminos dentro de dicho par. Esta hipótesis es válida para redes sin congestión o poco congestionadas donde las proporciones de viajes  $p_{i,a}$  son independientes del volumen en el arco.

En el caso de asignación no proporcional se tienen en cuenta los efectos de la congestión y, por tanto, las proporciones de viaje  $p_{i,a}$  dependen del nivel de tráfico en la red de transporte.

Se considera que las técnicas de asignación no proporcional son más realistas para condiciones de congestión. Por otra parte, la asignación no proporcional requiere la estimación conjunta o iterativa de las proporciones  $p_{i,a}$  y de las matrices de viajes para que ambas sean consistentes. La ventaja de los métodos de asignación proporcional es que puede separarse la parte de asignación (cálculo de  $p_{i,a}$ ) de la estimación de matrices, aunque esta hipótesis de independencia tiene limitaciones inherentes como se indica en Yang et al. (1992): “Debido a que la matriz O-D es estimada a partir de flujos observados en los arcos con proporciones de viajes constantes, y la matriz O-D es asignada a la red haciendo uso del concepto de equilibrio de usuario, hay una inconsistencia al utilizar unos valores de proporciones de viaje para obtener una matriz O-D a partir de los aforos, y otros para obtener la distribución de flujo asignando la matriz O-D a la red”. Por otra parte, según Yang et al. (1994), los errores derivados de las proporciones de rutas constantes y las observaciones de tráfico, pueden dar lugar a un sistema con ecuaciones de restricción de conteo inconsistente, por lo que debe llevarse a cabo algún tipo de preproceso para estimar flujos en arcos de forma consistente a partir de los aforos de tráfico para que los modelos de estimación sean factibles.

### 3.3 DEFINICIÓN DEL PROBLEMA

Las distintas formas de recogida de información para la estimación de matrices descritas en el apartado 3.2.3 presentan ciertas limitaciones:

- La matriz resultado de los procesos de encuestas, al recopilar información de una muestra de familias y expandirla sobre el conjunto de la población, no tiene en cuenta los viajes de todos los usuarios. Aunque, al considerar una muestra estadísticamente representativa, esta matriz contiene información muy importante en cuanto a la estructura de la matriz real.
- Los modelos de distribución zonal modelan el comportamiento de los usuarios pero no son capaces de explicar todos los viajes.
- Por sí solos los datos de volúmenes de aforos representan un sistema indeterminado por lo que no es una información válida para generar una matriz O-D. Aun así estos datos

de volumen pueden combinarse con cualquiera de los métodos anteriores para aportar mayor fiabilidad a la matriz resultado.

El objetivo que se persigue es diseñar un algoritmo de estimación de matrices de viaje tal que partiendo de una matriz de demanda desactualizada, ya sea resultado de un proyecto anterior, venga de un proceso de encuesta o sea la salida de un modelo de distribución, se ajuste para que cuando ésta sea asignada a la red, los flujos modelados se asemejen a los volúmenes medidos en los arcos. El periodo de tiempo definido será suficiente como para considerar el problema estático. Así, se buscará una única matriz que represente las mismas condiciones de demanda durante todo el periodo.

En la práctica, actualizar una matriz de referencia utilizando datos de volúmenes medidos en los aforos es un proceso muy económico. Se parte de una matriz ya construida y se conocen multitud de datos de aforos en las vías principales ya que las administraciones recogen valores de flujo automáticamente en muchos puntos de la red. Durante los últimos 40 años esta metodología ha sido la opción más atractiva y ha recibido especial atención en la literatura técnica dedicada a la planificación del transporte.

Si se denota por  $\bar{A}$  el subconjunto de arcos aforados del conjunto  $A$  de arcos de la red para el periodo objeto de estudio, la obtención de la matriz O-D para dicho periodo se plantea inicialmente en términos del sistema de ecuaciones (3.1):

$$v_a = \sum_{i \in P} p_{i,a} \cdot g_i = \bar{v}_a \quad \forall a \in \bar{A} \subseteq A \quad (3.3)$$

donde  $\bar{v}_a$  es el volumen de vehículos medido (observado) en cada arco del subconjunto  $\bar{A}$ .

El sistema de ecuaciones planteado en (3.3) generalmente es un problema incompatible. Los valores recogidos en los puntos de aforo presentan inconsistencias que llevan a un problema de ajuste sin solución. Estas inconsistencias se deben a errores en el proceso de medición como son: *i)* las mediciones en los puntos de aforo pueden realizarse en diferentes horas, incluso días, *ii)* el número de mediciones tomadas en cada aforo es diferente, teniendo más fiabilidad cuantas más horas de medición se encuentren disponibles, *iii)* existe tráfico recursivo imposible de discriminar, *iv)* el sistema de aforo puede arrojar valores erróneos, etc.

Para salvar los problemas de incompatibilidad e indeterminación, los métodos de ajuste se formulan como problemas de optimización matemática. Así, se selecciona la mejor solución, entre un conjunto de soluciones, maximizando o minimizando un criterio determinado. Puede

concluirse que los distintos métodos para el problema de estimación de matrices de viajes a partir de volúmenes de tráfico desarrollados en la literatura, presentan la forma genérica siguiente:

$$\underset{g,v}{\text{Minimizar}} \quad Z(g,v) = \gamma_1 F_1(g, \bar{g}) + \gamma_2 F_2(v, \bar{v}) \quad (3.4)$$

$$s.a. \quad v = P(g)g \quad (3.5)$$

$$g \in \Omega \quad (3.6)$$

siendo  $\gamma_1$  y  $\gamma_2$  factores de peso, donde  $\gamma_1 \geq 0$ ,  $\gamma_2 \geq 0$ ,  $\gamma_1 + \gamma_2 = 1$ . Las restricciones (3.5) representan el modelo de asignación. Cada fila de la matriz  $P(g)$  está formada por las fracciones de viajes correspondientes a cada par O-D para los diferentes arcos aforados.  $\Omega$  es el conjunto de las matrices O-D factibles, normalmente definido por las restricciones de no negatividad de los elementos aunque es posible añadirle ciertas restricciones lineales como la limitación de las generaciones y atracciones de los centroides.

Las funciones  $F_1(g, \bar{g})$  y  $F_2(v, \bar{v})$  representan medidas de distancia generalizadas entre la matriz O-D estimada  $g$  y la matriz de partida  $\bar{g}$ , y entre el flujo en el arco estimado  $v$  y el observado  $\bar{v}$ , respectivamente. En resumen, se minimiza la discrepancia respecto de dos parámetros  $g$  y  $v$  con el fin de realizar el ajuste de la matriz de viajes. Se supone que las funciones son convexas, por lo general definidas como cuadráticas, entrópicas o de máxima probabilidad y están diseñadas para tener en cuenta la aleatoriedad de la información recogida.

Los factores de peso  $\gamma_1$  y  $\gamma_2$  en (3.4) reflejan la confianza (o incertidumbre) relativa en la información proporcionada por  $\bar{g}$  y  $\bar{v}$ . Si la matriz previa se considera más fiable y precisa en comparación con los conteos de tráfico, entonces  $\gamma_1 \gg \gamma_2$ , se obtiene como resultado una matriz  $g$  muy cercana a  $\bar{g}$  donde se acepta una desviación mayor entre  $v$  y  $\bar{v}$ . En el caso contrario,  $\gamma_1 \ll \gamma_2$ , la matriz previa no tendrá influencia y los volúmenes estimados serán muy similares a los observados. Existe un caso especial cuando la matriz previa es fiable y los conteos de tráfico se consideran libres de errores. Entonces la formulación anterior se reduce a:

$$\underset{g}{\text{Minimizar}} \quad F_1(g, \bar{g}) \quad (3.7)$$

$$s.a. \quad \bar{v} = P(g)g \quad (3.8)$$

$$g \in \Omega \quad (3.9)$$

Previamente se ha clasificado la estimación de matrices O-D en dos categorías según si las proporciones de viajes en arco dependen de la matriz de viajes o no. Existe un elevado número de referencias bibliográficas en cada categoría. En el caso de asignación proporcional las restricciones (3.5) se formulan como:

$$v = P \cdot g \quad (3.10)$$

donde  $P$  es la matriz de proporciones que es constante durante todo el proceso.

Para los métodos que emplean asignación no proporcional, Fisk (1988) reconoció el problema de estimación de matrices como un problema de programación binivel. En el nivel superior se realiza el ajuste de la matriz y en el nivel inferior se resuelve un problema de asignación bajo el principio de equilibrio de usuario. La solución se alcanza a partir de la aplicación de métodos iterativos que resuelven alternativamente los dos niveles e intercambian parámetros entre ellos.

Aunque cada nivel está compuesto por un problema de programación convexa con una única solución, la programación binivel es generalmente no convexa. Las soluciones obtenidas a través de los métodos existentes, en el mejor de los casos, son óptimos locales. Aun así, la programación binivel al menos asegura consistencia entre las hipótesis de asignación en la estimación y las hipótesis en la extrapolación de los valores de aforos en los arcos no medidos (Bell y Iida (1997)).

Estimar las matrices O-D en cualquier caso no evita la indeterminación del problema. Existe un gran número de matrices que cuando son asignadas a la red representan exactamente los volúmenes aforados. Por tanto, la estimación de matrices se reduce a seleccionar una solución particular dentro de un sistema de ecuaciones no lineal e indeterminado. Esto le proporciona complejidad al problema y supone un desafío para los investigadores. Todos los trabajos en este campo se centran en encontrar una formulación que discrimine dentro del conjunto de matrices O-D factibles la que, reproduciendo los valores de aforo, represente con mayor probabilidad el sistema de transporte actual.

El problema ideal es aquel en el que la matriz de viajes inicial no está muy desactualizada. En ese caso, las diferencias entre el flujo modelado y aforado son mínimas. Así, con pequeñas modificaciones se puede alcanzar una solución más precisa y fiable.

Si por el contrario, la matriz O-D inicial está muy desactualizada ya sea porque el tráfico ha crecido fuertemente o la encuesta no ha recogido un grupo de viajeros, las distancias entre el flujo modelado y aforado serán elevadas. Así en el problema del ajuste, debido a su

indeterminación, podrían calcularse variaciones elevadas de elementos de la matriz modificando su estructura inicial y por tanto perdiendo fiabilidad.

En la práctica, la matriz O-D de partida está frecuentemente sesgada a la baja. Esta subestimación de la matriz real, en las matrices O-D obtenidas en las encuestas domiciliarias se debe a la falta de respuestas y en las matrices O-D desactualizadas son el resultado de un incremento en el total de viajes hasta la fecha del proyecto (Cascetta y Nguyen, 1988; Yang et al., 1992). En estos casos los modelos de estimación llegan de forma sistemática a soluciones fuertemente sesgadas incluso si la mejora respecto de la matriz O-D de partida es muy significativa. Para reducir esta desviación o distorsión Fisk y Boyce (1983), Doblas y Benítez (2005) y otros investigadores han tratado el problema añadiéndole restricciones tanto a elementos de la matriz como a filas (generación), columnas (atracción) o suma total de viajes. Yang et al. (1992) sugieren una formulación modificada de la función objetivo en el nivel superior. Esta modificación es equivalente a un factor de crecimiento en general como el introducido por Ortúzar y Willumsen (2002) para multiplicar la matriz O-D inicial antes de utilizar un estimador.

### **3.4 ESTADO DEL ARTE DE LOS MODELOS DE ESTIMACIÓN DE MATRICES A PARTIR DE DATOS DE AFOROS**

En los últimos 40 años han sido desarrolladas diversas técnicas con el objetivo de encontrar la relación existente entre los flujos medidos en los puntos de aforos y la demanda de viajes generada y atraída por cada zona en las que se divide la red de transporte. De forma breve, y sin exhaustividad, se describen las formulaciones más importantes. Éstas se clasifican según la función objetivo que implementen:

- Modelos gravitacionales.
- Modelos de máxima entropía/mínima información.
- Modelos de máxima verosimilitud.
- Modelos de inferencia bayesiana.
- Modelos de mínimos cuadrados generalizados.

La expresión (3.4), formada por dos términos que recogen unas funciones de distancia genéricas para las matrices O-D y para los volúmenes de tráfico, engloba todas las formulaciones anteriores eligiendo convenientemente las formas funcionales que la definen.

### 3.4.1 Modelos Gravitacionales

El calibrado de modelos gravitacionales fue uno de los primeros métodos empleados para la estimación de matrices de viajes. La idea básica es plantear una forma particular del modelo de gravedad y examinar lo que ocurre cuando se asigna a la red (Ortúzar y Willumsen, 2002). Este tipo de metodología fue utilizada por los primeros investigadores en este área, Low (1972), Robillard (1975), Högborg (1976) y Holm et al. (1976), y más tarde por Wills (1986), Tamin y Willumsen (1989) entre otros.

La principal diferencia de este método respecto de los demás radica en que el proceso de ajuste se lleva a cabo aplicando la estructura del modelo de gravedad (Erlander y Stewart, 1990). En este caso el modelador restringe el espacio posible de soluciones para la matriz estimada imponiendo una estructura particular que se obtiene habitualmente de un modelo de demanda de viajes existente, por ejemplo un modelo de gravedad o de demanda directa. Además, una vez calibrado el modelo de demanda, puede utilizarse también para la prognosis suponiendo que los valores de generación y atracción de viajes estén disponibles o sean de fácil estimación.

El principal inconveniente de los modelos gravitacionales según Van Zuylen y Willumsen (1980) es que obligan a la matriz de demanda a seguir un cierto patrón de gravedad y no hace un uso completo de la información que aportan los aforos siendo valores observados. Otro inconveniente es su coste computacional, supone un gran esfuerzo ya que es necesario calibrar múltiples parámetros.

### 3.4.2 Modelos de máxima entropía/mínima información

Debido a que la información obtenida sobre un conjunto pequeño de aforos es insuficiente para determinar una única matriz de viajes, diversos autores (Van Zuylen y Willumsen, 1980; Bell, 1983; Nguyen, 1977; LeBlanc y Farhangian, 1982) utilizan una matriz de referencia para resolver el problema de seleccionar la matriz de viajes que con más probabilidad reproduzca los aforos. Para ello en las dos primeras referencias se emplean métodos de mínima información/máxima entropía. Estos métodos se desarrollan desde el punto de vista de la obtención de una matriz de viajes a partir de una de referencia, pero dándole menos importancia a esta última.

Los modelos de mínima información/máxima entropía suponen que la fuente de información más fiable en la red de transporte proviene de los aforos. La matriz de referencia añade la mínima información posible mientras se tienen en cuenta las restricciones entre los volúmenes aforados y modelados (Abrahamsson, 1998). Según Ortúzar y Willumsen (2002) la

maximización de la entropía produce una matriz O-D sencilla y poco sesgada que es consistente con la información disponible, a diferencia de los modelos de gravedad donde sus restricciones representan información sobre viajes totales zonales y costes totales.

El problema de maximización de entropía donde se tiene en cuenta una matriz de referencia, responde a una formulación con la siguiente función objetivo:

$$\underset{T}{\text{Maximizar}} \quad Z = - \sum_{i,j} \left( T_{ij} \log \left( \frac{T_{ij}}{\bar{T}_{ij}} \right) - T_{ij} + \bar{T}_{ij} \right) \quad (3.11)$$

donde  $T = [T_{ij}]$  es la matriz a estimar y  $\bar{T} = [\bar{T}_{ij}]$  la matriz de referencia.

Para la aplicación de este modelo es necesaria la existencia de una matriz O-D de referencia. El problema de minimización formulado tiene múltiples soluciones, siendo importante para el resultado final la solución inicial de partida.

Una limitación que tienen estos métodos fue advertida por Maher (1983) y es el hecho de que consideran los aforos de tráfico como observaciones sin error. El modelo supone completa credibilidad a los aforos y utiliza la matriz de partida sólo para compensar el hecho de que los aforos no contienen suficiente información para la estimación. La matriz de partida es simplemente una condición inicial y estos enfoques emplean la mínima información posible que proporciona esta condición.

Para compensar esta dificultad, la función de entropía puede ser empleada como una medida de distancia; en este caso la función objetivo se formula como sigue (Brenninger-Gothe y Jornsten, 1989; Willumsen, 1984):

$$\underset{T,v}{\text{Minimizar}} \quad Z = \gamma_1 \sum_{i,j} T_{ij} \left( \log \left( \frac{T_{ij}}{\bar{T}_{ij}} \right) - 1 \right) + \gamma_2 \sum_{a \in A} v_a \left( \log \left( \frac{v_a}{\bar{v}_a} \right) - 1 \right) \quad (3.12)$$

donde  $v = [v_a]$ ,  $\bar{v} = [\bar{v}_a]$  son los vectores de flujo estimado y aforado respectivamente, y  $\gamma_1$ ,  $\gamma_2$  los factores de peso. Estos factores ponderan la expresión para tener en cuenta la incertidumbre relativa entre los flujos medidos en los aforos y la matriz de viajes.

### 3.4.3 Modelos de máxima verosimilitud

Además del nivel de confianza relativo entre los aforos y la matriz de viajes, muchos autores han indicado la necesidad de considerar que la información recogida presenta inconsistencias que llevan al incumplimiento de la restricción de conservación del flujo. Este problema puede encontrarse tanto en los puntos de aforos como en los datos usados para generar la matriz de viajes. En la captura de datos de aforos pueden generarse errores bien en el conteo de los vehículos o bien por pérdida de datos; además a menudo los aforos se realizan en momentos diferentes y para diferentes arcos (horas, días, semanas), lo que hace que aparezcan inconsistencias entre los volúmenes medidos en los puntos de aforos. A su vez la matriz O-D de partida, que suele obtenerse de encuestas o estudios pasados, contiene un cierto grado de error debido, por ejemplo, a errores de inferencia, cambios de actividades empresariales, variación de la población, etc.

Los métodos de estimación de matrices que se describen a continuación han sido formulados bajo el enfoque de la inferencia estadística. Éstos tienen en cuenta la naturaleza estocástica de los datos de viaje introduciendo un término de error en las ecuaciones. En este caso se supone que la intensidad en los arcos sigue una distribución de probabilidad concreta y las observaciones disponibles son muestras de dicha distribución. Estos métodos no están limitados por tanto por los problemas de independencia y consistencia.

Una interpretación intuitiva para los métodos de estimación de matrices de tipo estadístico se basa en encontrar una matriz de viajes cercana a la de partida y que, una vez asignada a la red, los flujos reproducidos en los arcos sean lo más aproximados posible a los volúmenes aforados.

El enfoque de máxima verosimilitud maximiza la verosimilitud de observar la matriz O-D de partida y los volúmenes de aforos, estos últimos condicionados a la matriz estimada, bajo la hipótesis de que estas variables, matriz O-D y aforos, son estadísticamente independientes.

Debido a la hipótesis de independencia, maximizar la verosimilitud de observar ambas variables es igual a maximizar el producto de la verosimilitud de cada variable por separado. Con la convención de que  $0 \cdot \ln(0) = 0$ , es posible maximizar el logaritmo del producto.

La matriz O-D de referencia y los datos de aforos están formados por dos elementos, el valor observado en la fase de adquisición más un componente implícito en dicho valor que es la suma de los errores que se han podido cometer en dicha adquisición.

$$\begin{aligned}\bar{T} &= T^{obs} + \varepsilon_{\bar{T}} \\ \bar{v} &= v^{obs} + \varepsilon_{\bar{v}}\end{aligned}\quad (3.13)$$

De forma general, se supone que éstas son variables aleatorias con medias igual a la matriz O-D estimada y los flujos modelados, respectivamente. Esto implica que los errores tienen media nula.

$$\begin{aligned}E(\varepsilon_{\bar{T}}) &= 0 \\ E(\varepsilon_{\bar{v}}) &= 0\end{aligned}\quad (3.14)$$

Con todo lo anterior y suponiendo que las variables del problema siguen una determinada distribución de probabilidad se obtiene una expresión para la función objetivo. Las distribuciones de probabilidad más empleadas son la de Poisson y la normal multivariante. Spiess (1987) emplea la distribución de probabilidad de Poisson formulando el modelo de máxima verosimilitud como:

$$\underset{T,v}{Maximizar} \quad Z = \sum_i \sum_j \left( -\alpha_{ij} t_{ij} + \bar{t}_{ij} \cdot \ln(t_{ij}) \right) + \sum_{a \in A} \left( \bar{v}_a \cdot \ln(v_a(T)) - v_a(T) \right) \quad (3.15)$$

Si en su lugar la distribución de probabilidad es normal multivariante se obtiene la siguiente función objetivo:

$$\underset{T,v}{Minimizar} \quad Z = \frac{1}{2} (\bar{T} - T)^T \Sigma_{\bar{T}}^{-1} (\bar{T} - T) + \frac{1}{2} (\bar{v} - E(v))^T \Sigma_{\bar{v}}^{-1} (\bar{v} - E(v)) \quad (3.16)$$

donde  $E(v)$  es el vector de valores esperados de los flujos en arcos,  $\Sigma_{\bar{T}}^{-1}$  y  $\Sigma_{\bar{v}}^{-1}$  son las matrices de varianzas-covarianzas de los valores esperados de los elementos de la matriz de viajes y de los flujos en arcos, respectivamente.

Para profundizar más en este tema véase Cascetta y Nguyen (1988) y Willumsen (1991).

### 3.4.4 Modelos de inferencia bayesiana

Conocida la probabilidad de que se den ciertos volúmenes por los arcos de la red dado que se tiene la matriz de viajes, aplicando el teorema de Bayes podría conocerse la probabilidad de obtener una cierta matriz de viajes si se conocen los volúmenes en los arcos de la red, siempre conociendo la probabilidad de que se dé la matriz de viajes inicial.

Los llamados estadísticos bayesianos permiten probabilidades subjetivas. El teorema puede servir entonces para indicar cómo deben modificarse las probabilidades subjetivas cuando se recibe información adicional de un experimento. Los modelos de inferencia bayesiana consideran la matriz O-D de referencia como una función de probabilidad a priori de la matriz O-D estimada y los datos de aforos son considerados otra fuente de información experimental. Por tanto, el teorema de Bayes proporciona un método para combinar ambas fuentes de información, rigiéndose por:

$$P(T|\bar{v}) \propto P(\bar{v}|T)P(T) \quad (3.17)$$

La matriz de viajes se puede encontrar maximizando el logaritmo de la expresión (3.17). Generalmente se establece la hipótesis de que los volúmenes de aforo siguen una distribución de probabilidad de Poisson o normal multivariante. Las expresiones para el logaritmo de  $P(\bar{v}|T)$  son: *i*) segundo término de la expresión (3.15) si se supone que  $\bar{v}$  sigue una distribución de probabilidad de Poisson y, *ii*) segundo término de la expresión (3.16) si se supone que  $\bar{v}$  sigue una normal multivariante. Para la función de probabilidad  $P(T)$  podría suponerse una distribución multinomial (3.18), Poisson (primer término en (3.15))(3.19) o normal multivariante (primer término en (3.16)). En estos casos es útil la aproximación de Stirling para el logaritmo de  $P(T)$ :

$$\ln P(T) = -\sum_{ij} t_{ij} \ln \left( \frac{t_{ij}}{t_{ij}} \right) + c \quad (3.18)$$

$$\ln P(T) = -\sum_{ij} t_{ij} \left( \ln \left( \frac{t_{ij}}{t_{ij}} \right) - 1 \right) + c \quad (3.19)$$

El modelo más conocido es el de Maher (1983) que emplea una distribución normal multivariante para  $P(T)$  y  $\bar{v}$  con covarianzas conocidas  $\Sigma_{\bar{T}}^{-1}$ ,  $\Sigma_{\bar{v}}^{-1}$ . Éste es aplicable únicamente para redes de transporte de muy pequeño tamaño.

Los métodos bayesianos, por tanto, amplían los métodos de máxima entropía/mínima información, siendo iguales cuando los elementos de la matriz O-D de partida se encuentran distribuidos equitativamente. Además tienen la ventaja de que puede incluirse información recogida de otras fuentes de información. Otro dato a destacar es que además de la matriz de viajes, es posible obtener información sobre la variabilidad de dicha matriz, probabilidad, intervalos de confianza y otros estadísticos.

La principal desventaja de estos métodos es que emplean el enfoque de asignación proporcional, por lo que únicamente son aplicables a redes con bajos niveles de congestión.

En las últimas dos décadas han aparecido numerosos estudios sobre esta temática (Cascetta y Nguyen, 1988; Hazelton, 2001; Li, 2005; Sánchez-Cambronero, 2008)

### 3.4.5 Modelos de mínimos cuadrados generalizados

Al igual que el modelo de máxima verosimilitud, el modelo de los mínimos cuadrados generalizados es otro estimador dentro de la teoría clásica de la inferencia estadística. Éste fue inicialmente propuesto para el problema de estimación de matrices O-D por Cascetta (1984). Proporciona una estimación de la matriz O-D a partir de un sistema lineal de ecuaciones estocásticas obtenido al combinar los valores de demanda con los de aforo. Como en los métodos anteriores, las variables se suponen según las expresiones (3.13) y (3.14). Supone que la varianza de los errores viene dada por:

$$\begin{aligned} Var[\varepsilon_{\bar{T}}] &= \Sigma_{\bar{T}} \\ Var[\varepsilon_{\bar{v}}] &= \Sigma_{\bar{v}} \end{aligned} \quad (3.20)$$

donde  $\Sigma_{\bar{T}}$  y  $\Sigma_{\bar{v}}$  son las matrices de varianzas-covarianzas. El modelo de los mínimos cuadrados generalizados puede ser expresado como:

$$Minimizar_{T,v} \frac{1}{2}(\bar{T} - T)^T \Sigma_{\bar{T}}^{-1} (\bar{T} - T) + \frac{1}{2}(\bar{v} - v(T))^T \Sigma_{\bar{v}}^{-1} (\bar{v} - v(T)) \quad (3.21)$$

En esta expresión se minimiza la suma de los cuadrados de las diferencias entre la matriz de viajes estimada y la de referencia, y entre los flujos obtenidos en el problema de asignación y los observados en los aforos, corregidos por las matrices de varianzas-covarianzas.

Una propiedad importante del método de los mínimos cuadrados generalizados es que permite de forma simple combinar las dos fuentes de información, matriz de demanda y volúmenes de aforos. Si algún elemento dentro de las matrices de varianzas-covarianzas está próximo a cero, reflejando una gran fiabilidad en la información observada, su inversa tendrá valores elevados. Así, el modelo le asignará un gran peso y dicha información será reproducida en la solución final.

Además de la independencia de hipótesis de distribuciones de probabilidad para las partes aleatorias, las buenas propiedades de las funciones formadas por sumas de cuadrados y su

relación con la distribución normal hacen que este método sea muy conveniente desde el punto de vista estadístico.

Un problema que presenta este método es que bajo determinadas circunstancias produce celdas negativas en la matriz de viajes, especialmente cuando la matriz de referencia tiene valores próximos a cero. Por tanto, es necesario agregar restricciones de no negatividad para los elementos de la matriz de viajes.

Puede observarse que para la formulación de este modelo no es necesaria la hipótesis de ninguna distribución de probabilidad para las componentes aleatorias de las variables. Sólo es requerida la existencia de las matrices de varianzas-covarianzas. En ausencia de matrices fiables generalmente se emplean matrices unitarias o matrices diagonales eliminando las covarianzas entre elementos. Esto se aconseja cuando el cálculo de las covarianzas es muy complejo y/o es preciso disminuir los requerimientos de memoria y tiempos de computación. En este caso la expresión (3.21) se simplificaría quedando:

$$\underset{T,v}{\text{Minimizar}} \quad \frac{1}{2} \sum_{ij} \frac{(\bar{T}_{ij} - T_{ij})^2}{\text{Var}[\varepsilon_{\bar{T}_{ij}}]} + \frac{1}{2} \sum_a \frac{(\bar{v}_a - v_a(T))^2}{\text{Var}[\varepsilon_{\bar{v}_a}]} \quad (3.22)$$

Este método ha sido profusamente empleado para el ajuste de matrices ya que es posible aplicarlo tanto para problemas con asignación proporcional como no proporcional (Yang et al., 1992). Aunque los algoritmos que emplean asignación no proporcional son heurísticos, éstos permiten abordar redes de grandes dimensiones con costos computacionales aceptables.

Si se supone que los términos de error de los flujos de aforos y la matriz de viajes ( $\varepsilon_{\bar{v}}$  y  $\varepsilon_{\bar{T}}$ ) siguen una distribución normal multivariante, puede demostrarse que el método de mínimos cuadrados generalizados coincide con los métodos de máxima verosimilitud y de inferencia bayesiana. Bell (1984) muestra como el método de mínima información de Van Zuylen y Willumsen (1980) se aproxima también al de mínimos cuadrados generalizados si los volúmenes de aforo son recogidos con alto grado de fiabilidad.

La semejanza entre los tres métodos estadísticos según Cascetta (2009) es únicamente formal ya que la matriz O-D de referencia tiene diferentes interpretaciones. Para los métodos de mínimos cuadrados generalizados y máxima verosimilitud la matriz O-D de partida es un estimado directo de la matriz de viajes mientras que el método de inferencia bayesiana considera que los elementos de la matriz O-D de partida son variables aleatorias con una distribución inicial dada.

Una discusión más amplia de esta metodología puede consultarse en Cascetta (1984), Cascetta y Nguyen (1988), Bell (1991), Bierlaire y Toint (1995).

### 3.5 MÉTODOS DE PROGRAMACIÓN BINIVEL

La literatura técnica referida a la estimación de matrices O-D es muy extensa, existe multitud de formulaciones que tratan de encontrar la matriz que con mayor exactitud reproduce los valores de aforo y con mayor fiabilidad tiene en cuenta el comportamiento de los usuarios en la red. Cada formulación además tiene asociada diferentes variantes según el tipo de asignación empleada. Para todas estas combinaciones formulación – tipo asignación, a su vez, han sido publicadas distintas metodologías de resolución.

Este apartado se centra exclusivamente en la descripción de los modelos de programación binivel formulados para resolver el problema de estimación de matrices O-D. Ésta es la formulación más extendida gracias a las propiedades teóricas y computacionales que presenta.

Desde el punto de vista teórico esta metodología emplea asignación no proporcional por lo que es apta para abordar redes de transporte con problemas de congestión y ofrece gran flexibilidad ya que puede ser implementada bajo cualquier método de asignación ya sea equilibrio de usuario determinista, estocástico, equilibrio de sistema, etc. A su vez, empleando esta metodología la solución generada (matriz O-D) es más consistente con los valores de aforo y con las hipótesis de comportamiento de los usuarios supuestas.

Desde el punto de vista de la práctica, los algoritmos que implementan modelos de programación binivel para la estimación de matrices O-D presentan la ventaja de ser computacionalmente abordables. Éstos son aplicados a redes de gran escala, obteniéndose resultados razonables con una significativa mejora en la bondad del ajuste. Las técnicas, o soluciones algorítmicas, utilizadas para la resolución de modelos binivel planteados hasta el momento por la comunidad científica, en el terreno de la ingeniería del transporte, son heurísticas y no tienen en cuenta la falta de diferenciabilidad del problema.

Yang et al. (1992) de acuerdo con Bard (1988) define el problema binivel como sigue:

$$\begin{aligned} & \underset{x}{\text{Minimizar}} && F(x, y) \\ & \text{s.a.} && G(x, y) \leq 0 \end{aligned} \tag{3.23}$$

donde  $y$  es obtenida resolviendo el problema (3.24):

$$\begin{aligned} & \underset{y}{\text{Minimizar}} && f(x, y) \\ & \text{s.a.} && g(x, y) \leq 0 \end{aligned} \tag{3.24}$$

En el nivel superior, la matriz de viajes define las variables de decisión que se modifican para minimizar la función objetivo y están sujetas a las restricciones de no negatividad y las condiciones de asignación en equilibrio de la matriz O-D correspondiente. Estas condiciones de asignación en equilibrio son obtenidas resolviendo el nivel inferior donde las variables de decisión son el flujo en los arcos.

Para definir un problema binivel es necesario especificar qué modelo, o formulación para la estimación de matrices O-D, se implementa en el nivel superior y qué método se emplea para asignar la matriz de viajes a la red en el nivel inferior. La experiencia sugiere que el problema de programación binivel, en el que se resuelve el modelo de los mínimos cuadrados generalizados en el nivel superior y la asignación en equilibrio de usuario determinista en el inferior, converge en la práctica.

La resolución de estos modelos se aborda a través de algoritmos iterativos. Cada iteración de este problema consta de dos fases; en la primera fase se resuelve el problema de nivel inferior (problema de asignación en equilibrio) y en la segunda se aborda el problema de nivel superior (problema de optimización) considerando que las variables calculadas en el nivel inferior permanecen constantes. La información aportada por la matriz O-D de referencia es tomada como una solución inicial en el problema de estimación de matrices y es ajustada o modificada con el fin de reproducir los aforos de tráfico.

