

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA CIVIL
SECCIÓN DE POSGRADO



**METODOLOGÍA PARA MEJORAR EL PROCESO
DE ASIGNACIÓN DE TRÁFICO A UNA RED DE
TRANSPORTE**

T E S I S

Para optar el Grado de Maestro en Ciencias
con Mención en Ingeniería en Transportes

Lic. Rolando Gandhi Astete Chuquichaico

LIMA - PERÚ

2011

Índice

Resumen	VI
Lista de Figuras	VII
Lista de Tablas	VIII
Lista de Siglas y Símbolos	IX
Introducción	X
1. Marco Metodológico	1
1.1. Tipo de Investigación	1
1.2. Realidad y problemática	1
2. Marco Teórico	2
2.1. Descripción de Conceptos Previos	2
2.2. Red de Transporte	2
2.3. Modelo de 4 Etapas	6
2.4. Modelos de Programación	7
2.4.1. Programación Lineal	7
2.4.2. Programación no Lineal	12
2.5. Método Simplex	14
2.5.1. Método Simplex Revisado	17
2.6. Método del Gradiente	18
2.7. Método del gradiente conjugado	19
3. Marco Filosófico y Tecnológico	22
3.1. introducción	22

3.2. Fundamentos de la asignación de tráfico	24
3.2.1. Modelos de asignación de tráfico	25
3.3. Modelos de optimización	27
3.3.1. Modelo de Transporte	28
4. Diseño de la Metodología	30
4.1. Generalidades	30
4.2. Formulación del Problema de Asignación de Tráfico	30
4.3. El método de Frank Wolfe	33
4.3.1. Algoritmo de Frank-Wolfe	36
5. Implementación Computacional	47
5.1. Introducción	47
5.2. Implementación del modelo	47
5.3. Aplicación	49
6. Conclusiones y Recomendaciones	53
6.1. Conclusiones	53
6.2. Recomendaciones	53
Apéndice	55
Programación en MATLAB Wolfenson-1	55
Archivo simplex-revisado.m	55
Archivo raiz-wolfenson.m	56
Archivo intercambia-simplex.m	56
Archivo basesale-simplex.m	56
Archivo datosinicial.m	57
Bibliografía	58

Resumen

El proceso de planificación del transporte consta de varias fases, una de ellas es la etapa de asignación, la cual sirve para crear nuevos escenarios para la toma de decisiones. Los problemas de asignación de tráfico se resuelven optimizando los flujos en una determinada red de transporte, los algoritmos resuelven problemas de asignación de un origen contra varios destinos.

Las funciones que describen el flujo de transporte, son funciones no lineales con restricciones lineales, estas funciones son efectivas cuando consideran solo algunos parámetros principales involucrados en los problemas de flujos de redes, tales como el costo de desplazamiento y el tiempo de desplazamiento. En la presente Tesis se implementa el algoritmo de Frank-Wolfe para optimizar el proceso de asignación a una red de transporte.

Summary

The transportation planning process consists of several phases, one of them is the assignments stage, which is used to create new scenarios for decision making. The assignments problems of traffic are resolved optimizing network flows in a certain network transport, the algorithms solve problems from a origin to multiple destinations

The functions that describe the flow of transport, are nonlinear functions with linear constraints, these functions are effective when they consider only some main parameters involved in the problems of flows in networks, such as the replacement cost and displacement time. In the present thesis is implemented the Frank-Wolfe algorithm to optimize the process of assignment to a transport network.

Lista de Figuras

2.1. Digrafo conexo	4
2.2. Función convexa	5
2.3. Modelo de 4 etapas	6
4.1. Digrafo conexo	35
4.2. Región Factible del problema	37
4.3. Dirección factible del ejemplo	38
4.4. Nuevo Punto de Iteración x_1	39
4.5. Determinando la dirección factible y_1 desde x_1	40
4.6. Nuevo Punto de Iteración x_2	41
4.7. Determinando la dirección factible y_2 desde x_2	42
4.8. Nuevo Punto de Iteración x_3	42
4.9. Determinando la dirección factible y_3 desde x_3	43
4.10. Nuevo Punto de Iteración x_4	44
4.11. Determinando la dirección factible y_4 desde x_4	45
4.12. Nuevo Punto de Iteración x_5	46
5.1. Diagrama de Flujo	49
5.2. Red vial	50
5.3. Red vial (Problema 2)	51

Lista de Tablas

5.1. Parámetros para las funciones BPR(Problema 1)	51
5.2. Estimación de los viajes(Problema 1)	51
5.3. Parámetros para las funciones BPR(Problema 2)	52
5.4. Estimación de los viajes(Problema 2)	52

Lista de Siglas y Símbolos

$G(A, V)$	Digrafo conexo y sin lazos
C_a	tiempo de viaje en el arco a en función del flujo
c_a^0	tiempo de viaje libre en el arco
f_a	flujo en el arco a
k_a	Capacidad de arco(Vehículos/hora)
a	arco
h_r	flujo en la ruta r
Z_{min}	Función objetivo mínimo
Z_{max}	Función objetivo máximo
$C_a(f_a)$	función de coste que indica el retraso en el arco $a \in A$, para cada arco $(i, j) \in A$, como función del flujo total f_a que lleva el mismo arco a
\mathbb{R}^n	Espacio vectorial columna n -upla de los reales
v^T	Denota la traspuesta del vector columna $v \in \mathbb{R}^n$
v_j	Denota el j -ésimo elemento
$\sum_{j=1}^n v_j$	La suma de los elementos v_j desde $j = 1$ hasta $j = n$
\mathbf{x}_B	Partición del vector \mathbf{x} correspondiente a las variables básica
\mathbf{x}_N	Partición del vector \mathbf{x} correspondiente a las variables no básica
$(\mathbf{B} \ \mathbf{N})$	Partición de la matriz A
$(\mathbf{c}_B^T \ \mathbf{c}_N^T)$	Partición del vector c vector de coeficientes de la Función objetivo correspondiente a las variables básicas y no básicas
Ω	Región convexa de \mathbb{R}^n
∇	Operador nabla
∇f	Gradiente de la función
$\ \nabla f(\mathbf{x}^{(t)})\ $	Norma del vector gradiente de f

Introducción

En las ciudades modernas el crecimiento de la población hace que se incremente incesantemente el flujo de personas, mercancías, etc. Este crecimiento genera problemas de congestión que trae consigo pérdidas económicas, afectando potencialmente al ser humano y al medio ambiente, en consecuencia el crecimiento de la demanda y de la oferta del sistema de transporte es más rápido que la infraestructura urbana [13].

Podemos ver como en los últimos años el incremento del tráfico urbano está afectando directamente en la calidad de vida del ciudadano. Hay una mayor preferencia por los vehículos privados frente al transporte público, aumentando la cantidad de vehículos por familia y el número de desplazamientos realizados.

Debido a la complejidad del sistema de Transporte es necesario contar con herramientas analíticas y tecnológicas que permitan a los organismos encargados de la planificación y de la toma de decisiones disponer de una adecuada metodología acorde con el avance tecnológico del Sistema de Transporte para poder hacer simulaciones y plantear alternativas de solución.

Para la representación conceptual y simbólica de la realidad es importante desarrollar instrumentos, conceptos o herramientas necesarias que permitan lograr tales objetivos. Para lograr una representación abstracta del carácter y funcionamiento del sistema de transporte, se desarrollan los modelos de simulación en computadora que combinan teoría, análisis de datos y algoritmos [13], estas herramientas permiten representar idealmente el sistema, estos modelos de simulación deben ser calibrados o ajustados con herramientas apropiadas de modo que representen con mayor exactitud posible el evento deseado. Luego el modelo puede usarse para hacer predicciones y correcciones acerca del comportamiento futuro del sistema.

El uso de los modelos de transporte forman parte de una herramienta importante para la planificación y zonificación de una ciudad [13]. La simulación urbana es también un problema de modelaje relativamente único. Los sistemas urbanos comúnmente representados en los modelos urbanos son notablemente complejos y difíciles de modelar. Los modelos de uso de suelo y los modelos de transporte se usan a menudo para dar soporte a las decisiones que influyen en el desarrollo de la ciudad y por ende en la vida de las personas.

No obstante el ambiente de la simulación es ahora apropiado para la difusión de nuevas ideas en la modelación urbana.

La tesis consta de los siguientes capítulos:

En el capítulo 1 se describe el marco metodológico en la cual se desarrolla el tipo

de trabajo que se presenta y la problemática de la realidad en el transporte.

En el capítulo 2. se describe el marco teórico correspondiente a los diversos modelos y métodos.

En el capítulo 3 se describe el marco filosófico, tecnológico y los conceptos de modelos de optimización, teoría de asignación de tráfico.

En el capítulo 4 se estructura e implementa la Formulación del problema de asignación y el algoritmo de Frank Wolfe

En el capítulo 5 se construye el modelo computacional para asignación de tráfico, se realiza una aplicación, utilizando el modelo propuesto

En el capítulo 6 se describen las principales conclusiones y recomendaciones.

En el apéndice se muestran los códigos en matlab de los programas.

1

Marco Metodológico

1.1. Tipo de Investigación

La presente investigación se desarrolla aspectos teóricos referentes a la optimización no lineal de redes de transporte, propone desarrollar el algoritmo de Franck-Wolfe para la optimización del proceso de asignación de viajes a una red de transporte, para una mejor programación del algoritmo se incluye en el el método de Simplex Revisado y para lograr la convergencia de la raíz del método se propone combinar el mismo con el método de la bisección por ser este un método de convergencia segura para funciones continuas si la raíz se encuentra en el intervalo de búsqueda la implementación numérica del algoritmo será elaborado en Matlab, así mismo se compararán ejemplos de método de Franck-Wolfe con el de multiplicadores de Lagrange, cabe destacar que el algoritmo de Franck-Wolfe es óptimo en procesos de varios datos por su relativamente fácil implementación numérica.

1.2. Realidad y problemática

Los problemas asociados con el transporte en las ciudades desarrolladas y en las ciudades en desarrollo se incrementan incesantemente, uno de estos problemas es la congestión, que ocasiona un elevado tiempo de desplazamiento, estos problemas traen consigo el elevado costo en horas hombres desperdiciados, trayendo con sigo el desmedro económico de sus habitantes. El crecimiento económico hace que la demanda de medios de transporte se incremente hasta sobrepasar la capacidad de los sistemas de transporte. La utilización de las herramientas modernas como los conceptos, equipos y software existentes en la actualidad permiten elaborar mejores herramientas para aminorar estos problemas.

2

Marco Teórico

2.1. Descripción de Conceptos Previos

En la actualidad el avance en las aplicación de la tecnología en las diversas áreas de la ingeniería permiten resolver problemas muy complejos, se presentan algunos conceptos y herramientas que permiten abordar el desarrollo del proyecto.

La modelación en el Sistema de Transporte está orientada generalmente a determinar la cantidad de viajes generados o atraídos por las diferentes zonas del área de estudio, establecer la distribución de estos viajes entre las distintas zonas, repartir los viajes entre los diferentes modos de transporte disponibles en el área, y finalmente los flujos de cada tramo de la red, es en este último tramo de la modelación del Sistema de Transporte que se orienta la presente investigación

El problema de asignación de tráfico comenzó a estudiarse en la década de los 50, del siglo pasado, hoy en día se van mejorando los algoritmos y modelos existentes. El Estado-del-Arte en la resolución del Problema de Asignación de Tráfico Simétrico lo constituye el algoritmo de Bar-Gera y Boyce desarrollado en su tesis doctoral [2]. Este algoritmo presenta una velocidad de convergencia superlineal, pero únicamente es aplicable al modelo separable que se formula como un problema de optimización.

2.2. Red de Transporte

El término genérico “red” hace referencia a un conjunto de entidades (objetos, personas, etc.) conectadas entre sí. Por lo tanto, una red permite que circulen elementos materiales o inmateriales entre estas entidades, según reglas bien definidas. así una red de transporte: es conjunto de infraestructuras y vehículos usados para transportar personas y bienes entre diferentes áreas geográficas

El problema de asignación de tráfico es una aplicación de digrafos ponderados al flujo (circulación) de un bien, desde una fuente (llamado origen), a un destino dado. Los bienes pueden ser por ejemplo litros de agua que fluyen por tuberías, llamadas telefónicas a través de un sistema de comunicación, vehículos, etc.

Definición 2.2.1. Un grafo G se define como un par $G = (V, A)$ formado por un conjunto finito V cuyos elementos son denominados vértices o nodos y A es un subconjunto de pares no ordenados de vértices y que reciben el nombre de aristas o arcos (no dirigidas o no orientadas). es decir

$$A \subset \{\{u, v\} / u, v \in V \wedge u \neq v\}$$

Si el conjunto de vértices es $V = \{v_1, v_2, \dots, v_n\}$, entonces los elementos de A se representa de la forma $\{v_i, v_j\}$, donde $i \neq j$.

Definición 2.2.2. Dos vértices v_i y v_j se dicen adyacentes o vecinos si existe una arista que los contiene, esto es si $\{v_i, v_j\} \in A$.

Un grafo $G = (V, A)$ se dice finito si V es un conjunto finito.

Definición 2.2.3. El grado de un vértice v es el número de aristas que lo contienen y este número se nota por $\text{deg}(v)$.

Definición 2.2.4. Sea $G = (V, A)$ un grafo. Un camino en el grafo G es una sucesión en la que aparecen de forma alternativa elementos de V y de A de la forma $v_0, e_1, v_1, e_2, v_2, \dots, v_n$ con $v_i \in V$ para $i = 0, 1, \dots, n$, y $e_j \in A$ para $j = 1, \dots, n$. y donde $v_{i.1}$ es adyacente con v_i mediante la arista e_i . El número de arcos que componen un camino se denomina longitud de dicho camino.

Definición 2.2.5. Un grafo es conexo si cada par de vértices se pueden unir por un camino, es decir, si para cualquier par de vértices (v_i, v_j) , existe al menos un camino posible desde v_i hacia v_j . Si no existe tal camino se dice que es desconexo

Definición 2.2.6. Sea $G = (V, E)$ un grafo, se dice que contiene lazos o bucle, si una de sus aristas tiene como punto inicial y final al mismo vértice.

Definición 2.2.7. Un digrafo es un grafo $G = (V, A)$ donde V es un conjunto finito y $A \subset (V \times V) - \Delta$, siendo $\Delta = \{(x, x) : x \in V\}$. A los elementos de V se les denomina vértices y a los de A aristas (dirigidas u orientadas)

Con gran frecuencia se desea modelar problemas prácticos utilizando grafos en los que se asocia a las aristas un entero no negativo llamado peso o costo, dichos números se asocian con información como la cantidad de material que puede embarcarse de un vértice a otro a lo largo de una arista la cual puede representar una carretera o ruta aérea, o hallar la forma de conectar todos los vértices al menor costo, en el caso de un circuito eléctrico por ejemplo. Estas técnicas surgen en el área de las matemáticas llamadas Investigación de Operaciones.

Nota 1. el peso de la arista será interpretado como la capacidad máxima que puede transportar dicha arista.

Definición 2.2.8. Sea $G(A, V)$ un digrafo conexo y sin lazos. Se dice que G es una Red de Transporte (arcos) si se verifican:

a) \exists un único vértice $f \in V / gr^+(f) = 0$ al cual no llegan flechas, se le llama fuente.

- b) \exists un único vértice $s \in V / gr^-(s) = 0$ del cual no salen flechas, se le llama sumidero.
- c) \exists una función $c : A \rightarrow N_0 /$ si $a = (v_i, v_j) \in A \Rightarrow c(a) = c_{ij}$ representa la capacidad de arista(arco).

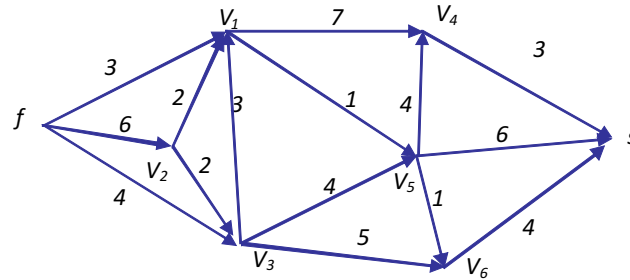


Figura 2.1: Digrafo conexo

Definición 2.2.9. Si $G(A, V)$ es una red de arcos. Se llama flujo de G a una función $F : A \rightarrow N_0 /$ (a cada arista (arco) se le asigna un número)

- a) $\forall a \in A$ se tiene $F(a) \leq c(a)$ (obs.: si $F(a) = c(a)$ se dice que la arista está Saturada)
- b) $\forall a \in A$ se tiene que el flujo de entrada es igual al flujo de salida

Nota 2. En la definición 2.2.9 se tiene que en:

- El inciso a) se indica que lo que se transporta por una arista no puede exceder la capacidad de la misma.
- El inciso b) se indica que lo que fluye (lo que llega) a un vértice distinto de los vértices fuente y sumidero debe ser igual a lo que fluye desde él (lo que sale).

Definición 2.2.10. El segmento que une dos puntos $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ es el conjunto

$$[\mathbf{a}, \mathbf{b}] = \{ \mathbf{x} \in \mathbb{R}^n / \mathbf{x} = \lambda \mathbf{a} + (1 - \lambda) \mathbf{b}, 0 \leq \lambda \leq 1 \}.$$

Definición 2.2.11. Se dice que un conjunto $\Omega \subset \mathbb{R}^n$ es convexo si $\forall \mathbf{x}_1, \mathbf{x}_2 \in \Omega$ se verifica que:

$$\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2 \in \Omega, \forall \lambda \in [0, 1].$$

Es decir, un conjunto es convexo si dados dos puntos cualesquiera del conjunto, el segmento que los une está contenido totalmente en el conjunto.

Definición 2.2.12. Dada la función f definida en una región convexa $\Omega \subset \mathbb{R}^n$, se dice que es convexa (estrictamente convexa) si, para cualquier par de puntos $\mathbf{x}_1, \mathbf{x}_2$ ($\mathbf{x}_1 \neq \mathbf{x}_2$) contenidos en Ω , y con $\alpha \in [0, 1]$ ($\alpha \in \langle 0, 1 \rangle$) si:

$$f(\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2) \leq \alpha f(\mathbf{x}_1) + (1 - \alpha) f(\mathbf{x}_2)$$

$$(f(\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2) < \alpha f(\mathbf{x}_1) + (1 - \alpha) f(\mathbf{x}_2)).$$

Definición 2.2.13. (*Punto extremo*) Sea S un conjunto convexo no vacío en \mathbb{R}^n . Un vector $\mathbf{x} \in S$ se denomina punto extremo de S si $\mathbf{x} = \lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2$ con $\mathbf{x}_1, \mathbf{x}_2 \in S$ y $\lambda \in \langle 0, 1 \rangle$ implica que $\mathbf{x} = \mathbf{x}_1 = \mathbf{x}_2$.

Teorema 2.2.1. (*teorema de representación de conjuntos convexos finitos*). Si un conjunto convexo está acotado y cerrado (o compacto), cualquiera de sus puntos puede escribirse como una combinación convexa de sus puntos extremos.

Definición 2.2.14. (*dirección*). Sea S un conjunto convexo, cerrado y no vacío en \mathbb{R}^n . Se dice que un vector unitario \mathbf{d} es una dirección de S , si para cada $\mathbf{x} \in S$, $\mathbf{x} + \pi \mathbf{d} \in S$ para todo $\pi \geq 0$.

