

# DISEÑO DE REDES DE COMUNICACIONES CON INCERTIDUMBRE EN LA DEMANDA

Julián Alberto Viera Méndez

Proyecto de Grado  
Ingeniería en Computación

Facultad de Ingeniería  
Universidad de la República

Montevideo, Marzo de 2004

## ***Resumen***

El problema de diseño óptimo de una red multicommodity tiene importantes aplicaciones en los campos de planificación de redes de transporte y distribución de mercancías y de diseño topológico de redes de computadoras y de telefonía.

Este trabajo está centrado en la planificación del diseño inicial de una red dorsal de comunicaciones, en la cual el parámetro básico que condiciona el diseño viene dado por la incertidumbre en la demanda inicial de los servicios de la red.

La red se modela como un grafo cuyas aristas representan un conjunto de enlaces potenciales, y las decisiones relevantes del problema consisten en elegir de forma óptima los enlaces de comunicación que efectivamente se instalarán, minimizando la suma de los costos fijos de instalación y los costos de flujo ó transmisión de los datos. Estos últimos resultan ser variables aleatorias dependientes de la distribución de probabilidad de la demanda de servicios inicial.

La técnica de programación matemática utilizada para la optimización bajo incertidumbre es la programación estocástica, en cuyo contexto se formulan los modelos y las metodologías de resolución, se investigan propiedades teóricas del problema y se evalúa la implementación computacional de distintas instancias de diseño.

# Tabla de Contenido

1	Introducción.....	1
2	Problema Determinístico de Diseño de Red.....	5
2.1	Problema Determinístico Sin Restricciones de Capacidad.....	6
2.1.1	Formulación del Modelo Determinístico del Problema UNDP.....	6
2.1.2	Propiedades del Problema UNDP.....	7
2.1.3	Formulación Alternativa del Problema UNDP.....	9
2.1.4	Formulación del Problema Dual del UNDP.....	10
2.1.5	Técnicas de Resolución del Problema UNDP.....	11
2.1.5.1	Branch & Bound estándar.....	11
2.1.5.2	Branch & Bound basado en Heurística Lagrangeana.....	12
2.1.5.3	Método de Ascenso Dual.....	13
2.1.5.4	Método de Descomposición de Benders.....	16
2.2	Problema Determinístico Con Restricciones de Capacidad.....	17
2.2.1	Formulación del Modelo Determinístico del Problema CNDP.....	17
2.2.2	Propiedades del Problema CNDP.....	18
2.2.3	Técnicas de Resolución del Problema CNDP.....	18
2.2.3.1	Branch & Bound estándar.....	18
2.2.3.2	Branch & Bound basado en Heurística Lagrangeana.....	18
2.2.3.3	Búsqueda Tabú con Vecindades basadas en Ciclos.....	20
3	Conceptos Básicos de Programación Estocástica.....	25
3.1	Problema Lineal en Dos Etapas con Recurso Fijo.....	25
3.2	Conjuntos Factibles.....	26
3.3	Formulación Extensiva del Modelo Estocástico.....	26
3.4	Valor del Modelo Estocástico.....	27
3.4.1	Problema de Distribución.....	27
3.4.2	Valor Esperado de la Información Perfecta.....	27
3.4.3	Problema de Valor Esperado.....	28
3.4.4	Relaciones de Valor Básicas.....	29
3.4	Técnicas de Resolución del Problema Estocástico.....	29
3.4.1	Dualización del Problema de Segunda Etapa.....	30
3.4.2	Método L.....	30
3.4.3	Método de Cortes Múltiples.....	33
3.4.4	Método L Entero.....	34
3.4.5	Método de Cortes Múltiples Entero.....	36
4	Problema Estocástico de Diseño de Red.....	39
4.1	Problema Estocástico Sin Restricciones de Capacidad.....	39
4.1.1	Formulación Extensiva del Problema SUNDP.....	39
4.1.2	Formulación del Modelo en Dos Etapas del Problema SUNDP.....	40
4.1.3	Propiedades del Modelo en Dos Etapas del Problema SUNDP.....	41
4.1.4	Factibilidad del Modelo en Dos Etapas del Problema SUNDP.....	42

