

ANÁLISIS DE LA CONVERGENCIA DE LOS PROGRAMAS TRANSYT Y SATURN PARA REDES NO LINEALES

Juan Enrique Coeymans A. (jec@ing.puc.cl), Pablo E. Hernández A. (phernana@ing.puc.cl)
Pontificia Universidad Católica de Chile. Av. Vicuña Mackenna 6860, Macul.
Santiago, Chile
Teléfono: 56 2 6864270. Fax: 56 2 5530281.

RESUMEN

El objetivo del presente estudio es mostrar empíricamente que los programas TRANSYT y SATURN convergen para el caso de redes complejas, a través de sucesivas simulaciones de una red determinada bajo diferentes situaciones de tráfico.

El procedimiento utilizado para mostrar esta convergencia se basó en una ejecución iterativa entre ambos programas, en la que los flujos asignados a la red y las programaciones óptimas de los semáforos para dichos flujos eran datos de entrada para ambos programas, y los factores de cambio entre una corrida y otra.

Se analizaron cuatro casos, que surgen de cruzar dos tipos de situaciones: flujos altos y bajos (y grados de congestión altos y bajos) y una situación base para SATURN, en la cual se cuenta con riqueza de datos de conteo para producir la matriz, y otra en que los datos no fueron tan ricos. Además, se hizo el ejercicio de ver la convergencia cuando TRANSYT optimiza el ciclo, y otro en que el ciclo era dato para este último programa.

Las principales conclusiones de este análisis se pueden resumir en que ambos programas convergen para las distintas situaciones de tráfico planteadas, y que las variables de tráfico influyen solamente en la velocidad de convergencia del proceso.

Asimismo, el criterio más evidente de convergencia resultó no ser la estabilidad en la asignación de los flujos, sino la estabilidad en las programaciones de semáforos en TRANSYT.

1. INTRODUCCIÓN

1.1. Objetivos del Trabajo

El objetivo del presente trabajo es mostrar empíricamente la convergencia entre los programas TRANSYT y SATURN, para el caso de redes no lineales, en diferentes situaciones de tráfico.

Se entenderá por redes no lineales aquellas que no representan un eje arterial solamente, sino que tienen una complejidad en forma de malla..

Corrientemente en la práctica profesional, el diseñador o analista se ve enfrentado a la disyuntiva de tener que trabajar con modelos como SATURN, que asigna flujos, y modelos de optimización de redes como TRANSYT, que optimiza los ciclos, repartos y desfases en intersecciones semaforizadas. Este trabajo, dado que ya se ha analizado en otras oportunidades la convergencia empírica para redes lineales, pretende entregar resultados que permitan suponer como viable, la convergencia para redes no lineales.

1.2. Alcances del Trabajo

Originalmente, esta investigación no pretende establecer un criterio de convergencia entre TRANSYT y SATURN, por lo que el criterio escogido para el análisis pudo haberse transformado en una limitante.

La versión ocupada del programa TRANSYT es la 8S, cuya modelación permite una mejor adaptación a las realidades locales.

Otra limitante del trabajo (contrario a lo que se podría pensar) es la cantidad de información real con la que se trabajó. Por lo general la calibración del programa SATURN se realiza con poca información de los conteos en la red (esto principalmente para la determinación de la Matriz Origen - Destino, como se explicará más adelante), lo cual resta enormemente grados de libertad al problema. Además, la calibración de la red para TRANSYT era un dato en este ejercicio, lo cual, al igual que lo anterior, limita la holgura del mismo.

Por último, este trabajo no espera demostrar la convergencia matemática entre ambos algoritmos, sino más bien mostrar empíricamente que la convergencia entre ambos programas se produce para diferentes situaciones de tráfico y de datos iniciales.

1.3. Contenido

Aparte de esta Introducción, el trabajo comprende otros 4 puntos. En el punto 2 se definen los parámetros iniciales de esta investigación. La convergencia de los dos modelos para las diferentes situaciones de tráfico analizadas se presenta en el punto 3. Finalmente, el punto 4 entrega las principales conclusiones obtenidas de este análisis.

2. DEFINICIÓN DE LOS PARÁMETROS INICIALES

2.1. Recolección de Información

La red a considerar fue obtenida de la UOCT¹, y corresponde a la zona centro de la comuna de San Bernardo², en Santiago (Chile). La principal ventaja de esta red fue el ya estar modelada y calibrada en el programa TRANSYT.



Figura N°1: Mapa de la Zona de Estudio

Otras ventajas de la elección de esta zona fueron:

- La red se encontraba funcionando en el óptimo (de TRANSYT) a la fecha del conteo.
- Se contaba con los flujos en los distintos arcos de la red, para dos periodos del día (punta mañana y punta tarde), tanto para vehículos livianos como para transporte público.
- Se contaba con los flujos de saturación para cada línea de parada.
- Se contaba con las programaciones de los semáforos para dos periodos el día: punta mañana y punta tarde. No se tuvo acceso a las programaciones de otros periodos.

Se tuvo que determinar, además, las frecuencias de los recorridos de transporte públicos existentes en la zona de estudio, los cuales se obtuvieron del Departamento de Transporte de la Pontificia Universidad Católica de Chile y de la Dirección de Tránsito de la Municipalidad de San Bernardo.

¹ Unidad Operativa de Control de Tránsito, Santiago, Chile.

² Santiago, Chile.

2.2. Definición de la Red TRANSYT

La UOCT proporcionó la red con la cual se había trabajado anteriormente. Sin embargo, debido a la extensión original de ésta, y la combinación de sectores lineales y no lineales en la misma codificación, fue necesario reducir la red original.

La red se limitó a un perímetro comprendido entre los siguientes límites:

- Por el Norte : Avenida Colón.
- Por el Sur : Avenida San José
- Por el Este : Barros Arana
- Por el Oeste : Urmeneta y Eyzaguirre

Se eligieron estos límites para formar una zona rectangular del área de estudio original, manteniendo aquellos bloques que presentan mayor interacción en la comuna, y que explican de mejor forma el comportamiento de la misma.

Con estos límites definidos, la red TRANSYT queda codificada de la siguiente manera³:

- 25 nodos.
- 109 arcos.
- 12 calles.

2.3. Codificación de la red SATURN

A partir de la red TRANSYT, se codificó la red SATURN correspondiente a la misma. A este modelo fue necesario agregarle nodos internos (para producir una coherencia mayor entre la realidad y el modelo), y además definir los nodos externos y la zonificación del área de estudio.

La red SATURN consta de las siguientes particularidades⁴:

- 32 nodos internos.
- 20 nodos externos.
- 111 arcos.
- 8 zonas internas.
- 15 zonas externas.

2.4. Definición del Criterio de Convergencia De Trabajo

Basándose en el estudio bibliográfico sobre la materia y a la investigación con especialistas en ambos programas, se definió el criterio de convergencia a utilizar en esta investigación. Se optó por basar este criterio en la diferencia de flujos en los arcos de la red entre una iteración y otra. Esto básicamente porque son los flujos los que determinan el comportamiento de la red, la

³ Para mayores detalles acerca de la codificación de la red, así como de los procedimientos empleados, ver Godoy et al (2000) y Vincent et al (1980).

⁴ Para mayores detalles acerca de la codificación de la red, así como de los procedimientos empleados, ver Godoy et al (2000) y Van Vliet (1985).

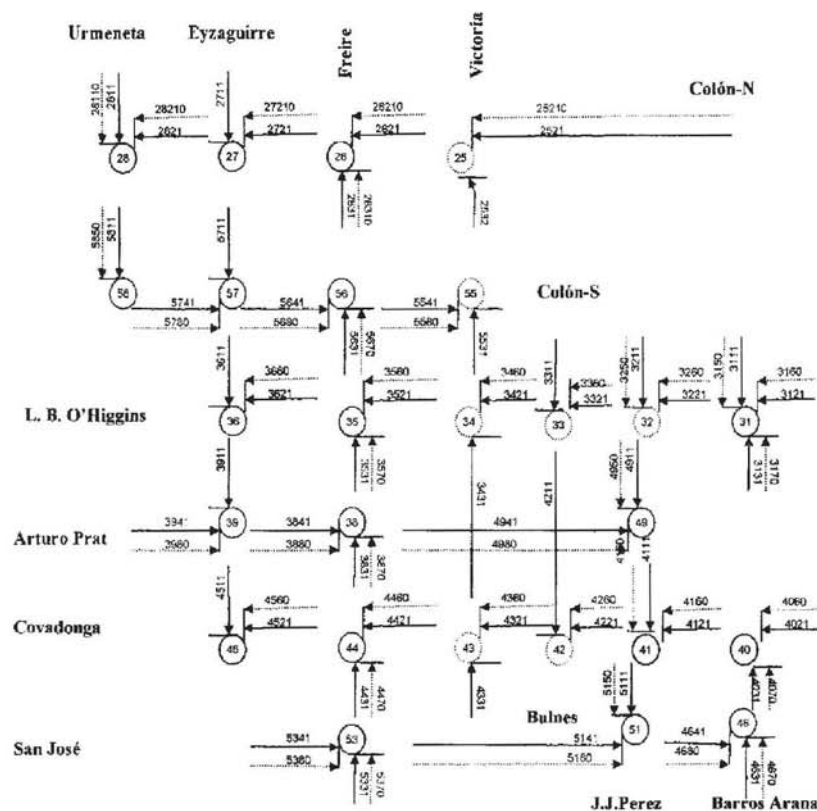


Figura N°2: Malla TRANSYT de la zona de proyecto

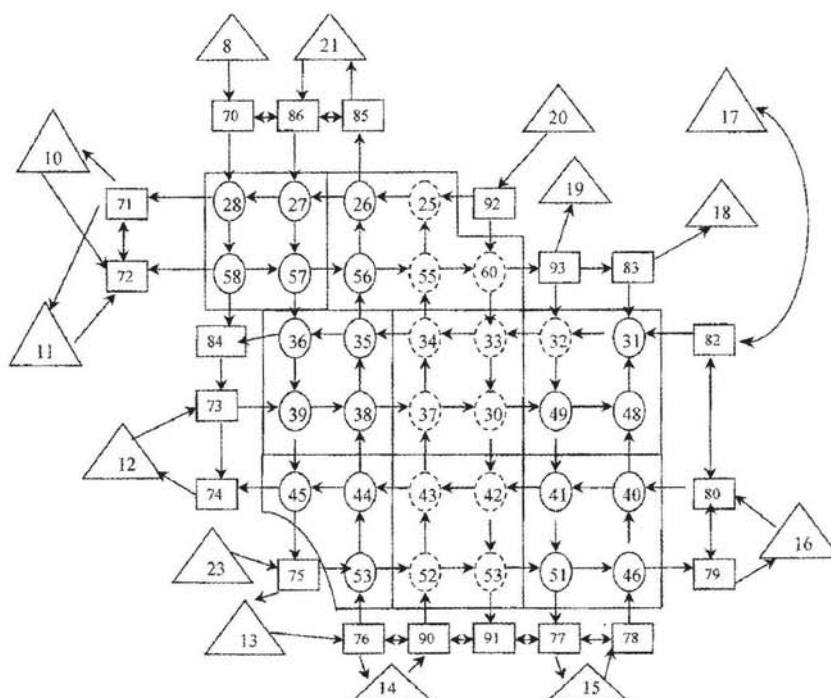


Figura N°3: Malla SATURN de la zona de proyecto

determinación de índices de rendimiento, asignación de reparto de verdes y desfases en la semaforización, así como el grado de saturación de los arcos.