Definición 2.2.15. (*dirección extrema*). Una dirección \mathbf{d} se denomina extrema si no puede expresarse como una combinación lineal positiva de dos direcciones distintas, esto es, si $\mathbf{d} = \pi_1 \mathbf{d}_1 + \pi_2 \mathbf{d}_2$ para $\pi_1, \pi_2 > 0$, entonces $\mathbf{d} = \mathbf{d}_1 = \mathbf{d}_2$.

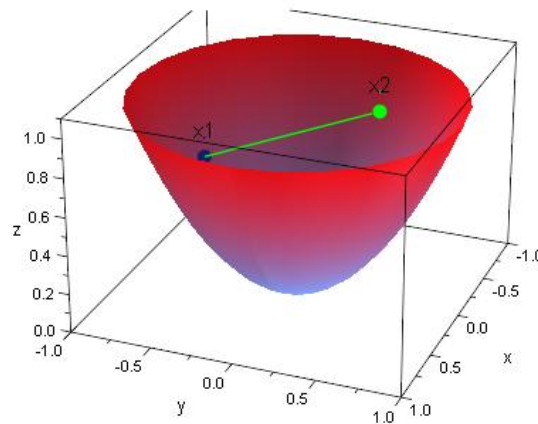


Figura 2.2: Función convexa

Definición 2.2.16. (*combinación lineal convexa*). Se dice que un vector \mathbf{x} es una combinación lineal convexa de los vectores de $A = \{\mathbf{a}_1, \dots, \mathbf{a}_k\}$ si y sólo si $\mathbf{x} = \lambda_1 \mathbf{a}_1 + \dots + \lambda_k \mathbf{a}_k$, donde $\lambda_i \in \mathbb{R}^n$ y $\sum_{i=1}^k \lambda_i = 1$.

El conjunto de todas las combinaciones lineales convexas de los vectores de A se denomina A_λ .

Definición 2.2.17. (*politopo*). Sea $A = \{\mathbf{a}_1, \dots, \mathbf{a}_k\}$. El conjunto $S = A_\lambda \in \mathbb{R}^n / x = \lambda_1 \mathbf{a}_1 + \dots + \lambda_k \mathbf{a}_k$ con $\lambda_i \geq 0$; $\sum_{i=1}^k \lambda_i = 1$ de todas las combinaciones lineales convexas de los vectores de A se denomina politopo o envoltura convexa generada por $\{\mathbf{a}_1, \dots, \mathbf{a}_k\}$. Si el conjunto de vectores $\{\mathbf{a}_1, \dots, \mathbf{a}_k\}$ es mínimo, se trata del conjunto de los k puntos extremos de S .

2.3. Modelo de 4 Etapas

Los planificadores y gestores de los sistemas de transporte buscan resolver estos problemas, intentando responder de forma adecuada a las necesidades de movilidad de la población. La planificación, gestión y control del tráfico emplea modelos matemáticos como herramienta analítica que auxilian la toma de decisiones. Estos modelos deben resolverse por medio de métodos computacionales y de forma eficiente. El proceso de planificación del transporte consta de las siguientes fases [13], [4]:

1. Recopilación de datos.
2. Análisis y ajuste de modelos.
 - a) Modelos de generación/atracción de tráfico.
 - b) Modelos de distribución zonal.
 - c) Modelos de distribución modal.
 - d) Modelos de asignación.
3. Previsiones de la demanda.
4. Evaluación de futuros escenarios

Este es un modelo clásico de simulación, el modelo de cuatro etapas en su proceso de aplicación se diferencia claramente la generación y producción de viajes, distribución de viajes, partición modal y la asignación de redes por modo. Como se indica en la figura (2.3):

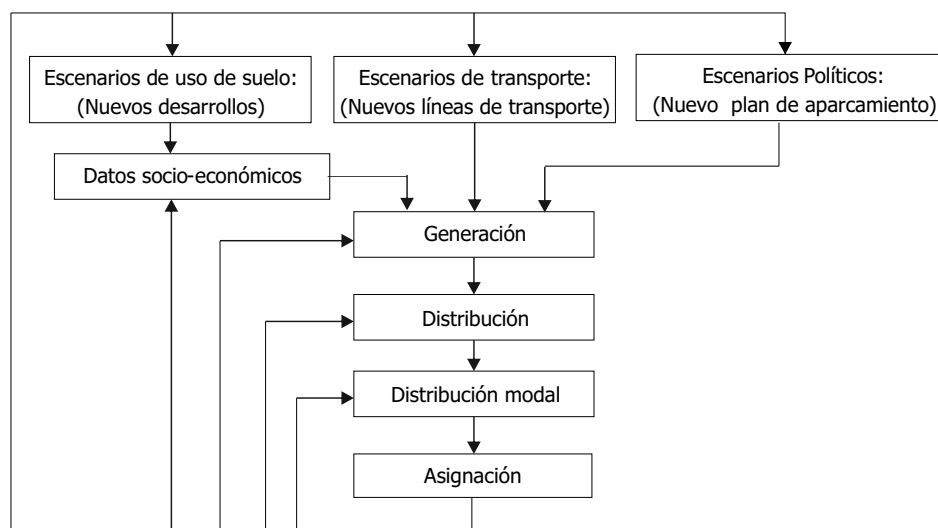


Figura 2.3: Modelo de 4 etapas

Existen dentro de los modelos indicados anteriormente modelos de optimización como el de Wardrop (modelo de asignación de viaje privado) que emplea una ecuación

variacional para obtener la típica función de costos ampliamente usada que es sugerida por Agencia de Carreteras Públicas (Bureau Public Road BPR) [3].

Uno de los modelos más ampliamente usado es el modelo de asignación de tráfico. El Problema de Asignación de Tráfico (TAP) modeliza el comportamiento de los usuarios en la elección de su ruta para satisfacer su viaje en redes de tráfico con congestión partiendo de la información de la configuración de la red y de la demanda (matriz origen-destino).

Los modelos que integran la fase de asignación en redes congestionadas modelizan un equilibrio Cournot-Nash, en la que los usuarios del sistema de transporte son los jugadores del juego de equilibrio, y en el que sus estrategias pueden consistir en la elección de la ruta, el modo de transporte, la realización o no del viaje, etc.

2.4. Modelos de Programación

La programación lineal es una técnica matemática relativamente reciente (siglo XX), que consiste en una serie de métodos y procedimientos que permiten resolver problemas de optimización en el ámbito, sobre todo, de las Ciencias Sociales.

La Programación es una técnica matemática utilizada para dar solución a problemas que se plantean muy comúnmente en diversas disciplinas como Economía, Ingeniería, Sociología, Biología, etc. En esencia trata de maximizar y/o minimizar una función de dos o más variables teniendo en cuenta que las mismas deben cumplir determinadas exigencias derivadas de la escasez de recursos disponibles en la realidad. El problema de asignar convenientemente recursos escasos es un problema conocido desde la antigüedad, especialmente en el mundo de la economía, aunque una solución matemática al mismo es relativamente reciente. Fue en la década de los años 40 del siglo XX que a través del trabajo de equipos formados por matemáticos, economistas y físicos, entre los cuales merece especial destaque George B. Dantzing, se sentaron las bases para la resolución de problemas de Programación Lineal y No Lineal [1].

2.4.1. Programación Lineal

La programación lineal (PL), que trata exclusivamente con funciones objetivos y restricciones lineales, es una parte de la programación matemática, y una de las áreas más importantes de la matemática aplicada. Se utiliza en campos como la ingeniería, la economía, la gestión, y muchas otras áreas de la ciencia, y la industria. Debemos notar que cualquier problema de programación lineal requiere identificar cuatro componentes básicos:

1. El conjunto de datos.
2. El conjunto de variables involucradas en el problema, junto con sus dominios respectivos de definición.

3. El conjunto de restricciones lineales del problema que definen el conjunto de soluciones admisibles.
4. La función lineal que debe ser optimizada (minimizada o maximizada).

Se ilustra de manera clara el alcance de la programación lineal, para familiarizarse con los cuatro elementos descritos.

El objeto de la programación lineal es optimizar (minimizar o maximizar) una función lineal de n variables sujeto a restricciones lineales de igualdades o desigualdades. Más formalmente, se dice que un problema de programación lineal consiste en encontrar el óptimo (máximo o mínimo) de una función lineal en un conjunto que puede expresarse como la intersección de un número finito de hiperplanos y semiespacios en \mathbb{R}^n . Considérense las siguientes definiciones.

Definición 2.4.1. (*problema de programación lineal*). La forma más general de un problema de programación lineal consiste en minimizar o maximizar la función f

$$z = f(\mathbf{x}) = \sum_{j=1}^n c_j x_j \quad (2.1)$$

Sujeto a:

$$\begin{aligned} \sum_{j=1}^n c_j x_j &= b_i, \text{ para } i = 1, \dots, p-1 \\ \sum_{j=1}^n c_j x_j &\leq b_i, \text{ para } i = p, p+1, \dots, q-1 \\ \sum_{j=1}^n c_j x_j &\geq b_i, \text{ para } i = q, q+1, \dots, m \end{aligned} \quad (2.2)$$

Donde p, q, m son enteros positivos tales que $1 \leq p \leq q \leq m$

Lo que distingue un problema de programación lineal de cualquier otro problema de optimización es que todas las funciones que en él intervienen son lineales. Una única función no lineal hace que el problema no pueda clasificarse como problema de programación lineal. Ténganse en cuenta además que se considera que los problemas tienen siempre un número finito de restricciones. La función lineal f de la ecuación (2.1) se denomina *función objetivo* o *función de coste*, y es la función que ha de optimizarse. Obsérvese que en el sistema de ecuaciones e inecuaciones (2.2) se presentan todas las posibles alternativas en lo que se refiere a los operadores que relacionan los dos términos de las *restricciones* (lineales), dependiendo de los valores p y q . Como casos especiales, el problema puede tener exclusivamente restricciones de igualdad, de desigualdad de un tipo, de desigualdad del otro tipo, desigualdades de ambos tipos, igualdades y desigualdades, etc., es decir el tipo de restricciones dependerá del problema que se plantea resolver.

Definición 2.4.2. (solución factible). Un punto $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ que satisface todas las restricciones (2.2) se denomina solución factible. El conjunto de todas esas soluciones es la región de factibilidad.

Definición 2.4.3. (solución óptima). Un punto factible $\tilde{\mathbf{x}}$ tal que $f(\mathbf{x}) \geq f(\tilde{\mathbf{x}})$ para cualquier otro punto factible \mathbf{x} se denomina una solución óptima del problema.

El objetivo de los problemas de optimización es encontrar un óptimo global. Sin embargo, las condiciones de optimalidad sólo garantizan, en general, óptimos locales, si éstos existen. Sin embargo, los problemas lineales presentan propiedades que hacen posible garantizar el óptimo global

- Si la región factible está acotada, el problema siempre tiene una solución (ésta es una condición suficiente pero no necesaria para que exista una solución).
- El óptimo de un problema de programación lineal es siempre un óptimo global.
- Si, \mathbf{x} , \mathbf{y} son soluciones óptimas de un problema de programación lineal, entonces cualquier combinación (lineal) convexa de los mismos también es una solución óptima. Obsérvese que las combinaciones convexas de puntos con el mismo valor de la función de coste presentan el mismo valor de la función de coste.
- La solución óptima se alcanza siempre, al menos, en un punto extremo de la región factible.

Teorema 2.4.1. (propiedad fundamental de la programación lineal). Si un problema de programación lineal tiene una solución óptima, es además una solución básica factible.

Problema de programación lineal en forma estándar

Un Problema de Programación Lineal puede plantearse de diversas formas, entonces para unificar su análisis, es conveniente transformarlo en lo que normalmente se llama forma estándar. A veces, esta transformación ha de realizarse antes de resolver el problema de programación lineal y determinar el óptimo. Para describir un problema de programación lineal en forma estándar son necesarios los tres elementos siguientes:

1. . Un vector $\mathbf{c} \in \mathbb{R}^n$
2. . Un vector no negativo $\mathbf{b} \in \mathbb{R}^m$
3. . Una matriz A de tamaño $m \times n$

Con estos elementos, el problema lineal asociado y en forma estándar tiene la siguiente forma. Minimizar

$$\text{mín } f(\mathbf{x}) = \mathbf{c}^T \mathbf{x} \quad (2.3)$$

$$\text{s.a. } A\mathbf{x} = \mathbf{b} \quad (2.4)$$

$$\mathbf{x} \geq \mathbf{0}$$

donde: $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{c}^T \mathbf{x}$ indica producto escalar de los vectores \mathbf{c} y \mathbf{x} , $A\mathbf{x}$ es el producto de la matriz A y el vector \mathbf{x} , y $\mathbf{x} \geq \mathbf{0}$ hace que todas las componentes de los vectores factibles sean no negativas. Los problemas de programación lineal se estudian normalmente en esta forma. Típicamente, n es mucho mayor que m . En resumen, un problema de programación lineal se dice que está en forma estándar si y sólo si

1. Es de minimización.
2. Sólo incluye restricciones de igualdad.
3. El vector b es no negativo.
4. Las variables x son no negativas.

Antes de analizar un problema de programación lineal dada en su forma estándar, es conveniente mostrar que cualquier problema expresado en la forma (2.1)-(2.2) también puede expresarse en forma estándar.

Transformación a la forma estándar

Cualquier problema de programación lineal puede expresarse siempre en forma estándar sin más que llevar a cabo una serie de manipulaciones algebraicas:

1. Las variables no restringidas en signo se pueden expresar como diferencias de variables que sí están restringidas en signo, es decir variables no negativas. Si algunas (o todas) de las variables no están restringidas en signo, éstas se pueden expresar mediante sus partes positiva y negativa. Las partes positiva y negativa de la variable x_i , se definen como $x_i^+ = \max\{0, x_i\}$ y $x_i^- = \max\{0, -x_i\}$, respectivamente. Se puede comprobar fácilmente que $x = x_i^+ - x_i^-$, $|x| = x_i^+ + x_i^-$ y que ambas x_i^+ y x_i^- son no negativas. Si el número de variables no restringidas en signo es r , el empleo de la regla anterior supone la necesidad de utilizar r variables adicionales.
2. Las restricciones de desigualdad pueden convertirse en restricciones equivalentes de igualdad introduciendo nuevas variables que se denominan variables de holgura:

- Si

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i$$

entonces, existe una variable $x_{n+1} \geq 0$ tal que

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n + x_{n+1} = b_i$$

- Si

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \geq b_i$$

entonces, existe una variable $x_{n+1} \geq 0$ tal que

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n - x_{n+1} = b_i$$

3. Un problema de maximización es equivalente a uno de minimización sin más que cambiar el signo de la función objetivo. En particular, maximizar

$$Z_{max} = \mathbf{c}^T \mathbf{x}$$

es equivalente a minimizar

$$Z_{min} = -\mathbf{c}^T \mathbf{x}$$

si ambos problemas han de cumplir las mismas restricciones. Obsérvese que ambos problemas alcanzan el óptimo en los mismos puntos, pero

$$Z_{max} = Z_{min}.$$

4. Una restricción con término independiente \mathbf{b} no positivo puede reemplazarse por otra equivalente cuyo término independiente es no negativo.

Soluciones básicas

Considérese un problema de programación lineal en forma estándar matricial (2.3)-(2.4), donde se supondrá, sin pérdida de generalidad, que el rango de la matriz A de dimensión $m \times n$ es m (recuérdese que $m \leq n$), y que el sistema lineal $A\mathbf{x} = \mathbf{b}$ tiene solución. En cualquier otro caso, el problema lineal es equivalente a otro con menos restricciones, o no tiene solución factible, respectivamente.

Definición 2.4.4. (matriz básica). Una submatriz no singular B de dimensión $m \times m$ de A se denomina matriz básica o base. B también se denomina matriz básica factible si y solo si $B^{-1}\mathbf{b} \geq \mathbf{0}$.

Cada matriz básica tiene un vector asociado que se denomina solución básica. El procedimiento para calcular esta solución es el siguiente. Sea \mathbf{x}_B el vector de las variables asociadas a las columnas de A necesarias para construir B . Las variables \mathbf{x}_B se denominan variables básicas y el resto se denominan variables no básicas. Asignando el valor cero a las variables no básicas

$$(B \ N) \begin{pmatrix} \mathbf{x}_B \\ \mathbf{0} \end{pmatrix} = \mathbf{b}, \quad B\mathbf{x}_B = \mathbf{b}, \quad \mathbf{x}_B = B^{-1}\mathbf{b} \quad (2.5)$$

donde N es tal que $A = (B \ N)$. Por tanto $B^{-1}\mathbf{b}$ nos permite obtener la solución básica asociada a B . Si B es una matriz básica factible, su solución básica se dice que es factible. El número de soluciones básicas factibles de un problema de programación lineal acotado con un número finito de restricciones es siempre finito, y cada una se corresponde con un punto extremo de la región de factibilidad. En concreto, el teorema siguiente establece la relación entre soluciones básicas factibles y puntos extremos.

Teorema 2.4.2. (caracterización de puntos extremos). Sea $S = \{\mathbf{x} : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$, donde A es una matriz $m \times n$ de rango m , y \mathbf{b} es un vector de dimensión $m \times 1$. Un punto \mathbf{x} es punto extremo de S si y sólo si A puede descomponerse en (B, N) tal que

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} B^{-1}\mathbf{b} \\ \mathbf{0} \end{pmatrix} \quad (2.6)$$

donde B es una matriz de dimensión $m \times m$ invertible que satisface $B^{-1}\mathbf{b} \geq 0$.

Una demostración de este teorema puede encontrarse en el texto de Bazaraa[1] El teorema siguiente muestra por qué las soluciones básicas factibles son importantes.

Teorema 2.4.3. (*propiedad fundamental de la programación lineal*). Si un problema de programación lineal tiene una solución óptima, es además una solución básica factible.