4.1.5	Existencia de la Solución Óptima del Problema SUNDP .....	42
4.1.6	Aplicación del Método L Entero al Problema SUNDP.....	43
4.1.7	Valor de la Solución Estocástica en el Caso del Problema SUNDP.....	46
4.2	Problema Estocástico Con Restricciones de Capacidad .....	48
4.2.1	Formulación Extensiva del Problema SCNDP.....	48
4.2.2	Formulación del Modelo en Dos Etapas del Problema SCNDP.....	49
4.2.3	Propiedades del Modelo en Dos Etapas del Problema SCNDP.....	50
4.2.4	Factibilidad del Modelo en Dos Etapas del Problema SCNDP.....	50
4.2.5	Existencia de la Solución Óptima del Problema SCNDP .....	51
4.2.6	Aplicación del Método de Cortes Múltiples Entero al SCNDP.....	52
4.2.7	Valor de la Solución Estocástica en el Caso del Problema SCNDP.....	55
5	Implementación de Casos de Prueba de Diseño.....	56
5.1	Implementación en GAMS – OSL.....	56
5.1.1	Caso de Diseño 1.....	57
5.1.2	Caso de Diseño 2.....	63
5.1.3	Caso de Diseño 3.....	66
5.2	Implementación de la Heurística de Ascenso Dual.....	70
5.2.1	Aplicación de la Heurística de Ascenso Dual a los Casos de Diseño .....	71
6	Conclusiones.....	73
7	Referencias.....	75
	Anexo A Ejemplo de Aplicación del Método L Entero al SUNDP .....	76
	Anexo B Valor de la Solución Estocástica para el SCNDP.....	79
	Anexo C Formulación GAMS del Modelo Estocástico Extensivo.....	83
C.1	Estructuras de Datos en GAMS .....	83
C.2	Ejemplo de Formulación del Modelo en GAMS .....	84
C.3	Codificación GAMS del Problema de Factibilidad de Diseño de Red.....	87
	Anexo D Codificación GAMS y Soluciones OSL de Casos.....	89
D.1	Codificación GAMS de los Casos de Prueba.....	89
D.2	Soluciones del Solver OSL de los Casos de Prueba .....	93
	Anexo E Heurística de Ascenso Dual .....	96
E.1	Ejemplo de Aplicación de la Heurística de Ascenso Dual al UNDP.....	96
E.2	Código Fuente C++ del Programa que Implementa la Heurística.....	99
E.3	Archivos de Entrada y Salida de la Heurística en los Casos de Prueba .....	107

# 1 Introducción

El problema de diseño inicial óptimo de una red de comunicación de datos, ya sea una red de computadoras o una red telefónica, puede verse como una aplicación particular del problema de diseño óptimo de una red multicommodity (multimercancía).

Una red multicommodity se modela naturalmente como un grafo, en el que se tienen nodos, enlaces (aristas ó arcos) entre nodos y commodities (mercancías) que fluyen desde ciertos nodos origen a ciertos nodos destino.

Cada commodity se genera en un nodo origen y se consume en un nodo destino, y tiene asociada como atributo un nivel dado de requerimiento o demanda a satisfacer.

Los nodos de la red puede ser origen ó destino de commodities, ambas cosas a la vez o ninguna. En este último caso se denominan nodos de tránsito o transbordo.

Las aristas entre dos nodos tienen varias propiedades: un costo fijo de instalación, un costo variable unitario de transporte dependiente en principio del tipo de commodity y eventualmente una cierta capacidad máxima de transporte.