Debido a que el comportamiento en un arco es definitivamente distinto dependiendo de la cantidad de flujo que lo use (básicamente por el grado de saturación del mismo), se optó por definir un criterio de convergencia en dos etapas: para *alto flujo* y para *bajo flujo*. Como límite entre ambas categorías se definió arbitrariamente un flujo de 100 ADE/hr⁵, el que marcaría la separación entre *arcos poco saturados* y *arcos saturados* (sin importar su grado cuantitativo de saturación). Esto lleva a utilizar el siguiente criterio de convergencia:

- 50% de variación de los flujos entre una iteración y otra, para movimientos con menos de 100 ADE/hr.
- 10% de variación de los flujos entre una iteración y otra, para movimientos con más de 100 ADE/hr.

Cabe reforzar que este criterio es el punto de partida del análisis, independiente de las conclusiones a las que se llegaron posteriormente (sobre el criterio de convergencia).

Los porcentajes de variación también son arbitrarios, pero basados fundamentalmente en Dourthe, 1984. El rango de menos de 100 ADE/hr podría ser incluso menos exigente. Si se hubieran empleado criterios más exigentes, ni los rangos de división de flujos en los arcos ni los porcentajes de aceptación del criterio de convergencia influirían en los resultados este estudio, como se verá más adelante.

2.5. Calibración de la Red SATURN y Calibración de la Matriz Origen – Destino⁶

Una vez corroborada la programación de los semáforos otorgada por la UOCT con el programa TRANSYT, se procedió a la calibración de la red SATURN y de la matriz O-D para los flujos de punta mañana (período seleccionado de análisis). Tomando como datos iniciales la codificación original de la red SATURN, los conteos proporcionados por la UOCT y una matriz O-D estimada a priori de acuerdo a la realidad de la red en estudio, se entró en un proceso iterativo cuyo objetivo era encontrar la codificación óptima de la red, así como la matriz O-D más probable para la red en cuestión.

Lo más importante de esta etapa era encontrar una codificación que se ajustara a la realidad, que no modificara mayormente los flujos originales. Esto se obtuvo a través de las siguientes modificaciones a los datos originales, principalmente, introduciendo más zonas externas, que reflejaran los viajes no explicados dentro de la codificación interna de la red.

En cada iteración se hizo un ciclo entre las rutinas de SATURN (SATASS – SATME2 – M1) con el fin de ir estimando, en cada iteración, una mejor aproximación a la matriz O-D real⁷.

⁵ Se utilizó como unidad de referencia el ADE en vez del vehículo, por lo que se utilizarán ambas nomenclaturas indistintamente.

⁶ Para mayor detalle del procedimiento, ver Godoy et al, 2000.

⁷ Para mayores detalles acerca del procedimiento de calibración de la matriz O-D en SATURN, ver Van Vliet (1985).

Después de 11 iteraciones, en las cuales se fueron modificando los atributos de la matriz y de la red anteriormente descritos, finalmente se encuentra una matriz O-D satisfactoria.

3. ANÁLISIS EMPÍRICO DE LA CONVERGENCIA ENTRE LOS PROGRAMAS TRANSYT Y SATURN PARA DIFERENTES SITUACIONES DE TRÁFICO⁸

Una vez obtenida la matriz O-D más probable para la red en cuestión, se procedió a estudiar la convergencia propiamente tal de los programas TRANSYT y SATURN para la zona de análisis ya señalada.

El estudio de la convergencia de los programas TRANSYT y SATURN se realizó para dos casos. Se definieron los siguientes casos según el proceso de optimización definido en TRANSYT:

- a) *Optimización Simple*: Se optimizan tan sólo los desfases entre nodos semaforizados y los repartos de verde en dichos nodos para cada fase, manteniéndose el ciclo constante para todas las iteraciones.
- b) *Optimización Completa*: Se optimizan los desfases y los repartos al igual que para la optimización simple, pero además se pide a TRANSYT que optimice el ciclo a usar en cada iteración.

3.1. Procedimiento de Trabajo para la Verificación de la Convergencia

Partiendo de la situación actual entregada por la UOCT, es decir, considerando el ciclo original para la red (100 segundos) y los repartos de verde en cada intersección semaforizada, se procedió a la iteración entre ambos programas. En particular, se definió en TRANSYT que la optimización de los semáforos fuera tan sólo para desfases y repartos de verde.

Como ya se contaba con la matriz O-D más probable para la zona en estudio, se siguió el siguiente procedimiento para cada iteración:

- Ejecución de TRANSYT, con los flujos asignados en la iteración anterior por SATURN para cada arco, empezando por los entregados por la UOCT.
- Ingreso al archivo de definición de la red de SATURN las programaciones de los semáforos que entregó TRANSYT en el paso anterior.
- Ejecución de las rutinas SATNET, M1 y SATASS de SATURN para obtener las asignaciones en la red para la nueva situación de programación de semáforos.

Sin pérdida de generalidad, se escogió el período Punta Mañana como período de análisis para este ejercicio.

⁸ Para mayor detalle de los procedimientos utilizados para cada uno de estos ejercicios, ver Godoy et al, 2000.

3.2. Verificación de la Convergencia para el Caso de una Optimización Simple

Bajo el procedimiento ya descrito, se procedió a realizar las iteraciones entre ambos programas, obteniéndose la convergencia después de tres iteraciones.

Del análisis de datos, la convergencia se produjo no bajo el criterio planteado originalmente (criterio de los flujos), sino porque las programaciones de las iteraciones 2 y 3 son idénticas. Esto hace que las iteraciones siguientes sean idénticas (ya que nada varía), por lo que hay convergencia exacta.

Cabe aclarar que si bien se presentan variaciones porcentuales entre las iteraciones 2 y 3 para los flujos en los arcos, en la práctica éstas son cero: al ser iguales las programaciones de las iteraciones 3 y 4, los flujos entre las iteraciones 3 y 4 son idénticos también, por lo que la variación de los flujos entre las iteraciones 3 y 4 es nula. Los resultados obtenidos en esta etapa de trabajo se presentan en el Gráfico N°1.

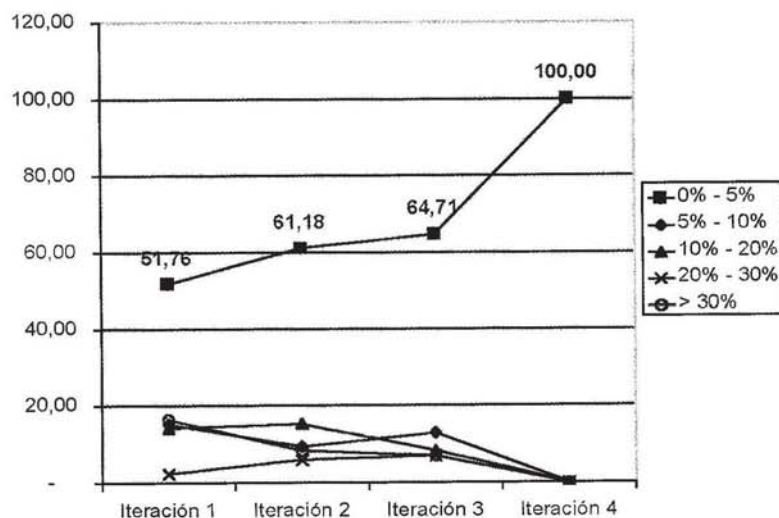


Gráfico N°1: Porcentajes de Variación de los Flujos Asignados por SATURN entre iteraciones, Caso Optimización Simple

Por lo tanto, el criterio de convergencia se cumple para el caso de *Optimización Simple*.

3.3. Verificación de la Convergencia para el Caso de una Optimización Completa

Bajo el procedimiento ya descrito, se procedió a realizar las iteraciones entre ambos programas, obteniéndose una convergencia relativa, que se procede a explicar a continuación.

A partir de la cuarta iteración la mayoría de los movimientos considerados tienden a estabilizarse alrededor de un valor, a excepción de un número mínimo de movimientos (alrededor del 15%) que empieza a oscilar alrededor de un valor promedio, pero que no satisfacen el criterio de convergencia planteado de antemano.

A pesar de lo mencionado, se puede decir que se produce una convergencia relativa para este ejercicio por los siguientes motivos:

- Porque la mayoría de los flujos cumple el criterio de convergencia planteado.
- Porque aquellos flujos que no cumplen con el criterio de convergencia oscilan alrededor de un valor promedio, valor que se podría considerar como de convergencia.
- Porque, si se adopta el criterio tan exigente de pedir que el 100% de los flujos cumpla el criterio de convergencia, bastaría tan solo con eliminar dichos conteos del archivo correspondiente de SATURN, con lo que se tendría una convergencia completa.
- Porque los flujos que presentan mayor oscilación (y que no han convergido hasta la novena iteración) son todos flujos menores a los 100 ADE/hr.
- Porque las programaciones de los semáforos, una vez que se estabiliza el ciclo óptimo para la red modelada, se mantienen estables. Éste es, al final, el verdadero criterio de convergencia.

Los distintos resultados obtenidos entre iteraciones para este caso, se aprecian en los Gráficos N°2 y N°3.