Las variables de decisión del nivel inferior son funciones implícitas de los viajes entre cada pareja O-D y son obtenidas en el procedimiento de asignación de tráfico.

$$v(g) = P(g)g \tag{3.25}$$

Esto significa que el problema de estimación de matrices (nivel superior) puede ser formulado únicamente en términos de variables de elementos de la matriz O-D ( $g_i$ ) utilizando funciones de distancia, véase (3.4).

Una característica particularmente importante dentro de la programación binivel es que desde el nivel superior no se puede intervenir en las decisiones del nivel inferior.

Las aportaciones más importantes en la programación binivel aplicada a la estimación de matrices O-D se desarrolla durante la década de los 90.

Fisk (1988, 1989) combina el modelo de máxima entropía con restricciones del tipo de equilibrio de usuario, requiriendo que sólo un subconjunto de arcos sea aforado. Para ello formula el problema con desigualdades variacionales.

Spiess (1990) sugiere un método heurístico que, aunque tiene débil base teórica, proporciona buenos resultados prácticos. Según la expresión (3.4) supone  $\gamma_1=0$ . Spiess propone un método de estimación de mínimos cuadrados y aproxima el gradiente de dicha función objetivo, respecto de la matriz O-D, bajo la hipótesis de que la matriz de proporciones o probabilidades  $P(g)$  es localmente constante. Cabe destacar la importancia práctica de esta aportación como punto de partida para formulaciones más elaboradas en los modelos de asignación no proporcional, ya que permite su aplicación a redes de transporte de gran tamaño. Los resultados presentados corresponden a un ejemplo de prueba con la base de datos de la ciudad Winnipeg (154 zonas y 2983 arcos) incluida en el paquete EMME/2 y a estudios en Basilea (330 zonas y 220 arcos), Berna (226 zonas y 2676 arcos), Suecia (522 zonas y 3879 arcos) y Finlandia (469 zonas y 12476 arcos).

Yang et al. (1992) muestran que es posible integrar las funciones de mínimos cuadrados generalizados o máxima entropía con un problema de asignación en equilibrio de usuario siguiendo un problema de optimización binivel. Yang (1995) encuentra una alternativa interesante al emplear la matriz de sensibilidades  $\nabla A(g)$  en el nivel superior en vez de la matriz de proporciones  $P(g)$ , y calcula el gradiente de la función objetivo respecto de la matriz O-D utilizando un método de análisis de sensibilidad (Tobin y Friesz, 1988); con él obtiene las derivadas parciales de los flujos en arcos, en equilibrio de usuario, respecto de los viajes en los pares O-D, (Ver además Yang y Yagar (1995)).

Drissi-Kaïtouni y Lundgren (1992) investigan diferentes gradientes. Uno de sus métodos, como Spiess (1990), considera que la función objetivo está compuesta únicamente por la función de distancia  $F_2$  ( $\gamma_1=0$  según (3.4)). Para mejorar el gradiente encontrado por Spiess utilizan información de segundo orden obtenida del Hessiano de la función objetivo. Los autores ponen especial énfasis en la calidad de la dirección de búsqueda y no en la eficiencia computacional del algoritmo. Según muestra Patriksson (2004) este método proporciona una aproximación del valor de la derivada direccional que es correcto cuando el flujo en arcos es diferenciable.

Florian and Chen (1993) reformulan el problema binivel dentro de un único nivel utilizando el concepto de función marginal. Estos autores proponen un método heurístico del tipo Gauss-Seidel.

Chen (1994) y Doblas y Benítez (2005) de forma independiente han propuesto resolver el problema mediante la técnica de la lagrangiana aumentada. Chen (1994), busca justificaciones teóricas que expliquen los resultados de Spiess y propone algoritmos alternativos para la resolución del problema. Estos algoritmos requieren a priori del conocimiento de todos los caminos empleados por los diferentes pares O-D, y por tanto, son aplicables a redes de pequeña escala. Doblas y Benítez (2005) tratan de disminuir la distorsión de la matriz de viajes incluyendo restricciones de sus elementos dentro de la función objetivo. Este método es eficiente y permite abordar redes de grandes dimensiones ya que minimiza la cantidad de información almacenada.

Denault (1994) propone una formulación basada en el método del gradiente, obtenida a partir del análisis de sensibilidad de Tobin y Friesz (1988), y la compara con el método presentado por Chen (1994). Además compara el gradiente propuesto, con el calculado por Drissi-Kaïtouni y Lundgren (1992), concluyendo que este último tiene un uso más sencillo y proporciona resultados más confiables. Estos métodos son aplicados a redes de pequeña y mediana escala.

Sherali et al. (1994) tratan la programación binivel incluyendo la idea de la estimación del flujo de los caminos entre los diferentes pares O-D. Para este enfoque se requiere el aforado de todos los arcos de la red. Posteriormente Sherali et al. (2003) relajan este requisito y proponen un modelo basado en un conjunto incompleto de conteos de tráfico. En ambos trabajos, las metodologías propuestas se aplican a redes sencillas.

Maher y Zhang (1999) proponen un procedimiento iterativo donde, en cada iteración, se calcula una nueva matriz O-D a través de los siguientes pasos: *i*) en un primer paso, suponiendo localmente constante la matriz de proporciones  $P(g)$ , se llega a una nueva matriz O-D de prueba. Esta matriz es empleada para definir una dirección de búsqueda. Asignando dicha matriz a la red, los autores obtienen una aproximación lineal de cómo la matriz de proporciones  $P(g)$  será afectada a lo largo de la dirección de búsqueda; *ii*) en un segundo paso, se finaliza la iteración con un procedimiento de búsqueda lineal.

Cascetta y Postorino (2001), Maher et al. (2001) y Yang et al. (2001) difieren de los anteriores métodos en que emplean equilibrio de usuario estocástico en el nivel inferior.

Codina y Barceló (2004) emplean un método de subgradiente para los problemas de optimización no diferenciables donde no es necesario ningún dato proveniente de cualquier análisis de sensibilidad.

Lo y Chan (2003) emplean en el nivel superior una estimación estadística de los elementos de la matriz O-D junto con los parámetros de dispersión  $\theta$ . El nivel inferior emplea un modelo de asignación logit.

Lundgren y Peterson (2008) parten de la idea inicial presentada por Drissi-Kaïtouni y Lundgren (1992). Estos autores muestran que bajo ciertas hipótesis es posible generar el gradiente real; así si en el punto actual la función objetivo es diferenciable, entonces el Jacobiano es exacto y este enfoque permite definir un gradiente. Sin embargo, la función objetivo no es diferenciable en todos los puntos por lo que no puede asegurarse que las derivadas direccionales constituyan un gradiente. En este caso, este enfoque proporciona una aproximación del gradiente.

Para más información relativa a la mayoría de los métodos expuestos, se recomienda consultar la recopilación de metodologías desarrollada por Abrahamsson (1998).

### 3.6 METODOLOGÍA DE RESOLUCIÓN PROPUESTA

La formulación general para el problema del ajuste de matrices ya se introdujo en el apartado 3.3 de este capítulo y es la siguiente:

$$\begin{aligned}
 \underset{g,v}{\text{Minimizar}} \quad & Z(g) = \gamma_1 F_1(g, \bar{g}) + \gamma_2 F_2(v(g), \bar{v}) \\
 \text{s.a.} \quad & v = \Lambda \cdot h \\
 & g = \Gamma \cdot h \\
 & g, v, h \geq 0
 \end{aligned} \tag{3.26}$$

donde  $\Lambda$  y  $\Gamma$  son las matrices de incidencia arcos-caminos y pares O-D-caminos, respectivamente, definidas en la sección 2.3.

La componente  $i$  del gradiente de  $Z(g)$  se expresa como:

$$\nabla_i Z(g) = \frac{\partial Z(g)}{\partial g_i} = \gamma_1 \frac{\partial F_1(g)}{\partial g_i} + \gamma_2 \frac{\partial F_2(v(g))}{\partial v_a} \cdot \frac{\partial v_a(g)}{\partial g_i} \tag{3.27}$$

Se observa que, conociendo  $\frac{\partial v}{\partial g}$ , es posible emplear cualquier tipo de función objetivo compuesta por  $F_1$  y  $F_2$ , siempre que cumpla la condición de que sea derivable respecto de  $g$  o  $v(g)$ , respectivamente.

La matriz Jacobiana  $\frac{\partial v}{\partial g}$ , en adelante Jacobiano, es una magnitud intrínseca de la red de transporte y del valor de la matriz de viajes en la situación actual. No depende en ningún caso de la expresión de la función objetivo.

Las variables que generalmente se emplean en los problemas de ajuste son los elementos de la matriz O-D,  $g_i$ , y el flujo en los arcos aforados,  $v_a(g)$ . Esta última variable puede sustituirse a través de las funciones de demora por los costes en arcos,  $s_a(v_a)$ , y el proceso de ajuste obtendrá soluciones similares ya que existe una relación biunívoca.

Aunque puede utilizarse independientemente una expresión u otra, es recomendable emplear el coste en arcos. Esto se debe a que, en la práctica, el coste en un arco es una función monótonamente creciente respecto de su demanda de viajes. El volumen en arco, una vez se ha superado el nivel de saturación, pierde el carácter creciente. En los puntos de aforo pueden observarse bajos valores de volumen en arcos; esto ocurre principalmente por dos razones: *i*) pasan efectivamente muy pocos vehículos por dicho arco porque hay poca demanda, *ii*) el arco está muy saturado y los vehículos se mueven muy despacio. En el segundo caso, el volumen en arcos observado puede interpretarse de forma errónea mientras que el coste en arcos no. En esta tesis se emplea como variable el flujo en arcos y se comprueba, a través de su coste, la saturación de cada arco aforado.

### 3.6.1 Expresión analítica para el cálculo del paso

La metodología desarrollada para resolver el problema de ajuste será el método del gradiente. El cálculo del paso óptimo puede llevarse a cabo resolviendo un problema de optimización unidimensional o empleando una expresión analítica. En este caso, al conocer la derivada de la función objetivo, se calculará el paso óptimo mediante una expresión analítica.

Siendo  $Z(g)$  la función objetivo empleada en el ajuste, la matriz de viajes que minimiza dicha función es:

$$g^{n+1} = g^n + \lambda \left. \frac{\partial Z}{\partial g} \right|_n \quad (3.28)$$

siendo  $\lambda$  el paso, el cual se quiere determinar de forma que garantice el máximo descenso de la función objetivo.

Conociendo la matriz  $g^{n+1}$ , el nuevo vector de flujos en arcos, desarrollando linealmente respecto al punto actual:

$$v^{n+1} = v^n + \frac{\partial v}{\partial g_n} \left( g^{n+1} - g^n \right) \quad (3.29)$$

Y aplicando (3.28) se tiene:

$$v^{n+1} = v^n + \lambda \frac{\partial v}{\partial g_n} \left| \frac{\partial Z}{\partial g_n} \right| = v^n + \lambda \tilde{v} \quad (3.30)$$

donde se ha definido un vector de volúmenes en arco  $\tilde{v}$  que representa la dirección de actualización de los volúmenes en arco de la iteración  $n$  a la siguiente  $n+1$ .

El valor de  $\lambda$  se obtendrá imponiendo la condición de un máximo descenso en la función objetivo  $Z$ :

$$\frac{\partial Z}{\partial \lambda} = \frac{\partial Z}{\partial v_n} \left| \frac{\partial v}{\partial \lambda} \right| = 0 \quad (3.31)$$

De la expresión (3.30) se despeja:

$$\frac{\partial v}{\partial \lambda} \Big|_n = \tilde{v} \quad (3.32)$$

Llamando  $f(\lambda) = \frac{\partial Z}{\partial v_n} \Big|_n$  y realizando un desarrollo en serie de  $f(\lambda)$  para  $\lambda = 0$ , donde se obvian los elementos de segundo orden o superior, se obtiene:

$$f(\lambda) \approx f(0) + \frac{\partial f(\lambda)}{\partial \lambda} \Big|_{\lambda=0} \cdot \lambda = f(0) + f'_0 \cdot \lambda \quad (3.33)$$

Sustituyendo las expresiones (3.32) y (3.33) en (3.31):

$$\frac{\partial Z}{\partial \lambda} = 0 \rightarrow (f(0) + f'_0 \cdot \lambda) \tilde{v} = 0 \quad (3.34)$$

Despejando el valor de  $\lambda$ :

$$\lambda^* = -\frac{f(0) \cdot \tilde{v}}{f'_0 \cdot \tilde{v}} \quad (3.35)$$

donde  $\tilde{v} = \frac{\partial v}{\partial g} \Big|_n \frac{\partial Z}{\partial g} \Big|_n$  y el valor del gradiente  $\frac{\partial Z}{\partial g} \Big|_n = \frac{\partial Z}{\partial v} \Big|_n \cdot \frac{\partial v}{\partial g} \Big|_n$ .

### 3.6.2 Particularización de las expresiones

En los algoritmos propuestos a lo largo de esta tesis se han considerado dos tipologías de la función objetivo  $Z$ , mínimos cuadrados generalizados y maximización de la entropía. En ambas se ha establecido un valor de  $\gamma_1 = 0$ , según la ecuación (3.4), aunque sin pérdida de generalidad. Para incluir un término que penalice la distorsión de la matriz previa se propone seguir la metodología de Doblas y Benítez (2005), donde el control de la distorsión se efectúa introduciendo restricciones adicionales al problema. En la Tabla 3-1 se resumen las expresiones empleadas.

Mínimos Cuadrados Generalizados	Maximización Entropía
$Z = \frac{1}{2}(v^n - \bar{v})^2$	$Z = v^n \cdot \ln\left(\frac{v^n}{\bar{v}}\right) - v^n + \bar{v}$
$f(0) = v^n - \bar{v}$	$f(0) = \ln\left(\frac{v^n}{\bar{v}}\right)$
$f'_0 = \tilde{v}$	$f'_0 = \frac{\tilde{v}}{v^n}$
$\frac{\partial Z}{\partial g} \Big _n = (v^n - \bar{v})^T \cdot \frac{\partial v}{\partial g} \Big _n$	$\frac{\partial Z}{\partial g} \Big _n = \ln\left(\frac{v^n}{\bar{v}}\right)^T \cdot \frac{\partial v}{\partial g} \Big _n$
$\tilde{v} = \frac{\partial v}{\partial g} \Big _n \cdot (v^n - \bar{v})^T \cdot \frac{\partial v}{\partial g} \Big _n$	$\tilde{v} = \frac{\partial v}{\partial g} \Big _n \cdot \ln\left(\frac{v^n}{\bar{v}}\right)^T \cdot \frac{\partial v}{\partial g} \Big _n$
$\lambda^* = -\frac{(v^n - \bar{v}) \cdot \tilde{v}}{(\tilde{v})^2}$	$\lambda^* = -\frac{\ln\left(\frac{v^n}{\bar{v}}\right) \cdot \tilde{v}}{\frac{\tilde{v}}{v^n} \cdot \tilde{v}}$

Tabla 3-1. Expresiones empleadas en el problema de estimación de matrices.

### 3.7 CÁLCULO DEL JACOBIANO $\frac{\partial v_a(g)}{\partial g_i}$

Recapitulando de anteriores apartados, la formulación binivel hace que los modelos de estimación para los problemas de ajuste de matrices sean ventajosos en tres aspectos:

- Siempre trabajan con soluciones factibles. La factibilidad del problema está garantizada incluso si los datos de aforo son inconsistentes.
- Pueden trabajar con un subconjunto de arcos aforados.
- Las proporciones de viaje son determinadas de forma endógena. Así la consistencia entre el flujo en arcos en equilibrio y la matriz O-D es simultánea.

Los problemas de programación binivel son generalmente difíciles de resolver ya que la evaluación de la función objetivo en el nivel superior requiere de la solución de un problema de optimización en el nivel inferior. Además, la formulación binivel presenta un carácter no convexo debido a que el nivel inferior tiene un carácter no lineal en condiciones de congestión. Esta no convexidad provoca la existencia de soluciones locales y hace muy compleja la búsqueda del óptimo global del problema (Friesz et al. (1990); Yang y Yagar et al. (1994)). A su vez, Yang (1995) escribe que, sin embargo, la función objetivo del nivel superior es fuertemente convexa con respecto de las variables  $g$  y  $v$ , por lo que un mínimo local podría ser probablemente un mínimo global. Esto particularmente podría darse si la matriz O-D de partida es muy cercana a la real.

Suponiendo que la asignación implementada en el nivel inferior sigue el primer principio de Wardrop y las funciones de coste en todos los arcos de la red  $s(v)$  son monótonas y diferenciables, existe una única distribución de flujos en arcos  $v(g)$  para cada matriz de demanda  $g$ . El éxito de la estimación de matrices dependerá enormemente en cómo se evaluará la función  $v(g)$ . En otras palabras, dependerá en gran medida de cómo se predecirán los cambios de los flujos en arcos para una iteración de ajuste de la matriz O-D.

En apartados anteriores se ha hecho referencia a la no linealidad del problema que forma el nivel inferior; esto se debe a que la distribución de flujos en arcos (3.25) es no lineal y no existe de forma explícita ninguna función que la represente. Esta no linealidad hace difícil predecir la evolución de la distribución de flujos en arcos desde el nivel superior. Para superar esta dificultad diferentes metodologías formulan una aproximación lineal de  $v(g)$  en cada solución intermedia alcanzada durante la resolución del problema ( $g^*$ ,  $v(g^*)$ ).

Los algoritmos que resuelven los problemas de programación binivel pueden dividirse en dos tipologías dependiendo del tratamiento que le den a la distribución del flujo en arcos en el nivel inferior:

- El primer tipo de algoritmos supone que los flujos calculados para la matriz O-D actual se mantienen proporcionalmente constantes durante la iteración. En este caso no es necesario predecir ninguna evolución de dichos flujos con respecto a la matriz de viajes. Así la función objetivo se simplifica y el problema de optimización que se resuelve es convexo con restricciones lineales.
- El segundo tipo supone que la distribución de flujos varía de forma lineal con la matriz de viajes. El problema de optimización a resolver vuelve a ser convexo aunque su función objetivo es más compleja ya que incluye el término  $v(g)$ . Estos algoritmos son más eficientes que los anteriores y, en la mayoría de los casos, arrojan resultados más precisos.

Considerando que los flujos en arcos pueden ser representados mediante una aproximación lineal en el punto actual, es posible calcular el gradiente de la función objetivo del nivel superior. Utilizando dicho gradiente puede obtenerse una dirección de búsqueda para un algoritmo de descenso que mejore la solución del problema.

La función objetivo  $Z(g)$  puede ser dividida en dos partes según dependa de  $g$  o  $v(g)$ , expresión (3.4):

$$Z(g) = \gamma_1 F_1(g) + \gamma_2 F_2(v(g))$$

su gradiente  $\nabla Z(g)$  puede ser expresado como

$$\begin{aligned} \nabla Z(g) &= \frac{\partial Z(g)}{\partial g_i} = \gamma_1 \frac{\partial F_1(g)}{\partial g_i} + \gamma_2 \frac{\partial F_2(v(g))}{\partial g_i} = \\ &= \gamma_1 \frac{\partial F_1(g)}{\partial g_i} + \gamma_2 \frac{\partial F_2(v(g))}{\partial v_{\bar{a}}} \cdot \frac{\partial v_{\bar{a}}(g)}{\partial g_i} \end{aligned} \quad (3.36)$$

donde  $i$  representa un elemento del conjunto de pares O-D  $P$  y  $\bar{a}$  pertenece al subconjunto de arcos aforados  $\bar{A}$ .

Las derivadas parciales  $\partial F_1(g)/\partial g_i$  y  $\partial F_2(g)/\partial v_a$  son fáciles de calcular para funciones objetivo y funciones de costes en arcos adecuadas. La dificultad se encuentra en el cálculo del Jacobiano  $J$ :

$$J = \frac{\partial v_a(g)}{\partial g_i} \quad \forall a \in A, i \in P \quad (3.37)$$

donde  $J$  representa la variación del flujo en arcos respecto de la variación de los elementos de la matriz de demanda y  $a$  es un elemento del conjunto de arcos de la red  $A$ . Anteriormente se ha apuntado la no linealidad y falta de forma funcional del problema, esto hace que la expresión de  $v(g)$  sea una aproximación lineal y, por tanto, el Jacobiano sea también aproximado. Esta aproximación le da un carácter heurístico a todos los algoritmos implementados hasta el momento.

Spieß (1990) realiza el ajuste de la matriz de viajes utilizando el método del gradiente. Para llevar a cabo el cálculo del gradiente formula el flujo en los arcos en función del flujo en caminos:

$$v_a = \sum_{i \in P} \sum_{k \in K_i} \delta_{ak} h_k \quad \forall a \in A \quad (3.38)$$

donde  $\delta_{ak}$  tiene valor 1 cuando el arco  $a$  está incluido dentro del camino  $k$ , y 0 en otro caso.

Definiendo la probabilidad de un camino  $p_k$  como el flujo de dicho camino dividido entre el número de viajes del par que conecta ( $p_k = \frac{h_k}{\sum_{i \in P, l \in K_i} h_l}$ ), la anterior expresión puede escribirse

como:

$$v_a = \sum_{i \in P} g_i \sum_{k \in K_i} \delta_{ak} p_k \quad (3.39)$$

Suponiendo que las probabilidades de los caminos son localmente constantes, Spieß obtiene una expresión para el Jacobiano.

$$\frac{\partial v_a}{\partial g_i} = \sum_{k \in K_i} \delta_{ak} p_k \quad \forall a \in A, i \in P \quad (3.40)$$

Yang et al. (1992) resuelven el nivel inferior siguiendo un método de asignación en equilibrio de usuario determinista y para el nivel superior proponen dos expresiones. La primera expresión

llamada BOM1 se reduce a resolver el problema de los mínimos cuadrados generalizados (Cascetta, 1984):

$$\mathbf{g}^{k+1} = \left( U^{-1} + P(\mathbf{g}^k) \cdot V^{-1} \cdot P(\mathbf{g}^k) \right)^{-1} \left( U^{-1} \cdot \mathbf{g}^k + P(\mathbf{g}^k) \cdot V^{-1} \cdot \mathbf{v}^k \right) \quad (3.41)$$

donde:  $U$  y  $V$  son matrices de ponderación, éstas pueden ser interpretadas como matrices de varianzas-covarianzas de los errores de  $\mathbf{g}$  y  $\mathbf{v}$ , respectivamente.  $P(\mathbf{g}^k)$  es la matriz de probabilidades o proporciones en arcos.

La segunda expresión (BOM2) según los modelos de máxima entropía (Willumsen, 1984):

$$g_i^{k+1} = g_i^k \exp \left( - \sum_{a \in \bar{A}} \lambda_a p_{ai}(\mathbf{g}^k) \right) \quad \forall i \in P \quad (3.42)$$

Para llevar a cabo el cálculo del gradiente expresa el flujo en los arcos en función de la matriz O-D como:

$$v(\mathbf{g}) = P(\mathbf{g}) \mathbf{g}, \quad \text{o} \quad v_a(\mathbf{g}) = \sum_{i \in P} p_{ai}(\mathbf{g}) g_i \quad \forall a \in \bar{A} \quad (3.43)$$

donde  $p_{ai}(\mathbf{g})$  son las proporciones de viajes para el par O-D  $i$  que pasan por el arco  $a$ . En este caso, Yang et al. suponen que las proporciones de viaje dependen de la matriz O-D. Así se tiene el siguiente Jacobiano:

$$\frac{\partial v_a(\mathbf{g})}{\partial g_i} = p_{ai}(\mathbf{g}) + \sum_{i \in P} g_i \frac{\partial p_{ai}(\mathbf{g})}{\partial g_i} \quad (3.44)$$

Sobre el segundo término de la expresión (3.44), Yang et al. (1992) reconocen que no puede ser correctamente evaluado. Además añade que si la matriz de partida es muy cercana a la matriz verdadera, el dominio de búsqueda estará limitado a una cierta vecindad de la solución óptima

por lo que es de esperar que el valor de  $\frac{\partial p_{ai}(\mathbf{g})}{\partial g_i}$  sea pequeño. Por tanto, el segundo término se

convierte en insignificante. Con esta hipótesis, Yang et al. (1992) obtienen la misma expresión del Jacobiano alcanzada por Spiess (1990).

Denault (1994) y Yang (1995) afrontan el cálculo del Jacobiano aplicando el análisis de sensibilidad de Tobin y Friesz (1988). Dicho análisis de sensibilidad proporciona una expresión

analítica del Jacobiano de los flujos en los arcos respecto de la matriz O-D que puede emplearse para el cálculo del gradiente.

Tobin y Friesz (1988) obtienen expresiones analíticas sobre la variación del flujo en arcos con respecto a la variación del coste de cada arco y con respecto a la variación de la matriz de viajes. El principal fin es permitir realizar análisis “what if” evitando tener que resolver en cada paso un problema de asignación en equilibrio que requiere un gran esfuerzo computacional. Si los costes en arcos son estricta y monótonamente crecientes con respecto a su flujo, el problema de asignación tiene solución única en arcos aunque en términos de flujos en caminos no se da esta unicidad. Tobin y Friesz demuestran que, si los costes en arcos son estrictamente monótonos, las expresiones propuestas pueden ser calculadas incluso teniendo diferentes soluciones en caminos. Formulan el problema en término de flujo en caminos aunque luego hacen la conversión a flujos en arcos. El segundo término de la expresión que presentan, correspondiente a la variación del flujo en arcos respecto de la matriz O-D, es el siguiente:

$$\nabla_{\varepsilon} f^0 = \Delta^0 \left[ \Delta^{0T} \nabla c(f^*, 0) \Delta^0 \right]^{-1} \Lambda^{0T} \left[ \Lambda^0 \left[ \Delta^{0T} \nabla c(f^*, 0) \Delta^0 \right]^{-1} \Lambda^{0T} \right]^{-1} \nabla_{\varepsilon} T(0) \quad (3.45)$$

Renombrando la expresión con la nomenclatura que se emplea en la tesis, donde:

$$\Delta^0 = \Lambda \rightarrow \text{Matriz incidencia arcos-caminos } (v = \Lambda \cdot h)$$

$$\Lambda^0 = \Gamma \rightarrow \text{Matriz incidencia pares O-D-caminos } (g = \Gamma \cdot h)$$

$$\nabla c(f^*, 0) = S \rightarrow \text{Jacobiano de costes en arcos } (\partial s / \partial v)$$

y definiendo una nueva variable como  $B = \Delta^{0T} \nabla c(f^*, 0) \Delta^0 = \Lambda^T S \Lambda$  donde  $B$  representa la variación del coste de los caminos respecto de su flujo. La expresión de Tobin y Friesz (1988) puede escribirse como sigue:

$$J = \frac{\partial v}{\partial g} = \Lambda B^{-1} \Gamma^T \left( \Gamma B^{-1} \Gamma^T \right)^{-1} \quad (3.46)$$

Esta formulación, en general, no tiene solución analítica. Según Bell y Iida (1997) y Patriksson (2004), si el número de rutas en la red es mayor al número de arcos, el caso más habitual en redes de mediano y gran tamaño, la matriz  $B$  es singular y por tanto no invertible.

Drissi-Kaïtouni y Lundgren (1992) y Lundgren y Peterson (2008) llevan a cabo la implementación de un gradiente que tiene como punto de partida el enfoque de Spiess (1990).

Estos autores reemplazan la aproximación lineal presentada por Spiess por una aproximación cuadrática. El Jacobiano se calcula como sigue. Para cada par O-D  $(i, j)$ , la derivada direccional de  $v$  en el punto actual  $v^*$  y en la dirección unitaria  $e_{ij}$  se obtiene de forma aproximada resolviendo el siguiente problema:

$$\begin{aligned} \underset{d}{\text{Minimizar}} \quad & \frac{1}{2} d^T S d \\ \text{s.a.} \quad & \Gamma^T x = e_{ij} \\ & d = \Lambda x \end{aligned} \tag{3.47}$$

donde:  $d = \frac{\partial v}{\partial T_{ij}}$ ,  $x = \frac{\partial h}{\partial T_{ij}}$  y  $S$  es una matriz cuadrada cuyos elementos son  $S_{ab} = \frac{\partial s_a}{\partial v_b}$ .

Según Patriksson (2004), si la función objetivo es diferenciable en el punto actual, este enfoque aporta gradientes exactos. Drissi-Kaïtouni y Lundgren (1992) aplican el método presentado en redes pequeñas y medianas. Lundgren y Peterson (2008) presentan un método para redes de gran escala; según los autores, el esfuerzo extra computacional requerido, en comparación con el algoritmo de Spiess, es marginal.

### 3.8 FORMULACIÓN PROPUESTA PARA EL CÁLCULO DEL

$$\text{JACOBIANO } \frac{\partial v_a(g)}{\partial g_i}$$

Como punto de partida de esta tesis se emplea el trabajo de Spiess (1990). La importancia de la aportación de Spiess radica en haber sido la primera implementación capaz de estimar matrices O-D en redes reales de gran tamaño en el contexto de la asignación en equilibrio y de una forma sencilla y eficiente, es decir, con un coste computacional más que razonable en aplicaciones prácticas. El autor propone el método del gradiente proyectado para resolver el problema y lo implementa en el programa EMME.

La eficiencia computacional mostrada por la metodología de Spiess se debe al uso de un método de descenso que actualiza una matriz de partida en cada iteración. El principal coste de estos métodos se encuentra en el cálculo de la dirección de descenso. Aunque, en este caso, la obtención del gradiente que minimiza la función objetivo es muy simple y rápida. Está demostrado que dicha metodología mejora el valor de la función objetivo en cada iteración, es decir, disminuye la discrepancia entre el valor medido y modelado en los puntos de aforo.

Al ser un método de descenso hay que poner especial atención en la obtención de la matriz inicial. El problema de estimación de matrices O-D tiene múltiples óptimos locales por lo que la solución final dependerá del punto de partida. Considerando que la matriz con la que parte el proyecto de planificación de transporte es bastante fiable, será muy interesante implementar un algoritmo que encuentre una solución próxima a la de partida. En este sentido, el método de Spiess por lo general obtiene soluciones muy diferentes respecto de la de inicio. Según Doblas y Benítez (2005) el algoritmo podría llevar a una desviación no controlada de la información contenida en la matriz de partida ya que no existen criterios explícitos que limiten las variaciones durante el proceso de optimización.

Al no estar garantizada la unicidad de la solución en el problema de estimación de matrices, es posible encontrar múltiples matrices que cumplen los criterios impuestos en la optimización. La función objetivo formulada por Spiess se centra únicamente en disminuir la discrepancia de los aforos. Al no imponer más criterios, el cálculo de la dirección de descenso se simplifica y permite abordar problemas de gran escala. En cambio, el proceso de ajuste no asegura el mantenimiento de la estructura de la matriz O-D dentro de unas cotas. Así, siendo un algoritmo eficiente desde el punto de vista computacional, tiene la desventaja de reportar soluciones poco fiables.

La metodología de Doblas y Benítez (2005) ha sido ampliamente estudiada y desarrollada por el Grupo de Ing. e Inf. de los Transportes de la Universidad de Sevilla. Ésta es una mejora del método de Spiess. Doblas y Benítez se centran en preservar la estructura de la matriz previa conocida tratando de no disminuir la eficiencia computacional del algoritmo. Para ello proponen una formulación que combina el método del lagrangiano aumentado (o de los multiplicadores) con el método de aproximación lineal de Frank & Wolfe. La expresión que utilizan para el cálculo del gradiente está basada en la de Spiess (1990), añadiendo los términos correspondientes a las nuevas restricciones introducidas. Para la obtención de la expresión analítica establecen igualmente la hipótesis de proporciones localmente constantes. Esta formulación, además, permite aumentar el valor de las celdas O-D inicialmente nulas en la matriz previa, lo cual es un enfoque realista dada la gran cantidad de pares O-D que no presentan viajes después de la expansión de los datos recogidos en la encuesta domiciliaria.

Centrando la atención en la aproximación que realizan Spiess (1990) y Yang et al. (1992) en el cálculo de la dirección de descenso, puede afirmarse que dicha aproximación no es válida para los problemas de transporte que presentan congestión. Estas aproximaciones no tienen en cuenta las relaciones existentes entre los pares O-D a través de los arcos saturados. El gradiente empleado por Spiess y más tarde por Yang et al., aunque mejora la función objetivo formulada

(dada la indeterminación del problema no es tarea compleja), también imprime desplazamientos elevados en la matriz de viajes como se demuestra en apartados posteriores.