En la Figura 1.1 se muestra el esquema topológico básico de una red multicommodity:

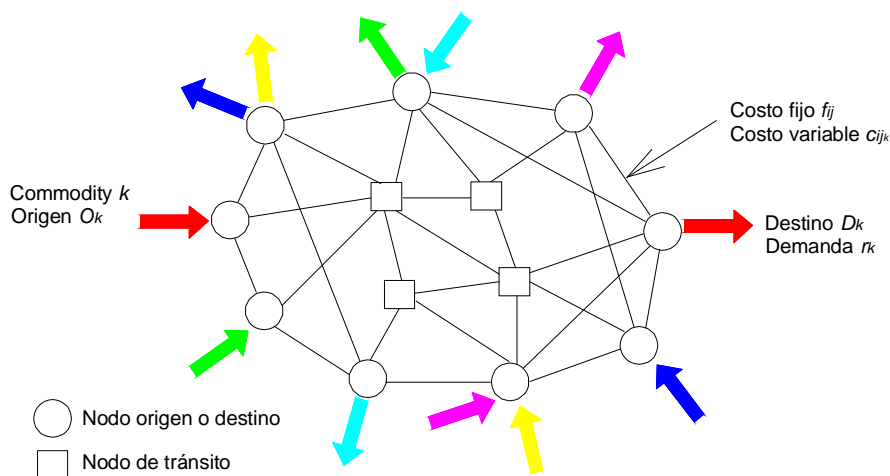


Figura 1.1 Esquema de Red Multicommodity

El problema de diseño óptimo de red consiste en seleccionar un cierto subconjunto de nodos y aristas de la red potencial y determinar los flujos de cada commodity en cada una de las aristas elegidas, de forma que:

- ❑ Se asegure la satisfacción de la demanda de todas las commodities.
- ❑ Se minimice el costo total suma de los costos fijos de las aristas más los costos de flujo por la red de las commodities

Es claro que el diseño óptimo contendrá todos los nodos que sean orígenes ó destinos de commodities ó ambas cosas a la vez. Los nodos de tránsito pueden ó formar parte de la solución óptima, en caso afirmativo se les denomina "nodos de Steiner" de la red.

En la Figura 1.2 aparece una posible solución óptima dada por las aristas en rojo y todos los nodos vinculados a alguna cualquiera de ellas .

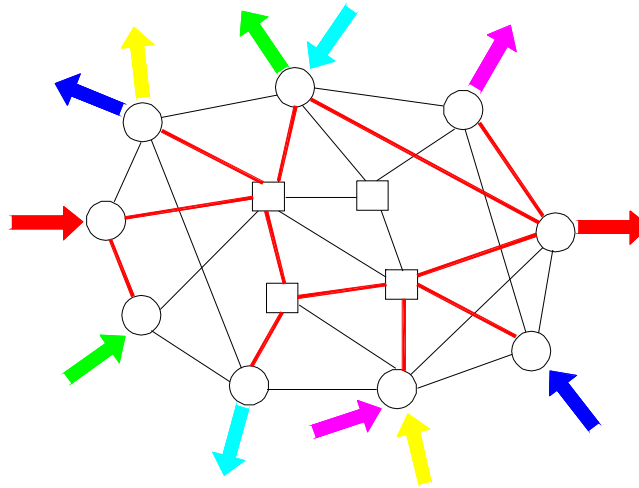


Figura 1.2 Solución del problema de diseño de Red

Se observa que una condición necesaria que debe cumplir el diseño solución es que debe haber camino entre todo par (origen, destino) de commodities.

Si la red no tiene restricciones de capacidad de transporte, los flujos de cada commodity siguen el camino de costo mínimo sobre la red óptima desde origen a destino.

Si en cambio cada arista tiene una capacidad máxima de transporte, la distribución de los flujos de las commodities en las aristas de la red óptima se obtiene como la solución de un problema de flujo multicommodity sobre dicha red óptima (multicommodity network flow problem).