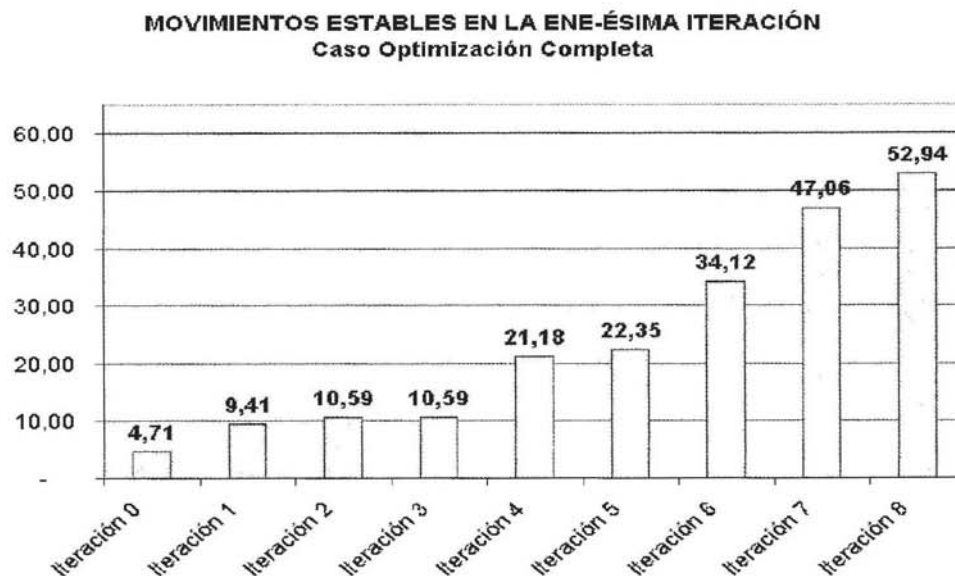


Gráfico N°2: Porcentaje de Movimientos cuyo Flujo no varía a partir de la Ene-Ésima Iteración. Caso Optimización Completa

Si bien no se produce una estabilización tan clara como la mostrada para el caso de la *Optimización Simple*, se puede apreciar que a partir de la 7ª iteración (iteración en que el ciclo óptimo para la red se estabilizó) una tendencia a la estabilización. Además, ya fue mostrado que, para el caso que el ciclo es fijo, el problema llega a converger. Por lo tanto, el criterio de convergencia se cumple para el caso de *Optimización Completa*.

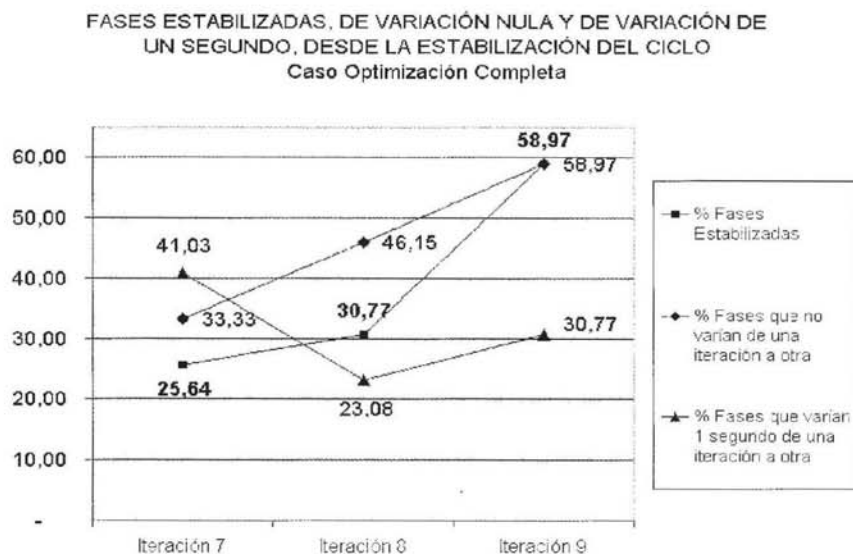


Gráfico N°3: Comportamiento de las fases de los semáforos en la red a partir de la 7ª Iteración. Caso Optimización Completa.

3.4. Caso de Flujos Menores a los Originales

El objetivo de este ejercicio fue determinar si el criterio de convergencia demostrado para el caso de los flujos de punta mañana se cumple también para el caso de flujos menores en la red⁹.

Para ello, y debido a no contar con una estructura de flujos menor a la original¹⁰, se procedió a la disminución de un 40% de los flujes originales. Esto llevó a la calibración de una nueva matriz O-D que reflejara la realidad de esta nueva situación.

Una vez calibrados los parámetros de entrada para ambos programas, se procedió a trabajar con el procedimiento de optimización completa descrito anteriormente (es decir, optimización de ciclo, desfases y repartos).

Para este ejercicio, la estructura convergió luego de la sexta iteración. Los distintos resultados obtenidos entre iteraciones para este caso, se aprecian en el Gráfico N°4.

⁹ Debido a que los flujos considerados *punta* suelen ser los más exigentes a que las redes de transporte suelen ser sometidos.

¹⁰ Se tenía información solamente sobre las estructuras de flujos de Punta Mañana y Punta Tarde en la zona, no difiriendo éstas considerablemente.

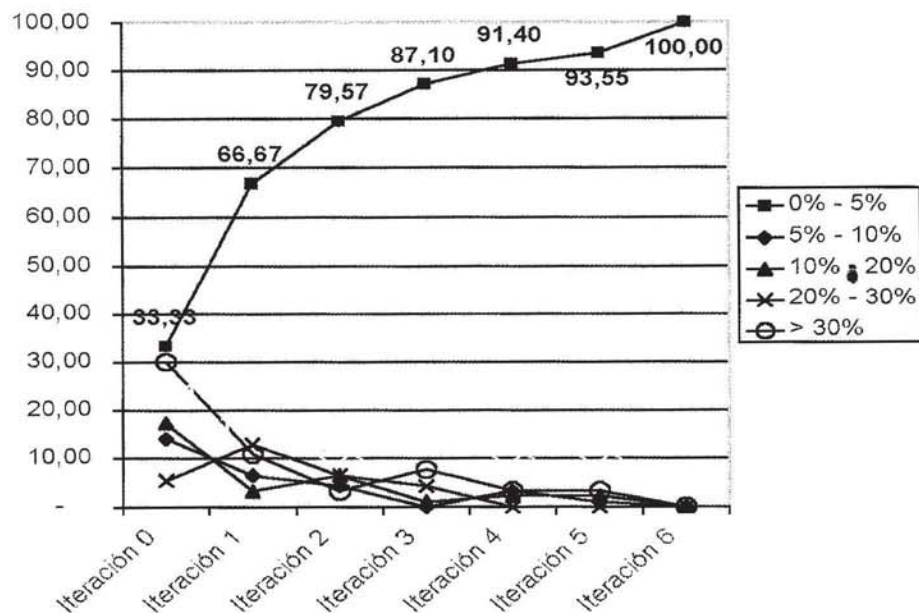


Gráfico N°4: Porcentajes de Variación de los Flujos Asignados por SATURN entre iteraciones. Caso Optimización Completa con Flujos Menores

La convergencia en este caso fue prácticamente perfecta ya que le mayor porcentaje de variación de flujos entre la penúltima y la última iteración no superó el 1%. Esto nos lleva a inferir que la cantidad de flujo incide negativamente en la velocidad de convergencia entre ambos programas, es decir, a flujos más altos menor cantidad de iteraciones son necesarias para la convergencia.

3.5. Caso de Matriz Mal Calibrada

Lo que se pretende a través de este ejercicio es determinar la importancia de la calibración de la matriz O-D y de la cantidad de datos usados en dicha calibración en la velocidad de convergencia de ambos programas.

Lo primero que se hizo para este ejercicio fue elegir una matriz O-D mal calibrada. Para ello, se escogió la primera calibración obtenida en el proceso de calibración para el ejercicio original. Esta matriz consta tan sólo de 15 zonas: las mismas 8 zonas internas del caso original de análisis (conectadas de distinta forma a la red original), pero tan sólo 7 zonas externas (a diferencia de las 15 zonas externas del problema original). Esta matriz fue rechazada previamente por otorgar 24 movimientos con una variación significativa entre la asignación y la realidad.

Para resolver este problema, fueron eliminados los movimientos cuya variación era significativa entre asignación y realidad, ajustándose así la matriz O-D en forma casi perfecta. Como se puede deducir del párrafo anterior, se usaron 61 conteos de movimientos en lugar de los 85 del ejercicio original.

La convergencia se produce, en este caso, en la quinta iteración. Esta convergencia es perfecta, es decir, los flujos obtenidos en la quinta y en la sexta iteración son idénticos.

Los distintos resultados obtenidos entre iteraciones para este caso, se aprecian en el Gráfico N°5.

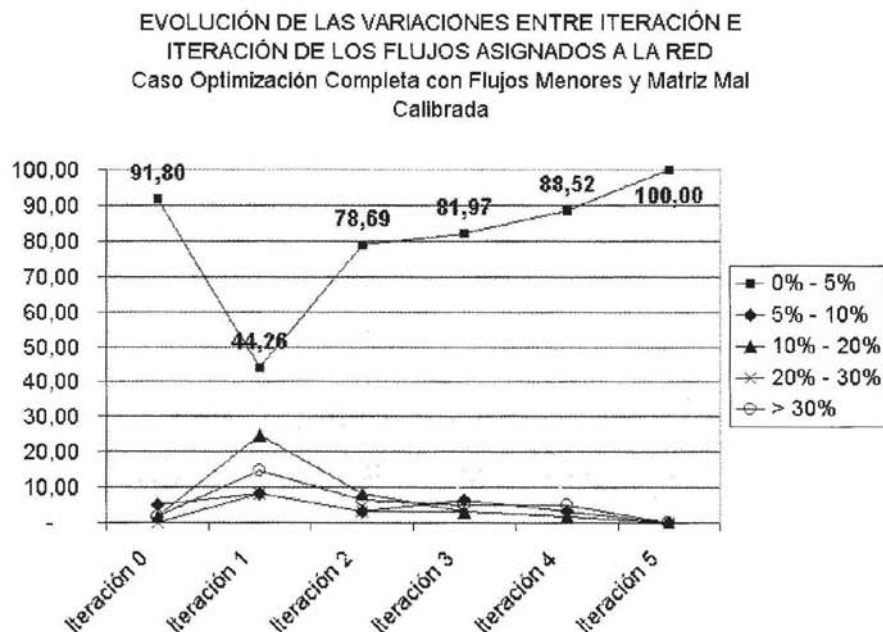


Gráfico N°5: Porcentajes de Variación de los Flujos Asignados por SATURN entre iteraciones. Caso Optimización Completa con Flujos Menores y Matriz O-D Mal Calibrada

4. CONCLUSIONES

La convergencia entre los programas TRANSYT y SATURN se mostró empíricamente para las distintas situaciones de tránsito presentadas en este paper, para el caso de redes no lineales tipo malla.

Es importante notar que dicha convergencia se produjo independientemente del proceso de optimización adoptado en TRANSYT para el ejercicio: ya sea optimizando tan sólo repartos y desfases, como optimizando ciclo, repartos y desfases, hay una convergencia de los flujos.

La definición de un criterio de convergencia apropiado para aseverar una convergencia total entre ambos programas es un aspecto que afecta tan sólo al número de iteraciones necesarias para demostrar la convergencia, y no una variable fundamental para validar o rechazar esta tesis. La convergencia se da entre ambos programas, independiente del criterio de convergencia que se utilice para el análisis.