Esta tesis trata de mejorar la dirección de búsqueda para los métodos de descenso con el fin de alcanzar un óptimo local del problema cercano a la matriz de partida. Inicialmente no se consideran restricciones que limiten el desplazamiento de la matriz de viajes dado que se espera que la dirección de descenso sea capaz de alcanzar dicho óptimo local. Si se desea controlar la distorsión que experimenta la matriz O-D respecto de la matriz previa se propone seguir la metodología de Doblas y Benítez (2005), que puede implementarse de forma directa en los algoritmos propuestos sin cambios metodológicos importantes.

### 3.8.1 Críticas a la dirección de descenso de Spiess (1990) y Yang et al. (1992)

En apartados precedentes se discute sobre la semejanza de los Jacobianos obtenidos por los dos autores (expresión (3.40)):

$$\frac{\partial v_a}{\partial g_i} = \sum_{k \in K_i} \delta_{ak} p_k \quad \forall a \in A, i \in P$$

De la expresión (3.40) puede interpretarse que la variación del flujo de un arco  $a$  respecto de un par O-D  $i$  es igual a la proporción de viajes, respecto del total del par O-D  $i$ , que pasan por el arco  $a$  y pertenecen a una ruta de dicho par O-D. La siguiente ecuación, expresada en función de los flujos en caminos, es equivalente a la anterior (3.40).

$$\frac{\partial v_a}{\partial g_i} = \frac{\sum_{k \in K_i} \delta_{ak} h_k}{\sum_{k \in K_i} h_k} \quad \forall a \in A, i \in P \quad (3.48)$$

#### 3.8.1.1 Ejemplo 1

Con el siguiente ejemplo se muestran las carencias de la expresión del gradiente empleado por Spiess (1990). Para ello se describe previamente la formulación empleada.

La matriz de demanda es actualizada en cada iteración siguiendo la dirección que minimiza la función objetivo  $Z(g)$ . Para un enfoque más realista Spiess realiza dicha actualización basándose en el cambio relativo de la matriz O-D.

$$\mathbf{g}^{k+1} = \mathbf{g}^k \left( 1 - \lambda^k \cdot \left. \frac{\partial Z(\mathbf{g})}{\partial \mathbf{g}} \right|_{\mathbf{g}^k} \right) \quad (3.49)$$

La función objetivo se define como la suma del cuadrado de la diferencia de los flujos modelados y aforados:

$$Z(\mathbf{g}) = \frac{1}{2} \sum_{a \in \bar{A}} (v_a - \bar{v}_a)^2 \quad (3.50)$$

El gradiente de la función objetivo es el siguiente:

$$\frac{\partial Z(\mathbf{g})}{\partial g_i} = \sum_{a \in \bar{A}} \frac{\partial v_a}{\partial g_i} (v_a - \bar{v}_a) \quad i \in P \quad (3.51)$$

Spiess (1990) a su vez proporciona una expresión para calcular la longitud del paso óptimo en cada iteración.

$$\lambda^k = \frac{\sum_{a \in \bar{A}} v'_a (\bar{v}_a - v_a)}{\sum_{a \in \bar{A}} v_a'^2} \quad (3.52)$$

donde  $v'_a = \frac{dv_a}{d\lambda} = \sum_{i \in P} \frac{dg_i}{d\lambda} \frac{\partial v_a}{\partial g_i} = - \sum_{i \in P} g_i \left( \sum_{k \in K_i} p_k \sum_{\bar{a} \in \bar{A}} \delta_{\bar{a}k} (v_{\bar{a}} - \bar{v}_{\bar{a}}) \right) \left( \sum_{k \in K_i} \delta_{ak} p_k \right)$ .

La expresión (3.52) equivale a la recogida en la Tabla 3-1 para el caso en el que la función objetivo del problema de ajuste es del tipo de mínimos cuadrados generalizados.

Conocida la formulación de Spiess (1990) necesaria para este pequeño ejemplo, se pasa a su resolución. En este caso, se dispone de una red simulada por un grafo formado por 7 arcos dirigidos y 4 centroides (A, B, C y D).

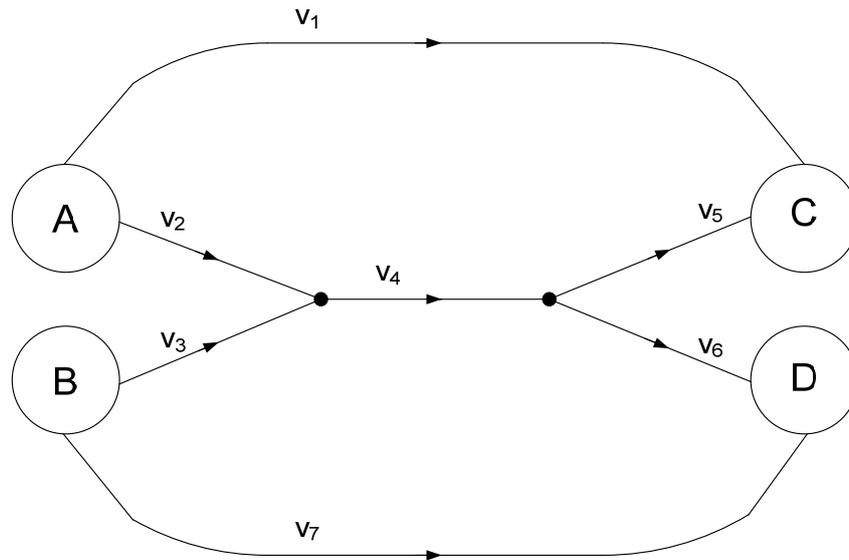


Figura 3-2. Ejemplo 1. Red simulada.

Los diferentes arcos de la red permiten conectar 4 pares O-D: A-C, A-D, B-C y B-D. Los pares A-C y B-D están conectados por 2 caminos cada uno, los dos restantes sólo están conectados por un solo camino.

Par O-D	Camino	Arcos que lo componen
A-C	$h_1$	$v_1$
	$h_2$	$v_2 - v_4 - v_5$
A-D	$h_3$	$v_2 - v_4 - v_6$
B-C	$h_4$	$v_3 - v_4 - v_5$
B-D	$h_5$	$v_3 - v_4 - v_6$
	$h_6$	$v_7$

Tabla 3-2. Ejemplo 1. Descripción de los caminos de cada par O-D.

Las capacidades de los arcos son 200 vehículos y su velocidad de flujo libre  $t_0 = \{3.5, 1, 1, 1, 1, 1, 10\}$ . La matriz O-D de partida se recoge en la Tabla 3-3:

Par O-D	Viajes
A-C	400
A-D	200
B-C	0
B-D	300

Tabla 3-3. Ejemplo 1. Matriz O-D de partida.

Resolviendo el problema de asignación de tráfico se alcanza la siguiente situación en equilibrio de usuario:

Flujo caminos	Flujo arcos	Coste caminos
$h_1 = 340$	$v_1 = h_1 = 340$	$C_1 = C_2 = U_{A-C}$
$h_2 = 60$	$v_2 = h_2 + h_3 = 260$	$C_3 = U_{A-D}$
$h_3 = 200$	$v_3 = h_4 + h_5 = 205$	$C_4 = U_{B-C}$
$h_4 = 0$	$v_4 = h_2 + h_3 + h_4 + h_5 = 465$	$C_5 = C_6 = U_{B-D}$
$h_5 = 205$	$v_5 = h_2 + h_4 = 60$	
$h_6 = 95$	$v_6 = h_3 + h_5 = 405$	
	$v_7 = h_6 = 95$	

Esta red sólo tiene un punto de aforo situado en el arco  $v_5$  donde se mide un total de 300 vehículos. Se observa en dicho arco una discrepancia de 240 vehículos, cuatro veces superior a lo modelado actualmente. Si se lleva a cabo un ajuste de la matriz de viajes según el método de Spiess se obtiene lo siguiente:

$\frac{\partial v_5}{\partial g_i}$	Valor
$i = \mathbf{A-C}$	0,15
$i = \mathbf{A-D}$	0
$i = \mathbf{B-C}$	0
$i = \mathbf{B-D}$	0

Tabla 3-4. Ejemplo 1.  $\frac{\partial v_5}{\partial g_i}$

$\frac{\partial Z(g)}{\partial g_i}$	Valor
$i = \mathbf{A-C}$	-36
$i = \mathbf{A-D}$	0
$i = \mathbf{B-C}$	0
$i = \mathbf{B-D}$	0

Tabla 3-5. Ejemplo 1.  $\frac{\partial Z(g)}{\partial g_i}$

Y por tanto la nueva matriz de viajes es igual a:

$$g^1 = g^0 \left( 1 - \lambda^* \cdot \frac{\partial Z(g)}{\partial g} \Big|_{g^0} \right) = \begin{pmatrix} 1840 \\ 200 \\ 0 \\ 300 \end{pmatrix}$$

Para resolver la expresión anterior es necesario calcular previamente:  $v_a' = 2160$  y  $\lambda^* = 0,1$ .

En la primera iteración se imprime un desplazamiento de la matriz de viajes muy grande aunque el valor de  $\lambda^*$  es pequeño. Spiess comenta en su artículo que el valor de  $\lambda^*$  debe ser seleccionado lo suficientemente pequeño para asegurar que el camino seguido por la matriz de demanda, a través del gradiente aproximado, sea suficientemente cercano al camino del gradiente verdadero. Como puede comprobarse en la primera iteración, el valor de  $\lambda^*$  es pequeño pero el desplazamiento de la matriz de viajes es elevado, luego no es cierto que dicho algoritmo asegure que no haya una desviación, respecto de la solución inicial, mayor que la necesaria.

Realizando 10 iteraciones el algoritmo alcanza la convergencia y obtiene la siguiente matriz de viajes ajustada.

Par O-D	Matriz Inicial	Matriz Final
A-C	400	767
A-D	200	200
B-C	0	0
B-D	300	300

Tabla 3-6. Ejemplo 1. Soluciones inicial y final.

La nueva situación de equilibrio es la siguiente:

Flujo caminos

$$h_1 = 467$$

$$h_2 = 300$$

$$h_3 = 200$$

$$h_4 = 0$$

$$h_5 = 64$$

$$h_6 = 236$$

Flujo arcos

$$v_1 = h_1 = 467$$

$$v_2 = h_2 + h_3 = 500$$

$$v_3 = h_4 + h_5 = 64$$

$$v_4 = h_2 + h_3 + h_4 + h_5 = 564$$

$$v_5 = h_2 + h_4 = 300$$

$$v_6 = h_3 + h_5 = 264$$

$$v_7 = h_6 = 236$$

Coste caminos

$$C_1 = C_2 = U_{A-C}$$

$$C_3 = U_{A-D}$$

$$C_4 = U_{B-C}$$

$$C_5 = C_6 = U_{B-D}$$

A continuación se representa la evolución seguida por el flujo en el arco  $v_5$  en las diferentes iteraciones del algoritmo.

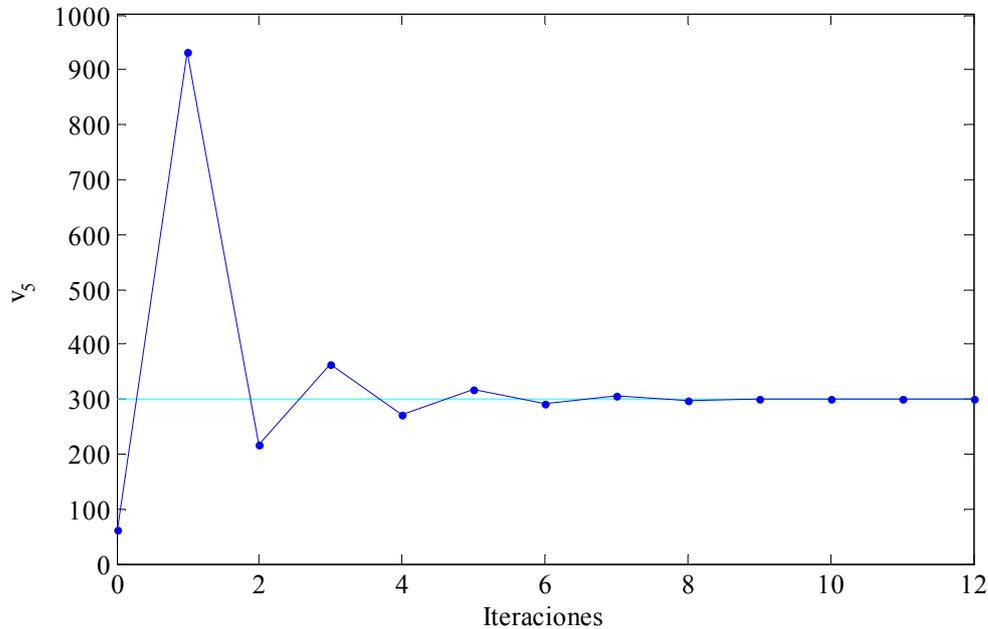


Figura 3-3. Ejemplo 1. Evolución del flujo en el arco  $v_5$ .

El excesivo desplazamiento de la matriz de viajes entre iteraciones consecutivas, sobre todo en las iteraciones iniciales, hace que no se verifique en ningún momento la hipótesis supuesta por Spiess, donde las probabilidades de los caminos  $p_k$  son localmente constantes.

En la solución final calculada por el método de Spiess se observa que el único par O-D modificado es el par  $g_{A-C}$ . Esto se produce porque sólo el par  $g_{A-C}$  tiene caminos que pasan por el arco aforado. El gradiente implementado por Spiess y Yang et al. sólo tiene en cuenta, durante el ajuste, los pares O-D interceptados por algún aforo, mientras que el resto de pares O-D no son ajustados. Ésta es la principal anomalía del gradiente ya que, en las redes de transporte con congestión, se supone que existe un equilibrio de usuario en el que interactúan los diferentes pares O-D a través de sus caminos. En una red congestionada en equilibrio, al modificar el número de viajes de un par O-D, el reparto de flujo entre las rutas del resto de pares puede verse modificado. Por tanto, un par O-D cuyo flujo no ha sido observado directamente en ningún punto de aforo puede alterar el reparto de flujo entre las rutas de otros pares. Así, dicho par O-D tiene el potencial para disminuir de forma indirecta la discrepancia de ciertas estaciones de aforo a través de dicha redistribución de flujos.

En el ejemplo anterior, aunque el gradiente de Spiess no lo pronostique, el par  $g_{B-D}$  puede influir sobre el volumen del arco  $v_5$ . Si se disminuye su valor en 150 viajes se obtiene la siguiente situación de equilibrio.

Par O-D	Viajes
A-C	400
A-D	200
B-C	0
B-D	150

**Tabla 3-7. Ejemplo 1. Modificación de la matriz O-D de partida.**

Flujo caminos	Flujo arcos	Coste caminos
$h_1 = 320$	$v_1 = h_1 = 320$	$C_1 = C_2 = U_{A-C}$
$h_2 = 82$	$v_2 = h_2 + h_3 = 282$	$C_3 = U_{A-D}$
$h_3 = 200$	$v_3 = h_4 + h_5 = 150$	$C_4 = U_{B-C}$
$h_4 = 0$	$v_4 = h_2 + h_3 + h_4 + h_5 = 432$	$C_5 = C_6 = U_{B-D}$
$h_5 = 150$	$v_5 = h_2 + h_4 = 82$	
$h_6 = 0$	$v_6 = h_3 + h_5 = 350$	
	$v_7 = h_6 = 0$	

La disminución del par  $g_{B-D}$  hace que el valor de  $v_5$  aumente desde 60 hasta 82 viajes.

De igual forma, el gradiente no tiene en cuenta los pares O-D que presentan valores nulos. Estos valores nulos pueden ser debidos a que, en el proceso de encuesta, no se han interceptado usuarios que recorran dichos pares O-D. Según Doblas y Benítez (2005) suele ser importante el tamaño del conjunto de pares O-D con valor nulo en las matrices de viaje iniciales. Este problema puede resolverse precargando los pares O-D nulos con una pequeña cantidad de viajes. Esta acción debe ser estudiada con profundidad ya que aumenta de forma evidente el número de viajes total de la matriz O-D.

Es sabido que para una matriz O-D dada y una red de transporte específica, donde los costes en arcos siguen una función monótonamente creciente, la solución en término de flujos en arcos es única, pero no en término de flujos en caminos. Según la expresión (3.48), el valor de  $\partial v_a / \partial g_i$  puede obtenerse conociendo la solución de flujos en caminos actual. Luego, para un mismo problema, pueden obtenerse diferentes valores de  $\partial v_a / \partial g_i$ , dependiendo de la solución en caminos obtenida. Esta solución dependerá del algoritmo de asignación empleado (Yang et al., 1992; Bierlaire y Toint, 1995). Por tanto, los métodos de ajuste de matrices que emplean esta

dirección de descenso pueden obtener, para un mismo problema, diferentes soluciones locales óptimas dependiendo de la solución en caminos alcanzada.

### 3.8.2 Análisis de sensibilidad para el equilibrio de usuarios con costes lineales en caminos

Para superar las limitaciones presentadas por el gradiente de Spiess y Yang et al. se lleva a cabo un análisis de sensibilidad con el fin de encontrar una nueva expresión del gradiente.

Dada una matriz de viajes y una red de transporte, el problema de asignación encuentra una solución de flujos en arcos en equilibrio. Es posible calcular las variaciones de dichos flujos, manteniendo el equilibrio de usuario, al modificar bien el valor de dicha matriz de viajes o bien las características de la red de transporte, sin la necesidad de volver a resolver el problema de asignación.

Para llevar a cabo este análisis se trabaja inicialmente en el espacio de caminos. Se supone que el coste de los caminos depende linealmente de su flujo y del flujo de los demás caminos. Llevando a cabo un desarrollo en serie de Taylor en torno a un punto actual  $(h^\phi, C^\phi)$  y despreciando los términos de segundo orden y superiores, se obtiene la expresión del coste de un camino respecto del flujo de los demás:

$$C_1 = C_1^\phi + \sum_k \frac{\partial C_1}{\partial h_k} (h_k - h_k^\phi) + O(2) = C_1^\phi - \sum_k \frac{\partial C_1}{\partial h_k} h_k^\phi + \sum_k \frac{\partial C_1}{\partial h_k} h_k \quad (3.53)$$

$$C(h) = A + B \cdot h \quad (3.54)$$

donde  $A$  es un vector que representa los valores constantes de coste en la iteración anterior y  $B$  es una matriz que representa la relación lineal entre el coste y el flujo de los caminos para el punto actual.

La matriz  $B$  está compuesta por las derivadas del coste en caminos respecto del flujo  $\partial C_k / \partial h_l$ . El coste de los caminos está formado por la suma de los costes de los arcos que lo componen.

$$C(h) = \Lambda^T \cdot s(v) \quad (3.55)$$

donde  $s$  representa el vector de coste de los arcos y  $\Lambda$  la matriz de incidencia arcos – caminos (2.7). Estos costes pueden obtenerse directamente evaluando las funciones de demora de los arcos. Para calcular los valores de las casillas de  $B$  se procede como sigue:

$$\frac{\partial C_k}{\partial h_l} = \Lambda_{F(k)}^T \cdot \frac{\partial s}{\partial h_l} = \sum_a \delta_{ak} \cdot \frac{\partial s_a}{\partial h_l} \quad (3.56)$$

donde  $\Lambda_{F(k)}$  se refiere a la fila de  $\Lambda$  que corresponde al camino  $k$ .  $\partial s_a / \partial h_l$  se desarrolla a continuación.

$$\frac{\partial s_a}{\partial h_l} = \sum_b \frac{\partial s_a}{\partial v_b} \cdot \frac{\partial v_b}{\partial h_l} = \sum_b \frac{\partial s_a}{\partial v_b} \cdot \Lambda_{F(b)}^T \cdot \frac{\partial h}{\partial h_l} = \sum_b \frac{\partial s_a}{\partial v_b} \cdot \delta_{bl} \quad (3.57)$$

Considerando que el problema de asignación es de costes separables en arcos, es decir, el coste de un arco depende únicamente de su propio volumen, la expresión (3.56) queda:

$$\frac{\partial C_k}{\partial h_l} = \sum_a \delta_{ak} \cdot \delta_{al} \cdot \frac{\partial s_a}{\partial v_a} \quad (3.58)$$

De forma matricial se tiene:

$$B = \Lambda^T \cdot \mathfrak{I} \cdot \Lambda \quad (3.59)$$

donde  $\mathfrak{I}$  es una matriz diagonal con los valores  $\partial s_a / \partial v_a$  para cada arco. De la expresión (3.58) se deduce que el elemento  $B_{kl}$ , que indica la relación lineal entre los caminos  $k$  y  $l$ , está formado por la suma de  $\partial s_a / \partial v_a$  para los arcos  $a$  comunes a dichos caminos.

Desde el punto de vista del equilibrio de usuario todos los caminos usados que pertenecen al mismo par tienen el mismo coste (Primer principio de Wardrop: ningún usuario puede mejorar su tiempo de viaje o costo generalizado cambiando unilateralmente de ruta), por lo que puede escribirse:

$$C(h) = \Gamma^T \cdot U + \pi \quad (3.60)$$

donde  $U$  es el vector formado por los costes mínimos de los caminos que unen los pares O-D y  $\Gamma^T$  la matriz de incidencia pares O-D – caminos (véase sección 2.3).  $\Gamma^T$  asigna a cada camino el coste mínimo de su correspondiente par O-D y  $\pi \geq 0$  es un vector que diferencia si los caminos son activos o no.

Se denominan caminos activos a aquellos que tienen flujo distinto de cero después del proceso de asignación. Estos caminos se caracterizan por tener el coste mínimo  $U$  para un mismo par O-D. En la ecuación (3.60), el valor de  $\pi_k$  es distinto de cero para los caminos  $k$  que no son activos.

$$\begin{aligned} Si \quad h_k \geq 0 &\rightarrow \pi_k = 0 \\ Si \quad h_k = 0 &\rightarrow \pi_k \geq 0 \end{aligned} \quad (3.61)$$

Igualando (3.54) y (3.60), y despejando el flujo en caminos, se tiene:

$$h = B^{-1} \cdot (\Gamma^T \cdot U + \pi - A) \quad (3.62)$$

Por lo general, el conjunto completo de caminos que pueden conectar los diferentes pares O-D tiene un tamaño muy elevado que hace complejo su cálculo y almacenamiento. Es por esto que en la práctica sólo se trabaja con el conjunto de caminos activos. Aunque considerar únicamente los caminos activos minimiza su cálculo y almacenamiento, las dimensiones del problema siguen siendo elevadas en comparación con los métodos de asignación en arcos. En adelante la formulación se limitará al conjunto de los caminos activos definidos mediante el superíndice asterisco (\*).

Considerando exclusivamente caminos activos la expresión anterior se simplifica.

$$h = B^{*-1} \cdot (\Gamma^{*T} \cdot U - A^*) \quad (3.63)$$

Además si se trabaja a nivel de incrementos los valores constantes desaparecen.

$$\Delta h = B^{*-1} \cdot \Gamma^{*T} \cdot \Delta U \quad (3.64)$$

Partiendo de la expresión anterior (3.64) y empleando la representación matricial de la ecuación de conservación del flujo (2.3) y la ecuación de transformación del flujo en caminos a flujo en arcos (2.5) se obtienen las dos siguientes expresiones:

$$\Delta v = \Lambda^* \cdot \Delta h = \Lambda^* \cdot B^{*-1} \cdot \Gamma^{*T} \cdot \Delta U \quad (3.65)$$

$$\Delta g = \Gamma^* \cdot \Delta h = \Gamma^* \cdot B^{*-1} \cdot \Gamma^{*T} \cdot \Delta U \quad (3.66)$$

En estos momentos, con las ecuaciones (3.65) y (3.66) se tendrían los flujos en arcos y el pseudovector O-D definidos a partir de los costes mínimos en los caminos que unen los pares O-D. Es decir, los flujos en arcos y la matriz de viajes estarían definidos desde el punto de vista del equilibrio de usuario.

Despejando  $\Delta U$  de la ecuación (3.66) e introduciéndolo en la ecuación (3.65), se obtendrían los flujos en arcos en función de los viajes del pseudovector O-D ( $\partial v / \partial g$ ).

$$\Delta v = \Lambda^* \cdot B^{*-1} \cdot \Gamma^{*T} \cdot \left( \Gamma^* \cdot B^{*-1} \cdot \Gamma^{*T} \right)^{-1} \cdot \Delta g \quad (3.67)$$

Esta expresión fue publicada por Tobin y Friesz (1988). En dicho artículo, formulando el problema de asignación a partir de desigualdades variacionales, se calcula la influencia que tienen los viajes de cada par O-D y la variación de los parámetros de la red sobre los flujos en arcos. La ecuación (3.67) es sólo la componente de la expresión de Tobin y Friesz (1988) que tiene interés para esta tesis. Con esta expresión se relacionan los flujos en arcos con los viajes de los pares O-D. A continuación se detalla el significado físico de dicha expresión:

$$\frac{\partial v}{\partial g} = \underbrace{\Lambda^* \cdot B^{*-1} \cdot \Gamma^{*T}}_{\text{I}} \cdot \underbrace{\left( \Gamma^* \cdot B^{*-1} \cdot \Gamma^{*T} \right)^{-1}}_{\text{II}} \quad (3.68)$$

III

- I. Representa las aportaciones de  $\Delta g$  a los costes mínimos de los caminos de cada par O-D,  $\left( \frac{\partial U}{\partial g} \right)$ .
- II. Representa como se modifican los costes de los caminos a partir de  $\Delta g$  siempre desde el punto de vista del equilibrio de usuario,  $\left( \frac{\partial C}{\partial g} = \frac{\partial C}{\partial U} \cdot \frac{\partial U}{\partial g} \right)$ .
- III. Representa como se reconfiguran los flujos de los caminos conocido  $\Delta C$ ,  $(\Delta h = B^{*-1} \cdot \Delta C)$ .

Por último queda definir los flujos en arcos a partir del flujo en caminos,  $(\Delta v = \Lambda^* \cdot \Delta h)$ .

Varios autores critican esta formulación por los problemas de singularidad que presenta la matriz  $B^*$  (ver Bell y Iida (1997), Sección 5.4. y Patriksson (2004), Sección 2.2.).

Por ejemplo, Patriksson (2004) expone que si el número de caminos es mayor al número de arcos, el rango de la matriz caminos – arcos ( $\Lambda^*$ ) no es completo, es decir, el rango es menor al número de arcos. De la expresión (3.59) se aprecia que si el rango de  $\Lambda^*$  no es completo, la matriz  $B^*$  es singular. Los problemas en los que el número de caminos sea mayor al de arcos son muy frecuentes en la práctica.

Como se comprueba en la ecuación (3.58),  $B_{kl}^*$  representa el sumatorio de los costes marginales de los arcos comunes entre los caminos  $k$  y  $l$ . Luego  $B^*$  permite conocer cuál será la influencia que tiene un camino sobre el coste de otro con el que comparte arcos cuando aumenta o disminuye su flujo. Aplicando además equilibrio de usuario a través de la ecuación (3.60) se calculan las influencias que se producen entre los diferentes caminos aún sin compartir arcos. Esto se debe a que todos los caminos de un par O-D se relacionan entre sí a través del coste mínimo.

La información contenida en  $B^*$  tiene mucho valor aunque a priori no es posible encontrar su inversa. La opción tomada por ciertos autores es la de simplificar dicha matriz manteniendo únicamente la diagonal (Verastegui, 2006). De esta forma las expresiones pueden ser evaluadas de forma sencilla pero a cambio dejan de considerarse las influencias entre caminos que comparten arcos, dependiendo el coste de los caminos únicamente de su propio flujo. En este caso, es posible calcular explícitamente la expresión (3.68).

$$\frac{\partial v_a}{\partial g_i} = \sum_{k \in K_i} \delta_{ak} \cdot \frac{\frac{1}{B_{kk}^*}}{\sum_{l \in K_i} \frac{1}{B_{ll}^*}} \quad (3.69)$$

De la expresión (3.69) puede obtenerse el valor que asignan los métodos de Spiess (1990) y Yang et al. (1992) a los elementos de la diagonal de  $B^*$  que es igual a  $B_{kk}^* = \frac{U_i}{h_k}$ , donde el camino  $k$  pertenece al par O-D  $i$ . Al emplear el gradiente de Spiess o Yang et al. se realizan por tanto dos hipótesis; una primera, donde el coste de los caminos sólo depende de su flujo, y una segunda, donde el coste de los caminos guarda la relación  $\frac{U_i}{h_k}$  con su flujo:

$$\Delta C_k = \frac{U_i}{h_k} \Delta h_k \quad (3.70)$$

### 3.8.3 Indeterminación del problema de estimación de matrices

La indeterminación que se da en el ajuste de la matriz de demanda es un inconveniente intrínseco al problema de transporte debido a la topología de las redes. Continuando con el apartado precedente en el que se ha realizado un análisis de sensibilidad del problema, y como extensión del apartado 3.2.4 donde se describe el problema de indeterminación, se lleva a cabo

un análisis de dicha indeterminación presentando las ecuaciones y variables que introducen las ecuaciones empleadas en la estimación de matrices.

Sin realizar un análisis exhaustivo de las ecuaciones, el número de relaciones y variables que introducen se muestran en la Tabla 3-8.

Ecuación	Definición	Nº Ecuaciones	Nº Variables
$\Delta C = B \cdot \Delta h$	Linealidad de costes	$ K $	$2 \cdot  K $
$\Delta C = \Gamma^T \cdot \Delta U$	Igualdad de costes. Eq. de usuario	$ K $	$ P $
$\Delta g = \Gamma \cdot \Delta h$	Restitución de la matriz de viajes	$ P $	$ P $
$\Delta v = \Lambda \cdot \Delta h$	Definición de los flujos en arcos	$ A $	$ A $
		$2 \cdot  K  +  A  +  P $	$2 \cdot  K  +  A  + 2 \cdot  P $

**Tabla 3-8. Planteamiento inicial del grado de indeterminación del problema.**

Se observa que la indeterminación del problema es  $|P|$ . Esta indeterminación se solventa partiendo de los valores de la matriz de viajes encontrados en la iteración anterior.

El problema descrito en la Tabla 3-8 es un caso especial que no suele darse en la práctica. Generalmente en los problemas de transporte el número de caminos que conectan con igual coste las diferentes parejas O-D es mucho mayor al número de arcos de la red. Esto hace que exista dependencia lineal entre los diferentes caminos. Según el rango de la matriz arcos-caminos ( $\Lambda$ ) el número de caminos linealmente independientes es igual al número de arcos linealmente independientes. Aplicando las leyes de Kirchhoff sobre un grafo, el número de arcos linealmente independientes es  $|N| - 1$ . Siendo  $|N|$  el número de nodos que forman el grafo. Este valor es una cota máxima del número de caminos linealmente independientes ya que puede ocurrir que, en la solución de equilibrio de usuario alcanzada en la red, existan arcos y nodos por los que no pase ningún camino. Con esto, la tabla anterior queda como sigue:

Ecuación	Definición	Nº Ecuaciones	Nº Variables
$\Delta C = B \cdot \Delta h$	Linealidad de costes	$ N  - 1$	$2 \cdot  K $
$\Delta C = \Gamma^T \cdot \Delta U$	Igualdad de costes. Eq. de usuario	$ K $	$ P $
$\Delta g = \Gamma \cdot \Delta h$	Restitución de la matriz de viajes	$ P $	$ P $
$\Delta v = \Lambda \cdot \Delta h$	Definición de los flujos en arcos	$ A $	$ A $
		$ K  +  A  +  P  +  N  - 1$	$2 \cdot  K  +  A  + 2 \cdot  P $

**Tabla 3-9. Indeterminación del problema.**

A medida que aumenta el número de caminos respecto del número de arcos independientes, aumenta el número de combinaciones lineales entre los diferentes caminos y, por tanto, la indeterminación que, partiendo de una matriz de viajes anterior, es del orden de:  $|K| - |N|$ .

### 3.8.4 Cálculo del Jacobiano $\frac{\partial v_a(g)}{\partial g_i}$ mediante diferencias finitas

Teniendo presente que la matriz  $B^*$  es singular, es preciso encontrar una formulación alternativa donde no sea necesario invertir dicha matriz. Es por esto que se proponen otros métodos para el cálculo del Jacobiano, siempre cumpliendo las ecuaciones empleadas hasta el momento.

La expresión (3.68) calcula el valor de la matriz  $\partial v_a / \partial g_i$ . Ésta refleja cómo afectará el aumento o disminución de viajes de los pares O-D ( $g_i$ ) en el flujo de los diferentes arcos de la red ( $v_a$ ). A partir de dicha matriz ( $\partial v / \partial g$ ), el proceso de ajuste calcula una dirección de descenso que pronostica el valor de los diferentes pares O-D para que reproduzcan el volumen de flujo medido sobre los arcos aforados.