Demostración. Una región poliédrica siempre puede escribirse como

$$\mathbf{x} = \sum_i \rho_i v_i + \sum_j \beta_j w_j + \sum_k \lambda_k q_k$$

donde $\rho_i \in \mathbb{R}$, $\beta_j \in \mathbb{R}^+$, y $0 \leq \lambda_k \leq 1$; $\sum_j \lambda_k = 1$. La región factible de un problema de programación lineal en forma estándar no contiene un espacio vectorial debido a las restricciones $\mathbf{x} \geq \mathbf{0}$. En este caso especial, el conjunto poliédrico puede escribirse como suma de un politopo y un cono; siendo el conjunto mínimo de generadores del politopo el conjunto de puntos extremos, y el mínimo conjunto de generadores del cono el conjunto de direcciones extremas. Por tanto, el valor de la función objetivo a minimizar $Z = \mathbf{c}^T \mathbf{x}$ se puede expresar como

$$Z = \mathbf{c}^T \mathbf{x} = \sum_j \beta_j \mathbf{c}^T w_j + \sum_k \lambda_k \mathbf{c}^T q_k$$

□

Definición 2.4.5. *politopo es una región finita del espacio n -dimensional encerrado por un número finito de hiperplanos, es la generalización del concepto polígono en (2D) y poliedro en 3D)*

2.4.2. Programación no Lineal

Un modelo de Programación no Lineal es aquel donde las variables de decisión se expresan como funciones no lineales ya sea en la función objetivo y/o restricciones de un modelo de optimización. Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$ una función continuamente diferenciable (es decir sus derivadas parciales existen y son continuas) y $\Omega \subseteq \mathbb{R}^n$ un conjunto convexo. El problema de optimización consiste en

$$\begin{array}{ll} \text{mín} & f(\mathbf{x}) \\ \text{s.a.} & \mathbf{x} \in \Omega \end{array}$$

Se dice que el problema no es lineal si f o las funciones que se definen en el conjunto Ω son no lineales. Una solución óptima de este problema se dice que es mínimo global de f en Ω . En la práctica la determinación de tal punto $\bar{\mathbf{x}} \in \Omega$ es difícil, por lo que en la mayoría de los algoritmos de optimización no lineal procuran obtener un mínimo local de f , es decir un mínimo $\bar{\mathbf{x}}$ de f en una vecindad de $\bar{\mathbf{x}}$. El concepto de punto estacionario de f en Ω está asociado a la determinación de un mínimo local o global de f en Ω . Se dice que $\bar{\mathbf{x}}$ es un punto estacionario de f en Ω si

$$\nabla f(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}}) \geq 0 \quad \text{para todo } \mathbf{x} \in \Omega \quad (2.7)$$

esto es, si $\bar{\mathbf{x}}$ es mínimo global del problema de optimización

$$\begin{aligned} \text{mín} \quad & \nabla f(\bar{\mathbf{x}})^T \mathbf{x} \\ \text{s.a.} \quad & \mathbf{x} \in \Omega \end{aligned}$$

Donde, $\nabla f(\mathbf{x})$ representa el gradiente de f y es definido como $\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x_i} \right) \in \mathbb{R}^n$

Para la ilustración de este concepto considere el siguiente problema de programación lineal

$$\begin{aligned} \text{mín} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.a.} \quad & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

entonces

$$\Omega = \mathbf{x} \in \mathbb{R}^n : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}$$

es el conjunto convexo y $\bar{\mathbf{x}}$ es el punto estacionario de f en Ω si y solo si

$$\mathbf{c}^T \bar{\mathbf{x}} \leq \mathbf{c}^T \mathbf{x}, \quad \forall \mathbf{x} \in \Omega$$

Así mismo para programas lineales, $\bar{\mathbf{x}}$ es punto estacionario de f en Ω si y solo si $\bar{\mathbf{x}}$ es mínimo global de f en Ω .

Para problemas de optimización no lineales, el resultado no es en general válido, más es posible establecer algunas propiedades que ilustran la importancia de los conceptos de punto estacionario en optimización no lineal diferenciable. Para establecer necesitamos los siguientes conceptos

Teorema 2.4.4. *Si f es continuamente diferenciable en \mathbb{R}^n , entonces para cualquier $\mathbf{x}, \bar{\mathbf{x}} \in \mathbb{R}^n$, existe por lo menos un $\tilde{\mathbf{x}} \in \langle \bar{\mathbf{x}}, \mathbf{x} \rangle$ tal que*

$$f(\mathbf{x}) = f(\bar{\mathbf{x}}) + \nabla f(\tilde{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}})$$

Teorema 2.4.5. *Una función f continuamente diferenciable es convexa en el conjunto convexo Ω (esto es $f(\lambda \mathbf{x}^1 + (1 - \lambda)\mathbf{x}^2) \leq \lambda f(\mathbf{x}^1) + (1 - \lambda)f(\mathbf{x}^2)$ para todo $\lambda \in [0, 1], \mathbf{x}^1, \mathbf{x}^2 \in \Omega$), si y solo si para cualquier $\mathbf{x}, \bar{\mathbf{x}} \in \Omega$,*

$$f(\mathbf{x}) \geq f(\bar{\mathbf{x}}) + \nabla f(\bar{\mathbf{x}})^T(\mathbf{x} - \bar{\mathbf{x}})$$

Como consecuencia de estos dos teoremas es posible probar las siguientes propiedades.

Teorema 2.4.6. *Si $\bar{\mathbf{x}}$ es mínimo local o global de f en Ω , entonces $\bar{\mathbf{x}}$ es punto estacionario de f en Ω*

Teorema 2.4.7. *Si f es convexa en Ω , entonces $\bar{\mathbf{x}}$ es punto estacionario de f en Ω si y solo si $\bar{\mathbf{x}}$ es mínimo global de f en Ω*

2.5. Método Simplex

Existen muchas variantes del método simplex, aunque todas ellas se basan en la misma idea central. En este ítem se describe una de tales versiones. El método simplex se aplica a un problema de programación lineal en el formato estándar siguiente (ver [1]).

$$\text{mín} \quad f(\mathbf{x}) = \mathbf{c}^T \mathbf{x} \quad (2.8)$$

$$\text{s.a.} \quad A\mathbf{x} = \mathbf{b}; \mathbf{b} \geq 0 \quad (2.9)$$

$$\mathbf{x} \geq 0 \quad (2.10)$$

donde: $\mathbf{c} = (c_1, c_2, \dots, c_n)^T$ es la matriz columna de los coeficientes de la función objetivo, $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ es el vector columna de las variables iniciales, y A es una matriz de tamaño $m \times n$ que contiene los coeficientes de las restricciones. Los teoremas (2.4.2) y (2.4.3), respectivamente, aseguran que la solución óptima de un problema de programación lineal, si existe, se alcanza en una solución básica factible (un punto extremo). El Método Simplex genera una sucesión ordenada de soluciones básicas factibles que progresivamente mejora el valor de la función objetivo. El Método Simplex opera en dos fases:

1. Una etapa de iniciación en la que
 - a) El conjunto inicial de restricciones de igualdad se transforma en otro equivalente del mismo tipo (restricciones de igualdad), asociado a una solución básica.
 - b) Los valores de las variables básicas se transforman en valores no negativos (se obtiene así una solución básica factible). Este proceso se llamará fase de regulación.
2. Una etapa iterativa estándar, en la que los coeficientes de la función objetivo se transforman en valores no negativos y el valor del coste se mejora progresivamente hasta que se obtiene la solución óptima, no se detecta ninguna solución factible, o aparecen soluciones no acotadas. En este proceso iterativo, se calculan distintas soluciones básicas factibles. Para este fin se usa la operación elemental de la pivotación.

Etapa de iniciación

Una peculiaridad esencial del Método Simplex consiste en que incorpora una nueva variable Z , idéntica a la función objetivo del problema, y la restricción asociada

$$Z = c_1x_1 + \dots + c_nx_n \quad (2.11)$$

Usando la notación típica en el Método Simplex, las ecuaciones para las restricciones se expresan como

$$(B \ N) \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \mathbf{b} \quad (2.12)$$

y la función objetivo como

$$Z = (\mathbf{c}_B^T \ \mathbf{c}_N^T) \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} \quad (2.13)$$

o

$$Z = (0 \ \mathbf{c}_B^T \ \mathbf{c}_N^T) \begin{pmatrix} 1 \\ \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} \quad (2.14)$$

donde $(B \ N)$ es una partición de la matriz A y \mathbf{x}_B y \mathbf{x}_N definen otra partición de \mathbf{x} , en las variables básicas y no básicas, respectivamente. usando la ecuación (2.12) se llega a:

$$B\mathbf{x}_B + N\mathbf{x}_N = \mathbf{b} \quad (2.15)$$

$$\mathbf{x}_B = B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N = \mathbf{v} + U\mathbf{x}_N \quad (2.16)$$

donde

$$\mathbf{v} = B^{-1}\mathbf{b} \quad (2.17)$$

$$U = -B^{-1}N \quad (2.18)$$

Por otro lado, de las ecuaciones (2.13) y (2.16), se tiene

$$Z = \mathbf{c}_B^T(\mathbf{v} + U\mathbf{x}_N) + \mathbf{c}_N^T\mathbf{x}_N \quad (2.19)$$

$$Z = \mathbf{c}_B^T\mathbf{v} + \mathbf{c}_B^T(U + \mathbf{c}_N^T)\mathbf{x}_N \quad (2.20)$$

$$Z = w_0 + \mathbf{w}\mathbf{x}_N \quad (2.21)$$

donde

$$w_0 = \mathbf{c}_B^T\mathbf{v} \quad (2.22)$$

$$\mathbf{w} = \mathbf{c}_B^T U + \mathbf{c}_N^T \quad (2.23)$$

de las ecuaciones (2.16) y (2.21)

$$\begin{cases} Z = u_0 + \mathbf{w}\mathbf{x}_N \\ \mathbf{x}_B = \mathbf{v} + U\mathbf{x}_N \end{cases} \quad (2.24)$$

finalmente

$$\begin{pmatrix} Z \\ \mathbf{x}_B \end{pmatrix} = \begin{pmatrix} u_0 & \mathbf{w} \\ \mathbf{v} & U \end{pmatrix} \begin{pmatrix} 1 \\ \mathbf{x}_N \end{pmatrix} \quad (2.25)$$

El Método Simplex comienza con un conjunto de restricciones mostradas en (2.9) y (2.11) escritas como

$$\begin{pmatrix} Z \\ \text{---} \\ \mathbf{x}_B \end{pmatrix} = \begin{pmatrix} u_0^{(0)} & | & \mathbf{w}^{(0)T} \\ - & + & - \\ \mathbf{x}^{(0)} & | & U^{(0)} \end{pmatrix} \begin{pmatrix} 1 \\ \text{---} \\ \mathbf{x}_N \end{pmatrix} = Z^{(0)} \begin{pmatrix} 1 \\ \text{---} \\ \mathbf{x}_N \end{pmatrix} \quad (2.26)$$

donde $\mathbf{x}_B \cup \mathbf{x}_N$ es una partición del conjunto de variables x_1, x_2, \dots, x_n , las matrices $\mathbf{v}^{(0)}$ y $U^{(0)}$ se obtienen resolviendo el sistema (2.9) para \mathbf{x}_B , y teniendo en cuenta las ecuaciones (2.17) y (2.18). Sustituyendo $\mathbf{v}^{(0)}$ y $U^{(0)}$ en (2.8) se obtiene:

$$u_0^0 = \mathbf{c}_B^T \mathbf{v}^{(0)}, \quad \mathbf{w}^{(0)} = \mathbf{c}_B^T U^{(0)} + \mathbf{c}_N^T \quad (2.27)$$

donde \mathbf{c}_B^T y \mathbf{c}_N^T son los coeficientes de la función objetivo asociados a \mathbf{x}_B y \mathbf{x}_N , respectivamente. Ahora, el Método Simplex sustituye el sistema (2.26), en cada iteración, por un nuevo conjunto equivalente de restricciones con la misma estructura

$$\begin{pmatrix} Z^{(t)} \\ \text{---} \\ \mathbf{x}_B^{(t)} \end{pmatrix} = \begin{pmatrix} u_0^{(t)} & | & \mathbf{w}^{(t)T} \\ - & + & - \\ \mathbf{v}^{(t)} & | & U^{(t)} \end{pmatrix} \begin{pmatrix} 1 \\ \text{---} \\ \mathbf{x}_N^{(t)} \end{pmatrix} = Z^{(t)} \begin{pmatrix} 1 \\ \text{---} \\ \mathbf{x}_N^{(t)} \end{pmatrix} \quad (2.28)$$

donde t indica el número de la iteración y $t = 0$ se refiere a la iteración inicial. La transformación más importante del método simplex es la operación elemental de pivotación

El Método Simplex llega a la solución del Problema de programación lineal mediante operaciones elementales, que inicialmente actúan sobre $Z^{(0)}$, para obtener $Z^{(1)}$ y sucesivamente $Z^{(2)}, \dots, Z^{(t)}$, hasta que se alcanza la solución o se concluye que ésta no existe. La idea consiste en sustituir una variable básica por una no básica, y el fundamento teórico de este cambio se da en el siguiente teorema.

Teorema 2.5.1. *(propiedad de invarianza de la transformación de pivotación). Sea $z_{\alpha\beta}^{(t)}$ un elemento no nulo de $Z^{(t)}$, denominado pivote. Si se transforma el sistema de restricciones $y_B^{(t)} = Z^{(t)} y_N^{(t)}$ en $y_B^{(t+1)} = Z^{(t+1)} y_N^{(t+1)}$, mediante*

$$z_{ij}^{(t+1)} = \begin{cases} \frac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si } i \neq \alpha, j = \beta \\ z_{ij}^{(t)} - \frac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} & \text{si } i \neq \alpha, j \neq \beta \\ \frac{1}{z_{\alpha\beta}^{(t)}} & \text{si } i = \alpha, j = \beta \\ -\frac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si } i = \alpha, j \neq \beta \end{cases} \quad (2.29)$$

entonces ambos sistemas de restricciones son equivalentes, esto es, tienen las mismas soluciones (mismo conjunto factible).

Identificación de una solución óptima

Cuando el problema de programación lineal se expresa en la forma (2.28) se puede saber fácilmente si se ha encontrado una solución óptima. El siguiente resultado responde a esta cuestión.

Teorema 2.5.2. *(se ha encontrado una solución). Si*

$$\mathbf{w}^{(t)} \geq \mathbf{0}, \mathbf{v}^{(t)} \geq \mathbf{0} \quad (2.30)$$

entonces, una solución óptima del PPL es

$$Z^{(t)} = u_0^{(t)}; \mathbf{x}_B^* = \mathbf{v}^{(t)}, \mathbf{x}_N = \mathbf{0} \quad (2.31)$$

2.5.1. Método Simplex Revisado

Toda la argumentación precedente nos permite construir nuestra versión particular del algoritmo simplex revisado, que trabaja con la matriz $U(t)$, hasta obtener la solución óptima o detectar la falta de puntos factibles o la no acotación (ver [1]).

Algoritmo 2.5.1. *(el algoritmo simplex)*

- **Entrada.** *La tabla inicial, $\{U^{(0)}, v^{(0)}, w^{(0)}, u^{(0)}\}$ del PPL, conteniendo la función objetivo, y las restricciones del problema.*
- **Salida.** *La solución del problema de minimización, o un mensaje advirtiendo sobre la no admisibilidad o la falta de acotación.*

Iteración reguladora

- **Paso 1 (iniciación).** *Si $v^{(0)} \geq 0$, ir al paso 4. En otro caso, transformar $v^{(0)}$ para que todos sus elementos sean no negativos. Con este fin, comprobar si existe una columna β en $U^{(0)}$ que verifique una de las dos condiciones*

$$\begin{cases} u_{i\beta}^{(0)} > 0 & \text{si } v_i^{(0)} < 0 \\ u_{i\beta}^{(0)} \geq 0 & \text{si } v_i^{(0)} \geq 0 \end{cases} \quad \forall i \quad (2.32)$$

Si la respuesta es afirmativa, continuar en el paso 2. En otro caso, introducir la variable artificial $x_{n+1} \geq 0$, modificar el valor de la función objetivo, sumando el término Mx_{n+1} , donde M es una constante positiva grande, sumar el término x_{n+1} a todas las restricciones (variables básicas), y elegir $x_\beta = x_{n+1}$.

- **Paso 2 (búsqueda del pivote).** *Encontrar la fila del pivote α usando*

$$\frac{v_\alpha^0}{u_{\alpha\beta}^0} = \min \frac{v_i^0}{u_{i\beta}^0} c_B^T v^{(0)}, \quad w^{(0)} = \mathbf{c}_B^T U^{(0)} + \mathbf{c}_N^T \quad (2.33)$$

e ir al paso 3

Iteraciones estándar

- **paso 3 (Pivotación).** Realizar la transformación de pivotación

$$z_{ij}^{(t+1)} = \begin{cases} \frac{z_{i\beta}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si } i \neq \alpha, j = \beta \\ z_{ij}^{(t)} - \frac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} z_{i\beta}^{(t)} & \text{si } i \neq \alpha, j \neq \beta \\ \frac{1}{z_{\alpha\beta}^{(t)}} & \text{si } i = \alpha, j = \beta \\ -\frac{z_{\alpha j}^{(t)}}{z_{\alpha\beta}^{(t)}} & \text{si } i = \alpha, j \neq \beta \end{cases} \quad (2.34)$$

- **paso 4 (selección de la variable entrante x_β).** Si la condición de solución

$$\mathbf{w}^{(t)} \geq 0, \mathbf{v}^{(t)} \geq \mathbf{0} \quad (2.35)$$

es cierta y se verifica además la condición, $\mathbf{x}_{n+1} > \mathbf{0}$, el problema es no factible; si la condición de solución se cumple con \mathbf{x}_{n+1}

$$\mathbf{x}_B^* = \mathbf{v}^{(t)}; \mathbf{x}_N^* = \mathbf{0} \quad (2.36)$$

y el algoritmo concluye. En otro caso, se selecciona la variable entrante x_β , formando parte de \mathbf{x}_N , mediante

$$w_\beta^{(t)} = \min_{w_j^{(t)} < 0} w_j^{(t)} \quad (2.37)$$

- **Paso 5 (selección de la variable saliente).** Si todo $w_{i\beta} \geq 0$, el problema es no acotado. En otro caso, se selecciona la variable saliente x_α que deja \mathbf{x}_B utilizando

$$\frac{v_\alpha^{(t)}}{u_{\alpha\beta}^{(t)}} = \max_{u_{i\beta}^{(t)} < 0} \frac{v_i^{(t)}}{u_{i\beta}^{(t)}} \quad (2.38)$$

Volver al paso 3

2.6. Método del Gradiente

El método del gradiente conjugado es un caso particular del método de descenso para lo cual se considera la función

$$f(\bar{x}) = \frac{1}{2} \bar{x}^t A \bar{x} - \bar{b}^t \bar{x} + c$$

Donde A es una matriz cuadrada, \bar{x} y \bar{b} son vectores y c es una constante escalar [12], [1].

El método del gradiente es un método de descenso que comienza a iterar en un punto arbitrario x_0 y se continua siguiendo la línea de máximo descenso de la función

obteniéndose una sucesión de puntos $\bar{x}^1, \bar{x}^2, \dots$, hasta que se obtiene un punto lo suficientemente cercano a la solución \bar{x} . En cada iteración se elige la dirección para la cual f decrece más rápidamente, es decir en la dirección contraria a $\nabla f(\bar{x}^i)$. Así si, si se tiene la ecuación $\nabla f(\bar{x}) = A\bar{x} - \bar{b}$, entonces dirección buscada es $-\nabla f(\bar{x}^i) = \bar{b} - A\bar{x}^i$

Lema 2.6.1. *Sea $f : \mathbb{R}^n \rightarrow \mathbb{R}$ diferenciable en $\mathbf{x} \in \mathbb{R}^n$. Sea \mathbf{d} un vector de \mathbb{R}^n . Si $\nabla f(\mathbf{x})^T \mathbf{d} < 0$, entonces \mathbf{d} es una dirección de descenso de f en \mathbf{x} .*

El método del gradiente emplea como dirección de descenso en el punto $\mathbf{x}^{(t)}$ la dirección $\mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)})$, que, empleando el lema (2.6.1), se puede mostrar que es de descenso, en efecto. Si $\nabla f(\mathbf{x}^{(t)}) = \mathbf{0}$

se cumple la siguiente expresión

$$\nabla f(\mathbf{x}^{(t)})^T \mathbf{d}^{(t)} = -\nabla f(\mathbf{x}^{(t)})^T \nabla f(\mathbf{x}^{(t)}) = -(\|\nabla f(\mathbf{x}^{(t)})\|)^2 < 0 \quad (2.39)$$

Entonces, $\mathbf{d}^{(t)}$ es una dirección de descenso de f en $\mathbf{x}^{(t)}$.