El criterio de convergencia puede ser planteado de dos maneras, las dos equivalentes entre sí:

- *Convergencia en los flujos:* Adoptar como criterio de convergencia que la variación de los flujos para cada movimiento (y/o arco) entre una iteración y otra no varíe de un cierto porcentaje determinado. Este porcentaje debe depender de la cantidad de flujo original asignada a cada movimiento y/o arco, adoptándose divisiones macro (caso de este trabajo) o en tramos más pequeños (como en el caso de Dourthe, 1984). Independiente del número de tramos que se elijan, lo importante es tener claro que no se puede exigir el mismo porcentaje de diferencia a arcos con una diferencia significativa de grado de saturación.
- *Convergencia en las programaciones:* La convergencia se puede dar primero en las programaciones de los semáforos. Esto es, si en dos iteraciones sucesivas las programaciones de los semáforos (ciclo, desfases y repartos) no varían. Esto conlleva a que los flujos en los movimientos y/o arcos tampoco varíe (variación de un 0%), con lo que cualquier criterio de convergencia en los flujos se cumpliría. Es importante señalar que este último aparece como un mejor y más práctico criterio de convergencia.

La definición de los porcentajes de convergencia así como los tramos de flujo para los cuales se definen son variables de entrada para este problema, y es el modelador quien debe determinar a priori dichos valores, sobre la base del grado de profundidad del problema a ser estudiado, y de las variables que determinan la velocidad de convergencia de la iteración entre ambos programas.

Además, la convergencia entre ambos programas no depende de la cantidad de flujo a asignar en la red. Esta variable tan sólo determina la velocidad de convergencia entre ambos ejercicios.

Tampoco afecta a la convergencia la correcta codificación de la red. Es más, el tener numerosa información acerca de los flujos que recorren los distintos arcos de la misma tan sólo lleva a demorar el proceso de convergencia más que a ayudar a una correcta modelación del problema. Se puede inferir del análisis de redes mal codificadas que el exceso de información limita los grados de libertad del problema, y que es recomendable trabajar con conteos para aquellos movimientos importantes en la red en vez de con la totalidad de conteos en la misma¹¹.

Por lo tanto, se puede afirmar que, al menos para el tipo de casos simulados en este trabajo, los programas TRANSYT y SATURN convergen para el caso de redes no lineales, y que las variables que afectan la velocidad de convergencia son las siguientes:

- Cantidad de conteos considerados para la entrada en SATURN.
- Grado de saturación en los arcos de la red.
- Validación de la matriz O-D.
- Profundidad del criterio de convergencia.

A futuro, se recomienda plantear el criterio de convergencia sobre la base de las programaciones de los semáforos en la red, ya que es este fenómeno el que se produce primero en el estudio.

¹¹ Esta afirmación es válida para los conteos que requiere SATURN para asignar a la red, y no para los flujos que requiere TRANSYT en su proceso de optimización.

UN MÉTODO LOCAL PARA RESOLVER EL EQUILIBRIO DEL USUARIO EN REDES DE TRANSPORTE

Francisco J. Martínez C., Martín I. Matamala V.
Universidad de Chile
Casilla 228-3 Santiago, Chile
email:fmartine@cec.uchile.cl • mmatamal@dim.uchile.cl

RESUMEN

El problema del equilibrio del usuario en redes de transporte con demanda fija se ha resuelto por la aplicación directa de las condiciones de equilibrio de Wardrop para rutas alternativas entre zonas de origen y de destino (centroides), sujeto a satisfacer restricciones de red; esto caracteriza una estrategia de equilibrio global. En este estudio la red es descompuesta en redes elementales, llamadas regiones, a las cuales se les aplica las condiciones de Wardrop intercambiando flujos entre rutas. Ya que los flujos en arcos se almacenan por origen del viaje, se pueden aplicar muchas estrategias para seleccionar de qué origen se intercambia flujo en cada caso. Seleccionando una estrategia, el enfoque local resuelve en forma iterativa problemas elementales del equilibrio del usuario, convergiendo, eventualmente, a un equilibrio global para redes cíclicas. Este enfoque tiene la ventaja teórica de interpretar el problema del equilibrio global como compuesto por un agregado de equilibrios en redes locales, segundo, identifica los orígenes de los viajes en el flujo de cada arco y tercero, genera una familia completa de nuevos algoritmos. Pruebas preliminares muestran que el algoritmo propuesto es al menos tan eficiente como los métodos tradicionales.

1. INTRODUCCIÓN

La noción de equilibrio del tráfico en las redes de transporte fue definida en el destacado trabajo de Wardrop (1952). Para cada par origen-destino (par OD) se obtiene el equilibrio si los costos (percibidos) de los viajes de los usuarios en las rutas utilizadas son iguales unos a otros y no mayores que aquellas rutas no utilizadas. El problema matemático no lineal de encontrar el equilibrio del flujo en arcos se denomina problema del equilibrio del usuario de transporte (PEU).

Es usual utilizar algunos supuestos simplificadorios para generar algoritmos de solución para el PEU que limitan su complejidad. Estos disminuyen el consumo de recursos computacionales y garantizan convergencia. Estos son: los flujos se asumen como variables continuas; los costos de viaje en los arcos son funciones crecientes dependientes del nivel de flujo en los arcos e independientes del nivel de flujo en otros arcos. La demanda de viajes por par OD se supone elástica y depende de los costos de los viajes en el equilibrio, aunque el caso especial de la demanda fija es útil para ilustrar el mecanismo de los algoritmos de solución.

Una familia de enfoques de solución ha sido derivada de la formulación de programación no-lineal del PEU de Beckmann *et al.* (1956). Muchos autores han contribuido con técnicas de solución tanto para las formulaciones primales como duales (ver Fukushima, 1984). Hay dos elementos comunes en esta familia de algoritmos: primero, enfrentan la solución global, es decir, el equilibrio es concebido al nivel de origen-destino de los viajes, con la interacción entre los pares OD impuestas a través de las restricciones de la red. Segundo, resuelven el problema de asignación en función del flujo total de los arcos. Una excepción en este segundo elemento es el trabajo reciente de Bar-Gera (1999), quien propone un algoritmo para redes cíclicas donde la solución es una sub red a-cíclica y se representa por flujos en arcos basados en el origen y agregados a todos los destinos. Nuestro trabajo también obtiene resultados con flujos basados en el origen pero el algoritmo es radicalmente diferente en cuanto al primer elemento, pues Bar-Gera logra obtener un algoritmo más eficiente pero usando un enfoque global.

El enfoque local presentado en este estudio se aparta del enfoque global. En este, hemos descompuesto la red en un conjunto de componentes elementales denominados regiones. Estas son pares de rutas entre nodos intermedios que conectan un nodo llamado origen con otro llamado destino, incluyendo el caso restrictivo donde una de estas rutas no existe, llamado circuito. Luego, encontramos intercambios óptimos de flujo entre las rutas de las regiones definidos por el equilibrio local wardropiano para cada región. De acuerdo con el conocido principio Wardrop, el equilibrio se puede obtener o bien con flujos positivos en ambas rutas (equilibrio fuerte) o con una de las rutas vacía (equilibrio débil). El objetivo de nuestro enfoque es iterar este proceso de intercambio óptimo de flujo región por región hasta encontrar un óptimo global. Intuitivamente, mientras el enfoque clásico o global procede equilibrando los costos de las rutas para cada par OD, el enfoque local simplemente equilibra las regiones. Sin embargo, este enfoque necesita resolver tres grandes problemas: la interdependencia entre regiones, su potencial baja eficiencia debido al crecimiento exponencial del número de regiones con el tamaño de la red y la interacción entre los flujos de diferente origen.

La interdependencia ocurre porque los arcos son compartidos entre las regiones, haciendo que sus soluciones de equilibrio local sean dependientes entre ellas. Probaremos que cada paso de equilibrio local mejora la función objetivo de Beckmann *et al.* (1956). Entonces, si un algoritmo

equilibra cada una de las regiones, este resuelve el problema de la interdependencia. De este modo, en nuestro enfoque, la función objetivo de Beckmann *et al.* (1956) disminuye con cada paso, sin embargo, este hecho sólo no garantiza que este proceso converja a óptimos globales para cualquier red real. Proponemos un algoritmo llamado $M^2(1)$ que genera una secuencia que converge al óptimo global.

La eficiencia, en el caso del enfoque global, depende exponencialmente del tamaño de la red y del número de pares OD; esto también es válido para el caso de $M^2(1)$, aspecto que mejoraremos de modo significativo en el algoritmo $M^2(2)$. En esta versión mejorada, para cualquier red (incluidas aquellas con ciclos), encontraremos un conjunto de regiones con un tamaño reducido que es proporcional al tamaño de la red, denominado base de la red, a la cual se le aplica $M^2(1)$. Por la simpleza de encontrar los equilibrios en cada región y por el reducido tamaño de la base, la eficiencia de $M^2(2)$ aumenta enormemente. Aunque este método es suficiente en muchos casos de redes, no es correcto en los casos más generales donde las soluciones del equilibrio global de Wardrop contienen rutas vacías, simplemente debido a que el equilibrio débil no se conserva en las regiones agregadas; sólo el equilibrio fuerte se mantiene sobre la agregación. Por lo tanto, se propone un método correcto aunque aproximado, denominado $M^2(\epsilon)$, para modificar ligeramente las funciones de costo de arcos usando métodos de barrera, los cuales evitan los equilibrios débiles manteniendo la secuencia generada por el algoritmo dentro del dominio de los flujos estrictamente positivos. Intentaremos probar que la diferencia entre las funciones de Beckmann *et al.* en la solución original óptima y nuestra solución aproximada es limitada por un parámetro ϵ controlado por el modelador.

La interacción entre los orígenes se soluciona simultáneamente con el procedimiento de intercambio de flujo, al seleccionar una estrategia para elegir de qué origen se intercambian los flujos, la que se aplica a menos que el flujo de origen sea cero. Este criterio supone que los vehículos individuales son homogéneos excepto por su origen. Es claro que otros criterios alternativos podrían ser implementados, produciendo el mismo equilibrio de Wardrop, pero un patrón diferente de los flujos por ruta y origen.

Como se observa, a pesar que el enfoque local utiliza viejas ideas, tal como intercambiar flujos, además introduce algunos métodos nuevos. Por ejemplo, el enfoque local, que se basa en descomponer la red en regiones; la noción de optimización en un conjunto reducido de regiones que define la red base; y la aplicación del método de barrera para resolver configuraciones de red comunes. El método tiene el primer mérito de descomponer la red en sus unidades básicas, el segundo en producir arcos de flujo diferenciados por el origen mientras simultáneamente optimiza a todos los orígenes y un tercero de ser eficiente y correcto aunque aproximado.

En la siguiente sección se describe la estructura local, seguida en la tercera sección por la representación de un conjunto de algoritmos que conducen al equilibrio global. Luego, en la Sección 4 probaremos el funcionamiento de los algoritmos $M^2(\epsilon)$ con dos redes, Leblanc *et al.* (1975) y una sub red de la red de transporte de Puerto Montt al sur de Chile.