Sería posible diseñar un nuevo algoritmo para resolver el problema de asignación a partir de la ecuación (3.68), basándose en su capacidad para pronosticar nuevos flujos a partir de cambios en la matriz de demanda. Aunque dicha expresión sea una aproximación lineal, después de un cierto número de iteraciones se llegará a una solución final convergente. Dicho esto, no existe ningún método de asignación basado en esta expresión. Esto se debe a que la singularidad de la matriz  $B^*$  es insalvable y el costo computacional de manejar las diferentes matrices es muy elevado.

Reconociendo la analogía entre el ajuste de matrices y la asignación de viajes en la red a través de la matriz  $\partial v / \partial g$ , se propone buscar un algoritmo numérico, similar a una asignación, para el cálculo de dicha matriz. Estos algoritmos analizan de forma individual la influencia de cada par O-D sobre todos los arcos de la red.

Las metodologías propuestas se centran en alcanzar un nuevo estado de equilibrio una vez que se ha aportado un incremento infinitésimo a un determinado par O-D. Por tanto se engloban dentro de los métodos numéricos de “Diferencias finitas”. Con estos incrementos infinitesimales se obtienen los valores de la matriz  $\partial h / \partial g$ . Obtener una nueva situación de equilibrio de usuario es una acción simple ya que se parte de la solución anterior. Este paso es necesario realizarlo para cada par O-D de forma individual.

Inicialmente se plantean tres metodologías igualmente válidas ya que reportan análogas soluciones al problema planteado. Varios de los algoritmos propuestos se centran en el cálculo de  $\partial h / \partial g$ . La matriz de  $\partial h / \partial g$  puede calcularse de una forma más simple e intuitiva que  $\partial v / \partial g$  en aquellos casos en los que se utilice un algoritmo de asignación basado en caminos. Para conocer los valores de la matriz  $\partial v / \partial g$  sólo es necesario aplicar la expresión (3.71):

$$\frac{\partial v}{\partial g} = \Lambda^* \cdot \frac{\partial h}{\partial g} \quad (3.71)$$

Las metodologías propuestas se dividen en dos conjuntos según si resuelven problemas de asignación de tráfico o resuelven expresiones.

#### 3.8.4.1 Metodología por asignación de tráfico

La primera metodología resuelve un problema de asignación para cada par O-D. El algoritmo de asignación empleado es un caso especial llamado “Warm Start”. Este método de asignación es muy rápido convergiendo ya que el punto de partida, para cada par O-D, es la solución de flujo en arcos obtenida con la matriz de viajes actual.

Además de emplear “Warm Start” en la asignación, se realizan dos modificaciones asumiendo la siguiente hipótesis: dar incrementos infinitesimos a la matriz de viajes genera una desviación mínima respecto del estado de equilibrio por lo que el número de pares O-D afectado es limitado. Al mismo tiempo, ningún par O-D modificará su número de caminos. Esta hipótesis disminuye el coste computacional del problema:

- Se elimina el módulo que busca nuevos caminos mínimos para cada par O-D. Entendiendo que la variación de la matriz de demanda es mínima, se presupone que no cambia el conjunto de caminos activos.
- Se modifica el módulo que equilibra el coste de los caminos de cada par O-D. Éste equilibra de forma iterativa el coste de los caminos de un mismo par O-D para que coincidan y sean mínimos. Entendiendo que una variación infinitesimal de los viajes de un par O-D no afecta al conjunto completo de caminos, dicho módulo se centrará únicamente en los pares O-D cuyos caminos se vean influidos. Estos caminos pueden conocerse a través de la matriz  $B^*$ .

#### 3.8.4.2 Metodología analítica

Los dos siguientes algoritmos comparten los mismos pasos para el cálculo de la matriz  $\partial h / \partial g$ .

**Paso 1.** Inicialmente se selecciona un par O-D  $p$  para analizar su influencia sobre todos los caminos de la red.

$$\frac{\partial h_k}{\partial g_p} \quad \forall k \in K \quad (3.72)$$

**Paso 2.** Se aporta un incremento infinitésimo al número de viajes del par O-D  $p$  que provoca un aumento infinitésimo en el flujo (y coste) de los caminos de dicho par O-D.

**Paso 3.** Este pequeño aumento de flujo afectará a los demás caminos de la red con los que comparten arcos incrementando su coste. Dicho aumento de coste se calcula a través de la matriz  $B^*$ .

**Paso 4.** Llegado a este punto, mediante un módulo iterativo se alcanza una nueva situación de equilibrio de usuario. Así, se reconfigura el flujo de todos los caminos influenciados para que finalmente tengan igual coste mínimo dentro de cada par O-D.

Las ecuaciones que intervienen en el paso 2 son las siguientes:

$$\sum_j h_j = g_p + \delta g_p \quad \forall j \in K_p \quad (3.73)$$

$$C_j = C_j^\phi + \sum_{k \in K} \frac{\partial C_j}{\partial h_k} (h_k - h_k^\phi) + O(2) = U_p \quad \forall j \in K_p \quad (3.74)$$

donde  $\frac{\partial C_j}{\partial h_k} = B_{jk}^*$ .

Inicialmente se parte de la situación en equilibrio de usuario alcanzada en la asignación anterior por lo que el coste de los caminos del par O-D  $p$  sólo se ven influidos por el flujo de ellos mismos. Así, considerando que se desea calcular únicamente los incrementos infinitésimos de flujo y que son despreciables los términos de segundo orden, las expresiones (3.73) y (3.74) pueden escribirse como sigue:

$$\sum_{j \in K_p} \Delta h_j = \delta g_p \quad (3.75)$$

$$\Delta C_j = \sum_{i \in K_p} B_{ji}^* \cdot \Delta h_i = U_p \quad \forall j \in K_p \quad (3.76)$$

Una vez calculado  $\Delta h$  para todos los caminos del par O-D  $p$ , en el paso 3 se calcula el incremento de coste de los caminos que comparten arcos con dichos caminos, de la forma:

$$\Delta C_k = \sum_{j \in K_p} B_{kj}^* \cdot \Delta h_j \quad \forall k \in K \quad (3.77)$$

En el paso 4 se resuelve un módulo iterativo donde se van reconfigurando los costes de los caminos para cada par O-D  $q \neq p$ . En este caso, la suma de los incrementos de los flujos para los caminos de un mismo par es cero puesto que no se ha modificado el valor de  $g_q$ . Se han de tener en cuenta únicamente los pares O-D tales que el coste de alguno de sus caminos  $k \in K_q$  haya sido modificado. Las expresiones que se emplean en este paso son las siguientes:

$$\sum_{j \in K_q} \Delta h_j = 0 \quad (3.78)$$

$$\Delta C_j = \sum_{i \in K_q} B_{ji}^* \cdot \Delta h_i + \sum_{k \in K} B_{jk}^* \cdot \Delta h_k^n = U_q \quad \forall j \in K_q \quad (3.79)$$

En las ecuaciones anteriores se distinguen dos vectores de flujo. El primero es  $\Delta h^n$ , en él están recogidos los incrementos de flujos de todos los caminos en la iteración anterior  $n$  (incluidos los del par O-D  $p$ ). El segundo vector es  $\Delta h$  donde se recogen los nuevos incrementos de flujo, de los caminos del par O-D  $q$ , necesarios para que se alcance equilibrio de usuario en dicho par O-D.

Una vez se resuelven las ecuaciones (3.78) y (3.79), la solución final será la suma de los dos vectores empleados,  $\Delta h$  y  $\Delta h^n$ .

$$\Delta h^{n+1} = \Delta h^n + \Delta h \quad (3.80)$$

El algoritmo finaliza cuando se alcance la convergencia definida.

Las expresiones expuestas en los pasos 2 y 4 pueden resolverse utilizando dos métodos diferentes. En el primer método se calculan los incrementos de flujo de cada camino evaluando expresiones, en el segundo dichos incrementos son obtenidos resolviendo sistemas de ecuaciones.

#### 3.8.4.2.1 Primer método: equilibrado de los pares O-D empleando la diagonal de la matriz de costes

Estudiando los elementos de la matriz  $B^*$  se comprueba que los valores más importantes se encuentran en la diagonal. Esto es de esperar ya que el flujo que más influye sobre un camino es el propio. Un elemento  $B_{ii}^*$  de la diagonal de  $B^*$  está formado por el coste marginal de todos los

arcos que forman el camino  $i$ . Cualquier elemento  $ik$  con  $i \neq k$  fuera de la diagonal es menor, ya que sólo se suman los costes marginales de los arcos comunes entre los caminos  $i$  y  $k$ . Así, para el reparto inicial del flujo sobre los diferentes caminos de un mismo par O-D, se considerarán únicamente los valores de la diagonal.

En el paso 2, para cada camino  $i$  dentro del par O-D, se resuelven las siguientes expresiones:

$$\sum_{j \in K_p} \Delta h_j = \delta g_p \quad (3.81)$$

$$U_p = B_{ii}^* \cdot \Delta h_i = B_{jj}^* \cdot \Delta h_j \quad \forall j \in K_p \quad (3.82)$$

De la expresión (3.82) se despeja  $\Delta h$  para cada camino del par O-D  $p$ :

$$\Delta h_j = \frac{B_{ii}^*}{B_{jj}^*} \cdot \Delta h_i \quad \forall j \in K_p \quad (3.83)$$

Los valores obtenidos mediante la ecuación (3.83) se introducen en la ecuación (3.81), resultando la expresión:

$$\Delta h_i \cdot \left( 1 + \sum_{(j \in K_p) \neq i} \frac{B_{ii}^*}{B_{jj}^*} \right) = \delta g_p \quad \forall i \in K_p \quad (3.84)$$

quedando:

$$\Delta h_i = \frac{\delta g_p}{\left( 1 + \sum_{(j \in K_p) \neq i} \frac{B_{ii}^*}{B_{jj}^*} \right)} = \frac{\delta g_p}{\sum_{j \in K_p} \frac{B_{ii}^*}{B_{jj}^*}} \quad \forall i \in K_p \quad (3.85)$$

En el paso 4 se agrega en las expresiones los valores de flujo de los demás caminos de la red. Partiendo de la ecuación (3.79) y considerando únicamente la diagonal de la matriz  $B^*$ , la expresión para el camino  $i$  queda:

$$B_{ii}^* \cdot \Delta h_i + \sum_{k \in K} B_{ik}^* \cdot \Delta h_k^n = U_p \quad \forall i \in K_p \quad (3.86)$$

En la expresión anterior (3.86) se despeja  $\Delta h_i$  dividiendo por  $B_{ii}^*$  todos los términos.

$$\Delta h_i + \frac{1}{B_{ii}^*} \sum_{k \in K} B_{ik}^* \cdot \Delta h_k^n = \frac{1}{B_{ii}^*} U_p \quad \forall i \in K_p \quad (3.87)$$

Sumando las ecuaciones de todos los caminos del par O-D  $p$ , se obtiene:

$$\sum_{j \in K_p} \Delta h_j + \sum_{j \in K_p} \left( \frac{1}{B_{jj}^*} \sum_{k \in K} B_{jk}^* \cdot \Delta h_k^n \right) = \sum_{j \in K_p} \frac{1}{B_{jj}^*} U_p \quad (3.88)$$

Como  $\sum_{j \in K_p} \Delta h_j = 0$  (expresión (3.78)), despejando el valor de  $U_p$ :

$$U_p = \frac{1}{\sum_{j \in K_p} \frac{1}{B_{jj}^*}} \sum_{j \in K_p} \left( \frac{1}{B_{jj}^*} \sum_{k \in K} B_{jk}^* \cdot \Delta h_k^n \right) \quad (3.89)$$

Introduciendo el valor de  $U_p$  en la expresión (3.87) se obtiene finalmente la ecuación buscada:

$$\Delta h_i = \frac{\frac{1}{B_{ii}^*}}{\sum_{j \in K_p} \frac{1}{B_{jj}^*}} \sum_{j \in K_p} \left( \frac{1}{B_{jj}^*} \sum_{k \in K} B_{jk}^* \cdot \Delta h_k^n \right) - \frac{1}{B_{ii}^*} \sum_{k \in K} B_{ik}^* \cdot \Delta h_k^n \quad \forall i \in K_p \quad (3.90)$$

Esta forma de calcular la matriz  $\partial h / \partial g$  reporta soluciones exactas ya que, aunque se utilicen inicialmente sólo los elementos de la diagonal de  $B^*$ , es un método recursivo donde en cada iteración se tienen en cuenta las interacciones entre ellos al calcular los costes de los caminos (paso 3).

#### 3.8.4.2.2 Segundo método: equilibrado simultáneo de los pares O-D mediante sistemas de ecuaciones

En el paso 2 se resuelven las expresiones (3.75) y (3.76) mediante un sistema de ecuaciones que considera tanto los elementos de la diagonal de  $B^*$  como los de fuera de la diagonal. Así, en forma matricial el sistema a resolver es el siguiente:

$$\begin{bmatrix} 1 & 1 & \dots & 1 & 0 \\ B_{11} & B_{12} & \dots & B_{1n} & -1 \\ B_{21} & B_{22} & \dots & B_{2n} & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ B_{n1} & B_{n2} & \dots & B_{nm} & -1 \end{bmatrix} \cdot \begin{Bmatrix} \Delta h_1 \\ \Delta h_2 \\ \vdots \\ \Delta h_n \\ U_p \end{Bmatrix} = \begin{bmatrix} \delta g_p \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \forall \Delta h_i \mid i \in K_p \quad (3.91)$$

En este método se calcula el reparto de flujo entre los caminos del par O-D considerando las diferentes influencias entre ellos. De esta forma no es necesario iterar dentro del mismo par

O-D. Para considerar las influencias indirectas entre los caminos de los demás pares O-D es necesario generar un algoritmo iterativo (paso 4).

El sistema de ecuaciones que se resuelve en el paso 4 para cada par O-D es el siguiente:

$$\begin{bmatrix} 1 & 1 & \dots & 1 & 0 \\ B_{11} & B_{12} & \dots & B_{1n} & -1 \\ B_{21} & B_{22} & \dots & B_{2n} & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ B_{n1} & B_{n2} & \dots & B_{nn} & -1 \end{bmatrix} \cdot \begin{Bmatrix} \Delta h_1 \\ \Delta h_2 \\ \vdots \\ \Delta h_n \\ U_p \end{Bmatrix} = \begin{bmatrix} 0 \\ \sum_k B_{1k}^* \cdot \Delta h_k^n \\ \sum_k B_{2k}^* \cdot \Delta h_k^n \\ \vdots \\ \sum_k B_{nk}^* \cdot \Delta h_k^n \end{bmatrix} \quad \begin{matrix} \forall \Delta h_i \mid i \in K_p \\ \forall \Delta h_k^n \mid k \in K_p \end{matrix} \quad (3.92)$$

La solución de este sistema es una reconfiguración del flujo de los caminos de un mismo par O-D que sumado a los flujos anteriores (expresión (3.80)) va convergiendo hasta obtener un vector de flujos en caminos que cumple equilibrio de usuario.

Comparando las expresiones (3.91) y (3.92), se observa que las matrices de coeficientes coinciden en ambos casos. Es por esto que el método genera la matriz de coeficientes para cada par O-D una sola vez y luego la almacena para emplearla en las diferentes iteraciones.

Una vez ensayados los algoritmos y contrastadas sus soluciones, puede concluirse que las matrices  $\partial v / \partial g$  encontradas en las tres metodologías anteriores son exactamente iguales a la definida por Tobin y Friesz (1988). Por tanto, es posible resolver de forma numérica la expresión del Jacobiano definida en (3.68) mientras que no es posible resolverla analíticamente con las ecuaciones que se presentan en los problemas de transporte. Con estas metodologías se evita el problema de singularidad que presenta la matriz  $B^*$ .

### 3.9 ALGORITMOS IMPLEMENTADOS PARA EL CÁLCULO

#### DEL JACOBIANO $\frac{\partial v_a(g)}{\partial g_i}$

Partiendo de la formulación propuesta en el apartado 3.8, se han llevado a cabo varias implementaciones con el fin de encontrar un algoritmo que calcule el Jacobiano de un problema de tamaño real en un tiempo aceptable.

El principal inconveniente encontrado en las diferentes implementaciones es la dimensión del Jacobiano. Éste se compone de tantas columnas como pares O-D tiene el problema. En la

descripción del cálculo del Jacobiano, apartado 3.8.4, se comenta que la obtención de cada una de sus columnas supone la resolución de un problema de asignación de tráfico. Luego se resolverán del orden de  $|P|$  problemas de asignación en el cálculo de cada dirección de descenso.

Por otro lado, la forma de proceder de los algoritmos implementados es aplicar un incremento infinitésimo de los viajes de cada par O-D y volver a encontrar equilibrio de usuario en el sistema de transporte. Al ser un incremento ínfimo de los viajes de cada par O-D el estado hacia el que se mueve el sistema estará muy cerca del equilibrio por lo que el tiempo de cada asignación será pequeño.

El inconveniente radica en que al tener un problema de orden  $O(P)$  los tiempos de asignación deben ser realmente pequeños para que la metodología sea factible. Poniendo un ejemplo, si se tiene una red ficticia con 400 centroides con un tiempo estimado para el problema de asignación de 100 segundos, para que el tiempo de cálculo del Jacobiano sea del mismo orden que el de asignación cada columna debe obtenerse en un tiempo de  $100 / 400^2 = 0,000625$  segundos.

Gracias al avance computacional experimentado en la última década se destaca una ventaja en el problema de cálculo del Jacobiano como es la independencia en el cálculo de cada columna. De esta forma es posible lanzar de forma paralela el cálculo de tantas columnas como procesadores incorpore la computadora. Para el desarrollo de esta tesis se dispone de una computadora con 12 procesadores que serán empleados gracias a la parallel computing toolbox de Matlab. Así, los resultados que se muestran en los siguientes apartados presentarán unos tiempos de computación del orden de unas 12 veces inferiores a la ejecución en serie con un solo procesador.

Inicialmente se analizan los algoritmos implementados empleando la red de transporte de Tiergarten presentada en el apartado 2.8.1. Los tiempos de cálculo para esta red de pequeño tamaño suelen ser bajos por lo que se emplea además la red de transporte de Chicago Sketch. Esta red es de tamaño medio y se hace más costoso su análisis. Para los algoritmos que superan con tiempos razonables los cálculos de las redes de Tiergarten y Chicago Sketch, se realiza un análisis de la red de transporte de la comunidad de Madrid. Uno de los principales objetivos de la tesis es ajustar matrices O-D con escala semejante a esta última red. En la Tabla 3-10 se detallan las dimensiones de las redes empleadas.

<b>Características</b>	<b>Tiergarten</b>	<b>Chicago Sketch</b>	<b>Madrid</b>
<b>Arcos</b>	747	2.950	17.544
<b>Nodos</b>	344	933	7.966
<b>Centroides</b>	26	387	1.180
<b>Pares O-D</b>	650	149.382	1.391.220

**Tabla 3-10. Características de las redes de transporte empleadas.**

En total se han implementado cuatro algoritmos para obtener el Jacobiano del problema, que se describen a continuación.

### **3.9.1 Algoritmo 1. Asignación basada en caminos.**

En el apartado 2.7 se lleva a cabo una implementación del algoritmo de Florian et al. (2009) modificada. Con las variables de salida de este algoritmo, basado en caminos, es posible abordar el análisis de los tres tipos de formulaciones propuestos en el apartado 3.8.4. De las tres implementaciones descritas, se destaca el método definido en el apartado 3.8.4.2.2 donde se calcula el Jacobiano resolviendo sistemas de ecuaciones. Las restantes opciones alcanzan el mismo resultado pero emplean tiempos de computación superiores.

Al comparar los métodos 3.8.4.2.1 y 3.8.4.2.2, se observa que tienen una estructura muy similar. La diferencia se encuentra en el equilibrado del coste de los caminos de cada par O-D. El primer método equilibra el coste mediante un método recursivo que tiene en cuenta las influencias entre los diferentes caminos de forma indirecta, y el segundo método lo hace resolviendo un sistema de ecuaciones que considera todas las influencias en un solo paso. El método 3.8.4.2.1 se hace más costoso numéricamente respecto del método 3.8.4.2.2 al necesitar varias iteraciones para cada par O-D.

Comparando los métodos 3.8.4.1 y 3.8.4.2.2 se aprecia igualmente una gran similitud, en los dos métodos se equilibran los pares O-D resolviendo sistemas de ecuaciones. La diferencia se encuentra en la estructura algorítmica. El primer método resuelve la opción de Warm Start modificada. La estructura de este procedimiento es semejante a la del problema de asignación por lo que es necesario tratar sus variables de salida para generar el Jacobiano. El segundo método, en cambio, se diseña expresamente para que la salida sea el Jacobiano y las variables internas que emplea se diseñan para agilizar el proceso.

El esquema algorítmico que se implementa es el siguiente:

- Crear la matriz  $\frac{\partial v}{\partial g}$  que será la salida del algoritmo.
- Calcular  $B$  siguiendo la expresión (3.59).
- Para cada par O-D  $p$ :
  - Crear un vector  $\frac{\partial h}{\partial g_p}$ .
  - Crear el vector  $\Delta h$  que incluye todos los caminos del problema, inicialmente todos sus elementos son nulos.
  - Repartir un incremento infinitésimo de viajes entre los caminos del par O-D  $p$ .

$$\Delta h(k) = \delta g_p \quad k \in p$$

- Mientras  $Rel. Gap > Rel. Gap objetivo$ :
  - Actualizar el coste de los caminos de todos los pares O-D.

$$\Delta C = B \cdot \Delta h$$

- Enumerar los pares O-D desequilibrados, almacenar en  $P_{desq}$ .
- Para cada par O-D  $i$  en  $P_{desq}$ :
  - Extraer de  $B$  la matriz bloque  $B_{k \times k}$  perteneciente a los caminos del par O-D  $i$ .
  - Resolver el siguiente sistema:

$$B_{k \times k} \cdot x = C_k \quad \forall k \in K_i$$

donde  $x$  es la dirección del par O-D  $i$  y  $C_k$  es el coste de cada camino,  $C_k = B(k, :) \cdot \Delta h \quad \forall k \in K_i$ .

- Comprobar si  $\exists k \mid h_k + x_k < 0 \quad \forall k \in K_i$ . En ese caso, buscar  $k \mid h_k + x_k = \min_{j \in K_i} (h_j + x_j)$ . La nueva dirección se calcula como:

$$x \leftarrow \frac{h_k}{x_k} x$$

- Actualizar el vector  $\Delta h(k) = \Delta h(k) + x \quad \forall k \in K_i$ .

- Asignar  $\frac{\partial h}{\partial g_p} \leftarrow \Delta h$ .
- Asignar  $\frac{\partial v}{\partial g_p} \leftarrow \Lambda \cdot \frac{\partial h}{\partial g_p}$ .

La principal propiedad de los algoritmos diseñados que se analiza en este apartado es la factibilidad de su aplicación en problemas de tamaño real. Para ello se medirán tiempos de ejecución en las diferentes redes. En la Tabla 3-11 se muestran los tiempos de ejecución del cálculo del Jacobiano en segundos.

Red Transporte	Cálculo del Jacobiano (RGap=1e-10)	Cálculo del Jacobiano (RGap=1e-6)	Problema de Asignación (RGap=1e-10)
Tiergarten	12,36	-	2,60
Chicago Sketch	>15.000	8.159	57,40

Tabla 3-11. Tiempos de computación (segundos) correspondientes al algoritmo 1.

Se ha agregado una columna con los tiempos de ejecución del problema de asignación de tráfico. En la red de Tiergarten se observa que los tiempos de computación del cálculo del Jacobiano y del problema de asignación de tráfico tienen el mismo orden para un nivel de convergencia exigente, con un *relative gap* de  $10^{-10}$ . No ocurre lo mismo para la red de Chicago Sketch. El tiempo de cálculo del Jacobiano se dispara cuando se exige una alta convergencia. Exigir un *relative gap* mayor reduce el tiempo de cálculo y hace posible llevar a cabo un proceso de ajuste aunque estos tiempos pueden ser inviables para redes de mayor tamaño. Por ello, se relaja el *relative gap* del cálculo del Jacobiano hasta  $10^{-6}$  para la red de Chicago Sketch.

Para niveles de convergencia con *relative gap* inferiores a  $10^{-6}$  las diferencias entre los Jacobianos son pequeñas. Los gradientes calculados con los diferentes Jacobianos serán similares aunque menos exactos cuanto menos convergencia se le exija al algoritmo.

Una vez medidos los tiempos de cálculo del Jacobiano, se lleva a cabo un análisis de la solución encontrada al incluir dicho Jacobiano en un proceso general de ajuste de matrices O-D. En este caso se comprueba que el programa alcance una matriz de viajes que estime bien el volumen en los arcos aforados. El objetivo es evaluar la eficiencia de la dirección de búsqueda propuesta comprobando que el algoritmo global de ajuste minimiza la discrepancia de los aforos. Para ello se realiza un proceso de ajuste completo en la red de Tiergarten. La metodología de ajuste que se sigue es la descrita en la primera columna de la Tabla 3-1. Se realiza un total de 200 iteraciones y se exige tanto a la solución del problema de asignación como a la salida del Jacobiano una convergencia de *relative gap* =  $10^{-10}$ .

En la Figura 3-4 se representa la evolución de la función objetivo durante el ajuste. Puede observarse como la dirección de descenso empleada minimiza el valor de la función objetivo en cada iteración.

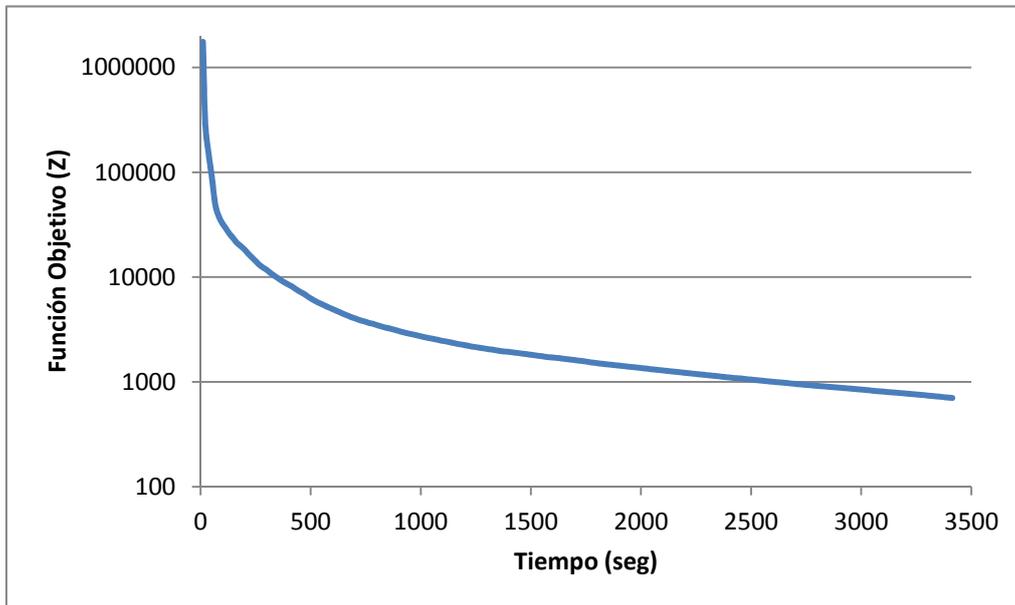


Figura 3-4. Algoritmo 1 - Evolución función objetivo.

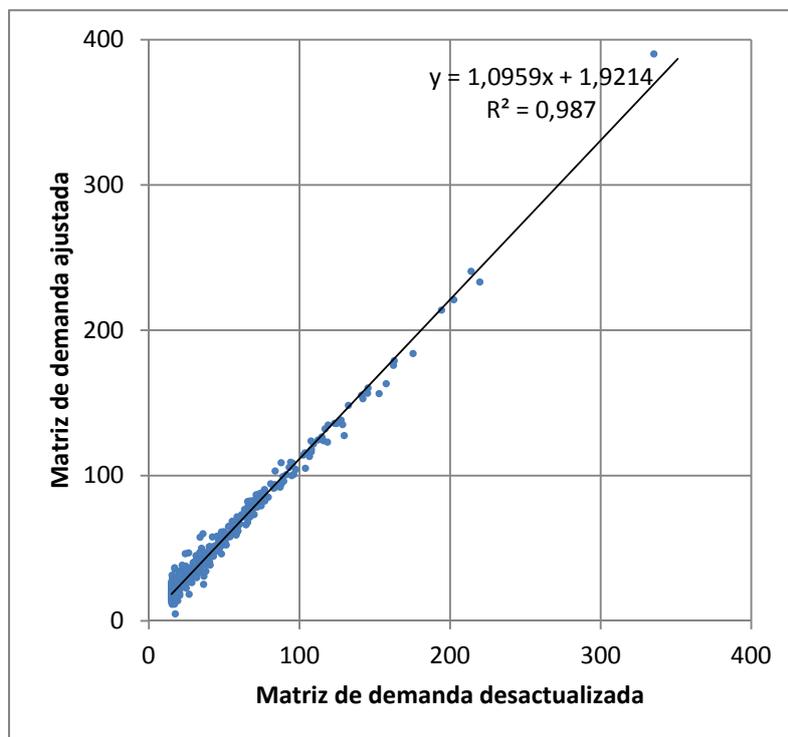


Figura 3-5. Algoritmo 1 - Matriz desactualizada frente a ajustada.

En la Figura 3-5 se muestran los elementos de la matriz de viajes desactualizada frente a los elementos de la matriz ajustada. De igual forma, la gráfica anterior muestra la dispersión entre

las dos matrices, la matriz de partida y la matriz solución. Se observa que el valor de  $R^2$  es cercano a 1, por lo que la variación de los elementos de la matriz desactualizada no ha sido excesiva. Así puede concluirse que el algoritmo no distorsiona apreciablemente la estructura de la matriz de partida.

### 3.9.2 Algoritmo 2. Asignación basada en orígenes.

El algoritmo de asignación basado en orígenes de Dial (2006) mejoró considerablemente los tiempos de ejecución de los demás algoritmos de asignación que existían hasta el momento. Por eso se decide emplear dicho algoritmo para acelerar el cálculo del Jacobiano. En este caso las variables de salida del algoritmo de asignación no permiten ensayar los métodos 3.8.4.2.1 y 3.8.4.2.2 por lo que solo se implementa el algoritmo descrito en el apartado 3.8.4.1 en el que se emplea la opción Warm Start modificada.

El esquema algorítmico que se implementa es el siguiente:

- Crear la matriz  $\frac{\partial v}{\partial g}$  que será la salida del algoritmo.
- Definir el vector  $v_{asig}$  como el patrón de flujo en arcos resultante de la asignación previa.
- Para cada origen  $i$ :
  - Resolver un algoritmo de ruta mínima desde  $i$  hacia todos los destinos.
  - Para cada destino  $j$  diferente de  $i$ :
    - Definir el par O-D  $p$  correspondiente al origen  $i$  y destino  $j$ .
    - Aplicar un incremento infinitésimo de viajes al camino de coste mínimo que conecta el par O-D  $p$ .
    - Ejecutar la opción de Warm Start modificada. Su salida son los nuevos flujos en arcos del problema  $v_{warm}$ .
    - Asignar  $\frac{\partial v}{\partial g_p} \leftarrow (v_{asig} - v_{warm})$

En la Tabla 3-12 se muestran los tiempos de ejecución en segundos del cálculo del Jacobiano.

Red Transporte	Cálculo del Jacobiano (RGap=1e-10)	Cálculo del Jacobiano (RGap=1e-6)	Problema de Asignación (RGap=1 e-10)
Tiergarten	9,63	-	1,23
Chicago Sketch	>15.000	7.869	32,26

Tabla 3-12. Tiempos de computación (segundos) correspondientes al algoritmo 2.

Se aprecia una mejora importante en los tiempos de cálculo del problema de asignación para la red de Chicago Sketch. El tiempo requerido para la asignación se ha reducido prácticamente a la mitad. Igualmente ocurre en el cálculo del Jacobiano para la red de Tiergarten. El inconveniente se encuentra en que repite los tiempos de cálculo del Jacobiano para redes de tamaño medio. Aunque esta nueva implementación presenta mejoras respecto de la anterior, continúa siendo un hándicap el cálculo del Jacobiano en redes de tamaño medio.

En las figuras Figura 3-6 y Figura 3-7 se presentan los resultados obtenidos en el proceso de ajuste de la matriz de Tiergarten. Las características del ajuste son las empleadas en el apartado anterior.