La red multicommodity puede modelar una variedad de problemas de optimización bien conocidos:

- ❑ Problema del Camino mas Corto (Shortest Path). En este caso se tiene una sola commodity que fluye desde el nodo origen al nodo destino, con costos fijos de cada arco iguales al largo del arco y con costos variables nulos.
- ❑ Problema del Arbol de Cubrimiento Mínimo (MST, Minimum Spanning Tree). Se tiene un solo origen de commodities y una commodity con destino en cada nodo de la red distinto del origen. Los costos fijos de los arcos representan el largo de cada arco y los costos variables son nulos.
- ❑ Problema de Steiner. Similar al MST pero con la inclusión de nodos de tránsito que no son origen ni destino de commodities.

El modelo también se emplea para el análisis, planificación y diseño de redes de transporte y distribución de mercancías. Un ejemplo es el caso de una empresa con una única planta de fabricación de un producto y un conjunto dado de clientes, que quiere optimizar la cantidad y ubicación de sus centros de distribución y determinar que clientes se atenderán desde los mismos. Este universo puede modelarse con la red siguiente:

- ❑ Se tiene un único origen de commodities en un nodo que representa a la planta de fabricación.

- ❑ Los nodos destino son los clientes finales de la empresa
- ❑ Los nodos de tránsito son nodos potenciales que simbolizan las posibles ubicaciones de los centros de distribución del producto.
- ❑ Existe un arco entre el nodo origen y cada uno de los nodos de tránsito, cuyo costo fijo esta dado por el costo de instalación y operación del centro de distribución y cuyo costo de transporte unitario representa el costo interno de distribución de la empresa. Este costo depende de la distancia entre la planta y el centro, siendo por ejemplo uno de sus componentes el costo de combustible.
- ❑ Existen arcos entre nodos de tránsito y nodos destino, y entre los nodos destino entre sí. Dicho arcos no tienen asignado costo fijo y sí tienen un costo de transporte unitario que modela el costo de distribución final.
- ❑ Las commodities son los pedidos de cada cliente y la demandas asociadas son los tamaños de dicho pedidos.

En el contexto del diseño de una red de comunicación de datos, la red multicommodity sirve por ejemplo para modelar un universo real con las siguientes características:

- ❑ Los nodos orígenes y destino se identifican con los *servidores* o *hosts*
- ❑ Los nodos de tránsito representan los *enrutadores* o *routers*
- ❑ Las aristas son los posibles enlaces o canales de comunicación (links) a instalar, constituidos por ejemplo por cables de fibra óptica
- ❑ Los costos fijos asociados a los enlaces son los costos de instalación del cable (zanjado) y el costo propio del cable y equipos necesarios (por ejemplo repetidores)
- ❑ Los costos variables unitarios reflejan los costos unitarios de utilización del ancho de banda del canal, son los costos de transmisión de 1 bit en cada canal.
- ❑ La capacidad máxima asignada a cada arista se interpreta como el ancho de banda finito del canal de comunicación, medido por ejemplo en Mbps.
- ❑ Las commodities son los paquetes de datos (bits) a transmitir entre hosts
- ❑ La demanda de cada commodity es el ancho de banda que ocupa el paquete de datos transmitido.

El modelado natural del problema de diseño de redes de comunicación toma en cuenta la capacidad finita de los enlaces. El modelado sin restricciones de capacidad es útil cuando los canales son de gran ancho de banda, como por ejemplo enlaces de fibra óptica de alta tasa de transmisión de bits.

El esquema básico del modelo presentado admite múltiples variaciones y extensiones:

- ❑ Modelado como grafo dirigido ó no dirigido.
- ❑ Limitaciones en la cantidad de flujo de cada commodity que puede ir por cada arco. Esto se impone para asegurar la existencia de más de un camino desde el nodo origen al nodo destino de la commodity, implicando la supervivencia de las conexiones frente a la contingencia de falla en un enlace.
- ❑ Posibilidad de tener más de un costo fijo asociado a cada arco, dependiendo de la capacidad del enlace a instalar. En redes de comunicación se emplea por ejemplo en los estudios de ampliación de la red cuando se necesita más capacidad de transmisión y se quieren añadir mas cables de fibra a los tendidos existentes. Se tiene así una estructura de costos fijos escalonada dependiendo del número de fibras a añadir y de su capacidad.
- ❑ Restricciones en el diseño de la topología de la red: árbol, anillo, etc.