Por tener velocidad lineal, la convergencia del método del gradiente es lenta, aún en el caso de funciones cuadráticas. Luego, se busca un método de convergencia más rápida. Una manera posible de acelerar esta convergencia es utilizar la noción de direcciones conjugadas u ortogonales [12].

2.7. Método del gradiente conjugado

El método de gradiente conjugado es un procedimiento iterativo originalmente propuesto por (Hestenes y Stiefel, 1950) [10] como un método directo para resolver sistemas lineales de la forma $A\mathbf{x} = \mathbf{b}$, donde $A \in \mathbb{R}^{n \times n}$ es una matriz simétrica y definida positiva y \mathbf{b} un vector en $\mathbb{R}^{n \times 1}$

Dado que en los modelos de optimización de Programación no Lineal las variables de decisión se expresan como funciones no lineales ya sea en la función objetivo y/o restricciones, esta característica particular de los modelos no lineales permite abordar problemas donde existen economías o deseconomías de escala (Las economías y deseconomías de escala existen cuando el costo unitario de producir un bien baja/sube a medida que aumenta/disminuye la tasa de producción) o en general donde los supuestos asociados a la proporcionalidad no se cumplen. En este sentido el método del gradiente (conocido también como método de Cauchy o del descenso más pronunciado) consiste en un algoritmo específico para la resolución de modelos de programación no lineal sin restricciones, perteneciente a la categoría de algoritmos generales de descenso, donde la búsqueda de un mínimo está asociado a la resolución secuencial de una serie de problemas unidimensionales.

El método de Gradiente Conjugado requiere trabajar con vectores para las cuales damos algunas definiciones.

Definición 2.7.1. *Se dice que $\{\mathbf{d}_k\}_{k=1}^n$ son vectores mutuamente conjugados respecto a una matriz G simétrica y positiva definida si:*

$$\mathbf{d}_k^T G \mathbf{d}_j = 0 \quad j \neq k \quad (2.40)$$

Definición 2.7.2. Dada una matriz G simétrica y definida positiva y sea el conjunto de vectores $\{v_1, v_2, \dots, v_n\}$ linealmente con $i = 1, 2, \dots, n$, se construye un conjunto de vectores $\{d_1, d_2, \dots, d_n\}$ mutuamente conjugados respecto a la matriz G , haciendo $d_1 = v_1$ y

$$d_{i+1} = v_{i+1} - \sum_{k=1}^i \frac{v_{i+1}^t G d_k}{d_k^t G d_k} d_k$$

Lema 2.7.1. Todo conjunto de vectores conjugados a una matriz G son linealmente independientes.

Lema 2.7.2. Si x y u son vectores cualesquiera en \mathbb{R}^n con $u \neq 0$ entonces

$$\operatorname{argmin}_{\alpha > 0} q(x + \alpha u) = \frac{r(x)^t u}{u^t A u}$$

donde $r(x) = -\nabla q(x) = b - Ax$.

Teorema 2.7.1. Sea $\{u_1, u_2, \dots, u_n\}$ un conjunto de A – conjugado de vectores no ceros. Si x_0 (arbitrario) es dado y

$$x_k = x_{k-1} + \alpha_k u_k \text{ para } 1 \leq k \leq n$$

donde α_k se escoge como el $\operatorname{argmin}_{\alpha > 0} q(x + \alpha u)$, entonces $Ax_n = b$

Teorema 2.7.2. Considerando las hipótesis del teorema (2.7.2). Para cada k se cumple que

$$q(x_k) = \min_{x \in x_0 + \exp\{u_1, \dots, u_k\}} q(x)$$

Teorema 2.7.3. El algoritmo de gradiente conjugado es un algoritmo de direcciones conjugadas.

Para denotar que conjunto de n vectores genera un espacio se usa la notación

$$\operatorname{Gen}\{v_1, v_2, \dots, v_n\}$$

Lema 2.7.3. Considerando la matriz A , las siguientes relaciones se cumplen en la k – sima iteración

1.

$$\operatorname{Gen}\{g_0, g_1, \dots, g_k\} = \operatorname{Gen}\{g_0, Ag_0, \dots, A^k g_0\} \quad (2.41)$$

2.

$$\operatorname{Gen}\{d_0, d_1, \dots, d_k\} = \operatorname{Gen}\{g_0, Ag_0, \dots, A^k g_0\} \quad (2.42)$$

3.

$$d_j A d_k = 0 \quad j = 0, \dots, k-1 \quad (2.43)$$

Nota 3. En el algoritmo, el cálculo de α se obtiene mediante la dirección de búsqueda del método, es decir se toma la dirección del gradiente negativo, la determinación de un valor apropiado para α en cada iteración es equivalente a resolver un problema de minimización en una dimensión

$$\min_{\alpha} f(x_k - \alpha \nabla f(x_k))$$

El método del gradiente conjugado debido a Fletcher y Reeves (1964) combina las características de la convergencia cuadrática del método de las direcciones conjugadas con las del método del gradiente. El método supone una importante mejora del método del gradiente con sólo un pequeño incremento en el esfuerzo de cálculo. El método del gradiente conjugado, esencialmente, combina la información obtenida del vector gradiente con la información acerca del vector gradiente de iteraciones previas. Lo que hace el método es calcular la nueva dirección de búsqueda utilizando una combinación lineal del gradiente en la etapa considerada y el de la etapa anterior. La principal ventaja del método es que necesita almacenar muy poca cantidad de información

Algoritmo 2.7.1. (algoritmo gradiente conjugado)

- **Paso 1** Dado $x_0 \in V_\delta(x^*)$ (punto inicial) y una tolerancia $rtol > 0$ calcular $f(x_0)$ y calcular $s_0 = -\nabla f(x_0)$
- **Paso 2** Para $k = 0, 1, \dots$ y calcular

$$x_{k+1} = x_k + \lambda_k s_k$$

con $\lambda \in \mathbb{R}$ que satisfice

$$F(x_k + \lambda_k s_k) \leq F(x_k + \lambda s_k), \forall \lambda \in \mathbb{R}$$

minimizando λ mediante una búsqueda unidireccional de la dirección s_k .

- **Paso 3** La dirección s_{k+1} de búsqueda es una combinación lineal de s^k y $\nabla f(x^{k+1})$:

$$s^{k+1} = -\nabla f(x^{k+1}) + \beta_k s^k$$

donde

$$\beta_k = \frac{\nabla^T f(x_{k+1}) \nabla f(x^1)}{\nabla^T f(x^0) \nabla f(x^0)}$$

para la etapa k -ésima la relación es

$$s^{k+1} = -\nabla f(x^{k+1}) + s^k \frac{\nabla^T f(x^{k+1}) \nabla f(x^{k+1})}{\nabla^T f(x^k) \nabla f(x^k)}$$

Para una función cuadrática se puede demostrar que dos direcciones de búsqueda son conjugadas. Después de n iteraciones conviene comenzar otra vez desde el principio tomando el último punto $k = n$ como nuevo punto de partida.

- **Paso 4** Realizar el test de convergencia, (la función objetivo ha disminuido), y terminar el algoritmo cuando s^k sea menor que alguna tolerancia preestablecida,

3

Marco Filosófico y Tecnológico

3.1. introducción

En el presente capítulo se describen los conceptos previos y necesarios para la elaboración del modelo, el problema de transporte es un problema antiguo y complicado de resolver. La mayor dificultad ocurre cuando se trata de modelar un problema que lleva en consideración las condiciones de saturación de la red, o cuando la función de costo de viaje sobre el arco depende, además del flujo del propio arco, de los flujos de otros arcos, de una vía con arcos de tránsito en dos sentidos, formación de filas en las intersecciones señalizadas, tiempos de espera en los paraderos o intersecciones, etc.

Los modelos de asignación de tráfico requieren de expresiones funcionales que proporcionen el tiempo/coste de viaje en cada arco en función del vector de flujo en los arcos de la red. Las funciones de coste dependen de un conjunto de parámetros. Algunos de ellos son estimados basándose en consideraciones físicas de los propios arcos, como su longitud, su capacidad o la velocidad media en el arco sin flujo; pero para otros existe un cierto grado de libertad.

La complejidad de los sistemas de transporte actuales ha mostrado la conveniencia de contar con herramientas analíticas que permitan al planificador disponer de un adecuado conocimiento del sistema y poder hacer predicciones. El uso de modelos matemáticos, ha sido de gran utilidad en la identificación de soluciones o políticas orientadas a lograr los objetivos propuestos para mejorar funcionamiento del sistema, constituyéndose en una gran ayuda para los procesos de planificación y de toma de decisiones, ya que el objetivo de los análisis, en la planificación del tráfico urbano, es la obtención de descripciones macroscópicas de los flujos de tráfico a partir del comportamiento de los conductores o usuarios. Tradicionalmente se ha considerado que el proceso de planificación del transporte consta de las fases siguientes

- *Recopilación de datos.* Comprende cualquier tipo de datos que explique como se comporta el sistema de transporte. Generalmente se incluye en este apartado:
 - a. Datos relativos a la infraestructura de transporte y su evolución
 - b. Datos relativos a las necesidades de transporte y a la utilización de la infraestructura.

- c. Factores tales como la utilización del terreno y distribución de la renta y tipo de empleo y estructura de las ciudades y conurbaciones (unión de ciudades debido a su crecimiento).
- *Análisis y ajuste de modelos.* Los modelos que se utilizan, y que son generados a partir de los datos de la fase anterior, habitualmente se aplican secuencialmente y se clasifican en:
 - *Fase de generación de viajes/atracción de viajes.* El esquema empieza considerando una zonificación del área de estudio, una codificación de la red de transporte y la obtención de una base de datos para el estudio. Estos datos están referidos al nivel de actividad económica y demográfica de cada zona que incluye el nivel de empleo, localización de centros comerciales, zonas recreativas, centros escolares, etc. y son empleados para estimar el número de viajes generados y atraídos por cada zona considerada en el estudio. Tras esta fase se obtiene una modelización de la red de transporte mediante un grafo $G = (\mathcal{N}, \mathcal{A})$ donde \mathcal{A} y \mathcal{N} son el conjunto de arcos (dirigidos) y nodos respectivamente. El significado de los arcos depende de si la red es de tráfico o de transporte público. En el primer caso los arcos están asociados a las calles y los nodos a las intersecciones. En el segundo caso cada nodo está asociado a una parada y cada arco representa los posibles desplazamientos entre paradas que un usuario puede efectuar. Hay arcos asociados a movimientos en el vehículo, andando o esperando.
 - *Fase de distribución zonal.* En esta fase se obtiene la distribución de los viajes sobre el espacio, esto es, se obtiene el número de viajes que se efectúan de una zona a otra, obteniéndose la denominada matriz de viajes origen-destino ($0 - D$). En esta fase se obtiene un conjunto de pares ordenados de $N \times N$ y la demanda de viajes (que inicialmente consideraremos fija). Denotamos este conjunto de pares de demandas por W y cada par $0 - D$ por $w = (i, j)$, donde i es el origen y j es el destino. Denotamos por \bar{g}_w la demanda total de viajes para el par w . [ítem *Fase de partición modal*. Cuando los usuarios eligen el modo de transporte con el que satisfacen sus necesidades de viaje. En esta fase se obtiene una matriz $0 - D$ para cada modo de transporte.
 - *Fase de asignación.* Finalmente, cada matriz de demanda $0 - D$ es asignada a un conjunto de rutas en la red de transporte. Usualmente se efectúa una asignación de tráfico por un lado (vehículos privados) y por otro lado una asignación a la red de transporte público.

Los modelos que integran la fase de asignación en redes congestionadas, ya sean éstas de tráfico o de transporte público, modelizan un modelo de equilibrio de Cournot-Nash, en la que los usuarios del sistema de transporte son los jugadores del juego de equilibrio, y en el que sus estrategias pueden consistir en la elección de la ruta, el modo de transporte, la realización o no del viaje, etc.

3.2. Fundamentos de la asignación de tráfico

El problema de asignación de los viajes describe a la red de tráfico. Durante este proceso se utiliza un conjunto de reglas y principios para signar una matriz O-D en la red de tráfico y así producir un conjunto de flujos en los arcos de éstas. Esto no es, sin embargo, el único producto relevante de la etapa de asignación; esta tiene varios objetivos los cuales son útil de considerar en detalle. No todos reciben en la práctica, el mismo énfasis en todas las situaciones y tampoco se puede alcanzar en todos el mismo nivel de certeza. Los objetivos primarios de los métodos de asignación son:

- Obtener buenas medidas *agregadas* de la red, por ejemplo flujos totales en los arcos, etc.
- Estimar costes (tiempos) de viajar de zona a zona para un nivel dado de demanda de viajes.
- Obtener flujos razonables en los arcos e identificar arcos muy congestionados.

Mientras que los objetivos secundarios son:

- Estimar las rutas utilizadas para cada par origen-destino.
- Analizar cuáles pares O-D utilizan un arco o un camino particular.

Los datos básicos necesarios para los modelos de asignación son

- Una matriz O-D, que expresa la demanda de viajes estimada para un intervalo de tiempo determinado (horas pico, horas valle, ...).
- Una red, específicamente conformada por los arcos y sus propiedades

La premisa básica en la asignación es la suposición de un viajero racional, esto es, un viajero que escoge la ruta que le ofrece los costes menores percibidos. Se cree que hay un buen número de factores que influyen en la elección de ruta entre dos puntos; estos incluyen el tiempo de viaje, la distancia, el coste monetario, la congestión y atascos. La construcción de una expresión generalizada de costes que incorpore todos estos elementos es una tarea difícil. Además, no es práctico tratar de modelar todos los factores en un modelo de asignación de tráfico, y, por tanto, las aproximaciones son inevitables. La aproximación más común es considerar sólo dos factores en la elección de rutas: el coste de tiempo y el coste monetario; además, se considera el coste monetario proporcional a la distancia del viaje. La mayoría de los programas de asignación del tráfico permiten que el usuario asigne pesos al tiempo de viajar y la distancia para representar las percepciones de los conductores en cuanto a estos dos factores. La suma ponderada de estos dos valores llega a ser un coste generalizado utilizado para estimar la elección de ruta. Por otra parte, hay evidencias que sugieren que, por lo menos en el tráfico urbano de vehículos privados, el tiempo es el factor dominante en la elección de rutas. Sin embargo, es un hecho que conductores diferentes escogen rutas diferentes cuando viajan entre los mismos dos puntos, esto puede deberse a dos tipos de razones

- Diferencias en las percepciones individuales de lo que constituye la “mejor ruta”; individuos diferentes no sólo pueden incorporar factores diferentes en su función generalizada de costes sino percibirlos de formas distintas.
- Los efectos de la congestión que primero afectan a las rutas más cortas y hacen sus costos generalizados comparable a rutas que inicialmente fueron menos atractivas.

Los distintos tipos de modelos de asignación son más o menos adecuados para considerar estas influencias. Se entiende que las hipótesis sobre las decisiones tomadas por los conductores para la elección de las rutas, la más natural es la de suponer que éstos eligen la ruta que les represente un coste mínimo hacia su destino bajo las condiciones de tráfico existente en la ruta a elegir. El resultado de la adopción de este criterio por los conductores, lleva a una situación en la que ningún conductor puede reducir el coste de su desplazamiento mediante cambio de su ruta, llegándose así a un equilibrio, que es consecuencia de su competición en igualdad de conocimiento de las posibilidades que ofrece el sistema de transporte. Este criterio se ha denominado criterio de “usuario-óptimo”, caracterizado éste por el hecho de que todas las rutas usadas poseen iguales costes marginales. Como luego veremos, Wardrop (1952a) fue el primero en establecer dos criterios de elección de ruta a los que se conoce como los dos principios de Wardrop. Sin embargo, han surgido algunas objeciones acerca de las hipótesis sobre el comportamiento de los usuarios y, por tanto, sobre la aplicabilidad del modelo de equilibrio de Wardrop.

3.2.1. Modelos de asignación de tráfico

El desarrollo de las herramientas tecnológicas y el crecimiento de las necesidades han permitido desarrollarse diversos modelos de asignación tales como:

Modelos estáticos de asignación de tráfico

Es uno de los más populares y conocidos. Los modelos estáticos asumen una situación estacionaria de la demanda. Esta suposición es válida para problemas planteados en un contexto de planificación estratégica, mientras que para aspectos operacionales se deberá recurrir a modelos dinámicos. Una vez conocida la movilidad en la zona de estudio, actual o futura, y definida la red de transporte que ha de soportarla, se trata de simular el comportamiento de ésta a través de la obtención de las cargas en la red que se deriven de la matriz de movilidad, o matriz $O - D$. A partir de la red vial, se selecciona un grafo, que garantice todos los posibles itinerarios coherentes entre cada par de demanda. La modelización de la red supone la descripción de ésta de acuerdo con unas normas establecidas.

A grandes rasgos, los modelos de asignación de tráfico estáticos se centran en unas pocas horas del día, como las horas puntas, y trabajan con valores medios (demandas, tiempos, flujos, etc.) durante el periodo de estudio

El problema partirá del número de usuarios que van a viajar en cada par origen-destino e intentará modelizar el comportamiento de éstos en la red de transporte,

intentando predecir las rutas que elegirán dentro de las distintas rutas o caminos posibles. Con esto conseguiremos modelizar el flujo de tráfico en cada uno de los tramos que componen la red viaria, evaluando la congestión de cada tramo.

- **Modelos de asignación determinista** Existen numerosos modelos que pertenecen a esta clase, todos ellos se pueden clasificar en función de que los costes en los arcos se consideren separables o no separables (asimétricos). Diremos que un modelo considera los costes en los arcos como separables cuando el coste de atravesar un arco de la red no depende del nivel de flujo en los restantes arcos; en caso contrario, consideraremos que los costes son no separables. Además, estos modelos se pueden clasificar según se considere o no que el coste de viaje en un arco depende o no del flujo en los arcos de la red, lo que origina a modelos con y sin congestión. Los primeros son adecuados para centros urbanos y los segundos para zonas interurbanas.

Los modelos sin congestión son modelos también conocidos como modelo de asignación todo o nada. En ellos primero se determinan los caminos o rutas de coste generalizado mínimo entre todos los pares origen-destino. Para cada par origen-destino se asignan todos los viajes de dicho par por dichos caminos. Finalmente, conocidos los flujos de cada ruta se obtienen los flujos en los arcos de la red.

- **Modelos de asignación estocástica** Análogamente a la clasificación anterior, podemos distinguir entre modelos con congestión y no congestión. En un entorno de no congestión, el esquema es común para todos los modelos. Primero realizan una determinación de todas las rutas existentes entre cada par origen-destino; especificando su coste. A continuación, para cada par origen-destino, en función del coste de las rutas, se crea la matriz de probabilidad de elección de ruta, después, para cada par, se asignan los viajes a la rutas creadas, en función de la matriz de probabilidad de elección de ruta; finalmente, conocidos los flujos de ruta, se obtienen los volúmenes en los arcos. La diferencia fundamental entre los distintos métodos de este tipo estriba en la forma en que calculan las probabilidades de elección de los caminos (rutas).