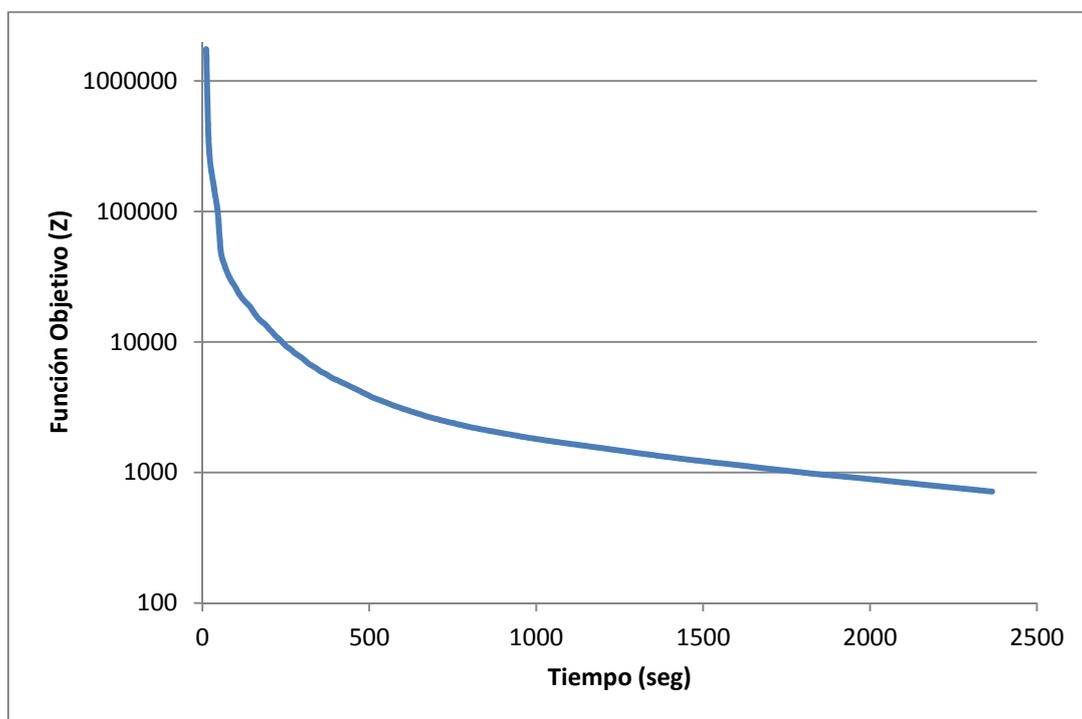


Figura 3-6. Algoritmo 2 - Evolución función objetivo.

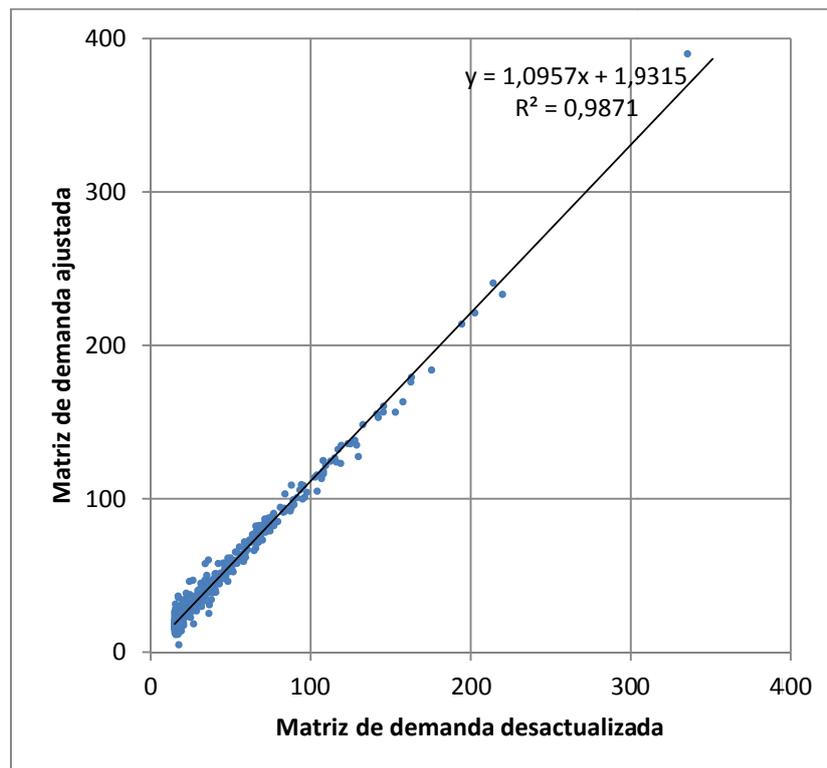


Figura 3-7. Algoritmo 2 - Matriz desactualizada frente a ajustada.

### 3.9.3 Algoritmo 3. Asignación basada en orígenes con almacenamiento de segmentos alternativos.

El algoritmo TAPAS de asignación basado en orígenes de Bar-Gera (2010) es una modificación del método de Dial (2006). Este nuevo algoritmo mejora los tiempos del anterior. La principal diferencia es el almacenamiento y gestión de parejas de segmentos alternativos (PAS). El algoritmo se centra en equilibrar exclusivamente los PAS, lo que implica una reducción del tiempo de computación. Esta aceleración del proceso se debe a que el conjunto de PAS que es necesario almacenar y equilibrar es muy reducido. Como ejemplo, la red de Chicago Sketch tiene 149.382 pares O-D y, cuando finaliza el algoritmo de asignación, el número de PAS es de tan solo 268.

Las variables de salida del algoritmo de Bar-Gera (2010) son los segmentos PAS y los flujos en arcos. Al igual que en el algoritmo 2 solo es posible implementar el método 3.8.4.1 para el cálculo del Jacobiano. Este método reproduce unos tiempos de computación similares a los que se dan en el apartado anterior por lo que en principio no aporta demasiado al problema del cálculo de la dirección de descenso.

Sin embargo, el conjunto de segmentos PAS ofrece la opción de diseñar una nueva implementación del cálculo del Jacobiano que no se recoge en el apartado 3.8.4. Los algoritmos

implementados hasta el momento proporcionan un incremento infinitesimal de viajes a cada par O-D y suponen que el conjunto de caminos no se modifica. De esta forma, suponiendo que el conjunto de PAS permanece constante, se alcanzará una nueva situación de equilibrio de usuario en la red equilibrando únicamente dicho conjunto. Inicialmente, esta nueva implementación mejorará las anteriores gracias a la disminución del tamaño del problema. En el caso de la red de Chicago Sketch será necesario equilibrar únicamente 268 PAS para cada incremento infinitesimal en cada par O-D. Al igual que los métodos de asignación esta implementación será recursiva ya que existen influencias entre los diferentes PAS.

El esquema algorítmico que se implementa es el siguiente:

- Crear la matriz  $\frac{\partial v}{\partial g}$  que será la salida del algoritmo.
- Definir el conjunto  $Q$  con todos los PAS resultantes de la asignación previa.
- Para cada origen  $i$ :
  - Resolver un algoritmo de ruta mínima desde  $i$  hacia todos los destinos.
  - Para cada destino  $j$  diferente de  $i$ :
    - Definir el par O-D  $p$  correspondiente al origen  $i$  y destino  $j$ .
    - Crear el vector  $\Delta v$  que incluye todos los arcos del problema, inicialmente todos sus elementos son nulos.
    - Aplicar un incremento infinitésimo de viajes al camino de coste mínimo que conecta el par O-D  $p$ .
    - Actualizar el coste de los arcos que han aumentado su flujo.
    - Mientras  $Rel. Gap > Rel. Gap objetivo$ :
      - Para cada PAS  $q \in Q$ 
        - Calcular el coste de los dos segmentos alternativos

$$C_S = \sum_{a \in S} s(v_a) \quad S \in \{S_1, S_2\} \equiv PAS_q$$

- Si  $C_{S_1} \neq C_{S_2}$ 
  - Equilibrar los flujos de los segmentos de  $PAS_q$ , obteniendo  $\Delta h_{S_1}$  y  $\Delta h_{S_2}$ .
  - Actualizar  $\Delta v = \Delta v + \Lambda_{S_1} \cdot \Delta h_{S_1} + \Lambda_{S_2} \cdot \Delta h_{S_2}$
  - Actualizar  $v = v + \Delta v$

- Asignar  $\frac{\partial v}{\partial g_p} \leftarrow \Delta v$

Los tiempos de cálculo del algoritmo se muestran en la Tabla 3-13:

Red Transporte	Cálculo del Jacobiano (RGap=1e-10)	Cálculo del Jacobiano (RGap=1e-6)	Problema de Asignación (RGap=1e-10)
Tiergarten	4,66	-	0,54
Chicago Sketch	6.496	1.332,5	31,6

Tabla 3-13. Tiempos de computación (segundos) correspondientes al algoritmo 3.

Esta implementación mejora las descritas anteriormente tanto en los tiempos de cálculo del problema de asignación como en los tiempos de cálculo del Jacobiano. La principal mejora se aprecia en la disminución de los tiempos de cálculo del Jacobiano para redes de tamaño medio. Este hecho reduce considerablemente el tiempo empleado en un proceso de ajuste de la matriz de viajes.

El tiempo del cálculo del Jacobiano sigue siendo muy superior al tiempo de ejecución del problema de asignación pero se ha reducido enormemente respecto de los algoritmos implementados anteriormente. Esto se debe a que el cálculo de cada columna del Jacobiano se reduce a equilibrar un pequeño conjunto de segmentos.

Llevando a cabo el proceso de ajuste de la matriz de viajes en la red de Tiergarten descrito en 3.9.1, se obtienen las Figura 3-8 y Figura 3-9.

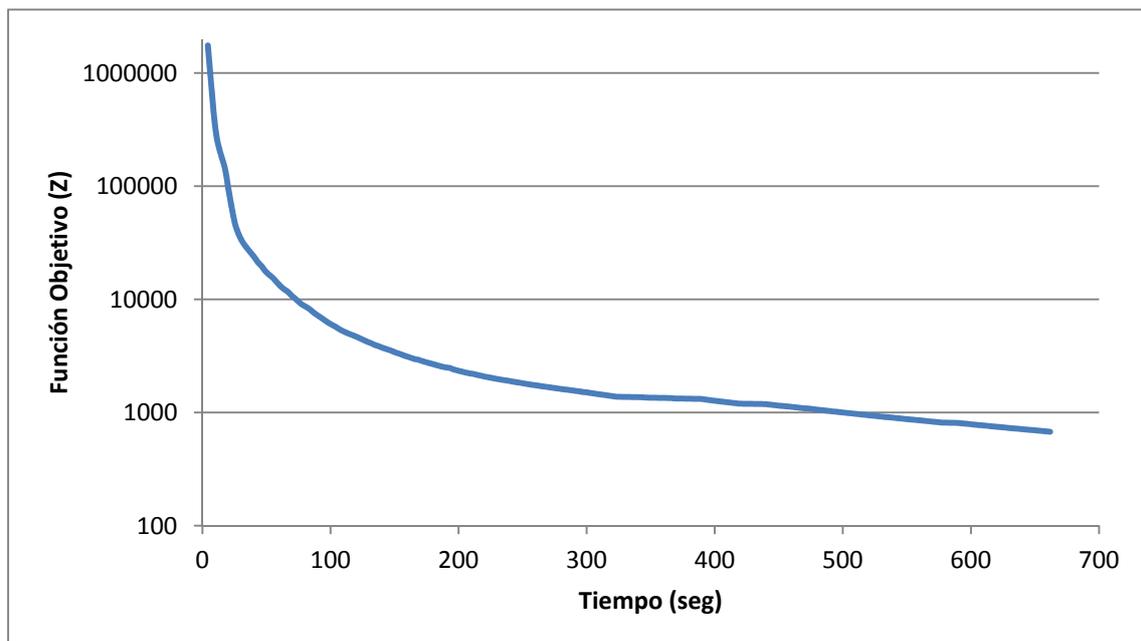


Figura 3-8. Algoritmo 3 - Evolución función objetivo.

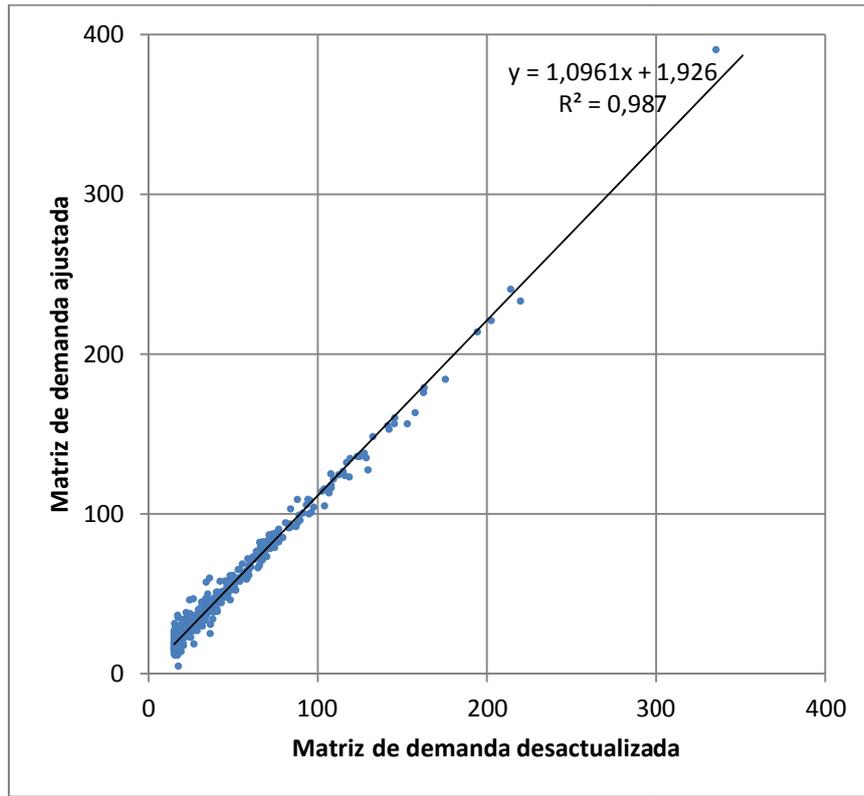


Figura 3-9. Algoritmo 3 - Matriz desactualizada frente a ajustada.

### 3.9.4 Algoritmo 4. Asignación basada en orígenes con equilibrado simultáneo del conjunto de PAS.

Este algoritmo se presenta como una mejora del algoritmo 3. En esta nueva implementación se sustituye el método recursivo de equilibrado de segmentos PAS por la resolución de un único sistema de ecuaciones. De este modo, se equilibra en un solo paso todo el conjunto de PAS.

El sistema de ecuaciones que se plantea es el siguiente:

$$\begin{aligned} \Delta h_{S_1} &= -\Delta h_{S_2} & \forall S \in \text{Segmentos PAS} \\ C_{S_1} &= C_{S_2} & \forall S \in \text{Segmentos PAS} \end{aligned} \quad (3.93)$$

donde:

$$\begin{aligned} C_{S_1} &= C_{S_1}^0 + B_{S_1} \cdot \Delta h_{S_1} \\ C_{S_2} &= C_{S_2}^0 + B_{S_2} \cdot \Delta h_{S_2} = C_{S_2}^0 - B_{S_2} \cdot \Delta h_{S_1} \end{aligned} \quad (3.94)$$

siendo  $B_{S_1}$  y  $B_{S_2}$  los subconjuntos de la matriz  $B$  correspondiente a los arcos que forman los segmentos alternativos  $S_1$  y  $S_2$ .

Sustituyendo las ecuaciones (3.94) en (3.93) se tiene:

$$C_{S_1}^0 + B_{S_1} \cdot \Delta h_{S_1} = C_{S_2}^0 - B_{S_2} \cdot \Delta h_{S_1} \quad (3.95)$$

De forma genérica el sistema de ecuaciones propuesto se representa como  $A \cdot \Delta h_{S_1} = b$  donde:

$$\begin{aligned} A &= B_{S_1} + B_{S_2} \\ b &= C_{S_2}^0 - C_{S_1}^0 \end{aligned} \quad (3.96)$$

A lo largo de todo el proceso de cálculo del Jacobiano la matriz de coeficientes  $A$  permanece constante, ya que el conjunto PAS sigue siendo el resultante de la asignación previa sin verse alterado por pequeños incrementos del número de viajes. El incremento infinitesimal en los viajes del par O-D solo afecta a los valores de  $C_{S_1}^0$  y  $C_{S_2}^0$ . Luego para cada par O-D se actualiza el vector de términos independientes  $b$  y se resuelve el sistema de ecuaciones.

El esquema algorítmico que se implementa es el siguiente:

- Crear la matriz  $\frac{\partial v}{\partial g}$  que será la salida del algoritmo.
- Definir el conjunto  $Q$  con todos los PAS resultantes de la asignación previa.
- Calcular  $A$  según la expresión (3.96).
- Calcular  $pinvA = pseudoinversa(A)$ .
- Para cada origen  $i$ :
  - Resolver un algoritmo de ruta mínima desde  $i$  hacia todos los destinos.
  - Para cada destino  $j$  diferente de  $i$ :
    - Definir el par O-D  $p$  correspondiente al origen  $i$  y destino  $j$ .
    - Crear el vector  $\Delta v$  que incluye todos los arcos del problema, inicialmente todos sus elementos son nulos.
    - Aplicar un incremento infinitésimo de viajes al camino de coste mínimo que conecta el par O-D  $p$ .
    - Actualizar el coste de los arcos que han aumentado su flujo.
    - Para cada PAS  $q$ 
      - Calcular el coste de los segmentos

$$C_S = \sum_{a \in S} s(v_a) \quad S \in \{S_1, S_2\} \equiv PAS_q$$

- Calcular el vector independiente  $b$  según la expresión (3.96).

- Calcular para el conjunto PAS

$$\Delta h = \text{pinv}(A) \cdot b$$

- Calcular  $\Delta v = \Lambda_{S_1} \cdot \Delta h_{S_1} + \Lambda_{S_2} \cdot \Delta h_{S_2}$
- Asignar  $\frac{\partial v}{\partial g_p} \leftarrow \Delta v$

Los tiempos de cálculo del algoritmo se muestran en la Tabla 3-14.

Red Transporte	Cálculo del Jacobiano (RGap=1e-10)	Problema de Asignación (RGap=1e-10)
Tiergarten	0,30	0,54
Chicago Sketch	5,3	31,6
Madrid	4.196,6	4.049

**Tabla 3-14. Tiempos de computación (segundos) correspondientes al algoritmo 4.**

En este caso la reducción de los tiempos en el cálculo del Jacobiano es extraordinaria. Los nuevos tiempos son muy inferiores, mejoran incluso los tiempos del problema de asignación. Así la aplicación de esta metodología en redes de tamaño medio es perfectamente plausible. Además, el algoritmo implementado incluso permite abordar un proceso de ajuste de una red de grandes dimensiones como es Madrid.

Esta mejora se ha alcanzado gracias a que con un sistema de ecuaciones de dimensiones razonables se equilibran todos los segmentos en una misma iteración. Con los anteriores procedimientos, cada par O-D se equilibra de forma individual y al existir influencias entre los diferentes pares O-D es necesaria la implementación de algoritmos iterativos.

En la Figura 3-10 se presenta la evolución de la función objetivo en el ajuste de la matriz de viajes para la red de Tiergarten. Las características del ajuste son las empleadas en el apartado 3.9.1.

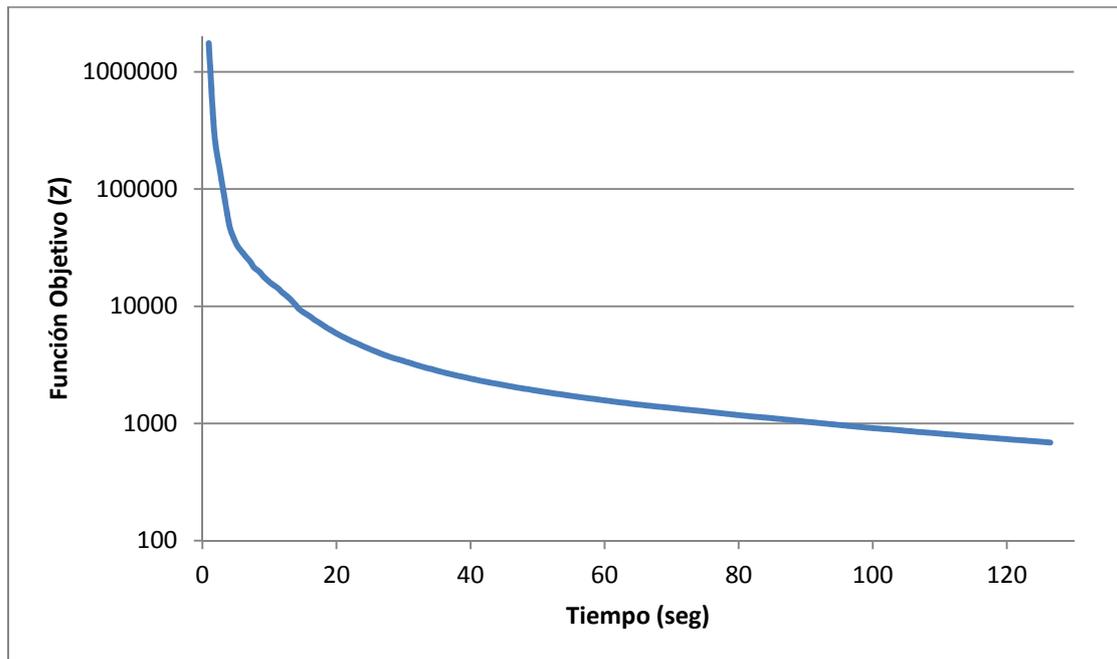


Figura 3-10. Algoritmo 4 - Evolución función objetivo.

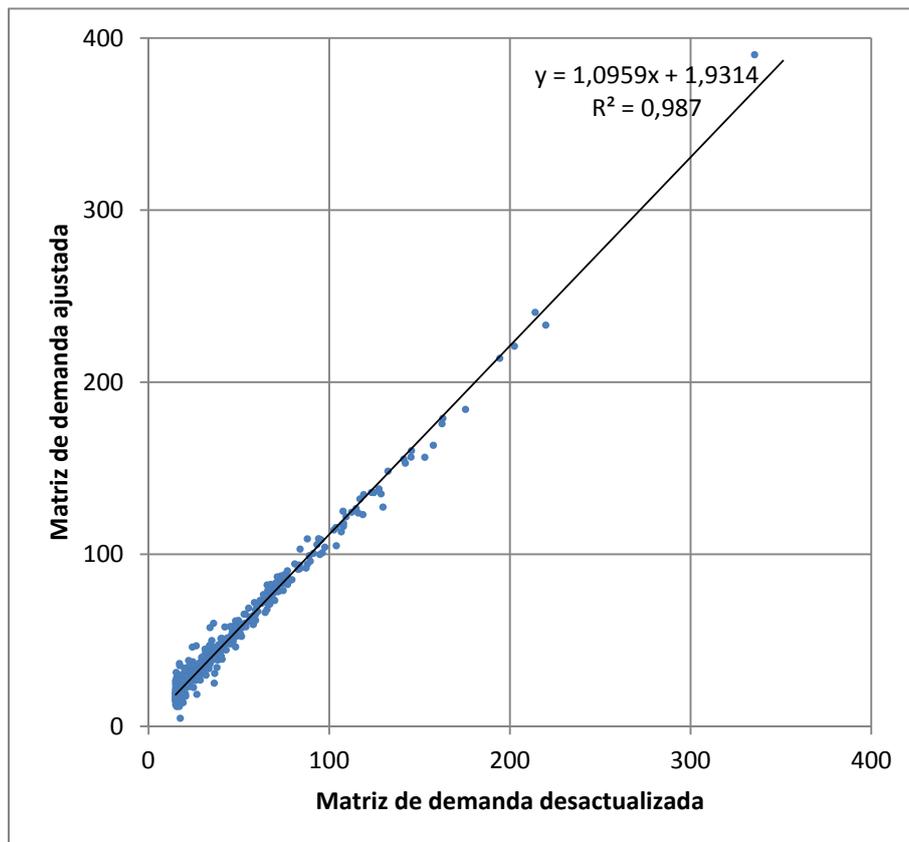


Figura 3-11. Algoritmo 4 - Matriz desactualizada frente a ajustada.

### 3.9.5 Análisis de algoritmos implementados

Se ha llevado a cabo la implementación de cuatro algoritmos diferentes y se ha realizado un ajuste sobre la red de Tiergarten con cada algoritmo. En este apartado se comparan los resultados obtenidos.

La principal diferencia que presentan los algoritmos entre sí es la metodología para el cálculo del Jacobiano. Cada metodología obtiene de distinta forma el mismo Jacobiano definido en 3.8.4. Esto hace que todos los algoritmos encuentren similares direcciones de búsqueda. En la Figura 3-12 se observa como todos los algoritmos repiten la evolución de la función objetivo en cada iteración.

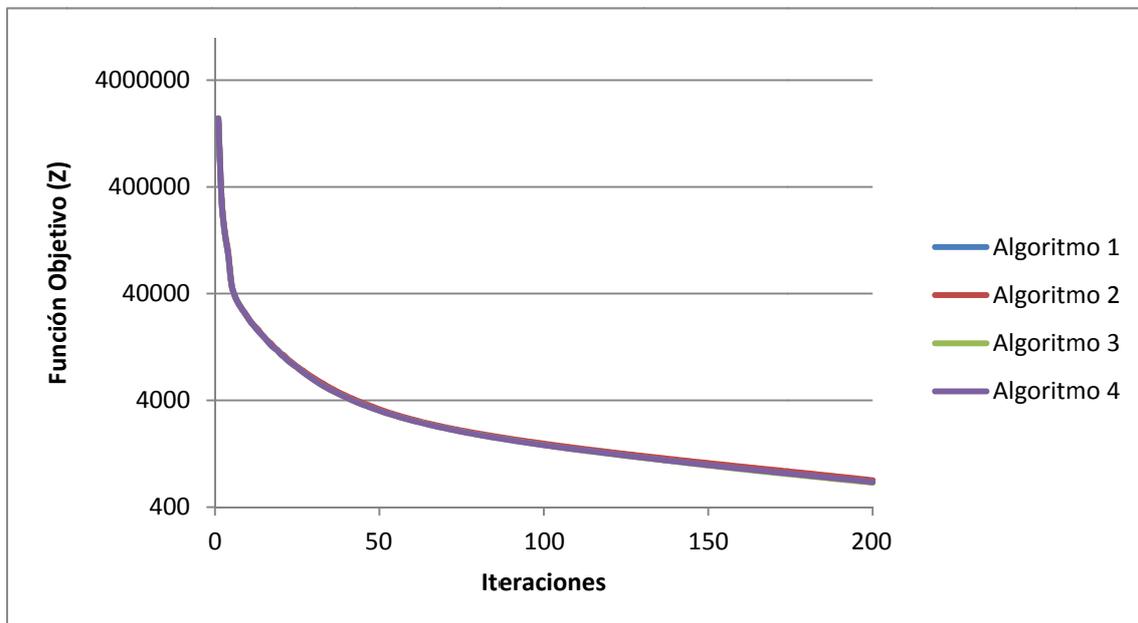


Figura 3-12. Análisis algoritmos. Función objetivo - iteraciones.

Aunque todos los algoritmos son válidos (generan igual solución) existen grandes diferencias en los tiempos de ejecución. En la Figura 3-13 se muestran las diferencias en los tiempos de computación de los procesos de ajuste; entre el algoritmo más lento y el más rápido hay diferencias de un 3000%.

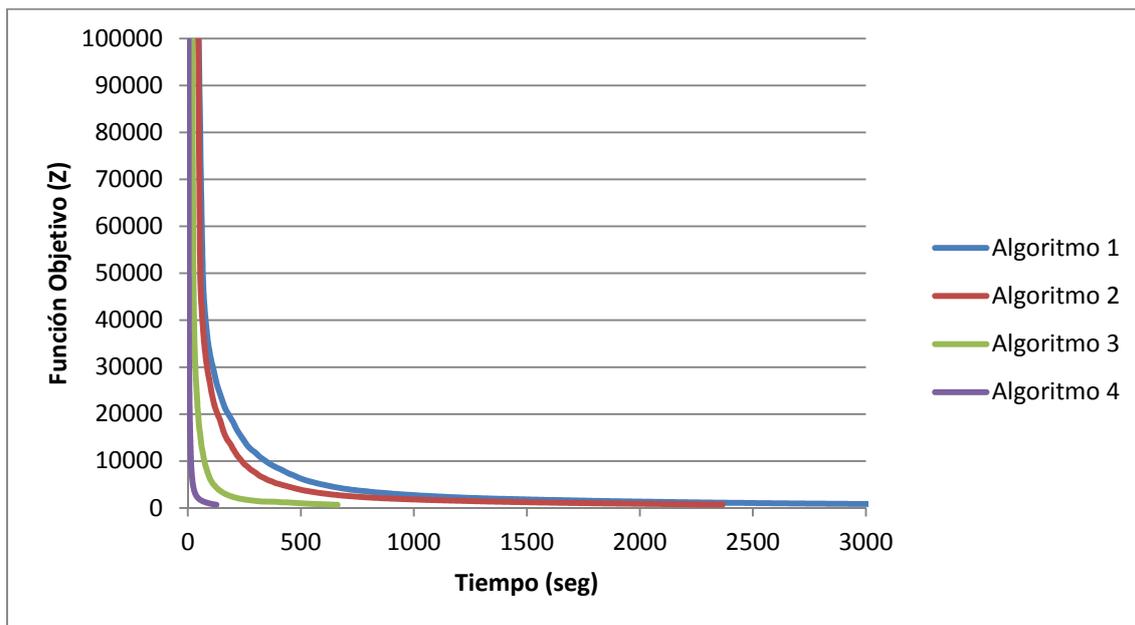


Figura 3-13. Análisis algoritmos. Función objetivo – tiempo de computación.

El algoritmo 4 realiza 200 iteraciones en menor tiempo que los demás. A continuación se presenta la Tabla 3-15 donde se comparan las diferentes metodologías en cuanto a: tiempos de ejecución del problema de asignación, tiempos de cálculo de un Jacobiano y tiempos del proceso de ajuste. Las casillas que no presentan datos se deben a que no es necesario su cálculo o son tiempos excesivos.

Método	Red Transporte	Problema de Asignación (RGap=1e-10)	Cálculo del Jacobiano (RGap=1e-10)	Cálculo del Jacobiano (RGap=1e-6)	Proceso ajuste (200 iteraciones)
Basado en caminos	Tiergarten	2,60	12,36	-	3.414
	Chicago Sketch	57,40	>15.000	8.159	-
Algoritmo de Dial	Tiergarten	1,23	9,63	-	2.366
	Chicago Sketch	32,26	>15.000	7.869	-
Bar-Gera Segmentos	Tiergarten	0,54	4,66	-	662
	Chicago Sketch	31,60	6.496	1.332,5	-
Bar-Gera Sistema	Tiergarten	0,54	0,30	-	127
	Chicago Sketch	31,60	5,30	-	11.526
	Madrid	4.196,6	4.049	-	-

Tabla 3-15. Resumen del tiempo de ejecución (segundos) de los algoritmos implementados.

En la última columna se muestran los tiempos que emplean los procesos de ajuste. Se destaca que la metodología nombrada como *Bar-Gera Sistema* reporta resultados en tiempos muy

asequibles. Con este algoritmo es posible llevar a cabo procesos de ajuste para la red de Chicago Sketch y Madrid. Por tanto, puede concluirse que esta metodología es válida para redes de grandes dimensiones.

Los tiempos de cálculo del Jacobiano en los algoritmos 1 y 2 no fueron los esperados. Inicialmente se tenía la idea de que al dar incrementos infinitésimos a cada par O-D el cálculo de cada columna sería muy rápido. Partiendo de la solución del problema de asignación con *relative gap* de  $10^{-10}$ , al dar un incremento infinitésimo de los viajes de un par O-D se obtiene una solución con *relative gap* de  $10^{-5}$  que es necesario equilibrar para alcanzar el nivel de convergencia previo. En el apartado 6.5 de Bar-Gera (2010) se habla de acoplamiento extremo. El autor describe un comportamiento peculiar encontrado en las redes de gran escala. Este comportamiento se da cuando el nivel de convergencia es alto y, por tanto, las diferencias de coste entre las diferentes rutas que conectan un par O-D son pequeñas. En esta situación el equilibrado de los PAS se hace menos efectivo y es necesario aumentar considerablemente el número de iteraciones para mejorar los niveles de convergencia. Este comportamiento puede ser la explicación de por qué los algoritmos 1, 2 y 3 realizan una gran cantidad de iteraciones para cada columna del Jacobiano.

Durante el desarrollo de esta tesis la implementación de cada nuevo algoritmo ha llevado a una mejora de los tiempos de ejecución del anterior. Cada algoritmo supera al anterior principalmente porque es más eficiente en el cálculo de cada columna del Jacobiano. Esta eficiencia se debe en gran medida a que cada algoritmo, aunque realiza aproximadamente el mismo número de iteraciones, implementa una gestión más optimizada de los datos que sus predecesores.