2. EL ENFOQUE LOCAL

2.1. Notación de la red

N : el conjunto de nodos, con n nodos.

U : el conjunto de arcos, con m elementos.

$(c_a)_{a \in U}$: funciones de costo de arcos, no-negativas continuas y crecientes.

$(K_a)_{a \in U}$: capacidad de arcos.

$O \subseteq N$: el conjunto de r orígenes.

$D \subseteq N$: el conjunto de destinos.

$T = (t_{ij})$: la matriz de viajes demandados, donde $(t_{ij}) = 0$ si $i \notin O$ o $j \notin D$ y $t_i = \sum_{j \in N} t_{ij} \quad \forall i \in O$

x_a^i : el flujo en arcos $a \in U$ con origen $i \in O$.

x_a : el flujo total en el arco $a \in U$.

2.2. El problema de optimización PEU

Consideremos la transformada de Beckmann et. al (1956) de las condiciones de equilibrio Wardrop en el siguiente problema de minimización equivalente:

$$v(TUE) = \min_{x \in X} F(x)$$

$$F(x) = \sum_{a \in U} \int_0^{x_a} c_a(w) dw \quad (P)$$

y X : arcos de flujo no-negativos $(x_a^i)_{a \in U, i \in O}$ satisfaciendo la demanda (t_{ij}) .

Aquí los flujos factibles en arcos se definen como vector diferenciando la contribución de cada origen al flujo total del arco, de tal manera que $x \in X : x = (x^1, x^2, \dots, x^r)$ donde cada $x^i, i \in O$, es un vector no-negativo de tamaño m con elemento x_a^i que satisface para cualquier $i \in N$ (Figura 1):

$$t_i \delta_{ij} + \sum_{a: a^- = i} x_a^i = \sum_{a: a^+ = j} x_a^i + t_{ij}$$

o de manera equivalente

$$Ax^i = d^i$$

para $i \in O$ y donde A es la matriz de incidencia nodo-arco (de tamaño $n \times m$) de la red y

$$d_j^i = \begin{cases} -t_{ij} & j = i \\ t_{ij} & j \neq i \end{cases} \text{ es decir, } d_j^i \text{ es el flujo neto del origen } i \text{ que deja la red en el nodo } j \text{ (cuando } j=i,$$

el flujo que ingresa la red en el origen i es $-d_j^i$).

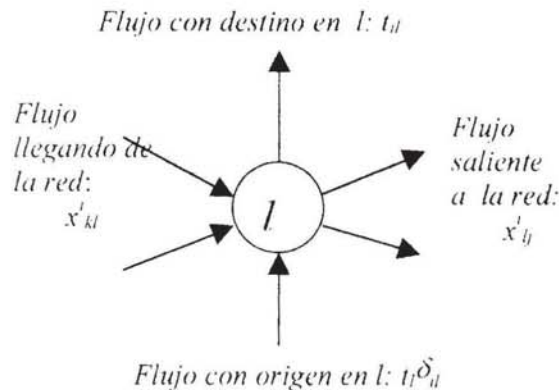


Figura 1: Notación de flujo asociado al origen i en el nodo l .

Interpretamos F como la función de energía del problema PEU. Nótese que F es una función aditiva de las integrales de costos, de tal manera que la contribución de cada arco para minimizar la energía es evidente. Esto motiva nuestro enfoque local basado en modificar una solución factible en un grupo de arcos de manera de mantener la factibilidad y reduciendo la función F . La estructura local natural en las redes de transporte es un par de rutas con el mismo origen y el mismo destino al que llamaremos una región.

Definición 2.1. Una región $R=(R^+, R^-)$ es un ciclo compuesto por dos rutas R^+ y R^- con un origen y destino común. Denotaremos por \mathbf{R} al conjunto de todas las regiones en la red de transporte.

Por ejemplo, en la red ilustrada en la Figura 2 tenemos la región R compuesta por las rutas $R^+=(1,2,6,13)$ y $R^-=(4,11,15,16)$ con origen A y destino K . El vector representativo de una región R es el vector $h \in \{-1,0,1\}^m$ definido por

$$h_a = \begin{cases} 1 & \text{si } a \in R^+ \\ -1 & \text{si } a \in R^- \\ 0 & \text{diferente} \end{cases}$$

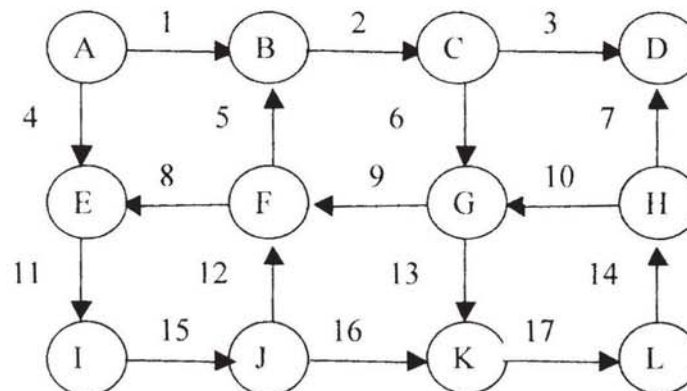


Figura 2: Ejemplo de red de transporte con 17 arcos y 12 nodos

El vector representativo de la región R compuesta por rutas $R^+ = (1,2,6,13)$ y $R^- = (4,11,15,16)$ en la red que aparece en la Figura 2 pertenece a $\{-1,0,1\}^{17}$ y está dado por

$$h_R = (1,1,0,-1,0,1,0,0,0,0,-1,0,1,0,-1,-1,0)$$

La matriz de incidencia nodo-arco A asociado a la red de la figura 2 es:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
B	-1	1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0
C	0	-1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
D	0	0	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0
E	0	0	0	-1	0	0	0	-1	0	0	1	0	0	0	0	0	0
$A=F$	0	0	0	0	1	0	0	1	-1	0	0	-1	0	0	0	0	0
G	0	0	0	0	0	-1	0	0	1	-1	0	0	1	0	0	0	0
H	0	0	0	0	0	0	1	0	0	1	0	0	0	-1	0	0	0
I	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	1	0	0
J	0	0	0	0	0	0	0	0	0	0	0	1	0	0	-1	1	0
K	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	-1	1
L	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	-1

donde las columnas están asociadas a arcos y las filas a los nodos. La columna asociada a arco (i,j) tiene un 1 en la fila i y un -1 en la fila j . Es un resultado conocido (ver Papadimitriou y Steiglitz, 1998) que para cualquier región R su vector representativo h_R satisface $Ah_R=0$, esto es, el vector h_R que pertenece al núcleo de la matriz A .

2.3. Método del intercambio de flujo

Ahora mostraremos que los cambios de flujo en las regiones nos permite reducir F manteniendo la factibilidad en los flujos. En términos del equilibrio de Wardrop, un punto factible \hat{x} es óptimo para el problema PEU si para todo par de origen o y destino k y cualquier par de rutas R^+ y R^- que conectan o y k , los costos de estas rutas son iguales o diferentes solamente si el más costoso no tiene flujo o el más barato alcanza su capacidad. De modo inverso, si la solución factible x no es óptima hay un origen o y un destino k y dos rutas R^+ y R^- entre ellas con costos diferentes, la más costosa tiene un flujo positivo y la más barata tiene flujo bajo la capacidad. Una estrategia para mejorar el equilibrio en estas dos rutas es cambiar el flujo de la más costosa a la más barata de tal modo que se reduce la diferencia en los costos.

Para la siguiente discusión se conviene en que el costo de R^- es menor que el de R^+ , es decir

$$\sum_{a \in R^+} c_a(x_a) - \sum_{a \in R^-} c_a(x_a) > 0 \quad (1)$$

Para intercambiar flujo de R^+ a R^- y mantener la factibilidad se necesita elegir un origen $o \in O$ y luego agregar un flujo apropiado λ del origen o a todo arco en R^- y restar el mismo flujo de todos los arcos en R^+ , en esta situación el costo en R^+ disminuye y aumenta en R^- . Entonces la diferencia entre los términos de la ecuación(1) disminuye.

Sea \tilde{x} el vector obtenido de x después del intercambio de flujo. Para el origen escogido o , $\tilde{x}_a^o = x_a^o + \lambda$, para $a \in R^-$ y $\tilde{x}_a^o = x_a^o - \lambda$ para $a \in R^+$ y $\tilde{x}_a^o = x_a^o$ para todos arcos a que no esté incluido ni en R^+ ni en R^- ; para todo otro origen u y todo otro arco a , $\tilde{x}_a^u = x_a^u$. Por lo tanto, podemos escribir $\tilde{x} = x - \lambda h$, donde $\hat{h} = h(R, o)$ es un vector de tamaño $r \cdot m$ dado por

$$\hat{h}_a^u = \begin{cases} 1 & u = o \text{ y } a \in R^+ \\ -1 & u = o \text{ y } a \in R^- \\ 0 & \text{diferente} \end{cases}$$

es decir $\hat{h} = (h^1, \dots, h^r)'$ donde para cada $i=1, \dots, r$ el vector h^i está en IR^m , para todo $i \neq o$, $h^i = \bar{0}$ y h^o es el vector representativo de la región R (el superíndice t denota vector transpuesto).

El cambio de flujo mejora el equilibrio local en la región formada por R^+ y R^- , sin embargo, aun no se puede decir nada sobre el mejoramiento global. Para analizar el efecto global del intercambio de flujos usaremos la gradiente de F en x , dada por $(\nabla F(x))_a^o = c_a^o$. Entonces

$$\nabla F(x) \cdot h(R, o) = \left(\sum_{a \in R^+} c_a(x_a) - \sum_{a \in R^-} c_a(x_a) \right) \quad (2)$$

Nuestro análisis previo puede ser resumido en lo siguiente. Si x no es un equilibrio de Wardrop entonces hay $R=(R^+, R^-)$ y $o \in O$ tal que $\tilde{x} = x - \lambda h(R, o) \in X$ para algún $\lambda > 0$ y $\nabla F(x) \cdot h(R, o) > 0$. Es decir \tilde{x} se obtiene de x al moverse en la dirección $-h(R, o)$. De la desigualdad (1) e igualdad (2) sabemos que $-h(R, o)$ es una dirección descendente para F en x . Es bien conocido el resultado de cálculo básico que es posible moverse en una dirección $-h(R, o)$ de tal manera que el valor de F disminuye; en el caso de las redes de transporte esta propiedad significa que por el intercambio de flujo de R^+ a R^- la función F disminuye. En el Teorema 2.2 mostraremos que mientras

$$\nabla F(x - \lambda h(R, o)) \cdot h(R, o) = \sum_{a \in R^+} c_a(x_a - \lambda) - \sum_{a \in R^-} c_a(x_a + \lambda) \geq 0$$

al intercambiar λ unidades de flujo de R^+ a R^- la función F disminuye.