Modelos dinámicos de asignación de tráfico

El problema de asignación dinámica de tráfico consiste en la estimación de los flujos de vehículos que utilizan los diferentes tramos de la red viaria, de manera variable con el tiempo, constituyendo por tanto una extensión del problema de asignación convencional o en equilibrio. Principalmente se han desarrollado modelos deterministas de asignación dinámica de tráfico bajo tres aproximaciones:

- Simulación
- Programación matemática
- Control óptimo sobre redes

Si bien, algunas de las formulaciones en programación matemática pueden contemplarse como problemas en control óptimo discreto y, a su vez, las formulaciones en control óptimo (continuas) se aproximan mediante programación matemática para su resolución quedando reducidas, en definitiva, a programación matemática.

Estos modelos consideran que los usuarios minimizan sus tiempos de viaje actualizando continuamente sus rutas elegidas de acuerdo con las condiciones del tráfico. El primer modelo de simulación por ordenador de la asignación dinámica de tráfico, de acuerdo con los principios de Wardrop, fue presentado por [16]

3.3. Modelos de optimización

Un problema de optimización viene caracterizado por tres elementos: las variables del problema que definen el conjunto de decisiones, la función objetivo que evalúa el coste o el beneficio de la decisión y el conjunto de soluciones que determina las decisiones válidas que pueden llevarse a cabo. Más formalmente, sea $f : S \rightarrow \mathbb{R}$ una función, X un subconjunto de S que se denomina conjunto factible de soluciones, entonces el problema de optimización (versión minimización) se formula como:

$$\begin{aligned} \min \quad & f(x) \\ \text{sujeto a} \quad & x \in X \end{aligned} \tag{3.1}$$

La forma de abordar el problema (3.1) es analizar situaciones particulares

Se van a exigir, tanto a f como a X , propiedades que sean suficientemente generales para poder ser utilizadas en las aplicaciones en estudio, y lo suficientemente fuertes para obtener resultados de interés. La hipótesis más débil que suele exigirse a X es que sea un conjunto cerrado. En los modelos que estudiaremos asumiremos que el conjunto X es convexo, incluso en la mayoría de los casos, tendremos propiedades aún más fuertes como la de ser un conjunto poliédrico (definido mediante restricciones lineales). Las condiciones para f son de dos tipos. Por un lado se exigen ciertas condiciones de regularidad local para poder caracterizar los extremos (mínimos) locales del problema y por otro, se exigen propiedades acerca del comportamiento global de la función, de modo que permitan garantizar que tales extremos locales son también extremos globales ahondar mas sobre el tema en [1].

Modelos de optimización aplican la teoría de optimización a simulaciones urbanas (asumen que la distribución de las actividades urbanas pueden ser ubicadas de tal manera que se optimicen algunas funciones objetivo, por ejemplo, el costo del transporte). Los modelos generalmente tienen contenidos que se ubican en ellos para asegurar que el sistema que está siendo simulado concuerde con las observaciones.

Para describir complejas situaciones humanas y sociales, esto debería escribirse en una expresión matemática que contenga una o más variables, cuyos valores deben determinarse. La pregunta que se formula, en términos generales, es qué valores deberían tener estas variables para que la expresión matemática tenga el mayor valor numérico posible (maximización) o el menor valor numérico posible (minimización). A este proceso general de maximización o minimización se lo denomina optimización

La optimización, también denominada programación matemática, sirve para encontrar la respuesta que proporciona el mejor resultado, la que logra mayores ganancias, mayor producción o felicidad o la que logra el menor costo, desperdicio o malestar. Con frecuencia, estos problemas implican utilizar de la manera más eficiente los recursos, tales como dinero, tiempo, maquinaria, personal, existencias, etc. Los problemas de optimización generalmente se clasifican en lineales y no lineales, según las relaciones del problema sean lineales con respecto a las variables.

La Programación Matemática, en general, aborda el problema de determinar asignaciones óptimas de recursos limitados para cumplir un objetivo dado. El objetivo debe representar la meta del decisor. Los recursos pueden corresponder, por ejemplo, a personas, materiales, dinero o terrenos. Entre todas las asignaciones de recursos admisibles, queremos encontrar las que maximizan o minimizan alguna cantidad numérica tal como ganancias o costos.

El objetivo de la optimización global es encontrar la mejor solución de modelos de decisiones difíciles, frente a las múltiples soluciones locales.

3.3.1. Modelo de Transporte

El objetivo de los modelos de transporte es encontrar la solución a un problema de coste de modo que éste sea mínimo para la realización de un plan de envíos, transporte o distribución, desde cualquier grupo de centros de abastecimiento llamados orígenes, a cualquier grupo de centros de recepción llamados destinos, es decir, determinar la cantidad de productos o mercancías que se deben enviar desde cada punto de origen a cada punto de destino, teniendo en cuenta las restricciones propias del problema referidas a las capacidades o disponibilidad de los centros de abastecimiento y las demandas de los centros de destino, de manera que se minimicen los costes totales de transporte o distribución. Los orígenes pueden ser fábricas, almacenes o cualquier punto o lugar desde el que se quiera enviar mercancías o productos. Los destinos son los puntos o lugares en donde se reciben dichas mercancías o productos.

Así el problema general del modelo de transporte se tiene m fuentes y n destinos, cada fuente y cada destino representa un nodo. Los arcos representan las rutas que enlazan las fuentes con los destinos. El arco (i, j) que une a la fuente i con el destino j conduce dos clases de información: el costo de transporte c_{ij} por unidad, y la cantidad transportada x_{ij} . La cantidad de oferta en la fuente i es a_i y la cantidad de demanda en el destino j es b_j . El objetivo del modelo es determinar las incógnitas x_{ij} que minimicen el costo total del transporte, y que al mismo tiempo satisfagan las restricciones de oferta y demanda.

En forma resumida podemos enunciar el problema de modo que, se desea enviar cierto producto en determinadas cantidades u_1, \dots, u_n , desde cada uno de m orígenes, y recibirse en cantidades v_1, \dots, v_n , en cada uno de n destinos. El problema consiste en determinar las cantidades x_{ij} , que deben enviarse desde el origen i al destino j , para conseguir minimizar el coste del envío. Los cuatro elementos principales de este problema son:

1. Datos

m : el número de orígenes.

n : el número de destinos.

u_i : la cantidad que debe enviarse desde el origen i .

v_j : la cantidad que debe ser recibida en el destino j .

c_{ij} : el coste de envío de una unidad de producto desde el origen i al destino j

2. Variables

x_{ij} : la cantidad que se envía desde el origen i al destino j .

Se supone que las variables deben ser no negativas:

$$x_{ij} \geq 0 \quad i = 1, \dots, m; j = 1, \dots, n; \quad (3.2)$$

Esto implica que la dirección de envío del producto está prefijada desde los distintos orígenes hasta los destinos. No obstante, otras hipótesis podrán tenerse en cuenta. Por ejemplo, podría no limitarse el signo de las variables $x_{ij} \in \mathbb{R}$, si no se quiere predeterminar cuáles son los puntos de partida y de llegada. Tener en cuenta, que los subíndices de las variables representan el origen y el destino respectivamente, y por ende el subíndice en sí no es una variable.

3. **Función Objetivo:** En la función objetivo que debe optimizarse en el problema de transporte, generalmente interesa minimizar los costos de envío (suma de los costos de envío por unidad de producto multiplicado por las cantidades enviadas), es decir:

$$\min z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (3.3)$$

4. **Restricciones:** Las restricciones de este problema son:

$$\sum_{j=1}^n x_{ij} = u_i \quad i = 1, \dots, m, \quad (3.4)$$

$$\sum_{i=1}^m x_{ij} = v_j \quad j = 1, \dots, n \quad (3.5)$$

4

Diseño de la Metodología

4.1. Generalidades

El implementación computacional de los modelos es muy importante en el estudio y desarrollo de la aplicación de la programación matemática lineal y no lineal los cuales ayudan a resolver los diversos problemas.(estudio de optimización de funciones). Un problema asociado ha esto es la asignación de tráfico.

Un problema de optimización viene caracterizado por tres elementos: *las variables del problema* que definen el conjunto de decisiones, la *función Objetivo* que evalúa el coste o el beneficio de la decisión y el conjunto de *soluciones* que determina las decisiones válidas que pueden llevarse a cabo. De manera formal, sea $f : S \rightarrow \mathbb{R}$ una función X un subconjunto de S que se denomina *conjunto factible de soluciones* entonces el problema de optimización se formula como:

$$\begin{array}{ll} \min & f(\mathbf{x}) \\ \text{sujeto a:} & \mathbf{x} \in X \end{array} \quad (4.1)$$

4.2. Formulación del Problema de Asignación de Tráfico

La planificación del tráfico ha motivado un buen número de modelos matemáticos. El análisis de estos modelos ayuda a planificar y predecir los efectos que determinados cambios en la red de tráfico tendrán un efecto en la buena marcha de la red.

La formulación del problema expresada en (4.1) es una expresión muy general para que su estudio conduzca a métodos satisfactorios de resolución. La forma de abordar esto es analizar situaciones particulares. Se exige tanto a f como a X , propiedades que sean suficientemente generales para poder ser utilizadas en las aplicaciones de estudio, y lo suficientemente fuertes para obtener resultados de interés. La hipótesis mas débil que se suele exigir a X es que sea un conjunto cerrado. En los modelos que estudiaremos asumiremos que el conjunto X sea un conjunto convexo (podemos

exigir que esté definido mediante restricciones lineales). Las condiciones para f son de dos tipos. Por un lado se exigen ciertas condiciones de regularidad local para poder caracterizar los extremos (mínimos) locales del problema y por otro lado se exigen propiedades acerca del comportamiento global de la función, de modo que permitan garantizar que tales extremos locales sean también extremos globales.

Teorema 4.2.1. (Condiciones de optimalidad(f, X)). El punto $\mathbf{x}^* \rightarrow X$ es una solución óptima de condiciones de optimalidad de f si y solo si cumple

$$\nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \geq 0, \quad \forall \mathbf{x} \in X \quad (4.2)$$

donde ∇f es el gradiente de la función f . (Ver, por ejemplo el teorema 3.3.4 de [1])

Teorema 4.2.2. (Condición suficiente para $F = \nabla f$). Sea $F : X \rightarrow \mathbb{R}^n$ de clase C^1 sobre un conjunto convexo y abierto $X_0 \in X$. Entonces F es el gradiente de una aplicación en X_0 si y solo si $\nabla F(\mathbf{x})$ es simétrica para todo $x \in X_0$.

Debe introducirse un principio que gobierne el comportamiento de los usuarios al elegir la ruta en la red. Wardrop fue el primero en enunciar formalmente este principio: “Bajo condiciones de equilibrio, el tráfico se organiza en redes congestionadas de tal modo que ningún vehículo puede reducir el tiempo de viaje mediante un cambio de ruta”.

Este principio se ha usado como punto de partida para confeccionar modelos de asignación en equilibrio. Un corolario de este principio es que si todos los viajeros perciben el tiempo de los viajes del mismo modo, bajo condiciones de equilibrio, todas las rutas utilizadas entre un par $O - D$ tienen el mismo tiempo mínimo mientras las no usadas requieren un tiempo igual o mayor.

Para la formulación matemática del problema de asignación de tráfico, se considera los siguientes elementos principales:

1. Datos

(\mathcal{N}, A) : un grafo dirigido (\mathcal{N}, A) , que se entiende como un modelo de la red de tráfico con un conjunto de nodos \mathcal{N} , y un conjunto de arcos A que representan las calles. El conjunto de nodos \mathcal{N} del grafo representan intersecciones o también los llamados centroides, que indican las zonas del estudio (orígenes y destinos).

W : el conjunto de pares orígenes-destinos.

d_w : dato de entrada, que representan el número de viajes en la red desde el origen i al destino j , para cada par origen-destino $w = (i, j)$. La matriz de pares origen-destino $\{d_w\}, w \in W$ se obtiene en la fase de distribución modal.

$C_a(f_a)$: una función de coste que indica el retraso en el arco $a \in A$, para cada arco $(i, j) \in A$, como función del flujo total f_a que lleva el mismo arco a .

\mathcal{R} : es el conjunto de rutas para el par $w = (i, j)$.

2. Variables

h_r : es el flujo en la ruta r

f_r : es el flujo en el arco a

3. **Restricciones.** El número de usuarios de un par origen-destino $w = (i, j)$ es la suma del número total de usuarios en caminos distintos que satisfacen tal par:

$$\sum_{r \in \mathcal{R}} h_r = d_w, \forall w \in W.$$

Además el flujo de cada camino debe ser no negativo:

$$h_r \geq 0, \forall r \in \mathcal{R}_w, \forall w \in W.$$

El flujo de cada arco a es la suma del flujo en todos los caminos que lo usan:

$$\sum_{w \in W} \sum_{r \in \mathcal{R}_w} \delta_{a,r} h_r = f_a, \forall a \in A.$$

Donde $\delta_{a,r} = 1$ si $r \in \mathcal{R}_w$ contiene al arco a y 0 en otro caso. A

4. **Función Objetivo.** El problema de asignación de tráfico minimiza la siguiente función:

$$z = \sum_{a \in A} \int_0^{f_a} C_a(x) dx.$$

Como resultado del volumen creciente de tráfico, la velocidad en los arcos tiende a disminuir. La función C_a , es decir el tiempo necesario para atravesar el arco a , tiene en cuenta este hecho. Estas funciones en el análisis de sistemas de tráfico son positivas, no lineales y estrictamente crecientes. Los parámetros que relacionan el tiempo de viaje, C_a , en el arco en función del flujo f_a en el arco, es el *tiempo de viaje libre de flujo*, c_a^0 , y la capacidad práctica del arco, k_a , que es una medida del flujo a partir del cual, el tiempo de viaje se incrementará muy rápidamente si el flujo aumenta. La expresión más común para $C_a(f_a)$, llamada la función BPR, es:

$$C_a(f_a) = c_a^0 + b_a \left(\frac{f_a}{k_a} \right)^{n_a} \quad (4.3)$$

Esta es una ecuación variacional, como resultado del volumen creciente de tráfico, la velocidad en los arcos tiende a disminuir. Podemos comentar lo siguiente de la función a optimizar:

Si se considera a los usuarios provistos de información completa sobre los costos de cada ruta, se puede suponer que en una distribución óptima ningún usuario está interesado en cambiar de ruta pues utiliza la que le resulta menos costosa. Esto da lugar al llamado Equilibrio de Wardrop, que postula que los costos (tiempo de recorrido) de las rutas utilizadas son todos iguales y no mayores que los de las rutas no utilizadas. Se puede obtener una formulación matemática equivalente al principio de Wardrop a través de una inecuación variacional, consistente en hallar $f^* \in K$; que verifique la siguiente desigualdad:

$$\langle C(f^*), f - f^* \rangle \geq 0 \forall f \in K.$$

Donde: K es el convexo determinado por los vectores de flujo sobre las rutas, no negativas, que verifican una ecuación de demanda y C es la función de costo sobre rutas. En el caso que el Jacobiano de $C(f)$ sea simétrico el problema anterior es equivalente a un problema de optimización. Una condición necesaria para ello es que el tiempo de recorrido de cada arco sea el mismo para todos los usuarios y dependa solamente del flujo sobre dicho arco.

La función a optimizar es:

$$\begin{aligned} Z &= \sum_{a \in A} C_a(f_a) = \sum_{a \in A} \int_0^{f_a} c_a(x) dx = \sum_{a \in A} \int_0^{f_a} [c_a^0 + b_a \left(\frac{x}{k_a}\right)^{n_a}] dx \\ &= \sum_{a \in A} [c_a^0 f_a + \frac{b_a}{n_a + 1} \left(\frac{x}{k_a}\right)^{n_a+1}] = \sum_{a \in A} c_a^0 f_a + d_a f_a^{m_a}. \end{aligned}$$

donde:

$$d_a = \frac{b_a}{(n_a + 1)k_a^{n_a+1}} \quad y \quad m_a = n_a + 1.$$

4.3. El método de Frank Wolfe

El algoritmo Frank-Wolfe es un algoritmo importante para solucionar problemas no lineales de programación. Es especialmente ventajoso en los problemas de asignación de tráfico porque no necesita enumerar todas las rutas factibles entre los orígenes y los destinos, que podría ser un trabajo muy pesado cuando el tamaño de la red aumenta. También, permite usar algoritmos de flujos de red muy eficientes como sus subproblemas. Sin embargo, tiene una tendencia en la que la convergencia se pone lenta cuando se acerca al punto óptimo. En este estudio, la modificación del algoritmo Frank-Wolfe se hace utilizando los algoritmos y programas descritos anteriormente. El algoritmo de Frank-Wolfe también, permite usar el algoritmo del camino más corto que es muy eficiente para resolver el problema de asignación de tráfico con función objetivo no lineal que es inevitable para considerar congestión de tráfico en enlaces de la red principal.

El algoritmo de Frank-Wolfe para optimización no lineal considera el problema de optimización de la forma

$$\begin{aligned} \text{mín} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s.a.} \quad & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

donde $f : \mathbb{R}^n \rightarrow \mathbb{R}$ es una función continuamente diferenciable en \mathbb{R}^n en por lo menos en un conjunto abierto que contiene al conjunto admisible

$$\Omega = \mathbf{x} \in \mathbb{R}^n : A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}$$

Donde A es una matriz de orden $m \times n$, con $m < n$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{x} \in \mathbb{R}^n$. Este algoritmo procura determinar un punto estacionario de f en el conjunto Ω resolviendo en cada iteración un programa lineal que surge de la definición de punto estacionario (puntos

donde la función toma sus valores extremos es decir donde la derivada de la función es cero).