Con el algoritmo 4 se ha conseguido evitar el equilibrado de cada columna del Jacobiano a través de métodos iterativos. Sin embargo, plantear y resolver un sistema de ecuaciones de grandes dimensiones consume muchos recursos. La principal característica del algoritmo 4 es que resuelve cada sistema de ecuaciones empleando la pseudoinversa, que es idéntica para todos los sistemas planteados. Así, obteniendo la pseudoinversa al inicio del algoritmo, el equilibrado de cada columna del Jacobiano se reduce al cálculo del vector independiente y su multiplicación por la pseudoinversa.

### **3.10 RESULTADOS/APLICACIONES**

Una vez analizada la eficiencia computacional de los diferentes algoritmos propuestos para el cálculo del Jacobiano, se selecciona como mejor solución algorítmica el denominado Bar-Gera

Sistema, descrito en el apartado 3.9.4. Éste emplea los tiempos de ejecución más bajos para resolver la formulación de diferencias finitas propuesta en 3.8.4.

A continuación se presentan varias aplicaciones en las que se estudian las bondades que ofrece la metodología de resolución que se propone en 3.6.

### 3.10.1 Ejemplo de aplicación 1: Red sencilla

En este apartado se resuelve el problema analizado en la sección 3.8.1.1 (Ejemplo 1). Para el cálculo de la dirección de descenso se emplea la formulación de diferencias finitas propuesta en esta tesis (apartado 3.8.4). La función objetivo que se emplea es la formulada por Spiess (1990)

$$Z(v(g)) = \frac{1}{2}(v(g) - \bar{v})^2 \quad (3.97)$$

La metodología de resolución lleva a cabo un proceso iterativo en el que cada iteración aportará una nueva matriz ajustada mediante la siguiente expresión:

$$g^{n+1} = g^n + \lambda \cdot \frac{\partial v_a^T}{\partial g}(v_a(g^n) - \bar{v}_a) \quad (3.98)$$

Para el cálculo del paso  $\lambda$  puede implementarse un proceso de búsqueda lineal aunque, en este caso, como se parte de una función objetivo sencilla, es posible alcanzar una expresión analítica para el cálculo del paso, como se desarrolla en los apartados 3.6.1 y 3.6.2.

$$v^{n+1} = v^n + \frac{\partial v}{\partial g}(g^{n+1} - g^n) = v^n + \lambda \cdot \frac{\partial v_a}{\partial g} \cdot \frac{\partial v_a^T}{\partial g}(v_a - \bar{v}_a) \quad (3.99)$$

$$\lambda = \frac{\sum_a (\bar{v}_a - v_a) \cdot v'_a}{\sum_a v_a'^2} \quad (3.100)$$

donde  $v' = \frac{\partial v_a}{\partial g} \cdot \frac{\partial v_a^T}{\partial g}(v_a - \bar{v}_a)$ . La expresión (3.100) es la particularización para la función objetivo (3.97) de la expresión genérica (3.35).

El valor de  $\lambda \cdot \frac{\partial Z}{\partial g}$ , es decir, el valor del desplazamiento global de la matriz O-D, debe ser lo suficientemente pequeño para que las hipótesis de linealización de costes necesarias para obtener la expresión del Jacobiano  $\partial v / \partial g$  continúen siendo válidas. El empleo de una

expresión analítica reduce el coste computacional del problema ya que, si se utiliza un método de búsqueda lineal, cada iteración de dicho método necesita resolver un problema de asignación.

**Primera iteración:**

$\frac{\partial v_5}{\partial g_i}$	Valor
$i = A-C$	0,681914
$i = A-D$	-0,136383
$i = B-C$	0,784171
$i = B-D$	-0,0240869

Tabla 3-16. Ejemplo de aplicación 1.  $\frac{\partial v_5}{\partial g_i}$ .

$\frac{\partial Z(g)}{\partial g_i}$	Valor
$i = A-C$	-163,66
$i = A-D$	32,732
$i = B-C$	-188,2
$i = B-D$	5,781

Tabla 3-17. Ejemplo de aplicación 1.  $\frac{\partial Z(g)}{\partial g_i}$ .

La nueva matriz de viajes es:

$$g^1 = g^0 \left( 1 - \lambda^* \cdot \frac{\partial Z(g)}{\partial g} \Big|_{g^0} \right) = \begin{pmatrix} 545 \\ 171 \\ 169 \\ 295 \end{pmatrix}$$

Para resolver la expresión anterior es necesario calcular previamente  $v_a' = 263,79$  y  $\lambda^* = 0,89$ .

Con la primera iteración se encuentra una solución óptima, recogida en la Tabla 3-18.

Par O-D	Matriz Inicial	Matriz Final
A-C	400	545
A-D	200	171
B-C	0	169
B-D	300	295

Tabla 3-18. Ejemplo de aplicación 1. Solución inicial y final.

La nueva situación en equilibrio de usuario es:

Flujo caminos	Flujo arcos	Coste caminos
$h_1 = 413$	$v_1 = h_1 = 413$	$C_1 = C_2 = U_{A-C}$
$h_2 = 132$	$v_2 = h_2 + h_3 = 303$	$C_3 = U_{A-D}$
$h_3 = 171$	$v_3 = h_4 + h_5 = 245$	$C_4 = U_{B-C}$
$h_4 = 169$	$v_4 = h_2 + h_3 + h_4 + h_5 = 548$	$C_5 = C_6 = U_{B-D}$
$h_5 = 76$	$v_5 = h_2 + h_4 = 301$	
$h_6 = 219$	$v_6 = h_3 + h_5 = 247$	
	$v_7 = h_6 = 219$	

Se observa como el flujo de  $v_5$  es muy aproximado al volumen aforado. Además, puede observarse que el par  $g_{B-C}$  es tenido en cuenta aún sin realizar una precarga inicial. Esta propiedad es muy importante ya que al no necesitar una precarga inicial de la matriz de viajes, no se satura innecesariamente la red de transporte.

Generalmente es inevitable realizar más iteraciones; el hecho de que en este caso sólo sea necesaria una única iteración es porque el desplazamiento de la matriz de viajes no es excesivo y, por tanto, se cumple en todo momento la hipótesis de linealización de los costes en caminos realizada en el análisis de sensibilidad.

En la Tabla 3-19 se comparan las soluciones obtenidas para este problema. Además se adhieren tres soluciones; dos casos en los que se emplea el método propuesto por Spiess (1990) precargados con uno y diez viajes las casillas de la matriz de partida con valor nulo, y un tercer caso en el que se emplea la formulación de diferencias finitas propuesta en esta tesis pero aplicando la función objetivo de tipo entrópico siguiente:

$$Z(v(g)) = v(g) \ln \left( \frac{v(g)}{\bar{v}} \right) - v(g) + \bar{v} \tag{3.101}$$

En adelante se emplea la denominación *método de probabilidad localmente constante* para referirse a la metodología propuesta por Spiess (1990) ya que es la principal hipótesis que adopta dicho autor para el cálculo del gradiente.

Par O-D	Matriz Inicial	Matrices Estimadas				
		Prob. L. Constante	Prob. L. Constante con Precarga 1	Prob. L. Constante con Precarga 10	Dif. Finitas Mínimos cuadrados	Dif. Finitas máxima entropía
A-C	400	767	767	749	545	545
A-D	200	200	200	200	171	171
B-C	0	0	0	13	169	168
B-D	300	300	300	300	295	295

**Tabla 3-19. Ejemplo de aplicación 1. Matrices O-D estimadas con los diferentes métodos.**

Los métodos de probabilidad localmente constante sin precarga y con precarga 1 proporcionan la misma solución. En ésta sólo se modifican los viajes del par  $g_{A-C}$ . El método de Spiess únicamente ajusta los pares O-D que han sido interceptados por algún aforo; no considera las influencias que tiene cada par O-D sobre los aforos que no son interceptados por sus caminos.

Si las casillas nulas de la matriz de viajes no son precargadas inicialmente, el método de probabilidad localmente constante no las tiene en cuenta. Además, según las matrices obtenidas en los casos donde las casillas nulas se precargan con 1 y 10 viajes, la precarga de la matriz debe ser considerable para que el algoritmo actúe sobre los pares O-D sin viajes. Esto se debe a que el algoritmo sigue una evolución oscilatoria. A modo de ejemplo, en la Figura 3-3 se representa el valor de  $v_5$  en cada iteración.

El número de iteraciones necesarias para alcanzar la convergencia en los 3 primeros métodos es 10. Para los métodos de diferencias finitas propuestos, mínimos cuadrados y máxima entropía, sólo es necesaria una única iteración ya que el problema es muy simple y la aproximación del Jacobiano aporta valores reales para el desplazamiento de la matriz O-D durante el ajuste.

En la Tabla 3-19 se comprueba que, para ajustar el flujo aforado en el arco  $v_5$ , los métodos propuestos han modificado los valores de todos los pares O-D. Esto se produce porque los métodos que emplean diferencias finitas consideran las influencias sobre el aforo desde todos los pares O-D. Además se observa que las matrices O-D ajustadas coinciden.

El número de viajes a aumentar en el arco  $v_5$  es de 240. El método de Spiess ha añadido 367 viajes a la matriz mientras que los métodos propuestos han aumentado el total de viajes de la

matriz en 279. Estos últimos resuelven el problema del ajuste produciendo desplazamientos menores en la matriz de viajes.

### 3.10.2 Ejemplo de aplicación 2: Red de Tiergarten

En este apartado se aplica el procedimiento de estimación propuesto a una zona de Berlín llamada Tiergarten. Esta red ha sido presentada en el apartado 2.8.1.1. Es una red de pequeño tamaño con 650 pares O-D (26 centroides), 747 arcos y 344 nodos.

En este caso no se conocen los valores de aforo reales por lo que es necesario llevar a cabo algún proceso de simulación que permita obtener tales datos. Para ello se sigue el siguiente procedimiento: *i)* Se supone una matriz de viajes como real, ésta incluye todas las decisiones de viajar de los usuarios de la red, *ii)* asignando dicha matriz es posible conocer los valores de aforo que se están dando en la red, *iii)* obtenidos los aforos, se lleva a cabo una desactualización hipotética de la matriz de viajes real, este proceso de distorsión consiste en multiplicar la matriz real por una matriz de coeficientes aleatorios entre 0,75 y 0,95, *iv)* a partir de la matriz desactualizada, el método de estimación ajustará los valores de flujo en los arcos aforados tratando de encontrar la matriz que con mayor probabilidad representa a los usuarios de la red.

Se ha tomado un número total de arcos aforados de 120. La selección de dichos arcos se ha realizado maximizando el número de pares O-D interceptados. En la Figura 3-1 se muestra la red de Tiergarten, los arcos aforados se marcan en rojo (trazado grueso).

Se obtienen resultados aplicando tres metodologías de estimación diferentes:

- Metodología propuesta empleando diferencias finitas para el cálculo del gradiente y una función objetivo de mínimos cuadrados (D. Finitas - Min C.).
- Metodología propuesta empleando diferencias finitas para el cálculo del gradiente y una función objetivo de maximización de entropía (D. Finitas - Entropía).
- Metodología empleando la hipótesis de probabilidad localmente constante de los caminos para el cálculo del gradiente (Spiess, 1990; Yang et al., 1992) (Prob. Constante).

Las tres metodologías utilizan como método de optimización el método del gradiente aunque difieren en su cálculo. La metodología de mínimos cuadrados y el método de probabilidad localmente constante comparten igual función objetivo. La función objetivo que implementa la metodología de maximización de entropía es diferente ya que afronta el problema de ajuste desde una perspectiva diferente. Para poder comparar los tres métodos se evalúa la función

objetivo de mínimos cuadrados en las diferentes iteraciones de la metodología de maximización de entropía. El resultado se muestra en las Figura 3-14 y Figura 3-15.

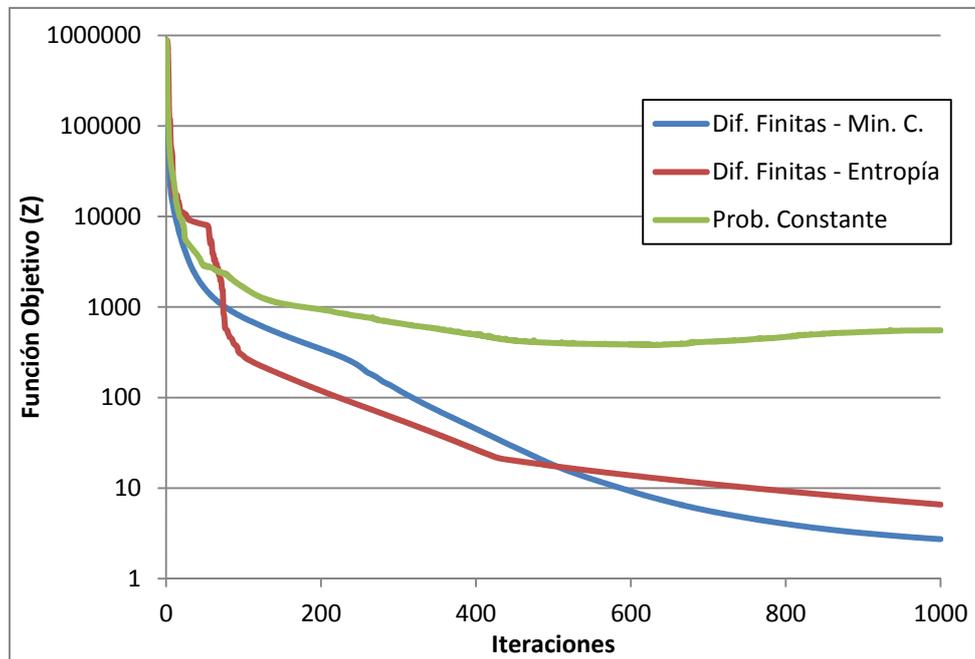
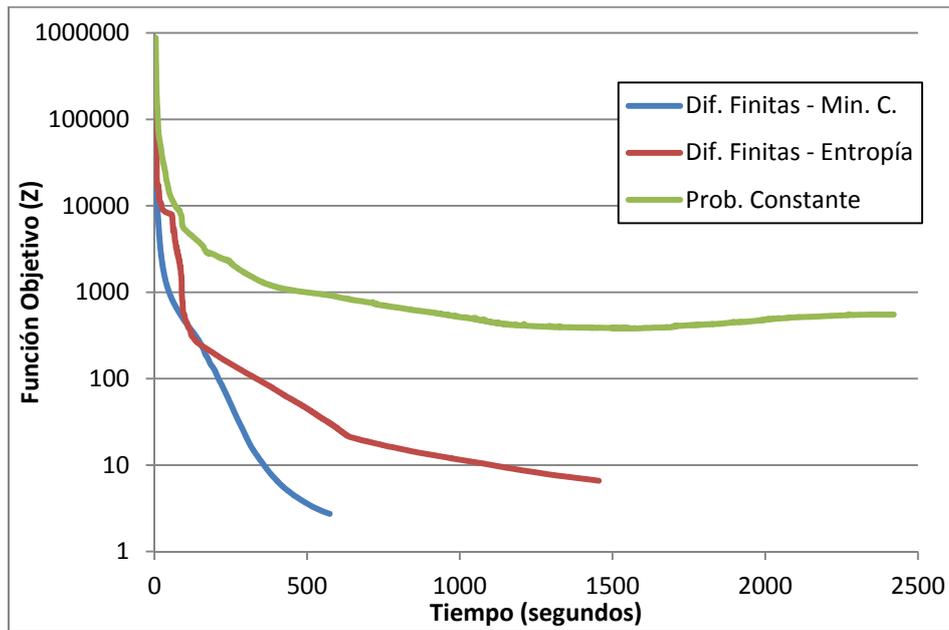


Figura 3-14. Ajuste Red de Tiergarten. Función objetivo – iteraciones.

En total se han realizado 1000 iteraciones en el proceso de ajuste. Las metodologías propuestas alcanzan valores muy bajos de la función objetivo. El método de probabilidad localmente constante llega a un mínimo en la iteración 628 y a partir de entonces obtiene peores soluciones en cada iteración.

Comparando los tiempos de computación mostrados en la Figura 3-15, la metodología de mínimos cuadrados ha alcanzado tiempos muy inferiores a los demás. La diferencia de tiempos entre las tres metodologías es debida principalmente a los tiempos de asignación en cada iteración. El método de Spiess utiliza un algoritmo de asignación basado en caminos que en general es más lento que el método basado en orígenes empleado en las metodologías propuestas. La diferencia de tiempos entre las metodologías propuestas se debe a las soluciones intermedias alcanzadas. Diferentes matrices de viajes llevan al algoritmo de asignación a diferentes tiempos de ejecución. En este caso cada metodología ha progresado por matrices de viajes completamente diferentes y los tiempos de asignación en la metodología de mínimos cuadrados han sido inferiores. Puede ocurrir que con otra selección de arcos aforados u otra red de transporte se intercambien los tiempos mínimos de ajuste.



**Figura 3-15. Ajuste Red de Tiergarten. Función objetivo – tiempo de computación.**

Al no conocer valores de aforo para esta red se ha llevado a cabo el proceso de simulación anteriormente descrito. En este proceso se supone conocida la matriz de viajes que incluye los viajes de todos los usuarios nombrada como matriz real. Esta matriz solo se emplea para el cálculo de los volúmenes de aforos. El proceso de ajuste parte de una matriz desactualizada y trata de ajustar los aforos. Debido a la indeterminación no es posible que la matriz ajustada (solución del problema de ajuste) se asemeje a la matriz real. Aun así se compara la evolución de la distancia entre la matriz ajustada en cada iteración y la matriz real.

Para el cálculo de la distancia entre matrices se ha analizado el comportamiento de diferentes medidas como son: la suma de las diferencias en valor absoluto de los elementos de las matrices, la suma de las diferencias en valor relativo y la entropía relativa. Empleando el valor absoluto tiene mucha importancia en el resultado las casillas de mayor valor y no se controla la distorsión de las casillas con bajo número de viajes en la matriz de partida. Empleando el valor relativo se le da mucha importancia a las casillas con menor valor. En este caso las casillas que parten con cero viajes darán valores infinitos. Los resultados obtenidos empleando la entropía relativa no se ven tan influenciados por la cantidad de casillas que parten con bajo o alto número de viajes como en las anteriores distancias y, por eso, será la medida que se emplee en el cálculo de la distancia entre matrices.

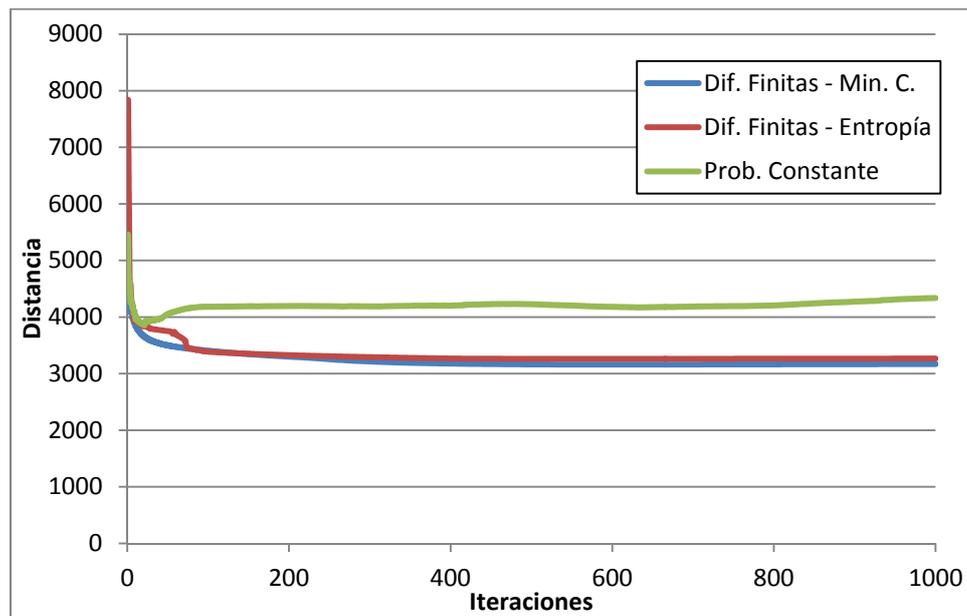


Figura 3-16. Ajuste Red de Tiergarten. Distancia entre matriz ajustada y matriz real.

Se observa una aproximación hacia la matriz real en cada método de ajuste. A partir de la iteración 300 se aprecia una estabilización de las soluciones a una cierta distancia de la real. Los algoritmos que usan diferencias finitas en el cálculo del gradiente generan soluciones más cercanas a la real.

Otra característica importante en los métodos de ajuste es la distorsión de la matriz de partida. Se parte de una matriz desactualizada en la que no se encuentran reflejados los datos de viajes de todos los usuarios de la red y por tanto su asignación no replica los volúmenes medidos en los puntos de aforo. La obtención de esta matriz supone un gran esfuerzo económico y generalmente se estima que sus elementos tienen una alta fiabilidad. Por ello es importante que el ajuste no imprima una distorsión excesiva a los valores de la matriz.

En la Figura 3-17 se muestra como las metodologías propuestas minimizan las discrepancias de los aforos llevando la matriz de viajes ajustada respecto de la de partida a una distancia inferior a la que imprime el método de probabilidad localmente constante.

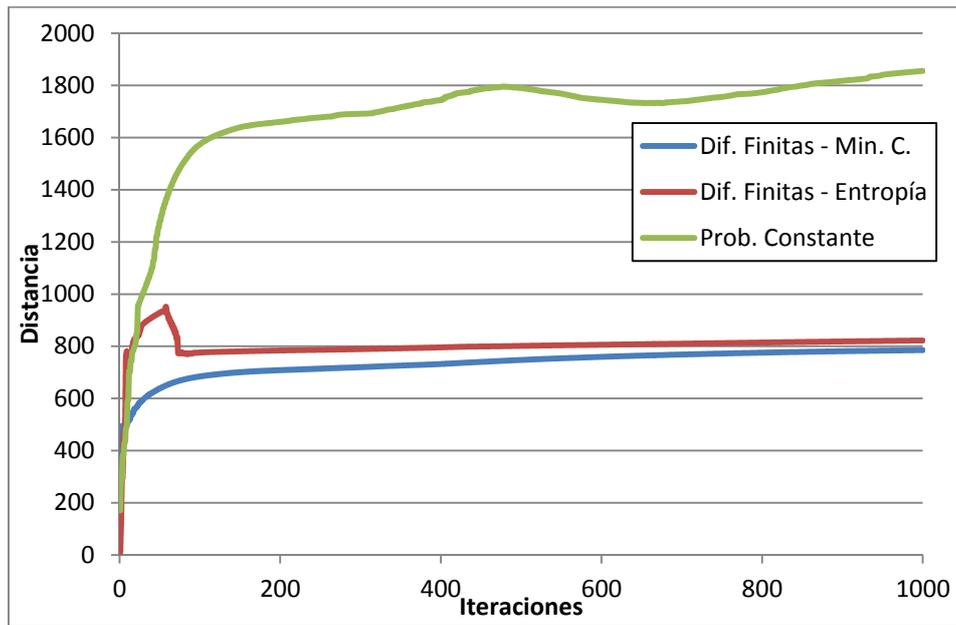


Figura 3-17. Ajuste Red de Tiergarten. Distancia entre matriz ajustada y matriz desactualizada.

Otra medida de distorsión de la matriz de partida es la correlación entre matrices.

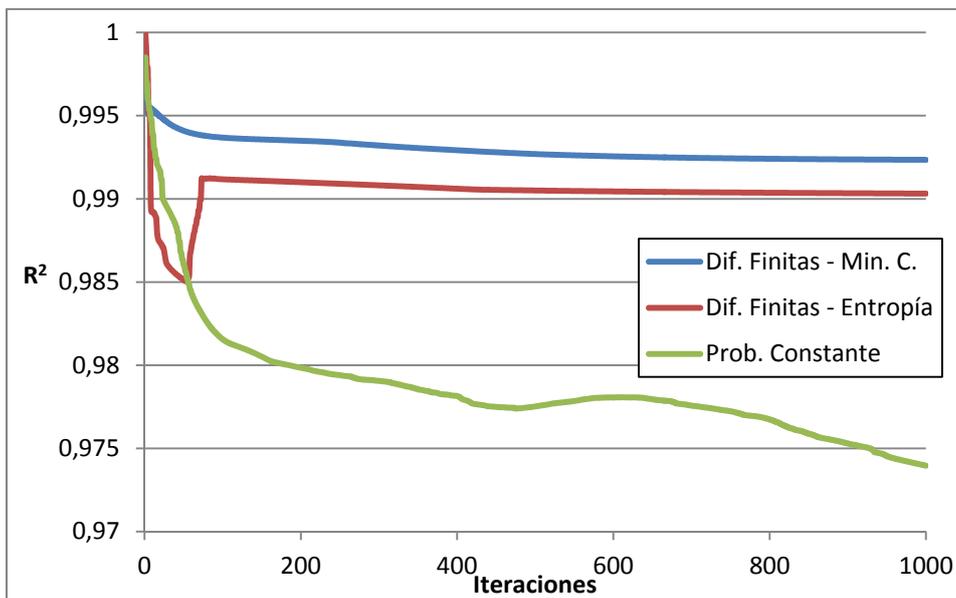


Figura 3-18. Ajuste Red de Tiergarten. Correlación matriz ajustada y matriz desactualizada.

Los valores de  $r^2$  son muy elevados en los tres casos por lo que se concluye que esta unidad de medida no es válida para medir la distorsión de la matriz de partida. Aun así puede observarse, al igual que en la Figura 3-17, las tendencias del comportamiento de cada uno de los métodos implementados.

Estudiando elemento a elemento las matrices solución de cada algoritmo es posible destacar ciertos datos que igualmente dan información sobre la distorsión que imprime cada método. En la Tabla 3-20 se muestran los más relevantes.

	<b>Dif. Finitas Mínimos cuadrados</b>	<b>Dif. Finitas Máxima entropía</b>	<b>Probabilidad Localmente Constante</b>
<b>Mayor variación de un elemento</b>	112%	159%	321%
<b>Valor medio de la variación de los elementos</b>	21,2%	21,4%	29,3%
<b>Suma de los incrementos en casillas</b>	3.642,60	3.663,03	4.286,69
<b>Suma de los decrementos en casillas</b>	-180,41	-189,81	-635,87

**Tabla 3-20. Ejemplo de aplicación 2. Datos de distorsión de las matrices ajustadas respecto de la inicial.**

En este caso al no restringir la variación de las casillas de las matrices ajustadas se alcanza hasta un 321% de incremento en el método de Spiess. Doblas y Benítez (2005), conociendo este problema, proponen un método para disminuir la distorsión de las matrices añadiendo restricciones tanto a las casillas como a filas y columnas. Las metodologías de diferencias finitas imprimen variaciones menores a las casillas aunque podría ser excesivo en ciertos proyectos y por tanto puede ser necesaria alguna metodología que restrinja la variación de las casillas. En este caso, siguiendo las directrices de Doblas y Benítez (2005), los algoritmos propuestos permiten incluir de forma directa y sin cambios metodológicos importantes restricciones que controlen dicha variación.

Puede observarse en las filas 3 y 4 de la Tabla 3-20 que el método de probabilidad localmente constante necesita, para ajustar los aforos, añadir más viajes a las casillas que incrementa y eliminar más viajes en las casillas que decrementa.

### **3.10.3 Ejemplo de aplicación 3: Red de Chicago Sketch**

A continuación se aborda un problema un tanto más complicado que los anteriores. La red de Chicago Sketch es de mediana escala y por tanto implica tiempos de ejecución superiores. Además, como se describe en el apartado 2.8.1.2, la indeterminación de este problema es muy alta y cada zona se conecta con las limítrofes a través de un único arco. Debido a esto, para llevar a cabo un problema de estimación de matrices adecuado, es necesario aforar todos los arcos de la red. En este caso solo se aforan 500 de los 2950 que componen la red por lo que no se esperan buenos resultados.

Este apartado no presenta un método de ajuste capaz de encontrar soluciones aceptables en redes muy indeterminadas ya que este tipo de problemas son inviables. Para alcanzar soluciones aceptables será necesario ejecutar previamente trabajos encaminados a disminuir la indeterminación. Lo que se pretende con este ejemplo es tratar de mostrar que puede llevarse a cabo un problema de estimación de matrices para redes de tamaño medio en un tiempo razonable. Además, se comprobará que el método actualiza de forma eficiente la matriz de viajes con el fin de minimizar la discrepancia en los aforos.

La metodología de ajuste que se aplica en este problema es la propuesta empleando diferencias finitas para el cálculo del gradiente y una función objetivo de mínimos cuadrados. Los resultados han sido obtenidos después de 200 iteraciones y 11.526 segundos de ejecución. La mayor parte del tiempo consumido se debe a la ejecución de 200 problemas de asignación.

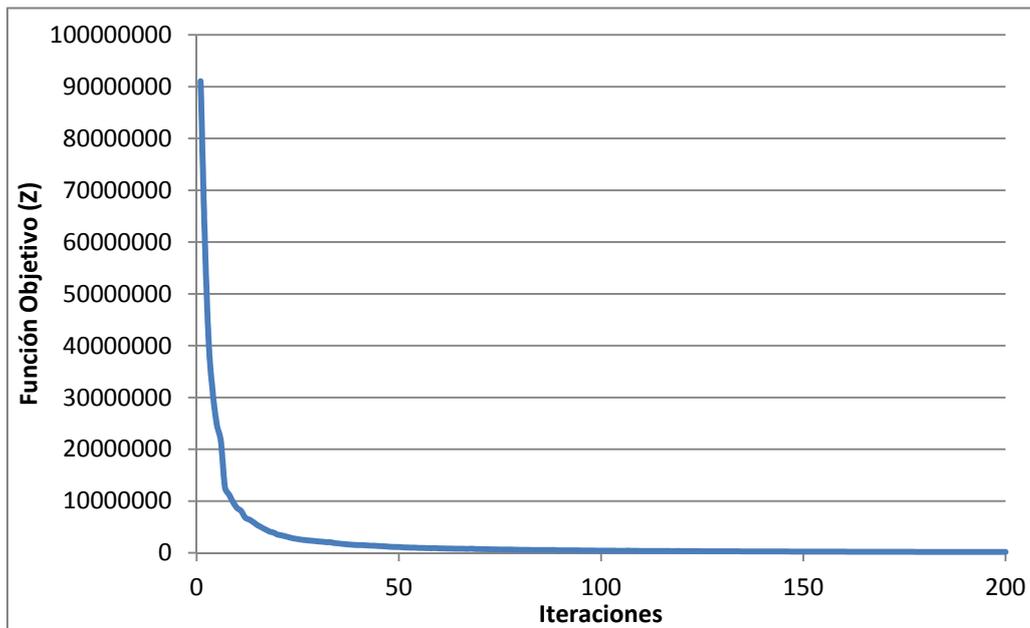


Figura 3-19. Ajuste Red de Chicago Sketch. Función objetivo - iteraciones.

La función objetivo (Figura 3-19) alcanza un mínimo razonable aunque a un precio muy alto ya que la matriz solución obtenida difiere en gran medida tanto de la matriz de partida como de la real que se supone conocida al igual que en el ejemplo anterior.

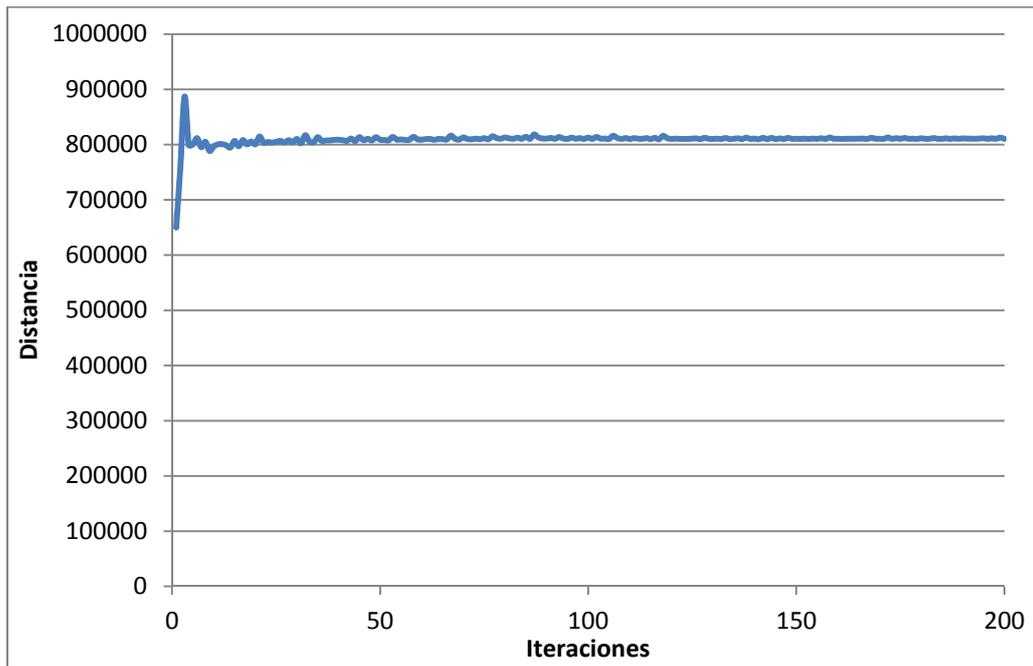


Figura 3-20. Ajuste Red de Chicago Sketch. Distancia entre matriz ajustada y matriz real.