Teorema 2.2. Sea $R=(R^-,R^+)$ una región compuesta por dos rutas R^- a R^+ , x un punto factible y $o \in O$ tal que $\nabla F(x) \cdot h(R,o) > 0$. If $x + \lambda_0 h(R,o) \in X$ y $\sum_{a \in R^+} c_a(x_a - \lambda_0) - \sum_{a \in R^-} c_a(x_a + \lambda_0) \geq 0$ para algún $\lambda_0 > 0$ entonces $F(x - \lambda_0 h(R,o)) - F(x) < 0$.

Demostración: Sea $\tilde{x} = x - \lambda h(R,o)$. Entonces

$$\tilde{x}_a = \sum_{i \in I} \tilde{x}_a^i = \sum_{i \in I} x_a^i - \sum_{i \in I} \lambda (h(R,o))_a^i = x_a - \lambda h(R,o)_a^o$$

y la variación en la función F es

$$\Delta(\lambda) = F(\tilde{x}) - F(x) = \sum_{a \in R} \int_{x_a}^{\tilde{x}_a} c_a(w) dw = \sum_{a \in R} \int_{x_a}^{x_a - \lambda h(R,o)_a^o} c_a(w) dw \quad (3)$$

Un punto clave es que $\Delta(\lambda)$ sólo depende de los costos de los arcos en la región $R=(R^-,R^+)$. Para analizar el comportamiento de $\Delta(\lambda)$ obtenemos $\Delta'(\lambda)$:

$$\Delta'(\lambda) = \sum_{a \in R} -h(R,o)_a^o c_a(x_a - \lambda h(R,o)_a^o) = \sum_{a \in R} c_a(x_a + \lambda) - \sum_{a \in R^+} c_a(x_a - \lambda)$$

Al utilizar el teorema del valor medio y $\Delta(0) = 0$ obtenemos $\Delta(\lambda) = \lambda \Delta'(\xi)$ con $\xi \in (0, \lambda)$. Al reemplazar en la derivada de $\Delta(\lambda)$ y considerando que $c_a(x_a + \lambda)$ aumenta para $a \in R^-$ y $c_a(x_a - \lambda)$ disminuye para $a \in R^+$ con λ , obtenemos que $\Delta'(\lambda) > \Delta'(\xi)$, lo que implica:

$$\Delta(\lambda) < \lambda \Delta'(\lambda) = \lambda \left(\sum_{a \in R} c_a(x_a + \lambda) - \sum_{a \in R^+} c_a(x_a - \lambda) \right)$$

Para $\lambda_0 > 0$ tal que $\sum_{a \in R^+} c_a(x_a - \lambda_0) - \sum_{a \in R^-} c_a(x_a + \lambda_0) \geq 0$ concluimos que $\Delta(\lambda_0) < 0$. Por lo tanto, $F(\tilde{x}) = F(x - \lambda_0 h(R,o)) < F(x)$.

En palabras esto prueba que mientras se intercambie flujo sin invertir la relación de costo original, se obtiene una ganancia neta en la función global F . Esto define una estricta relación entre el intercambio local hacia el equilibrio y el mejoramiento global en F .

En resumen, hemos definido un procedimiento basado en intercambios de flujo en una región, manteniendo la condición de factibilidad, mejorando el equilibrio de las regiones y simultáneamente la función de objetivo global F .

3. DEL EQUILIBRIO LOCAL AL GLOBAL

La aplicación reiterativa del procedimiento de intercambio para todos los pares de ruta (R^+, R^-) y todos los orígenes genera un algoritmo que mejora la función objetiva en cada paso. El algoritmo se detiene cuando no hay un par de rutas donde la solución factible \hat{x} pueda ser modificada, es decir los costos son equilibrados en el sentido de Wardrop para todos los pares de ruta (R^+, R^-) y todos los orígenes, de este modo \hat{x} es el equilibrio global.

Definición 3.1. Una región R está:

- (1) Equilibrada de forma fuerte en el flujo x si $\sum_{a \in R^+} c_a(x) = \sum_{a \in R^-} c_a(x)$. En este caso para todo origen $o \in O$ tenemos $\nabla F(x) \cdot h(R, o) = 0$.
- (2) Equilibrada de forma débil en el flujo x si $\sum_{a \in R^+} c_a(x) > \sum_{a \in R^-} c_a(x)$ y para cada $o \in O$ y cada $\lambda > 0$, $x - \lambda h(R, o) \notin X$.
- (3) Desequilibrado en flujo x si ni 1 ni 2 se cumple.

En términos de regiones y de acuerdo a las definiciones anteriores, tenemos que una solución factible \hat{x} es un equilibrio Wardrop si y sólo si no hay una región que sea desequilibrada, es decir, que el conjunto de regiones desequilibradas, de ahora en adelante denotadas como $DR(\mathbf{R}, x)$, es vacío.

$$\text{Definamos } \theta(\mathbf{R}, x) = \max_{o \in O, R \in DR(\mathbf{R}, x)} \nabla F(x) \cdot h(R, o) = \max_{o \in O, R \in DR(\mathbf{R}, x)} \left(\sum_{a \in R^+} c_a(x) - \sum_{a \in R^-} c_a(x) \right)$$

si $DR(\mathbf{R}, x) \neq \emptyset$ y $\theta(\mathbf{R}, x) = 0$ en otro caso. Por lo tanto $\theta(\mathbf{R}, x)$ es el valor máximo de la proyección de la gradiente de F en x en las direcciones asociadas a las regiones. Entonces \hat{x} es un equilibrio Wardrop si y solo si $\theta(\mathbf{R}, \hat{x}) = 0$.

Algoritmo 3.2. El Algoritmo "crudo" $M^2(1)$

- (1) Encontrar cualquier solución factible $x \in X$.
- (2) Mientras $DR(\mathbf{R}, x) \neq \emptyset$:
 - (a) Elija una región $R \in DR(\mathbf{R}, x)$ y $o \in O$ tal que $\nabla F(x) \cdot h(R, o) = \theta(\mathbf{R}, x)$.
 - (b) Calcular $\lambda_0 > 0$ tal que $\nabla F(x - \lambda_0 h(R, o)) \cdot h(R, o) = 0$.
 - (c) Fije el nuevo valor de x en $x - \lambda_0 h(R, o)$
- (3) Pare.

Este algoritmo produce una secuencia de soluciones factibles (x_n) . La solución x_{n+1} se obtiene de x_n al cambiar flujo en una región que está desequilibrada. El proceso finalmente se detiene si todas las regiones están equilibradas, pero esto podría no detenerse.

Teorema 3.3. *Si el algoritmo $M^2(1)$ se detiene con una solución \bar{x} , entonces es una solución óptima para el problema PEU.*

Demostración: Primero recuerde que el algoritmo produce soluciones factibles en cada paso. Cada paso encuentra un equilibrio wardropiano para la región analizada, aunque al realizarlo podría modificar las condiciones de equilibrio de otras regiones con arcos compartidos. Sin embargo, el algoritmo se detiene cuando todas las regiones están equilibradas simultáneamente, en particular aquellas regiones que incluyen centroides OD. Ya que las soluciones óptimas al problema PEU son exactamente las del equilibrio de Wardrop para la red, deducimos que \bar{x} es un mínimo de la función objetivo F .

Hasta ahora hemos probado que si el algoritmo se detiene encuentra la solución global. Ahora consideraremos el caso cuando el algoritmo no se detiene.

Teorema 3.4. *Bajo los supuestos de compacidad y la condición de Lipschitz sobre ∇F tenemos que si el Algoritmo $M^2(1)$ no se detiene entonces produce una secuencia de soluciones factibles cuyos puntos de acumulación son el equilibrio de Wardrop.*

Demostración: Si asumimos que el algoritmo no se detiene, una serie infinita de soluciones factibles (x_k) es producida, la que mejora la función objetivo. Bajo las suposiciones de compacidad (x_k) tiene una subserie que converge en \hat{x} . Necesitamos probar que \hat{x} es un equilibrio Wardrop.

Bajo la suposición de Lipschitz, podemos probar que para n suficientemente grande, $F(x_{n+1}) - F(x_n) \leq c(\theta(\mathbf{R}, \hat{x}))^2$ con $c > 0$. Si \hat{x} no es un equilibrio Wardrop hay una región R desequilibrada en el flujo \hat{x} . Entonces $\theta(\mathbf{R}, \hat{x}) \neq 0$ y la secuencia $F(x_n)$ no está acotada, lo cual es imposible ya que la función F es no-negativa.

No obstante, este algoritmo no es eficiente ya que el tamaño del conjunto \mathbf{R} crece muy rápido con el tamaño de la red, es decir, tiene que considerar demasiadas regiones en una red real. Entonces, necesitamos incrementar la eficiencia encontrando una representación compacta de las regiones.

3.1. La base de regiones

Hasta ahora los elementos del conjunto $\mathbf{H} = \{h(R, o) : R \text{ una región y } o \in O\}$, han sido utilizados como direcciones descendentes para la función F ; el principal problema de nuestro primer algoritmo es que el tamaño de \mathbf{H} podría ser excesivamente grande, exponencialmente proporcional al tamaño de la red.

Considere las regiones R , S y T en la Figura 2 dadas por

$$h_R = (1, 0, 0, -1, -1, 0, 0, 0, 0, 0, -1, -1, 0, 0, -1, 0, 0)$$

$$h_S = (1, 1, 1, -1, 0, 0, -1, 0, 0, 0, -1, 0, 0, -1, -1, -1, -1)$$

$$h_T = (0, -1, -1, 0, -1, 0, 1, 0, 0, 0, 0, -1, 0, 1, 0, 1, 1)$$

esto es

$$R^+ = (1)$$

$$R^- = (4, 11, 15, 12, 5)$$

$$S^+ = (1, 2, 3)$$

$$S^- = (4, 11, 15, 16, 17, 14, 7)$$

$$T^+ = (16, 17, 14, 7)$$

$$T^- = (12, 5, 2, 3)$$

Si restamos el vector h_S del vector h_R obtenemos el vector h_T . En palabras, la región T se puede obtener como una combinación lineal de las regiones R y S . Aun más, es fácil probar que si las regiones R y S están equilibradas de manera fuerte entonces T también lo está. Por lo tanto, en nuestro algoritmo anterior no es necesario considerar todas las regiones en \mathbf{R} . Mostraremos que es posible considerar un subconjunto \mathbf{B} de \mathbf{H} con un tamaño solamente proporcional al tamaño de la red mientras mantenemos el propósito de converger al equilibrio global.