Descripción del algoritmo de Frank-Wolfe, sea $\bar{x} \in \Omega$ una solución admisible del programa no lineal. Para verificar si \bar{x} es un punto estacionario de f en Ω , se resuelve el problema de programación lineal

$$\begin{aligned} \text{mín} \quad & \nabla f(\bar{x})^T \mathbf{x} \\ \text{s.a.} \quad & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in \Omega \end{aligned}$$

Sea $\mathbf{y} \in \Omega$ una solución óptima de este programa. entonces podemos discutir los casos:

(i). Si

$$\nabla f(\bar{x})^T \bar{x} \leq \nabla f(\bar{x})^T \mathbf{y} \quad (4.4)$$

entonces

$$\nabla f(\bar{x})^T \bar{x} \leq \nabla f(\bar{x})^T \mathbf{y} \leq \nabla f(\bar{x})^T \mathbf{x}$$

para todo $\mathbf{x} \in \Omega$ y por tanto \bar{x} es punto estacionario de f en Ω

(ii). Si (4.4) no se verifica, considérese el vector

$$\mathbf{d} = \mathbf{y} - \bar{x}$$

Entonces \mathbf{d} es una dirección descendente de f en \bar{x} , pues satisface

$$\nabla f(\bar{x})^T \mathbf{d} = \nabla f(\bar{x})^T (\mathbf{y} - \bar{x}) < 0$$

como f es continuamente diferenciable, existe $\bar{\alpha} > 0$ tal que

$$\forall 0 \leq \alpha \leq \bar{\alpha} \quad \nabla f(\bar{x} + \alpha \mathbf{d})^T \mathbf{d} < 0 \quad (4.5)$$

Entonces, por el teorema de los incrementos finitos (2.4.4), para $0 < \alpha < \bar{\alpha}$ entonces

$$\exists 0 < \bar{\alpha} < \alpha \quad f(\bar{x} + \alpha \mathbf{d}) = f(\bar{x}) + \nabla f(\bar{x} + \tilde{\alpha} \mathbf{d})^T (\alpha \mathbf{d})$$

Pero $\alpha > 0$ y

$$\nabla f(\bar{x} + \tilde{\alpha} \mathbf{d})^T \mathbf{d} < 0$$

por (4.5). Por tanto

$$\forall 0 \leq \alpha \leq \bar{\alpha} \quad \nabla f(\bar{x} + \alpha \mathbf{d}) < f(\bar{x}) \quad (4.6)$$

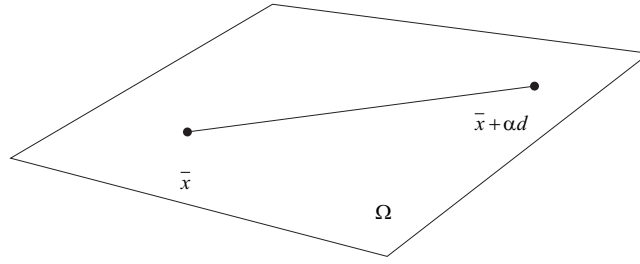


Figura 4.1: Digrafo conexo

Así se obtiene que un movimiento de $\bar{\mathbf{x}}$ a otro punto $(\bar{\mathbf{x}} + \alpha \mathbf{d}) \in \Omega$ en la dirección \mathbf{d} es conveniente para el objetivo de minimización de la función f en Ω .

El desarrollo del algoritmo para determinar un punto estacionario de f en Ω necesitamos un proceso práctico para calcular el valor de α que satisfaga la desigualdad (4.6). La primera exigencia para determinar tal valor de α es respecto a $(\bar{\mathbf{x}} + \alpha \mathbf{d}) \in \Omega$. Pero esto ocurre si

$$\begin{aligned} A(\bar{\mathbf{x}} + \alpha \mathbf{d}) &= \mathbf{b} \\ \bar{\mathbf{x}} + \alpha \mathbf{d} &\geq \mathbf{0} \end{aligned}$$

Como $\bar{\mathbf{x}} \in \Omega$, entonces $A\bar{\mathbf{x}} = \mathbf{b}$ y $\bar{\mathbf{x}} \geq \mathbf{0}$. Por lo tanto

$$\begin{cases} \alpha (A\mathbf{d}) = 0 \\ \bar{\mathbf{x}}_i + \alpha d_i \geq 0 \quad , i = 1, 2, \dots, n \end{cases}$$

Pero $y \in \Omega$, y por lo tanto

$$A\mathbf{d} = A(\mathbf{y} - \bar{\mathbf{x}}) = A\mathbf{y} - A\bar{\mathbf{x}} = 0$$

por lo que la primera restricción es verdadera. La segunda restricción es verdadera si $d_i \geq 0$ para todo $i = 1, 2, \dots, n$. De otro modo

$$\alpha \leq \frac{\bar{x}_i}{-d_i} \quad \text{para todo } i \text{ tal que } d_i < 0$$

Entonces α se debe escoger tal que sea menor o igual al número α_{max}

$$\alpha_{max} = \begin{cases} +\infty & \text{si } d_i \geq 0 \\ \min \left\{ \frac{\bar{x}_i}{-d_i} : d_i < 0 \right\} & \text{para todo } i = 1, 2, \dots, n \end{cases} \quad (4.7)$$

Así α debe ser un número positivo inferior o igual a α_{max} que satisfaga la desigualdad (4.6). En la práctica esa desigualdad es sustituida por la llamada Fórmula o Criterio de Armijo:

$$f(\bar{\mathbf{x}} + \alpha \mathbf{d}) \leq f(\bar{\mathbf{x}}) + \alpha \beta \nabla f(\bar{\mathbf{x}})^T \mathbf{d} \quad (4.8)$$

donde β es un número real positivo menor o igual que uno. se debe notar que si α satisface esta desigualdad, entonces también satisface $f(\bar{\mathbf{x}} + \alpha \mathbf{d}) < f(\bar{\mathbf{x}})$. La determinación de α se hace de la forma

$$\alpha = \theta^t \min \{1, \alpha_{max}\}, \quad t = 0, 1, \dots \quad (4.9)$$

con $0 < \theta < 1$ un número dado (normalmente $\theta = 1/2$) tal que satisface la desigualdad (4.8)

4.3.1. Algoritmo de Frank-Wolfe

El algoritmo del Frank-Wolfe es uno de los algoritmos de aproximación en la optimización no lineal. Genera una dirección factible que minimiza la función objetivo no lineal en cada iteración para encontrar la solución, para ello debe satisfacer un criterio predefinido de terminación. LeBlanc Et Al. (1975) uso inicialmente el algoritmo de Frank-Wolfe para solucionar el problema de asignación de tráfico en su investigación, el algoritmo ha sido usado ampliamente en el campo de transporte, pues determina de manera efectiva el flujo a ser asignada a la red y es un método relativamente efectivo en términos de método fácil y la cantidad moderada de almacenamiento de datos.

El método de Frank-Wolfe es un método que tiene como criterio linealizar la función objetivo, luego esta nueva función puede resolverse por cualquier otro método o criterio de programación lineal. Los pasos detallados del algoritmo son los siguientes:

Algoritmo 4.3.1. (*Algoritmo de Frank-Wolfe*)

En Términos generales consideremos el problema de minimización

$$\begin{aligned} \text{mín} \quad & f(\mathbf{x}) \\ \text{s.a.} \quad & \mathbf{x} \in \Omega \end{aligned} \quad (4.10)$$

Donde: Ω es un conjunto convexo factible, f es una función convexa y continuamente diferenciable sobre el conjunto Ω ,

0. *Elegir un punto inicial (solución inicial): \mathbf{x}_0*

- *Una solución factible, $\mathbf{x}_0 \in \Omega$, es un punto inicial arbitrario*
- *Hacemos $k = 0$.*

1. *Determinación de la dirección factible: \mathbf{d}_k Para determinar la dirección \mathbf{d}_k aproximación de la función f a su forma de serie de Taylor alrededor del punto \mathbf{x}_k es decir resolvemos el problema*

$$\begin{aligned} \text{mín} \quad & g_k(\mathbf{y}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T (\mathbf{y} - \mathbf{x}_k) \\ \text{s.a.} \quad & \mathbf{y} \in \Omega \end{aligned} \quad (4.11)$$

Esto es un problema lineal, y tiene un valor extremo \mathbf{y}_k , como un valor óptimo. De modo que la dirección buscada será $\mathbf{d}_k = \mathbf{y}_k - \mathbf{x}_k$, es el vector direccional que se inicia en el punto factible \mathbf{x}_k hacia el punto extremo, además es una dirección factible pues los puntos \mathbf{x}_k y \mathbf{y}_k y están dentro de Ω que es convexo.

2. *Determinación de la longitud del paso α_k tal que $f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) < f(\mathbf{x}_k)$. En este caso la longitud del paso será a lo más 1, porque para $\alpha > 1$ la solución sería no admisible; La línea buscada proviene de resolver*

$$\begin{aligned} \text{mín} \quad & f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \\ \text{s.a.} \quad & \alpha \in [0, 1] \end{aligned} \tag{4.12}$$

3. *Nuevo punto de iteración:*

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \tag{4.13}$$

4. *Si se cumple:*

$$\frac{f(\mathbf{x}_k) - g_k(\mathbf{y}_k)}{|g_k(\mathbf{y}_k)|} \leq \varepsilon, \quad \varepsilon > 0$$

Finalizar, y \mathbf{x}_{k+1} es una aproximación de \mathbf{x} . Y así el valor óptimo de la función es $f(\mathbf{x}_{k+1}) = f(\mathbf{x})$. De otro modo hacer $k = k + 1$, e ir al paso 1

Resolver el siguiente problema de minimización :

Ejemplo 1. *Consideremos el siguiente ejemplo para ilustrar el algoritmo de Frank-Wolfe*

$$\text{mín} \quad f(\mathbf{x}) = 3x_1^2 - 36x_1 + 3x_2^4 - 324x_2 \tag{4.14}$$

$$\text{s.a.} \quad 4x_1 + 3x_2 \leq 24 \tag{4.15}$$

$$x_1 - x_2 \leq 3 \tag{4.16}$$

$$x_1 \geq 0, x_2 \geq 0$$

En la figura (4.2) representamos como una región sombreada la región factible y se observa que el punto mínimo está fuera de la región

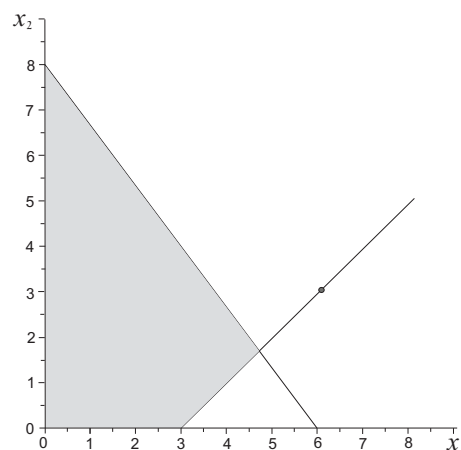


Figura 4.2: Región Factible del problema

Resolviendo el sistema homogéneo, formado por las derivadas parciales de la función objetivo se obtiene el mínimo de $f(\mathbf{x})$ que es $\mathbf{x}^ = (6, 3)$. Sin embargo, este punto*

“óptimo” está fuera de la región limitado por las restricciones (4.15) y (4.16), pues los puntos óptimos deben estar dentro de la región.

A continuación se hace una Descripción los pasos del algoritmo de Frank-Wolfe para encontrar las solución.

iteración 1.

0. elección del punto inicial: sea \mathbf{x}_0

sea $\mathbf{x}_0 = (x_1, x_2) = (0, 0)$

sea $k = 0$

1. Búsqueda de la dirección: \mathbf{d}_0

las derivadas parciales de la función objetivo $f(\mathbf{x})$ en el punto $\mathbf{x}_0 = (0, 0)$ son:

$$\frac{\partial f}{\partial x_1} = 6x_1 - 36 = -36 \tag{4.17}$$

$$\frac{\partial f}{\partial x_2} = 12x_2^3 - 324 = -324 \tag{4.18}$$

evaluando las derivadas parciales en \mathbf{x}_0 como coeficientes de costo del siguiente problema de minimización:

$$\text{mín } s_0(\mathbf{y}) = -36y_1 - 324y_2 \tag{4.19}$$

$$\text{s.a. } 4y_1 + 3y_2 \leq 24 \tag{4.20}$$

$$y_1 - y_2 \leq 3 \tag{4.21}$$

$$y_1 \geq 0, y_2 \geq 0$$

Siendo este un problema de minimización lineal cuya solución es $\mathbf{y}_0 = (y_1, y_2) = (0, 8)$ como se muestra en la figura (4.3) Una dirección facti-

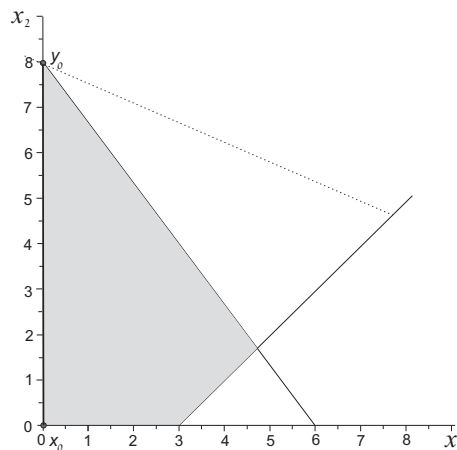


Figura 4.3: Dirección factible del ejemplo

ble es el vector $\mathbf{d}_0 = \mathbf{y}_0 - \mathbf{x}_0 = (0, 8)$. en el siguiente paso buscamos el punto mínimo entre \mathbf{x}_0 y \mathbf{y}_0 .

2. *Cálculo del siguiente punto de iteración: \mathbf{x}_1*
Realizar la búsqueda lineal el tamaño del paso α el cual dará el valor mínimo de la función f .

$$\begin{aligned} f(\mathbf{x}_0 + \alpha \mathbf{d}_0) &= f(0, 8\alpha) \\ &= 3(8\alpha)^4 - 324(8\alpha) \\ &= 12288\alpha^4 - 2592\alpha \end{aligned}$$

entonces

$$\text{mín } f(\mathbf{x}_0 + \alpha \mathbf{d}_0) = 12288\alpha^4 - 2592\alpha \quad (4.22)$$

$$\text{s.a. } 0 \leq \alpha \leq 1 \quad (4.23)$$

La solución es $\alpha^* = 0.375$. Por lo tanto la ecuación (4.13), el nuevo punto de iteración es $\mathbf{x}_1 = (0, 0) + 0.375(0, 8) = (0, 3)$.

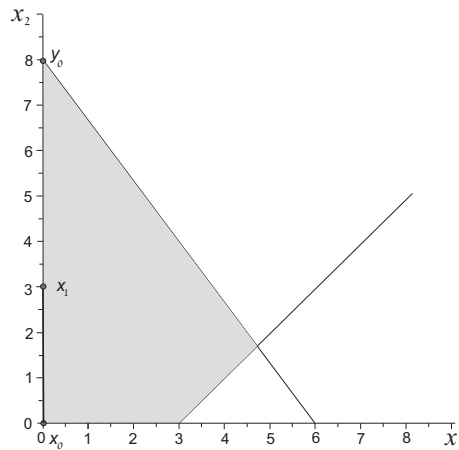


Figura 4.4: Nuevo Punto de Iteración \mathbf{x}_1

3. *Criterio de evaluación para finalizar. Calcular el estado de terminación*

$$\frac{f(\mathbf{x}_k) - f_k(\mathbf{x}_{k+1})}{|f_k(\mathbf{x}_{k+1})|} \leq \varepsilon$$

donde ε es un valor positivo pequeño. Entonces

$$\frac{f(\mathbf{x}_0) - f_k(\mathbf{x}_1)}{|f_k(\mathbf{x}_1)|} = \frac{0 - (-729)}{|-729|} = 1$$

después de lo cual se observa la mejora relativa no verifica la desigualdad, realizar otra iteración, para el propósito del ejemplo $\varepsilon = 0,01$. pues la relación es mayor que $\varepsilon = 0,01$ continuamos el algoritmo con un nuevo punto de iteración \mathbf{x}_1 y sea $k = 0 + 1 = 1$ ir al paso 1.

Iteración 2.

1. *Buscando la dirección: \mathbf{d}_1*
las derivadas parciales de la función objetivo $f(x)$ en el punto $\mathbf{x}_1 = (0, 3)$ son:

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= 6x_1 - 36 = -36 \\ \frac{\partial f}{\partial x_2} &= 12x_2^3 - 324 = 0 \end{aligned} \quad (4.24)$$

evaluando las derivadas parciales en \mathbf{x}_1 como coeficientes de costo del siguiente problema de minimización:

$$\text{mín } s_1(\mathbf{y}) = -36y_1 - 0y_2 \quad (4.25)$$

$$\text{s.a. } 4y_1 + 3y_2 \leq 24 \quad (4.26)$$

$$y_1 - y_2 \leq 3$$

$$y_1 \geq 0, y_2 \geq 0$$

Siendo este un problema de minimización lineal cuya solución es $\mathbf{y}_1 = (y_1, y_2) = (4.7143, 1.7143)$ como se muestra en la figura (4.5) Una dirección

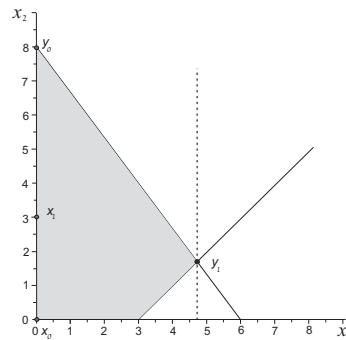


Figura 4.5: Determinando la dirección factible y_1 desde x_1

factible es el vector $\mathbf{d}_1 = \mathbf{y}_1 - \mathbf{x}_1 = (4.7143, 1.7143) - (0, 3) = (4.7143, -1.2857)$. en el siguiente paso buscamos el punto mínimo entre \mathbf{x}_1 e \mathbf{y}_1 .

2. Cálculo del siguiente punto de iteración: \mathbf{x}_2

Realizar la búsqueda lineal el tamaño del paso α el cual dará el valor mínimo de la función f .

$$\begin{aligned} \mathbf{x}_1 + \alpha \mathbf{d}_1 &= (0, 3) + \alpha(4.7143, -1.2857) \\ &= (4.7143\alpha, 3 - 1.2857\alpha) \end{aligned}$$

entonces la función a minimizar será:

$$\begin{aligned} f(\mathbf{x}_1 + \alpha \mathbf{d}_1) &= f(4.7143\alpha, 3 - 1.2857\alpha) \\ &= 3(4.7143\alpha)^2 - 24(4.7143\alpha) + 3(3 - 1.2857\alpha)^4 - 324(3 - 1.2857\alpha) \\ &= -729 - 169.7\alpha + 334.47\alpha^2 - 76.511\alpha^3 + 8.1975\alpha^4 \end{aligned}$$

Ahora resolvemos

$$\text{mín } -729 - 169.7\alpha + 334.47\alpha^2 - 76.511\alpha^3 + 8.1975\alpha^4 \quad (4.27)$$

$$\text{s.a. } 0 \leq \alpha \leq 1$$

La solución es $\alpha^* = 0.27945$. Por lo tanto la ecuación (4.13), el nuevo punto de iteración es $\mathbf{x}_2 = (0, 3) + 0.27945(4.7143, -1.2857) = (1.3174, 2.6407)$.

3. Criterio de evaluación para finalizar. Calcular el estado de terminación

$$\begin{aligned} \frac{f(\mathbf{x}_1) - f(\mathbf{x}_2)}{|f(\mathbf{x}_2)|} &= \frac{(-729 - (-751.92))}{|-751.92|} \\ &= 0.030482 \geq \varepsilon = 0.01 \end{aligned}$$

después de lo cual se observa la mejora relativa pero no cumple la desigualdad, entonces realizar otra iteración, para el propósito del ejemplo $\varepsilon = 0,01$.

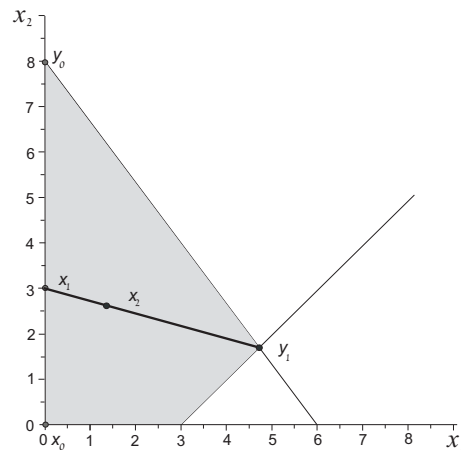


Figura 4.6: Nuevo Punto de Iteración x_2

pues la relación es mayor que $\varepsilon = 0,01$ continuamos el algoritmo con un nuevo punto de iteración x_2 y sea $k = 1 + 1 = 2$ ir al paso 1 para la siguiente iteración.