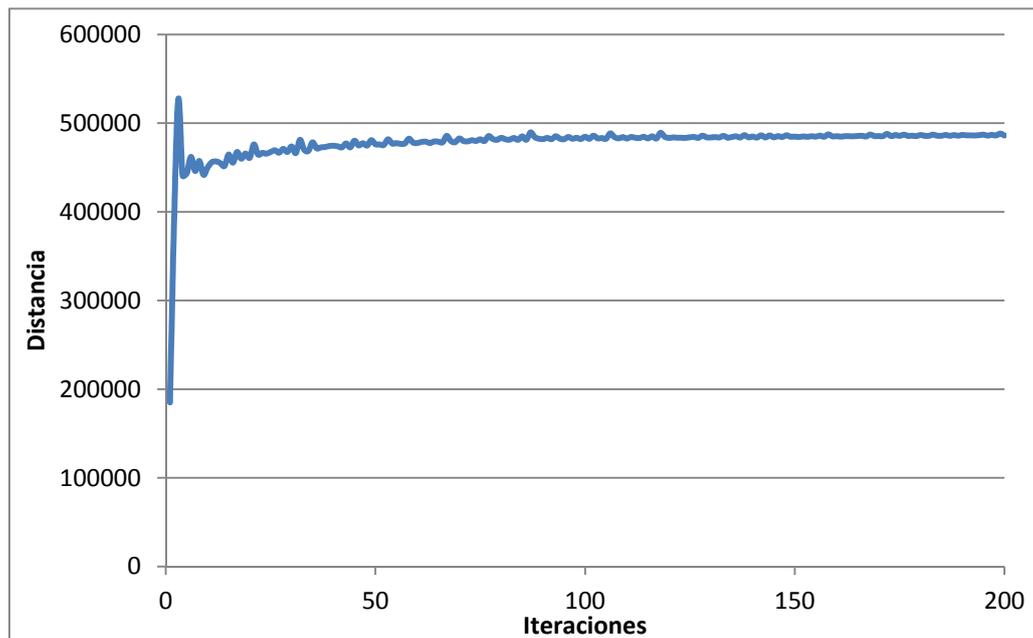


Figura 3-21. Ajuste Red de Chicago Sketch. Distancia entre matriz ajustada y matriz desactualizada.

La distancia entre la matriz ajustada y la real (Figura 3-20) no se reduce en ningún momento. Esto se debe a que la indeterminación del problema es grande e insalvable. En la Figura 3-21 se aprecia a su vez un alejamiento elevado de la matriz de partida.

Al existir una alta indeterminación, el número de matrices de viajes que ajustan el valor en los aforos es muy elevado. Con la escasa información inicial aportada, el gradiente no puede

encontrar una dirección de búsqueda que se acerque a la matriz denominada real aunque sí minimiza de forma eficiente el valor de la función objetivo. Tras pocas iteraciones de ajuste con un gradiente erróneo, el algoritmo encuentra un mínimo local de entre los casi infinitos mínimos locales.

Como conclusión conviene destacar que para que este ejemplo pueda ser abordado de una forma más eficiente es necesario disminuir el nivel de desagregación de las zonas de la red y tratar de conseguir un mayor número de arcos aforados, con lo que el nivel de indeterminación se reducirá. Solventados estos problemas se podrá llevar a cabo un ajuste de la matriz de viajes que obtendrá resultados en menos de 5 horas de ejecución.

#### 3.10.4 Ejemplo de aplicación 4: Red de Madrid

Ésta es una red de grandes dimensiones, sus características se detallan en el apartado 2.8.1.3. Al igual que en el apartado anterior, el objetivo principal es conocer la eficiencia computacional del algoritmo de ajuste propuesto. El modelo de red ha sido empleado por el Grupo de Ing. e Inf. de los Transportes de la Universidad de Sevilla en varios proyectos investigación. Los datos que se van a utilizar son los recogidos para las horas punta de mañana 8-9h y 9-10h.

La metodología de ajuste que se aplica en este problema es la propuesta empleando diferencias finitas para el cálculo del gradiente y una función objetivo de mínimos cuadrados. En total se han realizado 43 iteraciones del proceso de estimación. El tiempo empleado por el algoritmo ha sido de 5 días y 23 horas. Los resultados se muestran a continuación.

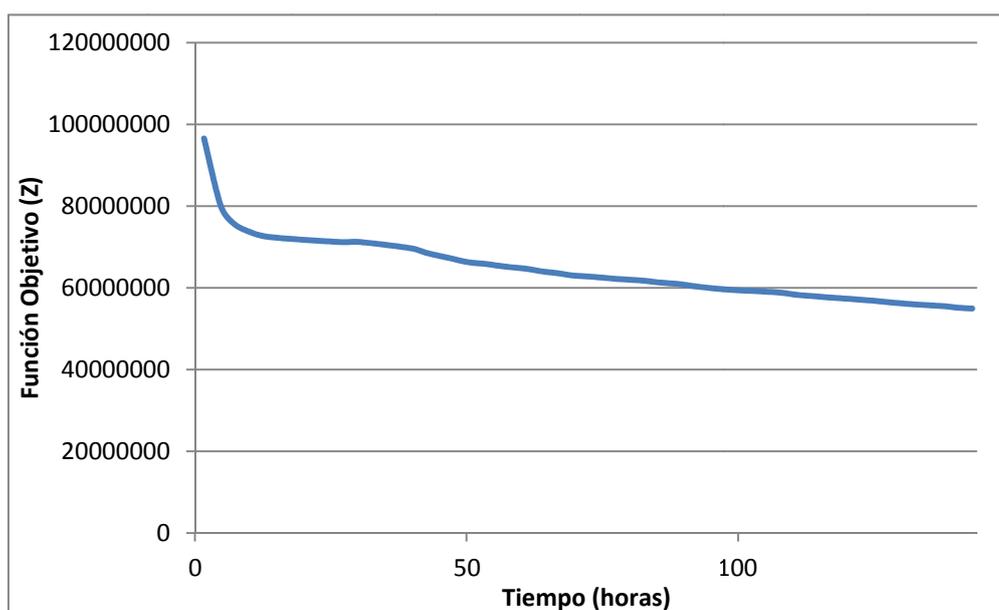


Figura 3-22. Ajuste Red de Madrid. Función objetivo – tiempo de computación.

Después de más de 140 horas de ejecución el algoritmo no ha alcanzado una solución convergente. Se aprecia, Figura 3-22, una tendencia en la que, en cada iteración, se minimiza el valor de la función objetivo. La evolución seguida por la solución es buena pero muy lenta debido a una falta de eficiencia de la dirección de búsqueda empleada.

A diferencia de los ejemplos prácticos anteriores, en este ejemplo de aplicación la dirección de búsqueda propuesta ha perdido eficiencia. El tiempo de ejecución es inevitablemente alto debido al tamaño del proyecto abordado. Aunque un gran porcentaje de este tiempo se atribuye a la falta de eficiencia de la dirección que es causada principalmente por dos factores: la inconsistencia de los aforos empleados (no se analizan en esta tesis) y la alta indeterminación que presenta el problema.

El proceso de ajuste ha tratado de actualizar el valor de 1.391.220 casillas de la matriz de partida con la información de únicamente 491 datos de aforo. La indeterminación es excesiva y es el principal motivo de falta de eficiencia del algoritmo.

La red de Madrid en estudio se ha dividido en un alto número de zonas de transporte y se quiere conocer la conexión entre todas ellas. En la práctica, debido al intervalo de tiempo considerado y a la distribución de zonas empleada, el porcentaje de casillas nulas en la matriz de viajes es muy alto. La mayoría de elementos O-D nulos, realmente, no presentan viajes porque para el intervalo de hora punta de mañana no existe la necesidad de viajar entre esas zonas. Por ejemplo, pares O-D cuyo origen sea una zona industrial y su destino sea una zona residencial. Por tanto, no es una estrategia recomendable considerar todos los pares O-D en el proceso de ajuste.

Estudiando los elementos de la matriz de partida, existen 1.086.542 pares O-D que no presentan datos y 277.280 pares O-D con valores residuales inferiores a 1. Esto se debe a que una encuesta o estudio anterior no ha recogido ningún viaje entre esos pares O-D y, además, algún proceso de ajuste ha añadido valores residuales en un subconjunto de casillas.

A continuación se aborda el problema considerando únicamente los pares O-D en los que se ha interceptado al menos un viaje durante el estudio o encuesta de generación de la matriz de partida. Para los demás pares O-D se supone que no hay usuario que necesite desplazarse entre ellos en el lapso de tiempo considerado. La indeterminación del problema sigue siendo grande pero en este caso se están actualizando las casillas más activas de la matriz de demanda.

En total se ejecutan 30 iteraciones con esta nueva estrategia de ajuste (trazado rojo, Figura 3-23) y se compara con las 30 primeras iteraciones de la metodología anterior (trazado azul, Figura 3-23).

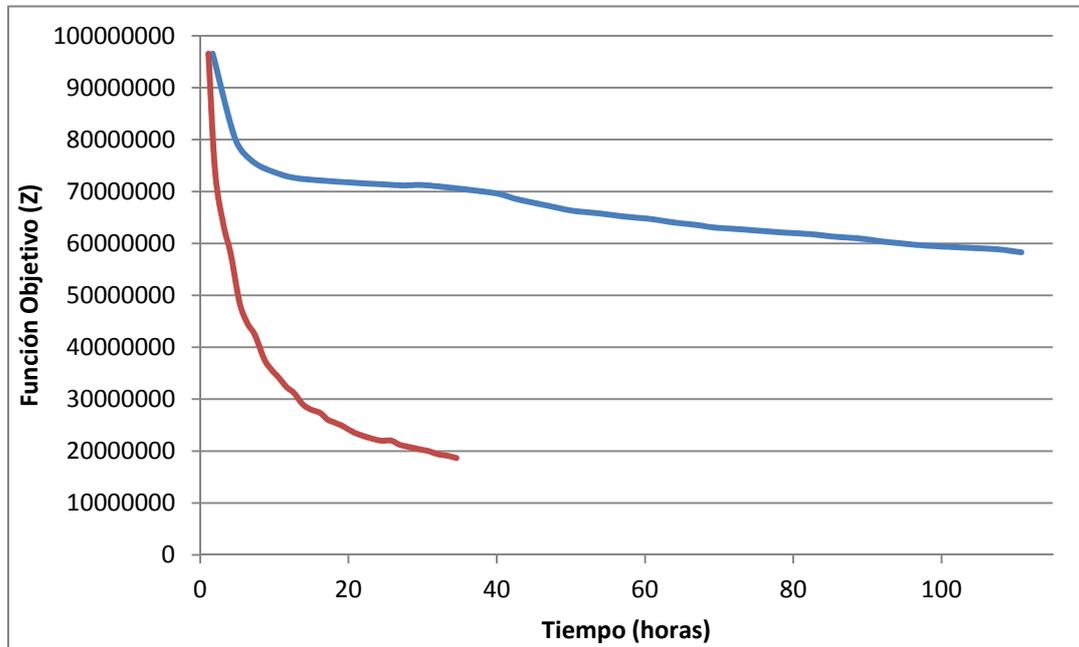


Figura 3-23. Ajuste Red de Madrid. Función objetivo – tiempo de computación.

Con esta nueva estrategia resaltan dos resultados muy significativos:

- Se obtiene una disminución del valor de la función objetivo varias veces inferior. Esto demuestra que se han empleado direcciones de búsqueda más eficientes.
- Se reduce el tiempo de ejecución del algoritmo. En los dos casos se calculan 30 direcciones de búsqueda diferentes y se resuelven 30 problemas de asignación de tráfico. La disminución del tiempo de ejecución se debe en gran medida a la reducción del tamaño del problema de cálculo de la dirección de descenso. Al considerar únicamente los pares O-D más activos, se ha reducido drásticamente el cálculo de las columnas del Jacobiano.

Con esta estrategia se está evitando el principal inconveniente en este tipo de problemas. Cuando el proyecto dispone de una gran cantidad de zonas e insuficientes valores de aforo, el ajuste de la matriz de viajes se ralentiza.

La formulación propuesta en esta tesis calcula el Jacobiano en el cual todos los elementos de la matriz de viajes presentan una dirección, incluso si los pares O-D no son interceptados por los aforos (apartado 3.8). Para el conjunto de aforos existe un número de pares O-D que son de

mayor importancia en el ajuste, los demás pares O-D influyen en menor grado. En el caso de Madrid hay más de 1.000.000 de pares O-D que afectan en menor medida al volumen de los arcos aforados, pero la dirección de estos pares no es nula y por tanto se modificará su valor en cada iteración. Ésta mínima modificación sobre la gran mayoría de casillas implica que el paso en la búsqueda monodimensional en las diferentes iteraciones sea muy bajo para no incorporar un alto número de viajes innecesarios a la red. Esta longitud del paso da lugar a una minimización de la función objetivo muy lenta.

A su vez, al restringir la actualización de la matriz de viajes a los pares O-D más activos se está logrando mantener la estructura principal de dicha matriz. Ya se ha comentado en esta tesis que en muchos proyectos la estructura de la matriz de viajes de partida es importante y por eso es prioritario que el método de ajuste no aplique distorsiones excesivas a la matriz O-D.

Aunque los resultados obtenidos al considerar los pares O-D activos mejoran a los alcanzados cuando se emplean todos los pares O-D, el valor de la función objetivo después de 30 iteraciones sigue siendo alto. Esto se debe a que existen valores de aforo erróneos e inconsistentes. La mejora de la solución final del proceso de ajuste pasa por realizar previamente un análisis de los datos de aforos empleados en el proyecto.

### **3.10.5 Mejora del proceso de ajuste para la red de Chicago Sketch**

A continuación se aplica a la red de Chicago Sketch la nueva estrategia de resolución propuesta para redes de gran escala. En el apartado 3.10.3 se concluye que es necesario reducir la indeterminación del problema para mejorar los resultados y disminuir el tiempo de ajuste. Al igual que para la red de Madrid, se va a estudiar el ajuste de la red de Chicago Sketch centrado el problema en actualizar sólo las casillas que presentan un número importante de viajes.

El total de pares O-D que se consideran en el problema son 149.382. De todos estos pares, 109.626 pares presentan menos de un viaje. Por tanto, se realizará un proceso de ajuste donde se actualizarán 39.756 elementos de la matriz de partida con datos de 500 aforos.

En la Figura 3-24 se muestran los resultados obtenidos con la nueva estrategia de ajuste (trazado rojo) y se compara con la metodología presentada en el apartado 3.10.3 (trazado azul).

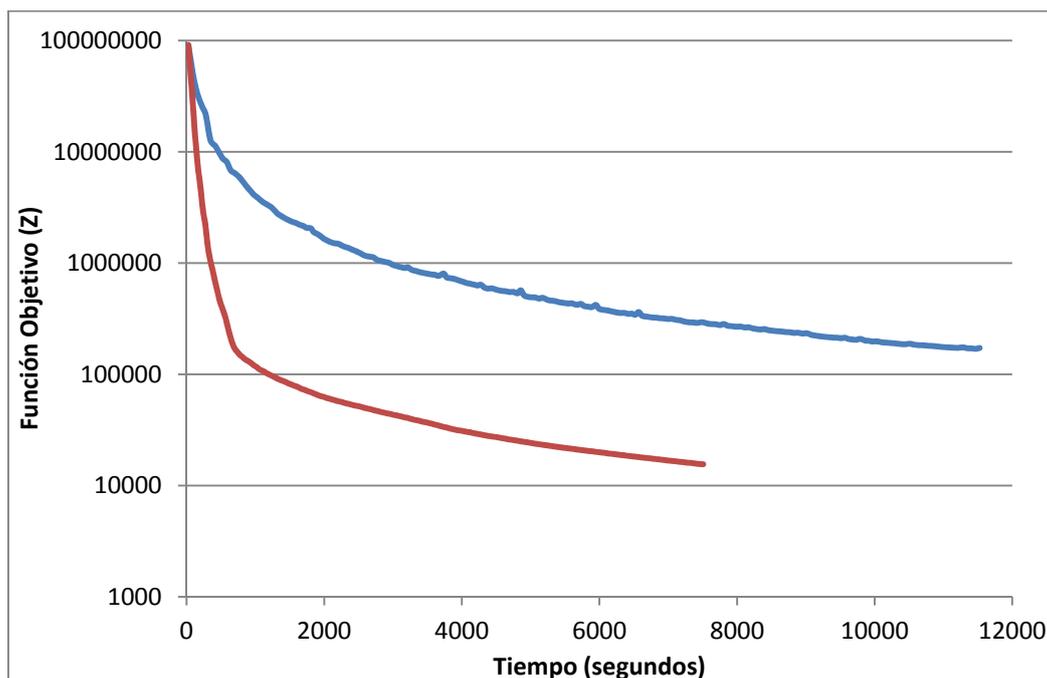


Figura 3-24. Mejora Ajuste Chicago Sketch. Función objetivo – tiempo de computación.

En los dos casos se han llevado a cabo 200 iteraciones de ajuste. Con la estrategia propuesta en este apartado se consigue alcanzar valores de la función objetivo inferiores y en menor tiempo de ejecución.

En el ajuste de la red de Chicago Sketch, el tiempo de ejecución no se ha reducido en las proporciones que se redujo el ejemplo de la red de Madrid ya que el tiempo del problema de asignación de tráfico constituye un gran porcentaje del proceso de ajuste. En la Tabla 3-21 se muestran los porcentajes que representan el problema de asignación y el cálculo del Jacobiano respecto del total del proceso de ajuste en ambos casos.

	Red de Chicago Sketch		Red de Madrid	
	Todos pares O-D	Pares O-D >1	Todos pares O-D	Pares O-D >1
<b>Asignación Tráfico</b>	77,1%	91,1%	55,2%	92,5%
<b>Cálculo Jacobiano</b>	21,5%	8%	42,1%	6,9%
<b>Tiempo total (seg.)</b>	11.525,6	7.508,15	398.721,2	52.888,2

Tabla 3-21. Mejora Ajuste Chicago Sketch. Porcentajes respecto del tiempo total de ajuste.

En ambos casos se reducen los tiempos de ejecución del algoritmo. La disminución de tiempo más importante ocurre en el cálculo del Jacobiano ya que se ha reducido el número de pares O-D a ajustar. En la Tabla 3-21 se comprueba como el problema de asignación de tráfico

constituye un gran porcentaje dentro de los procesos de ajuste binivel. Los tiempos de asignación han experimentado una muy leve disminución empleando la nueva estrategia.

Finalmente se analiza, para la red de Chicago Sketch, el desplazamiento de la matriz ajustada respecto de la matriz de partida. El objetivo en el problema de estimación de matrices, además de ajustar los valores de aforo en los arcos, es aplicar la menor distorsión posible a la matriz de viajes de partida. Para medir dicha distorsión se emplean varios parámetros que se encuentran detallados en el apéndice I.

	<b>Pares O-D &gt;1</b>	<b>Todos pares O-D</b>
<b>Entropía relativa</b>	81.255	125.544
<b>Coefficiente <math>\chi^2</math></b>	$2,70 \cdot 10^6$	$1,22 \cdot 10^{14}$
<b>Suma de los errores absolutos</b>	101.631,8	103.520,6
<b>Coefficiente <math>R^2</math></b>	0,99651	0,99836
<b>MSSIM</b>	0,980	0,981

**Tabla 3-22. Mejora Ajuste Chicago Sketch. Distorsión matriz ajustada.**

En la Tabla 3-22 se detallan diferentes indicadores de distancia entre la matriz de partida y las matrices ajustadas con las estrategias descritas. Para los tres primeros parámetros el valor nulo describe dos matrices completamente semejantes. Los parámetros  $R^2$  y MSSIM dan valores comprendidos entre 0 y 1, siendo 1 el valor que indica que las matrices de partida y ajustada son la misma matriz.

No se ha incluido en la tabla la media de los errores al cuadrado ni la desviación típica de los errores ya que tienen la desventaja de ponderar fuertemente los valores outliers (valores cuantitativamente distantes del resto de datos). En los procesos de ajuste no se aplica un incremento homogéneo a todas las casillas de la matriz de viajes sino que cada casilla modificará su valor según su influencia sobre el flujo de los arcos aforados. Es por esto que en este tipo de problemas generalmente existen casillas ajustadas con valores cuantitativamente superior al resto.

Puede apreciarse que tanto la entropía relativa como el coeficiente  $\chi^2$  son muy sensibles a las variaciones respecto de la matriz de partida. El coeficiente  $\chi^2$  aumenta exponencialmente. Estos valores indican una menor distorsión de la matriz de viajes cuando se tienen en cuenta únicamente los pares O-D con más de 1 viaje.

La suma de los errores absolutos, el coeficiente de correlación  $R^2$  y el índice MSSIM no aportan información concluyente sobre la distorsión de las matrices. En el caso del coeficiente  $R^2$  es

conocido que para matrices compuestas por elementos con elevados números de viajes, elementos con pequeñas cantidades de viajes y muy pocos elementos con viajes cercanos al valor medio, el valor de  $R^2$  se mantiene muy cercano a 1 incluso para distorsiones superiores al 150% (Smith y Hutchinson, 1981). El parámetro MSSIM, como se describe en el apéndice I, no es un buen indicador de distancia entre las matrices O-D inicial y ajustada.

Como conclusión conviene comentar que se ha propuesto una estrategia de resolución que mejora los tiempos de ejecución, alcanza valores de la función objetivo inferiores e imprime menor distorsión a la matriz de demanda. Posiblemente la selección de los pares O-D que van a ser ajustados deba ser más exhaustiva pero se ha demostrado que se mejoran los resultados al incluir en el ajuste los pares O-D con mayor número de viajes.



# CAPÍTULO 4

---

## CONCLUSIONES Y LÍNEAS FUTURAS

### 4.1 CONCLUSIONES

#### 4.1.1 Estimación de matrices origen – destino

En esta tesis se aborda el problema de la estimación de matrices de viajes en redes de tráfico reales, congestionadas y de grandes dimensiones. Éste es un tema de manifiesta importancia en la planificación del transporte dada la relación existente entre la precisión de las matrices O-D y la calidad de los resultados del proyecto.

El objetivo principal ha sido el de encontrar una metodología de ajuste de matrices que sea capaz de afrontar proyectos de grandes redes de tráfico congestionadas, incorporando los últimos modelos de asignación de tráfico de alta convergencia. La información de partida es la matriz de viajes del proyecto desactualizada y datos de aforos de un subconjunto de arcos de la red.

Una vez analizado el estado del arte de la estimación de matrices O-D se ha optado por desarrollar un método de programación binivel. Esta formulación ha sido ampliamente estudiada en diferentes campos de investigación, incluida la planificación del transporte. La mejora que presenta este trabajo se centra en el diseño de una nueva metodología para el cálculo de la dirección de descenso, además del empleo de nuevos algoritmos de asignación basados en caminos, orígenes o *bushes*. Mostradas las carencias de los Jacobianos implementados en los anteriores estudios de ajuste de matrices se plantea el cálculo de un nuevo Jacobiano dando incrementos infinitésimos para cada par O-D, por lo que la técnica puede englobarse en la metodología numérica de Diferencias Finitas.

El algoritmo de ajuste presentado resuelve iterativamente un problema de asignación de tráfico, cuya solución permite generar la dirección de descenso propuesta, y un problema de estimación, en el que se actualiza la matriz de viajes a través de la dirección de descenso.

Se han realizado pruebas numéricas donde se pone en evidencia las mejoras que ofrece el algoritmo:

- Se han abordado desde redes de pequeña escala hasta redes de grandes dimensiones como es la red de la comunidad de Madrid.
- Se ha comprobado que el cálculo del Jacobiano propuesto genera direcciones de búsqueda más eficientes que las propuestas en el método del gradiente de Spiess (1990) y de Yang et al. (1992). Esta eficiencia se hace patente en que se alcanzan matrices menos distorsionadas y en un menor número de iteraciones.
- Esta nueva formulación tiene en cuenta los efectos de la saturación de vehículos. El Jacobiano considera las influencias de todos los pares O-D, incluso los pares O-D no interceptados por ningún aforo y los que inicialmente no presentan viajes.
- Se ha reducido el tiempo de ejecución gracias a la inclusión de nuevos problemas de asignación de tráfico. A su vez, mediante el empleo de estos algoritmos de alta convergencia se ha mejorado la estabilidad, repetitividad y consistencia de las soluciones.

La metodología propuesta utiliza la misma información que los demás métodos formulados en la bibliografía, la matriz de partida y volúmenes de aforo en un conjunto de arcos. La principal diferencia se encuentra en que, al considerar la saturación de la red a través de las diferencias finitas, se está rescatando una mayor información de la matriz de partida.

Finalmente cabe destacar la utilidad práctica de la herramienta desarrollada, los resultados presentados ponen de manifiesto la amplia variedad de casos que puede tratarse con la formulación propuesta y la flexibilidad de la implementación desarrollada para adaptarse a esta casuística.

Esta formulación generalizada presenta además el atractivo, tal y como se ha mostrado en el CAPÍTULO 3, de poder ser implementada en el entorno de desarrollo integrado de Matlab con su propio lenguaje de programación. Desde Matlab existe la posibilidad de convertir su propio código de programación en cualquier código de programación como puede ser C, C++, C#, Java,... Luego en esta tesis se presenta una herramienta que puede ser empleada por todos aquellos profesionales que hayan de enfrentarse a este problema clásico de la Ingeniería del Transporte.

En líneas generales este trabajo aporta una formulación competitiva para el problema de estimación de matrices. En contraposición, durante el desarrollo de la tesis se han descrito

ciertos inconvenientes asociados al problema de ajuste, o a los proyectos de planificación del transporte, que no han sido solventados.

El principal problema que presentan las redes de transporte es la heterogeneidad. Esto lleva a los investigadores del transporte a realizar una gran cantidad de divisiones del sistema, lo que introduce en los modelos un gran número de variables a ser estimadas. El coste de la adquisición de datos asociados a las redes de transporte es excesivo si se quiere explicar con fiabilidad todas las variables consideradas en los modelos. En la práctica, la información recogida es escasa por lo que los problemas de planificación se caracterizan por ser muy indeterminados. La formulación propuesta no evita la indeterminación ya que es insalvable, aun así genera direcciones de búsqueda eficientes con el fin de imprimir pequeños desplazamientos de la matriz de viajes inicial. En este estudio no se ha analizado en detalle la disminución de la indeterminación y la reducción de la distorsión de las matrices mediante restricciones por lo que se proponen como líneas futuras de investigación a desarrollar.

#### **4.1.2 Problema de asignación de tráfico**

En este trabajo se ha puesto especial énfasis en el análisis e incorporación del problema de asignación en el ajuste de matrices de viaje. Se ha comprobado que supone un porcentaje muy alto del tiempo de ejecución en el proceso de ajuste y además aporta variables imprescindibles para el nivel superior de la formulación binivel.

Los algoritmos han sido implementados en código Matlab y, aunque inicialmente la generación de un código que pueda competir con los programas comerciales no es un hito en esta tesis, se ha alcanzado un nivel de eficiencia alto. Se han llevado a cabo diferentes estudios de eficiencia con el fin de mejorar los tiempos del proceso completo de ajuste. En el CAPÍTULO 2 se comparan los resultados con el software comercial TransCAD 5.0.

La idea inicial de la tesis es la de diseñar un algoritmo de asignación que aporte la mayor cantidad de información para el proceso de ajuste. Para este fin se desarrolla en la sección 2.7 un algoritmo de asignación de tráfico basado en caminos. Se supuso previamente esta tipología de algoritmos de asignación como la más adecuada ya que, al conocer el conjunto completo de caminos activos y sus variables, se tiene información completa de la distribución de vehículos sobre la red de transporte.

El algoritmo propuesto parte de la solución empleada por Florian et al. (2009) en la que se mejora el cálculo del traspaso de flujo entre los caminos de un mismo par O-D para equilibrar sus costes. Este algoritmo presenta buenos resultados para redes de tamaño bajo-medio. En el

estudio de redes de grandes dimensiones los tiempos de ejecución del problema de asignación y de ajuste se disparan. Es por esto que se ha investigado otras tipologías de algoritmos de asignación basadas en orígenes y *bushes* acíclicos.

Al explotar las posibilidades que ofrecen los nuevos algoritmos de asignación se han alcanzado mejoras importantes. La principal mejora que ha permitido que el ajuste de matrices pueda aplicarse a redes de grandes dimensiones ha sido desarrollada siguiendo la idea recogida en Bar-Gera (2010) donde se almacenan las parejas de segmentos alternativos PAS. Gracias a esto, cuando se da un incremento infinitésimo a cada par O-D, el problema se reduce a resolver un sistema de ecuaciones formado por un pequeño conjunto de PAS. Además, otra característica a destacar es que la matriz de coeficientes en el sistema de ecuaciones permanece constante para todos los pares O-D, la única diferencia se encuentra en el vector de términos independientes.

## 4.2 LÍNEAS FUTURAS

Durante el desarrollo de esta tesis ha surgido la necesidad de realizar diferentes investigaciones que tienen relación con la estimación de matrices de viajes. Éstas pueden ser desarrolladas en paralelo con el tema tratado y amplían el conocimiento del mismo. A continuación se proponen para un estudio posterior:

- *Incorporación de información medida desde otras fuentes de datos.*

Con la tecnología actual existe la posibilidad de realizar otros tipos de mediciones adicionales a las consideradas en este trabajo. Estas mediciones enriquecerán y darán más fiabilidad a la metodología presentada.

- *Acotamiento del desplazamiento de la matriz de viajes ajustada.*

El método del gradiente implementado no considera ningún tipo de restricción que minimice la distorsión de la matriz resultado. Este estudio es importante para los proyectos en los que la matriz de partida se considera que tiene una alta fiabilidad.

- *Localización de puntos de aforo.*

Para cada subconjunto de aforos considerado, los algoritmos de ajuste siguen una dirección de búsqueda de la solución diferente. Ciertos aforos colocados en vías principales, usadas por muchos pares O-D, contienen información de flujo muy

agregada y poco útil para el problema. La selección de los puntos de aforo puede mejorar la solución.

- *Reducción de la indeterminación del problema.*

En ciertos proyectos la información disponible es escasa. El nivel de detalle esperado en la matriz de viajes debe ir acorde con la información disponible. Se podría llegar a un acuerdo entre divisiones de la red e información de puntos de aforo disponible.

- *Estudio de la modelización de la red (tiempos de viajes, densidad...).*

En el estado de equilibrio de usuario, el reparto de los viajes de un par O-D en unas u otras proporciones depende en gran medida de los tiempos de viaje de los caminos que lo conectan. Modificando las características del modelo de red es posible obtener diferentes distribuciones de flujo en arcos. Por ello es importante que la caracterización sea fiable.

- *Estudio de los segmentos PAS empleados para el cálculo del Jacobiano.*

Estos segmentos forman el sistema de ecuaciones a resolver en el cálculo del Jacobiano. Se ha comprobado que existe la posibilidad de disminuir el tamaño de dicho conjunto de segmentos PAS y así reducir el tiempo de resolución del sistema de ecuaciones.

- *Implementación del algoritmo con otros lenguajes de programación.*

Los tiempos de computación pueden ser mejorados al racionalizar/optimizar la programación en determinados lenguajes de programación.

- *Estudio de programación en paralelo.*

El problema del ajuste tiene acciones que pueden ser paralelizadas. Mediante la programación en paralelo se ha conseguido reducir los tiempos de cálculo. El análisis llevado a cabo no ha sido exhaustivo ni muy sofisticado y con seguridad la eficiencia del algoritmo puede verse mejorada a través de un uso óptimo de las tecnologías de computación en paralelo.

- *Estudio de mejoras del algoritmo en cuanto a estructuras de datos, método para la actualización del bush, método para el equilibrado del bush, etcétera.*

Durante la ejecución del problema de asignación de tráfico los algoritmos recorren constantemente el grafo hasta alcanzar un patrón de flujos en equilibrio de usuario por sus arcos. Por ello, una mejora potencial de la eficiencia de los algoritmos implementados puede obtenerse en base a un buen diseño de la estructura de datos que modela la red de transportes. Con ello será posible implementar algoritmos más eficientes en los módulos de actualización y equilibrado de los *bushes*.