Considere $\hat{h} = h(R, o) \in \mathbf{H}$ y denote $\hat{h} = (h^1, \dots, h^r)'$. Recuerde que de la definición de $h(R, o)$ que para todo $i \neq o$, $h^i = \bar{0}$ y que $h^o = h_R$, donde h_R es el vector representativo de la región R . Sea A la matriz incidente nodo-arco de la red y sea $\hat{A} = \text{diag} (A, \dots, A)$ del tamaño $r \cdot n \times r \cdot m$. Ya que $\hat{A} \hat{h} = (Ah^1, \dots, Ah^r)'$ tenemos que para todo $i \neq o$, $(\hat{A} \hat{h})^i = A\bar{0} = \bar{0}$ y $(\hat{A} \hat{h})^o = Ah_R$. Es posible probar que $Ah_R = \bar{0}$ (ver Papadimitriou y Steiglitz, 1998) por lo tanto \hat{h} pertenece al núcleo de la matriz \hat{A} .

Considere el subespacio lineal $Y \subseteq IR^{mr}$ (donde m es el número de arcos de la red y r es el número de orígenes) generados por \mathbf{H} , que es el conjunto Y de todos los vectores obtenidos como combinaciones lineales de vectores en \mathbf{H} . Ya que cada elemento \hat{h} de \mathbf{H} satisface $\hat{A} \hat{h} = 0$ concluimos que $Y = \ker(\hat{A})$.

Se sabe que la dimensión del núcleo de la matriz A es $(m-n+1)$, donde n es el número de nodos de la red (ver Papadimitriou y Steiglitz, 1998). Por lo tanto la dimensión de Y es $p = r(m-n+1)$ y por consecuencia Y tiene una base de tamaño p , que existe en un conjunto $\mathbf{B} \subseteq Y$ de tamaño p tal que todos los vectores en Y pueden ser escritos como una combinación lineal de los vectores en \mathbf{B} . Más importante para nuestro propósito es que \mathbf{B} puede ser escogido como un subconjunto de \mathbf{H} . Al usar la base \mathbf{B} en vez del conjunto de todas las regiones \mathbf{R} , obtenemos los siguientes algoritmos.

Algoritmo 3.5. El Algoritmo $M^2(2)$ "eficiente aunque incorrecto"

- (1) Para cualquier $i \in O$, calcule un vector factible x^i tal que $Ax^i = d^i$.
- (2) Defina $x = (x^1, \dots, x^r)$.
- (3) Mientras $DR(\mathbf{R}, x) \neq \emptyset$:
 - (a) Elija un elemento $R \in DR(\mathbf{B}, x)$ y $o \in O$ tal que $\nabla F(x) \cdot h(R, o) = \theta(\mathbf{B}, x)$
 - (b) Equilibre R .
 - (c) Actualice el valor de x .
- (4) Pare.

Este algoritmo tiene dos características principales:

Eficiente: Explota la estructura de la red para reducir el espacio de las direcciones a un conjunto finito de tamaño p . De este modo la búsqueda clásica de direcciones descendentes, que necesita ser realizada en cada flujo x , aquí se reduce a un número finito que no dependen del flujo x . Esto constituye una evidente fuente de eficiencia del algoritmo cuando se compara con otros. Por ejemplo, el algoritmo de Bar-Gera (1999) realiza dos procedimientos básicos: uno estructural, que consiste en actualizar la subred de flujos no-nulos, y otro que podemos llamar operacional que consiste en intercambios de flujo en esa sub-red para lograr equilibrio. El procedimiento estructural es el más costoso en términos computacionales, mientras que el intercambio de flujo es menos costoso y se realiza por igual tanto en su método como en el nuestro. Bar-Gera reduce el número de cambios estructurales pero inevitablemente debe realizar un número significativo de ellos, en cambio en algoritmo $M^2(2)$ no actualiza redes porque siempre trabaja con misma base de regiones. Esta diferencia es una fuente de eficiencia fundamental de nuestro método.

Correcto: Lo cierto es que podría $M^2(2)$ detenerse con algunas regiones equilibradas de forma débil, en cuyo caso el equilibrio global no puede ser garantizado. Recuerde que un equilibrio débil ocurre en una región R cuando

$$\sum_{a \in R^+} c_a(x) > \sum_{a \in R^-} c_a(x)$$

y para todos los orígenes hay un arco a en R^+ con $x_a^l = 0$ o hay un arco a en R^- con $x_a = K_a$. Cabe notar que la condición de flujo cero en un arco a depende de la elección de las regiones en \mathbf{B} , de este modo podríamos cambiar la base \mathbf{B} para reparar el algoritmo $M^2(2)$. El problema con esta estrategia es que dado el tamaño del conjunto de bases perdemos eficiencia.

El dominio X tiene una breve descripción en términos de matriz \hat{A} : $x \in X$ si y solo si $\hat{A}x = \hat{d}$, donde $\hat{d} = (d^1, \dots, d^r)^t$, y x no es negativo. Además, tenemos la siguiente condición óptima: $x \in X$ es óptima si y sólo si para todo \hat{h} en el núcleo de \hat{A} la proyección del gradiente h es cero. Ya que \mathbf{B} es una base del núcleo de \hat{A} tenemos el siguiente teorema.

Teorema 3.6. Si x es una solución positiva factible (todas sus coordenadas son positivas) y todas las regiones en la base \mathbf{B} están equilibradas de manera fuerte en x , entonces x es una solución óptima.

Demostración: Nuestra suposición dice que $\nabla F(x) \cdot h(R, o) = 0$ para todas las regiones $h(R, o)$ en **B**. Ya que x pertenece a un interior relativo de X con respecto a los núcleos de \hat{A} y que **B** es una base para el núcleo de \hat{A} , concluimos que $\nabla F(x) \cdot \hat{h} = 0$ para todo \hat{h} en el núcleo de \hat{A} . Por lo tanto, x es una solución óptima.

Considerando este teorema, la estrategia para obtener un algoritmo es, obviamente, eliminar el equilibrio débil en regiones y producir soluciones factibles en el interior del dominio. Para tal propósito aplicamos un método de barrera, el que consiste en modificar ligeramente las funciones de los costos del arco agregando un término que mantenga la solución factible al interior del dominio y tal que todo proceso de equilibrio en regiones produzca un equilibrio fuerte.

Consideremos $\varepsilon > 0$ y $g: (0, +\infty) \rightarrow \mathbf{IR}$ una función estrictamente convexa con $\lim_{t \rightarrow 0^+} g'(t) = -\infty$. Considere el problema modificado P_ε dado por

$$\text{Min } F^\varepsilon(x) = F(x) + \varepsilon G(x)$$

$$\text{donde } G(x) = \sum_{a \in I} g(K_a - x_a) + \sum_{a \in I} \sum_{o \in O} g(x_a^o) \quad P(\varepsilon)$$

Bajo la hipótesis sobre la función g el problema P_ε tiene una sola solución ya que la función F^ε es estrictamente convexa. Denotamos esta solución por $x(\varepsilon)$.

Para entender como trabaja este método de barrera recuerde que en $M^2(1)$ el procedimiento de intercambio flujo se aplica a una región $R = (R^+, R^-)$ y a un origen $o \in O$, transfiriendo el flujo de R^+ a R^- tal que, ya sea:

- (1) los costos llegan a ser igualados (R llega a ser equilibrado de forma fuerte),
- (2) el flujo en R^+ asociado al origen llega a ser cero, o bien
- (3) el flujo total en algún arco en R^- alcanza capacidad.

Al utilizar la función de costo modificada arco-origen $\bar{c}_{a,o}$, dada por

$$\bar{c}_{a,o} = c_a + \varepsilon(g'(x_a^o) - g'(K_a - x_a))$$

evitamos las últimas dos opciones, porque el costo de la función $\bar{c}_{a,o}$ llega a infinito negativo cuando x_a^o llega a cero y llega a infinito positivo cuando x_a se acerca a la capacidad.

Algoritmo 3.7. El tercer Algoritmo $M^2(\varepsilon)$ "Eficiente y Correcto pero Aproximado"

- (1) Elija un nivel de precisión $\varepsilon > 0$.
- (2) Para cualquier $i \in O$ calcule un vector positivo x^i tal que $Ax^i = d^i$ (una solución factible para el origen i).
- (3) Defina $x = (x^1, \dots, x^r)$ (una solución global factible).
- (4) Mientras $DR(\mathbf{R}, x) \neq \emptyset$ para F^c :
 - (a) Elija un elemento $R \in DR(\mathbf{B}, x)$ y un origen $o \in O$ tal que $\nabla F^c(x) \cdot h(R, o) = \theta(\mathbf{B}, x)$
 - (b) Equilibre R con respecto a o .
 - (c) Actualice el valor de x .
- (5) Pare.

Teorema 3.8. *Si el algoritmo $M^2(\varepsilon)$ no se detiene genera una secuencia que converge al óptimo de F^c .*

Demostración: La prueba es similar a la del Teorema 3.4. Debido a la selección de $h(R, o)$ podemos asegurar que la secuencia generada por el algoritmo converge en puntos factibles x en el interior relativo de X y en el punto $\theta(\mathbf{B}, x) = 0$.

Para asegurar la calidad de la solución de P_ε tenemos que analizar el comportamiento de la solución óptima $x(\varepsilon)$ para cuando ε se va a cero.

Teorema 3.9. *Sea $x(\varepsilon)$ la solución óptima del problema P_ε . De este modo cualquier punto de acumulación de $(x(\varepsilon))_{\varepsilon > 0}$ es una solución óptima del problema P .*

Demostración: Sea z cualquier solución factible positiva de P (todas sus coordenadas son positivas). Ya que $x(\varepsilon)$ es la solución óptima del problema P_ε y para todo $\lambda \in [0, 1]$ el punto $\lambda z + (1 - \lambda)x^*$ es una solución factible positiva, tenemos que

$$F(x(\varepsilon)) + \varepsilon G(x(\varepsilon)) \leq F(\lambda z + (1 - \lambda)x^*) + \varepsilon G(\lambda z + (1 - \lambda)x^*)$$

Suponemos que hay una constante K tal que $K \leq G(x(\varepsilon))$ para todo $\varepsilon > 0$. Para mantener esta suposición es suficiente que todos los arcos tengan una capacidad finita. En esta situación tenemos

$$F(x(\varepsilon)) + \varepsilon K \leq F(\lambda z + (1 - \lambda)x^*) + \varepsilon G(\lambda z + (1 - \lambda)x^*)$$

Sea \bar{x} un punto de acumulación de $(x(\varepsilon))_{\varepsilon > 0}$. Dado que F es una función continua

$$F(\bar{x}) \leq F(\lambda z + (1 - \lambda)x^*) \quad (4)$$

Tomando el límite cuando λ tiende a cero en (4) obtenemos $F(\bar{x}) \leq F(x^*)$. Por lo tanto, \bar{x} es una solución óptima del problema P .

En resumen el algoritmo $M^2(\varepsilon)$ explota la estructura natural de una red de transporte lo que la hace eficiente, evita el equilibrio débil al perturbar la función objetiva empujando la solución al interior del dominio y produce una secuencia que converge en una solución aproximada del problema P como se establece en el Teorema 3.9.