Iteración 3.

1. Buscando la dirección: d_2

las derivadas parciales de la función objetivo $f(x)$ en el punto $x_2 = (x_1, x_2) = (1.3173, 2.6407)$ son:

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= 6x_1 - 36 = -28.096 \\ \frac{\partial f}{\partial x_2} &= 12x_2^3 - 324 = -103.03 \end{aligned} \quad (4.28)$$

evaluando las derivadas parciales en x_2 como coeficientes de costo del siguiente problema de minimización:

$$\begin{aligned} \text{mín} \quad s_2(\mathbf{y}) &= -28.096y_1 - 103.03y_2 \\ \text{s.a.} \quad &4y_1 + 3y_2 \leq 24 \\ &y_1 - y_2 \leq 3 \\ &y_1 \geq 0, y_2 \geq 0 \end{aligned}$$

Siendo este un problema de minimización lineal cuya solución es $\mathbf{y}_2 = (y_1, y_2) = (0, 8)$ como se muestra en la figura (4.7)

Una dirección factible es el vector $\mathbf{d}_2 = \mathbf{y}_2 - \mathbf{x}_2 = (0, 8) - (1.3173, 2.6407) = (-1.3173, 5.3593)$. en el siguiente paso buscamos el punto mínimo entre x_2 y \mathbf{y}_2 .

2. Cálculo del siguiente punto de iteración: x_3

Realizar la búsqueda lineal el tamaño del paso α el cual dará el valor mínimo de la función f .

$$\begin{aligned} \mathbf{x}_2 + \alpha \mathbf{d}_2 &= (1.3173, 2.6407) + \alpha(-1.3173, 5.3593) \\ &= (1.3173 - 1.3173\alpha, 2.6407 + 5.3593\alpha) \end{aligned}$$

entonces la función a minimizar será:

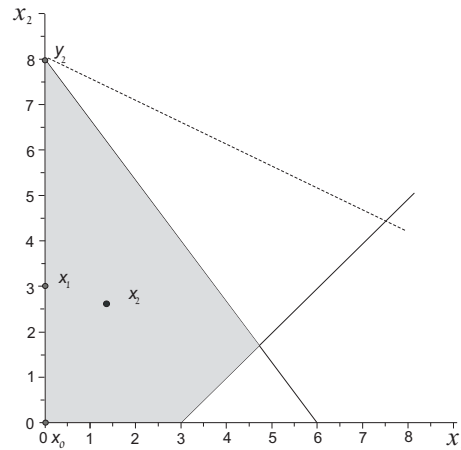


Figura 4.7: Determinando la dirección factible y_2 desde x_2

$$f(\mathbf{x}_2 + \alpha \mathbf{d}_2) = f(1.3173 - 1.3173\alpha, 2.6407 + 5.3593\alpha)$$

Ahora resolvemos

$$\begin{aligned} \text{mín} \quad & f(1.3173 - 1.3173\alpha, 2.6407 + 5.3593\alpha) \\ \text{s.a.} \quad & 0 \leq \alpha \leq 1 \end{aligned}$$

La solución es $\alpha^* = 0.062966$. Por lo tanto la ecuación (4.13), el nuevo punto de iteración es

$$\begin{aligned} \mathbf{x}_3 &= \mathbf{x}_2 + \alpha \mathbf{d}_2 \\ &= (1.3173, 2.6407) + 0.062966(-1.3173, 5.3593) \\ &= (1.2344, 2.97815). \end{aligned}$$

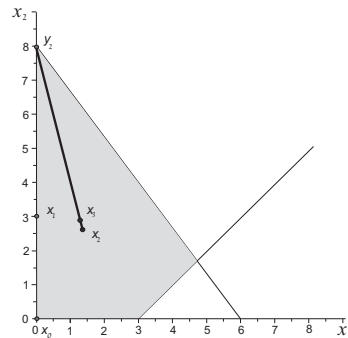


Figura 4.8: Nuevo Punto de Iteración x_3

3. Criterio de evaluación para finalizar. Calcular el estado de terminación, y comparar con la tolerancia

$$\begin{aligned} \frac{f(\mathbf{x}_2) - f(\mathbf{x}_3)}{|f(\mathbf{x}_3)|} &= \frac{(-751.92 - (-768.79))}{|-768.79|} \\ &= 0.021944 \geq \varepsilon = 0.01 \end{aligned}$$

después de lo cual se observa la mejora relativa pero aún no cumple la condición por lo que se requiere otra iteración, para el propósito del ejemplo $\varepsilon = 0,01$. pues la relación es mayor que $\varepsilon = 0,01$ continuamos el algoritmo con un nuevo punto de iteración \mathbf{x}_3 y sea $k = 2 + 1 = 3$ ir al paso 1 para la siguiente iteración.

Iteración 4.

1. *Buscando la dirección: \mathbf{d}_3*

las derivadas parciales de la función objetivo $f(x)$ en el punto $\mathbf{x}_3 = (x_1, x_2) = (1.2344, 2.97815)$ son:

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= 6x_1 - 36 = -28.594 \\ \frac{\partial f}{\partial x_2} &= 12x_2^3 - 324 = -7.0280 \end{aligned} \quad (4.29)$$

evaluando las derivadas parciales en \mathbf{x}_3 como coeficientes de costo del siguiente problema de minimización:

$$\begin{aligned} \text{mín} \quad & s_3(\mathbf{x}) = -28.594y_1 - 7.0280y_2 \\ \text{s.a.} \quad & 4y_1 + 3y_2 \leq 24 \\ & y_1 - y_2 \leq 3 \\ & y_1 \geq 0, y_2 \geq 0 \end{aligned}$$

Siendo este un problema de minimización lineal cuya solución es $\mathbf{y}_3 = (y_1, y_2) = (4.7143, 1.7143)$ como se muestra en la figura (4.12 Una dirección

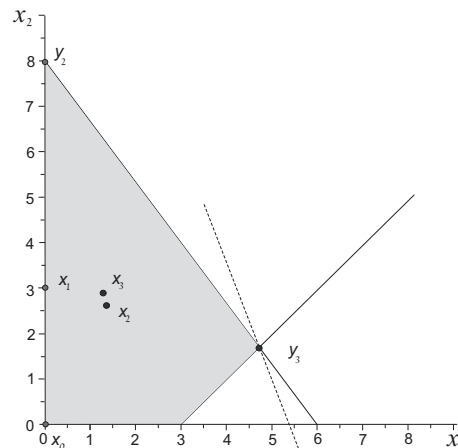


Figura 4.9: Determinando la dirección factible y_3 desde x_3

factible es el vector $\mathbf{d}_3 = \mathbf{y}_3 - \mathbf{x}_3 = (4.7143, 1.7143) - (1.2344, 2.97815) = (3.4799, -1.26386)$. en el siguiente paso buscamos el punto mínimo entre \mathbf{x}_3 y \mathbf{y}_3 .

2. *Cálculo del siguiente punto de iteración: \mathbf{x}_4*

Realizar la búsqueda lineal el tamaño del paso α el cual dará el valor mínimo de la función f .

$$\begin{aligned} \mathbf{x}_3 + \alpha \mathbf{d}_3 &= (1.2344, 2.97815) + \alpha(3.4799, -1.26386) \\ &= (1.2344 + 3.4799\alpha, 2.97815 - 1.26386\alpha) \end{aligned}$$

entonces la función a minimizar será:

$$f(\mathbf{x}_3 + \alpha \mathbf{d}_3) = f(1.2344 + 3.4799\alpha, 2.97815 - 1.26386\alpha)$$

Ahora resolvemos

$$\begin{aligned} \text{mín} \quad & f(1.2344 + 3.4799\alpha, 2.97815 - 1.26386\alpha) \\ \text{s.a.} \quad & 0 \leq \alpha \leq 1 \end{aligned}$$

La solución es $\alpha^* = 0.16547$. Por lo tanto la ecuación (4.13), el nuevo punto de iteración es

$$\begin{aligned} \mathbf{x}_4 &= \mathbf{x}_3 + \alpha \mathbf{d}_3 \\ &= (1.2344, 2.97815) + 0.16547(3.4799, -1.26386) \\ &= (1.8102, 2.7691). \end{aligned}$$

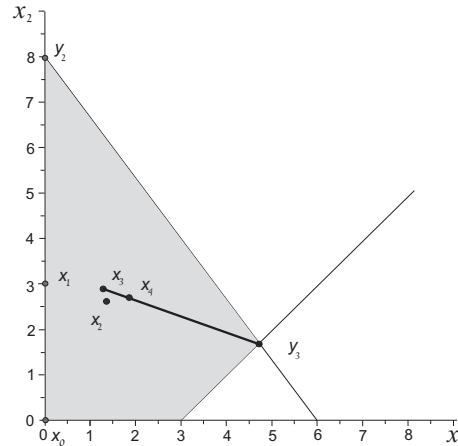


Figura 4.10: Nuevo Punto de Iteración x_4

3. *Criterio de evaluación para finalizar. Calcular el estado de terminación después de lo cual no la prueba de razón de la mejora relativa de la iteración de punto nuevo. Entonces*

$$\begin{aligned} \frac{f(\mathbf{x}_3) - f(\mathbf{x}_4)}{|f(\mathbf{x}_4)|} &= \frac{(-768.79 - (-776.13))}{|-776.13|} \\ &= 0.0095 \approx 0.01 \leq \varepsilon = 0.01 \end{aligned}$$

después de lo cual se observa la mejora relativa pero aún no cumple la condición por lo que se requiere otra iteración, para el propósito del ejemplo $\varepsilon = 0,01$. pues la relación es aproximadamente igual a $\varepsilon = 0,01$ continuamos el algoritmo con un nuevo punto de iteración \mathbf{x}_4 y sea $k = 3 + 1 = 4$ ir al paso 1 para la siguiente iteración.

Iteración 5.

1. *Buscando la dirección: \mathbf{d}_4*
 las derivadas parciales de la función objetivo $f(x)$ en el punto $\mathbf{x}_4 = (x_1, x_2) = (1.8102, 2.76902)$ son:

$$\begin{aligned} \frac{\partial f}{\partial x_1} &= 6x_1 - 36 = -25.139 \\ \frac{\partial f}{\partial x_2} &= 12x_2^3 - 324 = -69.223 \end{aligned} \tag{4.30}$$

evaluando las derivadas parciales en \mathbf{x}_4 como coeficientes de costo del siguiente problema de minimización:

$$\begin{aligned} \text{mín} \quad s_4(\mathbf{x}) &= -25.139y_1 - 69.223y_2 \\ \text{s.a.} \quad &4y_1 + 3y_2 \leq 24 \\ &y_1 - y_2 \leq 3 \\ &y_1 \geq 0, y_2 \geq 0 \end{aligned}$$

Siendo este un problema de minimización lineal cuya solución es $\mathbf{y}_4 = (y_1, y_2) = (0, 8)$ como se muestra en la figura (4.12 Una dirección factible es

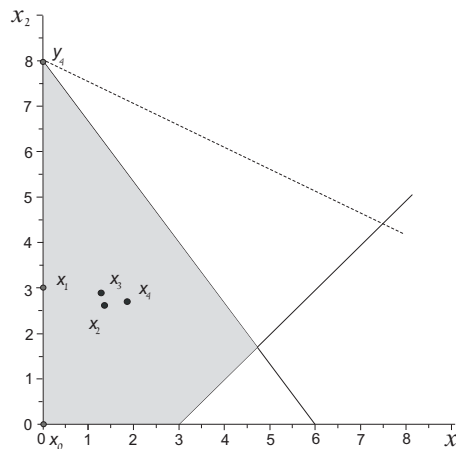


Figura 4.11: Determinando la dirección factible y_4 desde x_4

el vector $\mathbf{d}_4 = \mathbf{y}_4 - \mathbf{x}_4 = (0, 8) - (1.8102, 2.76902) = (-1.8102, 5.2309)$. en el siguiente paso buscamos el punto mínimo entre \mathbf{x}_4 y \mathbf{y}_4 .

2. Cálculo del siguiente punto de iteración: \mathbf{x}_5

Realizar la búsqueda lineal el tamaño del paso α el cual dará el valor mínimo de la función f .

$$\begin{aligned} \mathbf{x}_4 + \alpha \mathbf{d}_4 &= (1.8102, 2.76902) + \alpha(-1.8102, 5.2309) \\ &= (1.8102 - 1.8102\alpha, 2.76902 + 5.2309\alpha) \end{aligned}$$

entonces la función a minimizar será:

$$f(\mathbf{x}_4 + \alpha \mathbf{d}_4) = f(1.8102 - 1.8102\alpha, 2.76902 + 5.2309\alpha)$$

Ahora resolvemos

$$\begin{aligned} \text{mín} \quad & f(1.8102 - 1.8102\alpha, 2.76902 + 5.2309\alpha) \\ \text{s.a.} \quad & 0 \leq \alpha \leq 1 \end{aligned}$$

La solución es $\alpha^* = 0.03889$. Por lo tanto la ecuación (4.13), el nuevo punto de iteración es

$$\begin{aligned} \mathbf{x}_5 &= \mathbf{x}_4 + \alpha \mathbf{d}_4 \\ &= (1.8102, 2.76902) + 0.03889(-1.8102, 5.2309) \\ &= (1.7398, 2.97245). \end{aligned}$$

3. Criterio de evaluación para finalizar. Calcular el estado de terminación después de lo cual no la prueba de razón de la mejora relativa de la iteración de punto nuevo. Entonces

$$\begin{aligned} \frac{f(\mathbf{x}_4) - f(\mathbf{x}_5)}{|f(\mathbf{x}_5)|} &= \frac{(-776.13) - (-782.43)}{|-782.43|} \\ &= 0.00805 \leq \varepsilon = 0.01 \end{aligned}$$

para el propósito del ejemplo $\varepsilon = 0,01$. pues la relación es menor que $\varepsilon = 0,01$ al cumplirse la desigualdad terminamos el proceso y la solución se da en \mathbf{x}_5 y el valor de la función es $f(\mathbf{x}_5) = -782.43$.

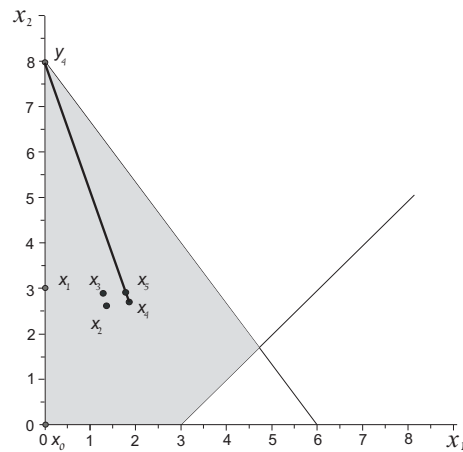


Figura 4.12: Nuevo Punto de Iteración x_5

En el ejemplo se consideró la tolerancia $\varepsilon = 0,01$ con lo cual se obtiene el punto $\mathbf{x} = (1.7398, 2.97245)$ donde la función objetivo toma el valor $f(\mathbf{x}) = -782.43$.

Si se considera una tolerancia mas pequeña, se requiere realizar más iteraciones, así, si consideramos por ejemplo $\varepsilon = 0.0003$ se obtiene tiene la solución con mayor precisión, así en el punto $\mathbf{x} = (3.074011, 2.9752)$ es donde la función objetivo toma el valor $f(\mathbf{x}) = -811.22$.

5

Implementación Computacional

5.1. Introducción

Debido al tamaño de la red en la situación de la vida real, el algoritmo propuesto debe ser implementado por una computadora. En esta sección se muestra la descripción general de la entrada de código de computadora junto con los requisitos y la estructura de las subrutinas, así para el desarrollo de ejemplos es esta tesis se implementa el algoritmo en Matlab

5.2. Implementación del modelo

Las herramientas para la elaboración de las rutinas de los modelos son diversas, en el presente trabajo la codificación esta elaborado en scripts de Matlab. La implementación computacional de los algoritmos representa la culminación de los procesos de estudio y análisis realizados para la elaboración del modelo, y la ejecución de estas rutinas permiten visualizar los resultados buscados.

La función objetivo del problema de asignación es una función no lineal. El algoritmo de Franck Wolfe es un algoritmo que permite trabajar con este tipo de funciones, en cada iteración del algoritmo la función no lineal es linealizada, obteniéndose así un nuevo problema cuya función es lineal, este nuevo problema se resuelve usando el método simplex mejorado, el siguiente paso del algoritmo de Franck-Wolfe es determinar el nuevo valor de la función objetivo, debido a que la región en la que se trabaja es una región convexa es facil encontrar el valor extremo, para asegurar la convergencia se usa el método de la bisección.

Las funciones y subrutinas elaborados son scripts desarrollados en Matlab cuyos archivos se guarda con extensión ".m". En el proceso de ejecución estos scripts devuelven valores, las variables cambian con su ejecución. La figura (5.2) muestra la relación entre los archivos de Matlab

Se describen los scripts desarrollados, en la cual el script principal es el archivo wolfenson.m. Los valores inicializados en cada script se modifican en cada iteración

que realizan el programa, hasta la obtención del valor óptimo.

El problema a resolver es de la forma:

$$\begin{array}{ll} \min & f(\mathbf{x}) \\ \text{sujeto a:} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in X \end{array}$$

- **Datos iniciales**

Los iniciales de ingreso: la matriz A , el vector \mathbf{b} , el vector de valores iniciales \mathbf{v}_0 y las tolerancias $tol1$ y $tol2$.

- **wolfenson.m**

Es el programa principal, que requiere de los datos iniciales, luego hace uso de los sub programas, que le permite linealizar la función, luego encontrar la dirección de desplazamiento, luego la longitud de este, y encontrar los nuevos valores de la función objetivo, verificando que cumplen las condiciones de tolerancia con la cual finaliza. dando como resultado el vector óptimo.

- **deriva1.m**

Este subprograma es llamado por el programa principal y para determinar y evaluar la derivada de función.

- **simplex-revisado.m**

con el problema ya linealizado este programa determina la dirección óptima de desplazamiento,

- **raizfw.m**

este sub programa determina la longitud de desplazamiento, y verifica la convergencia, entrega como resultado el vector de valores óptimos

Los códigos de los archivos se presentan en el apéndice.