- *Estudio de modelos con múltiples clases de usuarios con diferente opción de viajar.*

Este trabajo se centra en la movilidad del usuario que emplea únicamente el vehículo privado. Aunque representa un porcentaje elevado del total de desplazamientos, es interesante incluir otros tipos de usuarios con diferentes opciones de viajar.

- *Estudio de la eficiencia del algoritmo de asignación de tráfico.*

El problema de asignación constituye un gran porcentaje del tiempo de ejecución del proceso de ajuste. Por ello es importante mejorar la eficiencia de dichos algoritmos. Un elemento a optimizar puede ser la opción Warm Start. Ésta puede partir de situaciones cercanas al equilibrio reduciendo los tiempos de inicialización del algoritmo.

# BIBLIOGRAFÍA

---

Abrahamsson, T. (1998). "Estimation of origin–destination matrices using traffic counts – a literature survey". Technical Report, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria

Ahuja, R., T. Magnanti y J. Orlin (1993). "Network flows". Prentice Hall, Englewood, NJ.

Bard, J.F. (1988). "Convex two-level optimisation". *Mathematical Programming* 40, 15-27

Bar-Gera, H. (2002). "Origin-based algorithm for the traffic assignment problem". *Transportation Science* 36, 398-417

Bar-Gera, H. (2010). "Traffic assignment by paired alternative segments". *Transportation Research* 44B, 1022-1046

Beckman, M., C. McGuire y C. Winsten (1956). "Studies in the Economics of Transportation". Yale University Press, New Have

Bell, M.G.H. (1983). "The estimation of an origin-destination matrix from traffic counts". *Transportation Science* 17, 198-217

Bell, M.G.H. (1984). "Log-linear models for the estimation of origin-destination matrices from traffic counts: An approximation". *Proceedings of the Ninth International Symposium on Transportation and Traffic Theory, The Netherlands, Delft*, 451-470

Bell, M.G.H. (1991). "The estimation of origin-destination matrices by constrained generalized least squares". *Transportation Research* 25B, 13-22

Bell, M.G.H., y Y. Iida (1997). "Transportation network analysis". Chichester: John Wiley & Sons

Ben-Akiva, M., y S. Lerman (1985). "Mathematical Programs with Equilibrium Constraints". *Transportation Study*. MIT Press

Bernstein, D. (1990). "Programmable network equilibria". Ph.D. dissertation University of Pennsylvania

- Bertsekas, D.P. (1976). "Dynamic Programming and Stochastic Control". Academic Press, NY
- Bertsekas, D.P., y R. Gallager (1992). "Data Networks (second edition)". Prentice-Hall, Englewood Cliffs, New Jersey
- Bierlaire, M. y P.L. Toint (1995). "Meuse: An origin–destination matrix estimator that exploits structure". *Transportation Research* 29B, 47–60
- Boyce, D., D. Ralevic-Dekic y H. Bar-Gera (2004). "Convergence of Traffic Assignments: How much is enough?". *Journal of Transportation Engineering*, ASCE
- Braess, D. (1968). "Über ein paradoxon der verkehrsplanung". *Unternehmensforschung* 12, 258-268
- Brenninger-Göthe, M., K.O. Jornsten y J.T. Lundgren (1989). "Estimation of origin/destination matrices from traffic counts using multiobjective programming formulation". *Transportation Research* 23B, 257-269
- Caliper Corporation (2010). "What TransCAD users should know about new static traffic assignment methods". Caliper corporation Communication to Users, [http://www.caliper.com/Press/What\\_TransCAD\\_Users\\_Should\\_Know\\_about\\_Traffic\\_Assignment.pdf](http://www.caliper.com/Press/What_TransCAD_Users_Should_Know_about_Traffic_Assignment.pdf)
- Cascetta, E. (1984). "Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator". *Transportation Research* 18B, 289-299
- Cascetta, E., y S. Nguyen (1988). "A unified framework for estimating or updating origin/destination matrices from traffic counts". *Transportation Research* 22B, 437-455
- Cascetta, E., y M. Postorino (2001). "Fixed point approaches to the estimation of O/D matrices using traffic counts on congested networks". *Transportation Science* 35, 134–147
- Cascetta, E. (2009). "Transportation System Analysis: models and applications. 2nd edition". Springer
- Chen, A., D-H. Lee y R. Jayakrishnan (2002). "Computational study of state-of-the-art path-based traffic assignment algorithms". *Mathematics and Computers in Simulation* 59, 509-518

- Chen, Y. (1994). "Bilevel programming problems: analysis, algorithms and applications". Ph.D. Publication 984, Centre de Recherche sur les Transports (CRT), Université de Montréal, Montréal, Canada
- Codina, E., y J. Barceló (2004). "Adjustment of O–D trip matrices from observed volumes: an algorithmic approach based on conjugate directions". *European Journal of Operational Research* 155, 535–557
- Dafermos, S.C. (1980). "The traffic equilibrium and variational inequalities". *Transportation Science* 14, 42–54
- Denault, L. (1994). "Étude de deux méthodes d'ajustement de matrices origine-destination à partir des flots des véhicules observés (in French)". Report CRT-991, Mémoire D'étudiant, Centre de recherche sur les transports (CRT), Université de Montréal, Montréal, Canada
- Dial, R. (1996). "Bicriterion Traffic Assignment: Basic theory and elementary algorithms". *Transportation Science* 30, 93-111
- Dial, R. (2006). "A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration". *Transportation Research* 40B, 917-936
- Doblas, J., y F.G. Benítez (2005). "An approach to estimating and updating origin–destination matrices based upon traffic counts preserving the prior structure of a survey matrix". *Transportation Research* 39B, 565–591
- Drissi-Kaitouni, O., y J. Lundgren (1992). "Bilevel origin–destination matrix estimation using a descent approach". Technical Report lith-matr92-49, Department of Mathematics, Institute of Technology, Linköping, Sweden
- Djukic, T., S.P. Hoogendoorn y J.W.C. Van Lint (2013). "Reliability assessment of dynamic OD estimation methods based on structural similarity index". 92<sup>nd</sup> meeting of the Transportation Research Board, Washington D.C.
- Erlander, S. y N.F. Stewart (1990). "The Gravity Model in Transportation Analysis. Theory and Extensions". *Topics in Transportation*. VSP, Utrecht, The Netherlands
- Fisk, C.S. y S. Nguyen (1982). "Solutions algorithms for the network equilibrium models with asymmetric user costs". *Transportation Science* 16, 361-381

- Fisk, C.S., y D.E. Boyce (1983). "A note on trip matrix estimation from link traffic count data". *Transportation Research* 17B, 245-250
- Fisk, C.S. (1988). "On combining maximum entropy trip matrix estimation with user optimal assignment". *Transportation Research* 22B, 66-79
- Fisk, C.S. (1989). "Trip matrix estimation from link traffic counts: The congested network case". *Transportation Research* 23B, 331-356
- Florian, M. y H. Spiess (1982). "The convergence of diagonalisation algorithms for asymmetric network equilibrium problems". *Transportation Research* 16B, 477-483
- Florian, M., J. Guélat y H. Spiess (1987). "An efficient implementation of the "PARTAN" variant of the linear approximation method for the network equilibrium problem". *Networks* 17, 319-339
- Florian, M., y Y. Chen (1993). "A coordinate descent method for the bilevel O-D matrix adjustment problem". Publication 750, Centre de Recherche sur les Transports (CRT), Université de Montréal, Montréal, Canada
- Florian, M., y D.W. Hearn (1995). "Network equilibrium models and algorithms". M. O. Ball et al., Eds., *Handbooks in Operations Research and Management Science*, Vol. 8, chapter 6, pages 485-550. Elsevier, New York
- Florian, M., I. Constantin y D. Florian (2009). "A New Look at Projected Gradient Method for Equilibrium Assignment". *Transportation Research Record* 2090, 10-16
- Frank, M. y Wolfe, P. (1956). "An algorithm for quadratic programming". *Naval Research Logistics Quarterly* 3 (1-2): 95-110
- Friesz, T.L., R.L. Tobin, H.J. Cho y N.J. Metha. (1990). "Sensitivity Analysis Based Heuristic Algorithms for Mathematical Programs with Variational Inequality Constraints". *Mathematical Programming* 48, 265-28
- García, R. (2001). "Metodología para el diseño de redes de transporte y para la elaboración de algoritmos en programación matemática convexa diferenciable". Tesis doctoral en matemáticas, Universidad Politécnica de Madrid, Madrid, España
- Gentile, G. (2012). "Linear user cost equilibrium: A bush-based algorithm for traffic assignment". *Transportmetrica*. DOI: 10.1080/18128602.2012.691911

- Hazelton, M.L. (2001). "Inference for origin-destination matrices: estimation, reconstruction and prediction". *Transportation Research* 35B, 667-676
- Hearn, D.W., S. Lawphongpanich y J.A. Ventura (1985). "Finiteness in restricted simplicial decomposition". *Operations Research Letters* 4, 125-130
- Högberg, P. (1976). "Estimation of parameters in models for traffic prediction: a non-linear regression approach". *Transportation Research* 10, 263-265
- Hohenbalken, B.v. (1977). "Simplicial decomposition in nonlinear programming algorithms". *Mathematical Programming* 13, 49-68
- Holloway, C.A. (1974). "An extension of the Frank and Wolfe method of feasible directions". *Mathematical Programming* 6, 14-27
- Holm, J., T. Jensen, S. Nielsen, A. Christensen, B. Johnsen y G. Ronby (1976). "Calibrating traffic models on traffic census results only". *Traffic Engineering and Control* 17, 137-140
- Janson, B., y C. Zozaya-Gorostiza (1987). "The Problem of Cyclic Flows in Traffic Assignment". *Transportation Research* 21B, 299-310.
- Jayakrishnan, R., W.K. Tsai, J. Prashker y S. Rajadhyaksha (1994). "A faster path-based algorithm for traffic assignment". *Transportation Research Record* 1443, 75-83
- Knight, F.H. (1924). "Some Fallacies in the Interpretation of Social Cost". *Quarterly Journal of Economics* 38, 582-606
- Kullback, S. (1959). "Information theory and statistics". Wiley, New York.
- Larsson, T., y M. Patriksson (1992). "Simplicial decomposition with disaggregated representation for the traffic assignment problem". *Transportation Science* 26, 4-17
- Larsson, T., M. Patriksson y C. Rydergren (1997). "Applications of simplicial decomposition with nonlinear column generation to nonlinear network flows". In P. M. Pardalos, W. W. Hager, and D. W. Hearn (Eds.), *Network Optimization*, Number 450 in *Lecture Notes in Economics and Mathematical Systems*, 346-373. Berlin: Springer-Verlag
- LeBlanc, L., E.K. Morlok y W.P. Pierskalla (1975). "An efficient approach to solving the road network equilibrium traffic assignment problem". *Transportation Research* 9B, 309-318

LeBlanc, L., y K. Farhangian (1982). "Selection of a trip table which reproduces observed link flows". *Transportation Research* 22B, 83-88

Leurent, F. (1996). "The theory and practice of a dual criteria assignment model with a continuously distributed value-of-time". J.B. Lesort (ed), *Transportation and Traffic Theory*, 455-477, Elsevier Science, Ltd. Oxford.

Li, B.B. (2005). "Bayesian inference for origin-destination matrices of transport networks using the EM algorithm". *Technometrics*, 47:4:499-508

Lo, H.P., y C.P. Chan (2003). "Simultaneous estimation of an origin-destination matrix and link choice proportions using traffic counts". *Transportation Research* 37A, 771-788

Low, D.E. (1972). "A new approach to transportation system modelling". *Traffic Quarterly* 26, 391-404

Lundgren, J.T., and A. Peterson (2008). "A heuristic for the bilevel origin-destination matrix estimation problem". *Transportation Research* 42B, 339-354

Maher, M.J. (1983). "Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach". *Transportation Research* 20B, 435-447

Maher, M.J., y X. Zhang (1999). "Algorithms for the solution of the congested trip matrix estimation problem". *Proceedings of the 14th International Symposium on Transportation and Traffic Theory*, The Netherlands, Amsterdam, 445-469

Maher, M.J., X. Zhang y D. van Vliet (2001). "A bi-level programming approach for trip matrix estimation and traffic control problems with stochastic user equilibrium link flows". *Transportation Research* 35B, 23-40

Mitradjieva, M. y P. O. Lindberg (2012). "The stiff is moving-Conjugate direction Frank-Wolfe methods with applications to traffic assignment". *Transportation Science*, Articles in advance, 1-14.

Nie Y., J. Xie y Y. Xiaoguang (2012). "Does LUCE outperform OBA? A comparison study of two bush-based algorithms for the traffic assignment problem". 92<sup>nd</sup> meeting of the Transportation Research Board, Washington D.C.

Nguyen, S. (1977). "Estimating an O-D matrix from network data: A network equilibrium approach". Publication 87, Centre de recherche sur les transports (CRT), Université de Montréal, Montréal, Canada

Nguyen, S. y C. Dupuis (1981). "An Efficient Method for Computing Traffic Equilibria in Networks with Asymmetric Transportation Cost". Centre de recherche sur les transports (CRT), Université de Montréal, Montréal, Canada

Ortúzar, J. de D., y L.G. Willumsen (2002). "Modelling Transport (third edition)". Chichester (UK): John Wiley & Sons

Patriksson, M. (1994). "Traffic Assignment Problem. Models and Methods". VSP International Science, Utrecht, The Netherlands

Patriksson, M. (2004). "Sensitivity analysis of traffic equilibria". *Transportation Science* 38, 258–281

Rabinowicz, A. (2008). "Experimentos con métodos de asignación alternativos". VIII Congreso de Ingeniería de Transporte (CIT). A Coruña, España.

Robillard, P. (1975). "Estimating the O–D matrix from observed link volumes". *Transportation Research* 9, 123–128

Rose, G., M. Daskin y F. Koppelman (1988). "An examination of convergence error in equilibrium traffic assignment models". *Transportation Research* 22B, 261-274

Rosen, J.B. (1960). "The gradient projection method for nonlinear programming (Part 1: Linear constraints)". *SIAM J. Appl. Math.* 8, 181-217

Sánchez-Cambronero, S. (2008). "Traffic Prediction Models using Bayesian Networks and other tools". Ph.D. in civil engineering, University of Castilla La Mancha, Ciudad Real, Spain

Sheffi, Y. (1985). "Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods". Prentice-Hall, Englewood Cliffs, New Jersey

Sherali, H.D., R. Sivanandan y A.G. Hobeika (1994). "A linear programming approach for synthesizing origin-destination trip tables from link traffic volumes". *Transportation Research* 28B, 213-233

- Sherali, H.D., A. Narayanan y R. Sivanandan (2003). "Estimation of origin–destination trip-tables based on a partial set of traffic link volumes". *Transportation Research* 37B, 815–836
- Slavin, H., J. Brandon y A. Rabinowicz (2006). "An Empirical Comparison of Alternative User Equilibrium Traffic Assignment Methods". *Proceedings of the European Transport Conference 2006, Strasbourg, France*
- Smith, D.P. y B.G. Hutchinson (1981). "Goodness of fit statistics for trip distribution models". *Transportation Research* 15A, 295-303
- Smith, M.J. (1979). "The existence, uniqueness and stability of traffic equilibria". *Transportation Research* 13B, 295–304
- Smith, M.J. (1983). "An algorithm for solving asymmetric equilibrium problems with a continuous cost-flow function". *Transportation Research* 17B, 365–372
- Spiess, H. (1987). "A maximum-likelihood model for estimating origin-destination matrices". *Transportation Research* 21B, 395-412
- Spiess, H. (1990). "A descent based approach for the OD matrix adjustment problem". *Publication 693, Centre de recherche sur les transports (CRT), Université de Montréal, Montréal, Canada*
- Tamin, O.Z., y L.G. Willumsen (1989). "Transport demand model estimation from traffic counts". *Transportation* 16, 3-26
- Tobin, R.L., y T.L. Friesz (1988). "Sensitivity analysis for equilibrium network flows". *Transportation Science* 22, 242–250
- Van Zuylen, H., y L.G. Willumsen (1980). "The most likely trip matrix estimated from traffic counts". *Transportation Research* 14B, 281-293
- Verastegui, D. (2006). "Estimación de matrices origen - destino y calibración de parámetros en el problema de asignación de tráfico en redes congestionadas". *Tesis doctoral en ingeniería civil, Universidad de Castilla La Mancha, Ciudad Real, España*
- Wardrop, J. (1952). "Some theoretical aspects of road traffic research". *Proceedings of the Institute of Civil Engineers, Part 2, 325–378*

- Wills, M.J. (1986). "A flexible gravity-opportunities model for trip distribution". *Transportation Research* 20B, 89-111
- Willumsen, L.G. (1984). "Estimating time-dependent trip matrices from traffic counts". *Proceedings of Ninth international symposium on transportation and traffic theory*, The Netherlands, Utrecht, 397-411
- Willumsen, L.G. (1991). "Origin-destination matrix: static estimation". En M. Papageorgiou (ed.), *Concise Encyclopedia of Traffic & Transportation Systems*. Pergamon Press, Oxford.
- Yang, H., T. Sasaki, Y. Iida y Y. Asakura (1992). "Estimation of origin-destination matrices from link traffic counts on congested networks". *Transportation Research* 26B, 417-434
- Yang, H., Y. Iida y T. Sasaki (1994). "The equilibrium-based Origin-Destination matrix estimation problem". *Transportation Research* 28B, 23-33
- Yang, H., S. Yagar, Y. Iida y Y. Asakura (1994). "An algorithm for the inflow control problem on urban freeway networks with user-optimal flows". *Transportation Research* 28B, 123-139
- Yang, H., y S. Yagar (1995). "Traffic assignment and signal control in saturated road networks". *Transportation Research* 29A, 125-139
- Yang, H. (1995). "Heuristic algorithms for the bi-level origin-destination matrix estimation problem". *Transportation Research* 29B, 1-12
- Yang, H., Q. Meng y M.G.H. Bell (2001). "Simultaneous estimation of the origin-destination matrices and travel-cost coefficient for congested networks in a stochastic user equilibrium". *Transportation Science* 35, 107-123
- Zhou, Z. y M. Martimo (2010). "Computational study of alternative methods for static traffic equilibrium assignment". *Proc. World Conf. Transport Res. Soc. (WCTRS)*, Lisboa, Portugal



## **APÉNDICE I**

# **MEDIDAS DE DISTANCIA ENTRE MATRICES**

Existe una bibliografía muy extensa referida a los procesos de actualización de matrices O-D empleando información de volúmenes de aforo. Cada investigador aplica diferentes metodologías de ajuste alcanzando distintas matrices actualizadas. Si los volúmenes aforados son consistentes, es muy factible alcanzar un buen ajuste entre los volúmenes modelados y aforados. Debido a la indeterminación que presenta el problema, el número de matrices que ajustan los aforos es casi infinito. Por lo que cualquier procedimiento puede alcanzar altas convergencias de la función objetivo con diferentes matrices de demanda.

En esta tesis tiene un gran peso la información que introduce la matriz de partida, por lo que la matriz ajustada no debe distanciarse de la de partida. A continuación se estudia una serie de herramientas que computan las diferencias entre dos conjuntos de datos y pueden ser útiles para analizar distancias entre matrices.

### Coefficientes de correlación.

- *Coefficiente de correlación de Pearson.*

$$r_{TT^*} = \frac{\sum Z_T \cdot Z_{T^*}}{n^2} = \frac{\sum T \cdot T^* - \overline{T \cdot T^*}}{S_T \cdot S_{T^*}} = \frac{n^2 \sum_{ij} T_{ij} \cdot T_{ij}^* - \sum_{ij} T_{ij} \sum_{ij} T_{ij}^*}{\sqrt{n^2 \sum_{ij} T_{ij}^2 - \left(\sum_{ij} T_{ij}\right)^2} \sqrt{n^2 \sum_{ij} T_{ij}^{*2} - \left(\sum_{ij} T_{ij}^*\right)^2}} \quad (I.1)$$

Este índice mide el grado de correlación entre dos variables aleatorias cuantitativas relacionadas linealmente. Como se comprueba en Smith y Hutchinson (1981) este parámetro es muy poco sensible cuando se emplea como medida de distancia entre dos matrices de viajes. Existe un tipo de matrices en el que los valores de correlación se mantienen muy cercanos a 1 incluso con grandes distorsiones que se presenta cuando la matriz se puede dividir en dos grandes grupos, un grupo de casillas con bajo número de viajes y otro de casillas con alto número de viajes.

- *Expresión empleada en el test  $\chi^2$  de Pearson.*

$$\chi^2 = \sum_{ij} \frac{(T_{ij} - T_{ij}^*)^2}{T_{ij}^*} \quad (I.2)$$

Esta expresión es muy sensible para el cálculo de distancias. El principal inconveniente es el denominador  $T_{ij}^*$ . No pueden incluirse valores nulos. A su vez, este parámetro da mucho peso a las casillas que inicialmente tenían algún valor y durante el ajuste se ha reducido hasta valores cercanos a cero.

### Basados en la teoría de errores.

Suponiendo una matriz como teórica y la otra como observada, pueden considerarse las discrepancias entre casillas como errores de medida. Así es posible emplear los siguientes índices:

- *Error cuadrático medio.*

$$MSE = \frac{\sum_{ij} (T_{ij} - T_{ij}^*)^2}{n^2} \quad (I.3)$$

- *Raíz cuadrada del error cuadrático medio / Desviación estándar del error.*

$$RMSE = \sqrt{\frac{\sum_{ij} (T_{ij} - T_{ij}^*)^2}{n^2}} \quad (I.4)$$

- *Raíz cuadrada del error cuadrático medio normalizada.*

$$RMSE\_Norm = \frac{\sqrt{\frac{\sum_{ij} (T_{ij} - T_{ij}^*)^2}{n^2}}}{\frac{\sum_{ij} T_{ij}^*}{n^2}} \quad (I.5)$$

- *Suma de los errores absolutos*

$$ABSERR = \sum_{ij} |T_{ij} - T_{ij}^*| \quad (I.6)$$

- *Error absoluto medio.*

$$MABSERR = \frac{\sum_{ij} |T_{ij} - T_{ij}^*|}{n^2} \quad (I.7)$$

- *Error absoluto medio normalizado.*

$$MASBERR\_Norm = \frac{\sum_{ij} |T_{ij} - T_{ij}^*|}{\frac{\sum_{ij} T_{ij}^*}{n^2}} \quad (I.8)$$

Los índices MSE, RMSE y RMSE\_Norm presentan un comportamiento equivalente. Dan un valor proporcional a la desviación estándar de las discrepancias. En la teoría de errores es importante minimizar la varianza o desviación del error. En el problema de ajuste de matrices

esta minimización supondría la homogeneización de las discrepancias. El objetivo de un ajuste es actualizar las casillas de la matriz de viajes para ajustar el flujo en los aforos y una forma de minimizar estos parámetros es evitando grandes variaciones sobre ciertas casillas y aplicando pequeñas variaciones sobre una gran número de casillas. Éste no es un buen comportamiento para los métodos de ajuste, luego estos parámetros no son válidos para medir distancias entre matrices ajustadas.

Los parámetros ABSERR, MABSERR y MABSERR\_Norm computan la suma en valor absoluto de los desplazamientos de todas las casillas. Los valores MABSERR y MABSERR\_Norm están normalizados y pueden ser útiles para comparar distorsiones de matrices de diferentes redes de transporte. Para una misma red, ABSERR es muy útil ya que su valor puede compararse con la suma de las discrepancias de los arcos aforados:

$$\hat{d} = \sum_a (f_a - v_a) \quad a \in \text{Aforos} \quad (I.9)$$

donde  $f_a$  son los volúmenes medidos en arcos y  $v_a$  los volúmenes modelados.

Al partir de una matriz de viajes que no representa los flujos medidos en los puntos de aforo será necesario aplicar un desplazamiento sobre sus casillas para ajustar dichos aforos. El número de viajes que es necesario desplazar las matrices (ABSERR) es similar a la suma de las discrepancias iniciales de los arcos aforados ( $\hat{d}$ ). Si cada viaje aportado a la matriz se traduce en la disminución de las discrepancias en arcos, ABSERR puede ser inferior a  $\hat{d}$ .

Aunque no sea conocido el valor de  $\hat{d}$ , para dos matrices solución que han alcanzado igual nivel de función objetivo, es decir han ajustado los aforos en un mismo grado, ABSERR, MABSERR y MABSERR\_Norm pueden dar una idea de la distorsión aplicada a cada matriz. Mientras menor sea el valor de estos parámetros, menor es la distorsión aplicada a la matriz.

#### **Basados en la teoría de la información.**

- *Basada en la divergencia de Kullback-Leibler.*

$$I = T_{ij} \cdot \left| \ln \frac{T_{ij}}{T_{ij}^*} \right| \quad (I.10)$$

La divergencia de Kullback-Leibler (también llamada ganancia de la información o entropía relativa) es una medida de la similitud o diferencia entre dos funciones de

distribución. Ésta se formula empleando las probabilidades de las medidas. La expresión (I.10) es una modificación empleada por Smith y Hutchinson (1981). Tanto la divergencia de Kullback-Leibler como la expresión (I.10) son medidas no simétricas, por tanto el resultado depende del orden de las variables.

El término de primer orden del desarrollo en serie de Fourier de la expresión (I.10) coincide con la expresión (I.2) por lo que presenta inconvenientes semejantes. En este caso aplica grandes pesos, aunque menores que en la expresión (I.2), a las variables que se hacen casi nulas durante el ajuste. Este problema se suaviza aún más en los siguientes índices.

- *Máxima log-verosimilitud suponiendo que la matriz se distribuye bajo un modelo multinomial.*

$$\Delta L = \sum_{ij} T_{ij} \left[ \ln \frac{T_{ij}}{\sum_{ij} T_{ij}} - \ln \frac{T_{ij}^*}{\sum_{ij} T_{ij}^*} \right] \quad (I.11)$$

- *Estadístico psi (Kullback, 1959).*

$$\Psi = \sum_{ij} \left( T_{ij} \cdot \ln \frac{T_{ij}}{S_{ij}} + T_{ij}^* \cdot \ln \frac{T_{ij}^*}{S_{ij}} \right) \quad (I.12)$$

$$\text{donde } S_{ij} = \frac{T_{ij} + T_{ij}^*}{2}.$$

### **Similitud estructural.**

- *Índice de similitud estructural.*

El uso común de este parámetro es la medición de la similitud entre dos imágenes, está basado en la degradación estructural de la información de una imagen comparada con otra de referencia. Según Djukic et al. (2013) este índice puede ser visto como un indicador sobre la calidad de la matriz O-D ajustada si se supone que la matriz O-D de partida contiene el mejor patrón de información."

Para evaluar la similitud estructural entre dos matrices O-D se comparan dos vectores  $d$  y  $\hat{d}$  extraídos de la matriz de partida y ajustada, respectivamente. Estos vectores contienen los elementos de las matrices  $T$  y  $T^*$  que se encuentran dentro de una caja cuadrada de tamaño  $n \times n$ . Esta caja ocupa la misma posición en las dos matrices que se

están comparando y se mueve celda a celda hasta analizar todas las casillas de la matriz. Para cada posición de la caja, análogamente para cada par de vectores  $d$  y  $\hat{d}$ , se obtiene el siguiente valor:

$$SSIM(d, \hat{d}) = \left[ l(d, \hat{d})^\alpha \right] \left[ c(d, \hat{d})^\beta \right] \left[ s(d, \hat{d})^\gamma \right] \quad (I.13)$$

donde  $l$  es una medida de distancia que compara la media de los dos vectores,  $c$  compara la desviación típica de los vectores, y  $s$  compara la estructura de los vectores.

Siendo  $\mu_d = \frac{1}{N} \sum_{n=1}^N d_n$  la media de los valores de  $d$ , se define  $l(d, \hat{d})$  como sigue:

$$l(d, \hat{d}) = \frac{2 \cdot \mu_d \cdot \mu_{\hat{d}} + C_1}{\mu_d^2 + \mu_{\hat{d}}^2 + C_1} \quad (I.14)$$

El término  $c(d, \hat{d})$  toma una forma similar al anterior. Siendo  $\sigma_d = \sqrt{\frac{1}{N} \sum_{n=1}^N (d_n - \mu_d)^2}$

la desviación típica de los valores de  $d$ :

$$c(d, \hat{d}) = \frac{2 \cdot \sigma_d \cdot \sigma_{\hat{d}} + C_2}{\sigma_d^2 + \sigma_{\hat{d}}^2 + C_2} \quad (I.15)$$

El término asociado a la estructura  $s(d, \hat{d})$  se define como la correlación entre los vectores  $d$  y  $\hat{d}$  normalizados. Este término mide el grado de correlación lineal entre los vectores.

$$s(d, \hat{d}) = \frac{\sigma_{d\hat{d}} + C_3}{\sigma_d \cdot \sigma_{\hat{d}} + C_3} \quad (I.16)$$

donde  $\sigma_{d\hat{d}} = \frac{1}{N-1} \sum_{n=1}^N (d_n - \mu_d)(\hat{d}_n - \mu_{\hat{d}})$ .

Finalmente, la medida de calidad de la matriz ajustada viene dada por la media de los índices SSIM obtenidos durante el recorrido de la caja por toda la matriz.

$$MSSIM(T, T^*) = \frac{1}{M} \sum_m SSIM(d_m, \hat{d}_m) \quad (I.17)$$

Los parámetros  $\alpha > 0$ ,  $\beta > 0$  y  $\gamma = 0$  en la ecuación (I.13) son empleados para ajustar la importancia relativa de cada componente. Los parámetros  $C_1$ ,  $C_2$  y  $C_3$  evitan valores infinitos o no válidos en los diferentes términos.

Djukic et al. (2013) para simplificar la expresión de SSIM emplea los siguientes parámetros  $\alpha = \beta = \gamma = 1$  y  $C_3 = C_2/2$ , obteniendo:

$$SSIM(d, \hat{d}) = \frac{(2\mu_d\mu_{\hat{d}} + C_1)(2\sigma_{d\hat{d}} + C_2)}{(\mu_d^2 + \mu_{\hat{d}}^2 + C_1)(\sigma_d^2 + \sigma_{\hat{d}}^2 + C_2)} \quad (I.18)$$

Así, los valores que puede tomar la expresión (I.18) son:  $-1 \leq SSIM(d, \hat{d}) \leq 1$ . El máximo valor  $SSIM(d, \hat{d}) = 1$  se alcanza cuando  $d = \hat{d}$ .

Después de varios test con diferentes métodos de ajuste se plantea una serie de inconvenientes en la aplicación del índice estructural como indicador de similitud de matrices O-D.

Comparando una imagen con una matriz de viajes existen diferencias patentes. La primera es que cada pixel de una imagen puede tomar valores entre 0 y 255, y en las matrices O-D sus elementos tienen valor positivo pero no están acotados, existiendo grandes diferencias entre los elementos de mayor y menor valor. Por otra parte, generalmente los colores de una imagen se mantienen constantes por zonas, es decir, suele existir relación entre los valores de pixeles contiguos. En las matrices de viajes no existe una enumeración u orden concreto para los centroides por lo que la disposición de las filas y columnas en la matriz es aleatoria y no existe relación o dependencia de una casilla con la siguiente. Por tanto lo habitual es encontrar zonas de la matriz O-D con valores muy heterogéneos.

En las pruebas realizadas se ha comparado la matriz de viajes inicial con dos matrices ajustadas mediante dos métodos diferentes. Los indicadores obtenidos han sido muy semejantes y cercanos a la unidad. Comprobando que las matrices ajustadas están desplazadas respecto de la inicial puede concluirse que este indicador es muy poco sensible para el problema de estimación de matrices.

En el ajuste de matrices se parte de una matriz que está inicialmente desactualizada pero que contiene información en cuanto a su estructura por lo que es prioritario realizar procesos de ajuste donde no se apliquen distorsiones excesivas a dicha matriz de inicio. En la práctica, los procesos de ajuste modifican sólo un porcentaje de casillas y estas casillas varían un máximo de un 20% en valor absoluto. Analizando un par de vectores  $d$  y  $\hat{d}$  puede comprobarse que la mayoría de casillas no ha sido modificada y el resto han aumentado o disminuido como máximo un 20%. Debido a esto se tiene que  $\mu_d \simeq \mu_{\hat{d}}$ ,  $\sigma_d \simeq \sigma_{\hat{d}}$  y los vectores  $d$  y  $\hat{d}$  están muy correlacionados, obteniendo un valor de SSIM cercano a la unidad.