4. RESULTADOS EMPIRICOS

4.1. Características de la implementación

- En la implementación de algoritmo $M^2(\varepsilon)$ hemos modificado la parte 4a. En vez de elegir (y encontrar) una región y un origen tal que $\nabla F^*(x) \cdot h(R, o) = \theta(B, x)$ equilibramos todas las regiones en la base en un orden establecido. Esto es lo que denominamos una iteración.
- El equilibrio para una región R es ejecutado con respecto al origen o que presenta el desequilibrio máximo en R . Este es un criterio opcional que podía modificarse en el futuro.
- Para realizar del cambio de flujo calculamos el flujo mínimo en el origen o en los arcos de R^+ y el mínimo de la diferencia entre la capacidad de los arcos y el flujo del arco en R^- . Tomando el mínimo de estas dos cantidades definimos el cambio máximo de flujo π_M . De este modo encontramos un valor $\pi \in (0, \pi_M)$ tal que al cambiar las unidades de flujo π de R^+ a R^- la región R se equilibra. En esta implementación ignoramos la restricción capacidad.
- El criterio de detención es que la variación en la función objetiva de F (la original) sería menor que 1%.
- Usamos la función $g(x) = -\log(x)$.

4.2. Datos

Consideramos las redes de transporte de Sioux-Falls¹ (Leblanc *et al.* 1975) y una sub red de transporte en la ciudad Puerto Montt (Chile). Los parámetros importantes de estos ejemplos están dado en la siguiente tabla:

	Nodos	Arcos	Orígenes	Destinos	Elementos en la Base	Viajes Min. Min. $\{t_{ij}\}$	Viajes Máx. Máx. $\{t_{ij}\}$
Sioux-Falls	24	76	24	24	406	1	44
Puerto Montt	135	343	36	36	685	0.1	275.2

Tabla 1: Características de las redes

La función de costo utilizada en ambos casos es $c_a(x_a) = A_a + B_a x_a^J$

¹ Ciudad en Dakota del Sur, Estados Unidos.

4.3. Computación

Para obtener información de la dependencia del algoritmo $M^2(\varepsilon)$ sobre el volumen de flujo hemos aumentado los flujos OD en la red de Puerto Montt por factores 3, 7, 10 y 15. Esto nos entrega seis ejemplos que llamaremos Sioux-Falls, Puerto Montt, Puerto Montt 3, Puerto Montt 7, Puerto Montt 10 y Puerto Montt 15. Todos los cómputos fueron llevados a cabo en una Sun Ultra Sparc-4, 400Mhz, 1GRAM, con sistema operativo Solaris y el código fue escrito en C. Para cada ejemplo, realizamos los siguientes cómputos

- i) Para $\varepsilon \in \{0.01, 0.05, 0.10\}$ corrimos $M^2(\varepsilon)$, para verificar el efecto de la precisión del parámetro. En este caso $M^2(\varepsilon)$ corre para un ε fijo.
- ii) Para una secuencia (ε_n) que va a cero corrimos $M^2(\varepsilon)$, donde en la $n^{\text{ésima}}$ reiteración $\varepsilon = \varepsilon_n$, para $n \geq 1$. Aquí verificamos la hipótesis que al correr $M^2(\varepsilon)$ con un valor decreciente de ε en cada reiteración se mejora la solución.
- iii) Aplicamos el software Saturn 8 para tener una solución de referencia.

<i>Sioux Falls</i>	Iteraciones	CPU tiempo (seg.)	$F(x(\varepsilon))$
$\varepsilon = 0,01$	5	4,7	$895,10^3$
$\varepsilon \rightarrow 0$	5	5,3	$927,10^3$
Saturn	10	-	$923,10^3$
Le Blanc et al		-	$1210,10^3$
<i>Puerto Montt</i>			
$\varepsilon = 0,01$	2	4,2	$2492,10^3$
$\varepsilon \rightarrow 0$	3	5,7	$2486,10^3$
Saturn	3	-	$2371,10^3$
<i>Puerto Montt 3</i>			
$\varepsilon = 0,01$	4	6,4	$7657,10^3$
$\varepsilon \rightarrow 0$	4	6,4	$7584,10^3$
Saturn	3	-	$7317,10^3$
<i>Puerto Montt 7</i>			
$\varepsilon = 0,01$	5	7,0	$2418,10^4$
$\varepsilon \rightarrow 0$	6	7,6	$2405,10^4$
Saturn	3	-	$2271,10^4$
<i>Puerto Montt10</i>			
$\varepsilon = 0,01$	7	7,8	$6180,10^4$
$\varepsilon \rightarrow 0$	7	7,8	$6207,10^4$
Saturn	3	-	$5500,10^4$
<i>Puerto Montt15</i>			
$\varepsilon = 0,01$	8	8,2	$3114,10^5$
$\varepsilon \rightarrow 0$	9	8,5	$3091,10^5$
Saturn	3	-	$2700,10^5$

Tabla 2: Resultados de corridas

En la solución de equilibrio para la red de Puerto Montt hay un arco que está sumamente congestionado con respecto a los otros, mientras en la solución de equilibrio de la red de Sioux-Fall hay una congestión más homogénea. Esto explica en forma parcial porqué nuestro modelo funciona mejor en Sioux-Fall que Saturn, mientras en Puerto Montt el resultado de funcionamiento es el contrario. De acuerdo a estos resultados limitados se podría decir que en casos como Puerto Montt podría ser mejor una estrategia global, porque sólo se necesita tener cuidado con los arcos congestionados, mientras en casos con una congestión más homogénea todos los arcos importan por lo que la estrategia local es mejor.

Algunos resultados complementarios son:

- Ya que el valor de ε es pequeño, no obtenemos diferencias importantes entre $F(x(\varepsilon))$ y $F_\varepsilon(x(\varepsilon))$.
- En algunos casos como Sioux Falls y Puerto Montt 7 y 15, la opción de $\varepsilon \rightarrow 0$ no provee el mínimo de F . Este resultado resalta el hecho que el Teorema 3.8 sólo garantiza que el algoritmo $M^2(\varepsilon)$ converja en una solución óptima para P_ε , cuando ε es establecido; hasta ahora, no hay un teorema equivalente para $\varepsilon \rightarrow 0$. De hecho, el Teorema 3.9 se aplica a una secuencia de soluciones obtenidas como puntos límites del algoritmo $M^2(\varepsilon)$ (con ε fijo) y sólo garantiza que estas soluciones se acumulen en una solución global para P .

5. COMENTARIOS FINALES

Este estudio identifica una "estructura natural para las redes de transporte", denominadas regiones, que son un subconjunto de los ciclos utilizados en la Teoría de Gráfos. Ellas constituyen la unidad elemental de la red, de las cuales están compuestas las redes complejas de transporte. A nuestro modo de ver la mera descomposición de un objeto complejo como las redes de transporte es la primera contribución importante, sin embargo es más bien de interés académico.

Una primera aplicación es el desarrollo de un nuevo algoritmo para resolver el clásico problema del equilibrio del usuario del transporte (PEU). Sin embargo, esperamos que otras aplicaciones puedan ser desarrolladas en investigación aplicada. Por ejemplo, haciendo un zoom de la red a redes locales con una base de regiones a nivel municipal que sea más detallada.

La segunda contribución de interés es la identificación de una base de regiones, la cual tiene las siguientes características importantes:

- Es independiente al nivel de la demanda, lo que implica que una vez que es identificada, puede utilizarse para resolver el PEU, o para otras aplicaciones, con una demanda variable. Esta característica llega a ser primordial en un modelo dinámico, donde la demanda es en esencia variable con el tiempo.
- Es dependiente de la red y del conjunto de nodos de origen y destino. Esto hace una base específica a las características de la red. Sin embargo reconstruir (parchar) la base para considerar un número limitado de cambios provoca un esfuerzo extra menor.

- Constituye la estructura de la red que se mantiene inalterada durante el desarrollo del algoritmo M^2 , el que solamente realiza procedimientos de intercambio de flujo y no de estructura. Esta es la fuente principal de eficiencia en redes complejas ya que sólo se desarrollan cálculos unidimensionales (cambio de flujo). En forma adicional, por naturaleza se pueden introducir cálculos paralelos utilizando la descomposición de la red en regiones.

Hemos propuesto una familia de algoritmos caracterizados por un enfoque local que genera un gran número de algoritmos que necesitan de un estudio más detallado, lo cual es materia para una investigación futura. Por ejemplo, hemos implementado un método específico de cambio de flujo, pero se puede considerar otros, como es el paso 4a de $M^2(\epsilon)$ que se podría modificar en: la forma como se elige a una región, el conjunto de orígenes seleccionados para cambiar flujo en cada región y el criterio utilizado para definir la cantidad de flujo utilizado. Algunos de estos algoritmos óptimos pueden desarrollarse con gran eficiencia y precisión.

AGRADECIMIENTOS

Esta investigación fue apoyada por FONDAP en Matemática Aplicada y Fondecyt 1981206 y 1010422. Por cierto, agradecemos los comentarios de árbitros anónimos.

REFERENCIAS

- Bar-Gear, H. (1999) Origin-based Algorithms for Transportation Network Modeling. Technical Report 103, National Institute of Statistical Sciences, U.S.A.
- Beckmann, M., McGuire, C. B. y Winsten, C. B. (1956) **Studies in the Economics of Transportation**. Yale University Press, New Haven, CT.
- Fukushima, M. (1984) On the dual approach to the traffic assignment problem. **Transportation Research B**, 18B, Nr 3, 235-245.
- LeBlanc, L.J., Morlok, E.K. y Pierskalla, W. P. (1975) An efficient approach to solving the road network equilibrium traffic assignment problem. **Transportation Research** 9, 309-318.
- Papadimitriou, C. y Steiglitz, K (1998) **Combinatorial optimization algorithms and complexity**. Prentice-Hall 1982 second edition by Dover, (1998).
- Wardrop, J. G. (1952) Some theoretical aspects of road traffic research. **Proceedings Inst. Civil Eng.**, 1, Part II, 352-378.

REFERENCIAS

Dourthé, A., (1984) Modelo de Evaluación de Pistas Exclusivas para la Locomoción Colectiva Memoria para optar al Título de Ingeniero Civil, Universidad de Chile.

Godoy, C., Hernández, P., Lizama, X. (2000) Análisis de la Convergencia entre los Programas TRANSYT y SATURN para Redes No Lineales Seminario de Ingeniería de Transporte, Pontificia Universidad Católica de Chile.

Van Vliet, D. (1985) **SATURN 9.2: User Manual.** University of Leeds, England

Vincent, R.A., Mitchell, A.I., Robertson, D.I (1980) **User Guide to TRANSYT Version 8**, TRRL Laboratory Report 888.