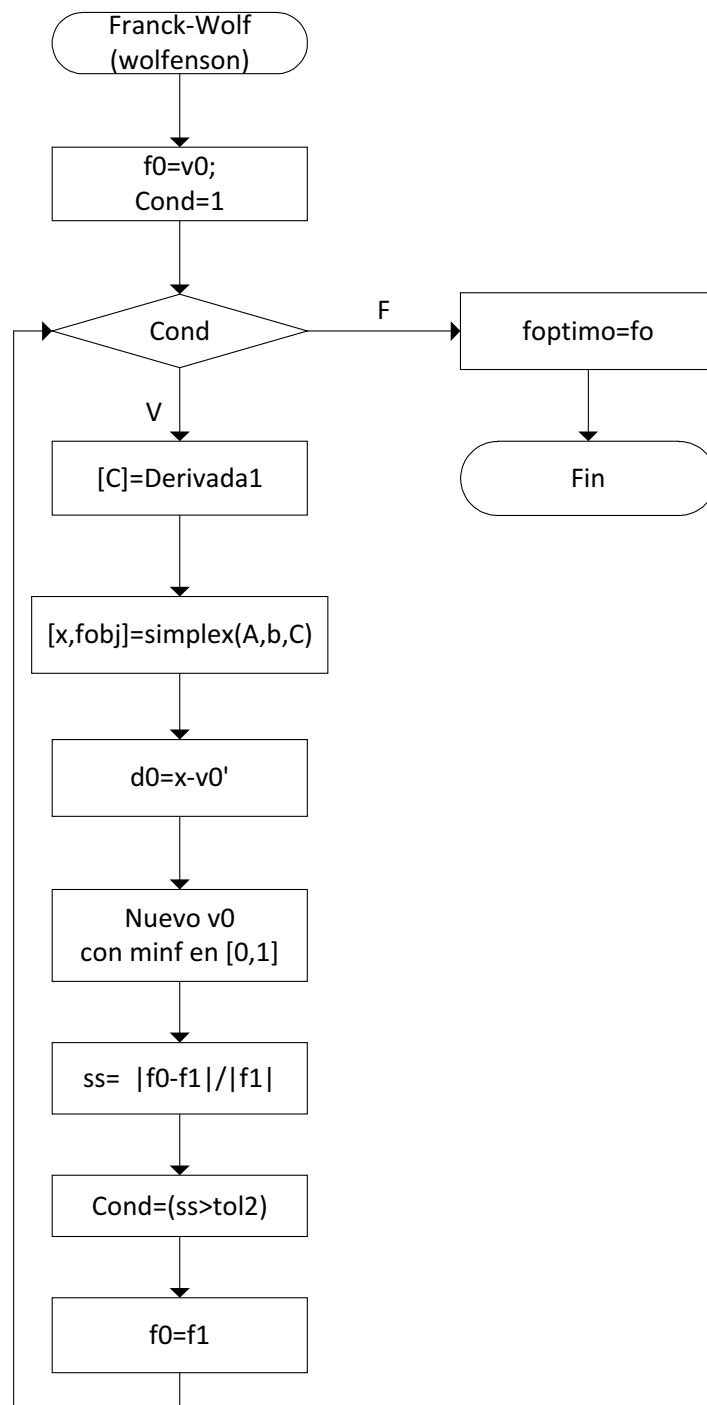


Figura 5.1: Diagrama de Flujo

5.3. Aplicación

Problema 1

Considérese el siguiente problema, una ciudad con una vía de circunvalación y varias rutas centrales como se ilustra en la figura (5.2). Imagínese que se realizan 4000 viajes

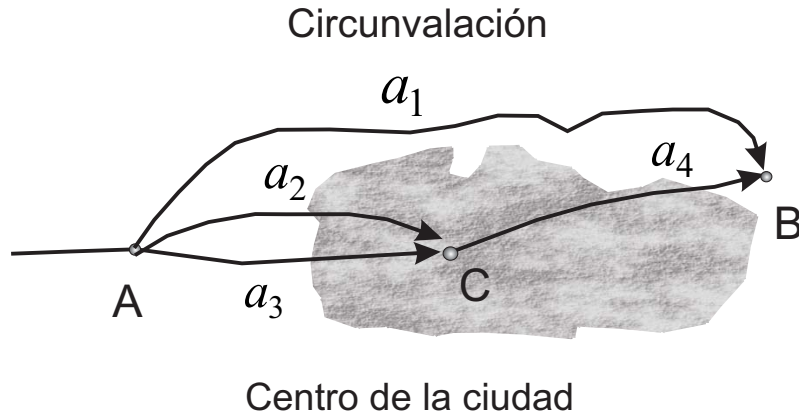


Figura 5.2: Red vial

desde A hasta B y 2500 desde A hasta C . Las rutas disponibles para satisfacer la demanda del par $w_1 = (A, B)$ son $r_1 = \{a_1\}$, $r_2 = \{a_2, a_4\}$, y $r_3 = \{a_3, a_4\}$, y las rutas para el par $w_2 = (A, C)$ son $r_4 = \{a_2\}$ y $r_5 = \{a_3\}$. En este ejemplo $W = \{w_1, w_2\}$, y $\mathcal{R}_{w_1} = \{r_1, r_2, r_3\}$ y $\mathcal{R}_{w_2} = \{r_4, r_5\}$. Las variables de flujo en los caminos son h_1, \dots, h_5 , y las variables de flujo en los arcos son f_1, \dots, f_4 . Como

$$\begin{aligned} \int_0^{f_a} C_a(x) dx &= \int_0^{f_a} c_a^0 + b_a \left(\frac{x}{k_a} \right)^{n_a} dx \\ &= c_a^0 f_a + \frac{b_a}{n_a + 1} \left(\frac{f_a}{k_a} \right)^{n_a + 1} \end{aligned}$$

La formulación completa es como sigue. Minimizar la función

$$Z = \sum_{i=1}^4 \int_0^{f_{a_i}} C_{a_i}(x) dx = \sum_{a \in A} c_{a_i}^0 f_{a_i} + \frac{b_{a_i}}{n_{a_i} + 1} \left(\frac{f_{a_i}}{k_{a_i}} \right)^{n_{a_i} + 1} \quad (5.1)$$

sujeto a

$$h_1 + h_2 + h_3 = 4000 \quad (5.2)$$

$$h_4 + h_5 = 2500 \quad (5.3)$$

$$h_1 = f_1 \quad (5.4)$$

$$h_2 + h_4 = f_2 \quad (5.5)$$

$$h_3 + h_5 = f_3 \quad (5.6)$$

$$h_2 + h_3 = f_4 \quad (5.7)$$

$$h_1 \dots h_5, f_1 \dots f_4 \geq 0 \quad (5.8)$$

y el valor óptimo de la función es: $f_{optimo} = 45813.316$

En este ejemplo se han usado las funciones de la BPR dada en la ecuación (4.3), y la tabla (5.1) muestra los parámetros. La solución correspondiente al problema se muestra en la tabla (5.4) la cual indica los flujos asignados a cada arco respectivo y los flujos en las rutas correspondientes, estos datos obtenidos verifican las condiciones establecidas. por ejemplo $a_2 = r_3 + r_5$

Enlaces	k_a	c_a^0	b_a	n_a
1	500	5	1	4
2	400	7	1	4
3	400	10	1	4
4	500	2	1	4

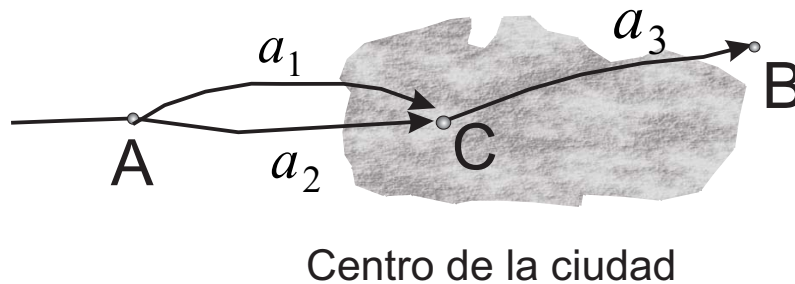
Tabla 5.1: Parámetros para las funciones BPR(Problema 1)

Flujo en los arcos (f_i)	Flujo en las rutas (h_i)	
a_1	3845.884	r_1 3845.883
a_2	2354.464	r_2 54.164
a_3	299.651	r_3 99.951
a_4	154.116	r_4 2300.299
		r_5 199.700

Tabla 5.2: Estimación de los viajes(Problema 1)

Problema 2

Considérese el *problema 1*, a cuya ciudad se le extrae la vía de circunvalación. ver figura(5.3).



Centro de la ciudad

Figura 5.3: Red vial (Problema 2)

Imagínese que se realizan 4000 viajes desde A hasta B y 2500 desde A hasta C . Las rutas disponibles para satisfacer la demanda del par $w_1 = (A, B)$ son $r_1 = \{a_1, a_2\}$, y $r_2 = \{a_2, a_3\}$, y las rutas para el par $w_2 = (A, C)$ son $r_3 = \{a_1\}$ y $r_4 = \{a_3\}$. En este ejemplo $W = \{w_1, w_2\}$, y $\mathcal{R}_{w_1} = \{r_1, r_2\}$ y $\mathcal{R}_{w_2} = \{r_1, r_3\}$. Las variables de flujo en los caminos son h_1, \dots, h_4 , y las variables de flujo en los arcos son f_1, \dots, f_3 .

La formulación es como sigue. Minimizar la función

$$Z = \sum_{i=1}^3 \int_0^{f_{a_i}} C_{a_i}(x) dx = \sum_{a \in A} c_{a_i}^0 f_{a_i} + \frac{b_{a_i}}{n_{a_i} + 1} \left(\frac{f_{a_i}}{k_{a_i}} \right)^{n_{a_i} + 1} \quad (5.9)$$

sujeto a

$$h_1 + h_2 = 4000 \quad (5.10)$$

$$h_3 + h_4 = 2500 \quad (5.11)$$

$$h_1 + h_3 = f_1 \quad (5.12)$$

$$h_2 + h_4 = f_2 \quad (5.13)$$

$$h_1 + h_2 = f_3 \quad (5.14)$$

$$h_1 \dots h_4, f_1 \dots f_3 \geq 0 \quad (5.15)$$

Enlaces	k_a	c_a^0	b_a	n_a
2	400	7	1	3
3	400	10	1	3
4	500	2	1	3

Tabla 5.3: Parámetros para las funciones BPR(Problema 2)

Flujo en los arcos(f_i)	Flujo en las rutas (h_i)
a_1	6499.999 r_1 3999.999
a_2	0.000 r_2 0.000
a_3	4000.000 r_3 2499.999
	r_4 0.000

Tabla 5.4: Estimación de los viajes(Problema 2)

y el valor óptimo de la función es $f_{optimo} = 71956.250$

Podemos observar que valor óptimo obtenido en problema 2 es mucho mayor que el obtenido en el problema 1, esta diferencia nos muestra que si no se considera la via de circunvalación el costo de viaje en la red es alto,

6

Conclusiones y Recomendaciones

6.1. Conclusiones

El algoritmo de Frank-Wolfe es una combinación del método del gradiente con un método de optimización lineal, por lo que es adecuado para la asignación de viajes en redes de transporte ya que estos últimos son un problema de optimización no lineal.

Para trabajar una matriz de asignación de viajes, el algoritmo resuelve un origen contra varios destinos lo que quiere decir que se debe de aplicar el algoritmo para cada origen nuevamente.

El método Simplex revisado permite resolver el óptimo del dominio local del problema ya que el comportamiento local no lineal y el comportamiento local lineal son similares.

El algoritmo de Frank-Wolfe es más fácilmente automatizable que otros algoritmos como el de multiplicadores de Lagrange.

Es especialmente ventajoso en los problemas de asignación de tráfico porque no necesita enumerar todas las rutas factibles entre los orígenes y los destinos, que podría ser un trabajo muy pesado cuando el tamaño de la red aumenta.

Cuando se compara una red con vía de evitamiento con otra sin vía de evitamiento el óptimo de la función de costo aumenta en 26000 unidades monetarias diarios lo que indica 780000 unidades monetarias más al mes y 9360000 unidades monetarias más al año, esto justifica la inclusión de vías de evitamiento en una red de transporte.

6.2. Recomendaciones

Para mejorar la rapidez de convergencia del método, se debe de probar con algoritmos de punto interior en vez del método simplex ya que este tiene una convergencia exponencial en el peor de los casos.

Se debe de hacer estudios sobre las funciones BPR en cada arco de la red de manera que consideren sentidos de subida, bajada ya que con esto se tendrían resultados más

aproximados a nuestro medio.

Se recomienda un estudio variacional de la formulación del problema ya que de esta manera se puede aproximar mejor las soluciones. Existen estudios como del Doctor Pedro Canales que dan ideas al respecto [6].

Se debe probar mejoras del método Frank-Wolfe con combinaciones del gradiente conjugado con métodos de punto interior como los de Merotra [8].

Apéndice

Programa Wolfenson-1

Los scripts correspondientes a los programas se presentan a continuación

Listing 1: Wolfenson-1.m

```
1
2
3 %Datos requeridos
4 %la matriz A, vector b, el vector inicial v0, y la tolerancias tol1, tol2
5 %
6 function [v0, foptimo]=Wolfenson_1(A,b,v0,tol1,tol2)
7 f0=f(v0);
8 cond=1;
9 while cond
10     C=derivada_1(v0);
11     [x, ff]=simplex_revisado(C,[],[],A,b,zeros(9,1),[]);
12     d0=x-v0';
13     [alfa, f1, v0]=raiz_wolfenson(v0',d0,0,1,tol1);
14     ss=abs(f0-f1)/abs(f1);
15     cond=(ss>tol2); f0=f1;
16 end
17 foptimo=f0;
```

Archivo simplex-revisado

Listing 2: simplex-revisado.m

```
1
2 function [CB, fobjetivo, posicion, b]=simplex_revisado(C,A,b)
3 [B,X,CT,n,nnb]=arranque_simplex(C,A,b); warning off
4 [CB,invB,CNB,Q, posicion, pos_entra]=inicializa_simplex(B,X,CT,C,A);
5 condicion=0;E=invB;
6 while condicion==0
7 bi=invB*A(:, pos_entra); b=E*b;
8 [pos_sale]=basesale_simplex(b, bi, nnb);
9 [CT]=intercambia_simplex(CT, pos_sale, pos_entra); CB=CT(1, length(C)+1:length(CT));
10 [posicion]=intercambia_simplex(posicion, pos_sale, pos_entra);
11 [invB, A, E]=matrices(pos_sale, pos_entra, nnb, n, A, invB);
12 Q=CB*invB; CNB=Q*A-CNB;
13 [num, pos_entra]=max(CNB); condicion=(max(CNB)>0);
14 end
15 b=E*b; fobjetivo=CB*b; X=[zeros(1, length(C)), b'];
```


Archivo raiz-wolfenson.m

Listing 3: raiz-wolfenson.m

```
1
2 function [ alfa , f1 , x0]=raiz_wolfenson(x0,d0, alfa1 , alfa2 , tol) %método de la bisección
3 xe1=x0+alfa1*d0;
4 xe2=x0+alfa2*d0;
5 condicion =1;
6 while condicion
7     alfap=alfa1+(alfa2-alfa1)/2;
8     xep=x0+alfap*d0;
9     p=derivada_1(xe1)*d0*derivada_1(xep)*d0;
10    if p>0
11        xe1=xep;
12        alfa1=alfap;
13    else
14        xe2=xep;
15        alfa2=alfap;
16    end
17    condicion=(abs( alfa2-alfa1)>tol);
18 end
19 alfa=alfap;
20 f1=f(x0+alfa*d0);
21 x0=(x0+alfa*d0)';
```

Archivo intercambia-simplex

Listing 4: intercambia-simplex.m

```
1
2 function [V]=intercambia_simplex(V, pos1 , pos2)
3 pivot=V(pos1);V(pos1)=V(pos2);
4 V(pos2)=pivot;
```

archivo basesale-simplex

Listing 5: basesale-simplex.m

```
1
2 function [ pos_sale]=basesale_simplex(b, ai , nmb)
3 division=b./ ai;[num, pos_sale]=min( division); pos_sale=pos_sale+nmb;
4 function valor=f(x)
5 valor=5*x(1)+0.2*(0.002*x(1))^5+7*x(2)+0.2*(0.0025*x(2))^5+10*x(3)+0.2*(0.0025*x(3))^5+
6 2*x(4)+0.2*(0.002*x(4))^5+zeros(1,5)*[x(5);x(6);x(7);x(8);x(9)];
7 %valor=3*x(1)^2-24*x(1)+2*x(2)^4-64*x(2);
```

Archivo datosincial

Listing 6: datosincial.m

```
1
2 % Matriz A
3 A=[0 0 0 0 1 1 1 0 0
4     0 0 0 0 0 0 0 1 1
5     -1 0 0 0 1 0 0 0 0
6     0 -1 0 0 0 1 0 1 0
7     0 0 -1 0 0 0 1 0 1
8     0 0 0 -1 0 1 1 0 0 ]
9
10 % Vector b
11 b=[4000;2500;0.0;0.0;0.0;0.0]
12
13 % Vector inicial
14 v0=[2000 2500 2000 2000 2000 1000 1000 1500 1000]
15
16 % Tolerancias
17 tol1=0.000000001;
18 tol2=0.000000001;
```

Bibliografía

- [1] Mokhtar S. Bazaraa, Hamif D. Sherali. “Nonlinear Programming Theory and Algorithms”, Third Edition. Georgia Institute of Technology.
- [2] Bar-Gera H., D. Boyce (2005) User-equilibrium route set analysis of a large road network, in *Transportation and Traffic Theory: Flow, Dynamics and Human Interaction*, H. Mahmassani (ed), Elsevier, Oxford, 673-692
- [3] Bureau of Public Roads (BPR) (1964), *Traffic Assignment Manual*, U. S. Department of Commerce, Washington DC.
- [4] Ceder, A y Wilson, N.H.M. *Bus Network desing*. “Transportation Research”.
- [5] Pedro Canales Garcia, “Redes de Transporte Urbano: Un Algoritmo para Estimar una Matriz de Demanda de Viajes”. *TECNIA* 12(2) 2004
- [6] Pedro Canales Garcia, “Asignación de Flujos de Tránsito a redes de Transporte Urbano”. *TECNIA* Vol. 13 No 2 Diciembre 2003
- [7] Dantzig G. B.- *Application of the Simplex Method to a Transportation Problem*. Monografía n.º 13 de la Cowles Commission. John Wiley, New York 1951
- [8] Flores Salinas, José, “Estudio del Algoritmo Proyectivo de Karmarkar y sus Aplicaciones en la Ingeniería”. Facultad de ciencias UNI, Lima Perú 2010
- [9] Antonio Galvao Novales. “Sistemas de Transportes” Vol1 Análise da Demanda. Editorial Edgard Blucher Ltda.1986
- [10] Hestenes, Magnus R.; Stiefel, Eduard (December, 1952). “Methods of Conjugate Gradients for Solving Linear Systems”. *Journal of Research of the National Bureau of Standards* 49 (6).
- [11] Hiller, F. J. Liberman. “Investigacion de operaciones”, México, McGraw-Hill, 2002.
- [12] D. G. Luenberger, *Linear and Nonlinear Programming*, Second Edition, John Wiley and Sons, 1984.
- [13] Ortuzar, J. De D. Y Willumnsen, L. “Modelling Transport”, John Wiley and Sons, Inc., 1996

- [14] Russo F. “Modelli e metodi per la programmazione dei servizi di trasporto pubblico locale: uno stato dell’arte” . Franco Angeli Editore Et. Al 2002.
- [15] Vargas G. “Modelos lineales en Investigación de Operaciones, Teoría y Aplicaciones” . Bogotá: Fondo Editorial, Universidad Distrital Francisco José Caldas, Bogotá Colombia ,1990 Et. Al 2002.
- [16] H. Yang, S. Yagar, Y. lida, and Y. Asakura. An algorithm for the inflow control problem on urban freeway networks with user-optimal flows. Transportation Research, 28B:123-139, 1994