- ❑ Restricciones en el número de saltos (hops) que pueden existir entre origen y destino.
- ❑ Restricciones en el número de enlaces en los que puede participar un nodo (grado).
- ❑ Asignación de costos a los nodos de tránsito potenciales.
- ❑ Incertidumbre en la demanda de las commodities.

La idea de este trabajo es investigar el problema de diseño de red en el caso en el que la demanda inicial de las commodities no es conocida con precisión, sino que se puede modelar como una variable aleatoria con una distribución de probabilidad conocida. Se trabaja con un número finito de escenarios discretos de demanda, cada uno con una cierta probabilidad de ocurrencia. La aleatoriedad se limita al monto o nivel de la demanda, asumiéndose fijos los orígenes y destinos de cada commodity. Se impone que la solución encontrada sea factible para todos los escenarios de demanda definidos, a diferencia de otras alternativas de modelado en que se consideran penalizaciones por no satisfacer demandas en ciertos escenarios. No se modela la dinámica del tráfico por la red.

Para formular el problema se emplea el modelo básico de la red multicommodity, considerándose ambos casos con y sin restricciones de capacidad en los enlaces, y enfocándose el diseño hacia una red dorsal (“backbone network”), con un número relativamente reducido de nodos.

Como metodología para resolver el problema de optimización en presencia de incertidumbre se utiliza la *programación estocástica*. Esta técnica minimiza el valor esperado del costo de la red, tomando en cuenta todos los escenarios de demanda posibles y proporcionando un diseño de red que en general no es óptimo para ninguna de las realizaciones individuales de dichos escenarios, pero cuyo costo tiene un valor esperado inferior al costo esperado que se obtiene diseñando la red con la demanda determinística de cualquiera de los escenarios de demanda. Uno de los principales objetivos planteados es investigar si el enfoque estocástico del problema tiene ó no valor agregado respecto a solucionar el problema determinístico en el que cada commodity tiene asignada su demanda media, resultado que es conocido como el *valor de la solución estocástica*.

La estructura de desarrollo elegida es la siguiente:

- ❑ En el Capítulo 2 se formulan los problemas de diseño con y sin restricciones para el caso de demanda determinística. Se investigan las propiedades de dichos problemas y se describen técnicas de solución, tanto exactas como basadas en heurísticas.
- ❑ En el Capítulo 3 se presentan los conceptos básicos de la teoría de programación estocástica, y se describen técnicas generales de solución del problema estocástico.
- ❑ En el Capítulo 4 se realiza la formulación del problema estocástico de diseño de red en sus dos versiones con y sin restricciones de capacidad, se estudian propiedades de las soluciones de dichos problemas, en particular el valor de la solución estocástica, y se adaptan los algoritmos de solución del problema estocástico general al problema de diseño de red con incertidumbre en la demanda.
- ❑ El Capítulo 5 está dedicado a la resolución práctica y análisis de resultados de casos de prueba de diseño de red, utilizando por un lado un solver comercial estándar e implementando una técnica heurística para el caso sin restricciones de capacidad.
- ❑ Finalmente, en el Capítulo 6 se presentan conclusiones provenientes tanto del análisis teórico como de las instancias de diseño generadas.



## 2 Problema Determinístico de Diseño de Red

Se considera el problema de diseño de una red representada por un grafo dirigido  $G = (\mathbf{N}, \mathbf{A})$  siendo  $\mathbf{N}$  el conjunto de nodos y  $\mathbf{A}$  el conjunto de arcos entre nodos de  $\mathbf{N}$ , empleada para el tráfico de un conjunto de commodities ó mercancías  $\mathbf{K}$ .

Cada commodity  $\mathbf{k} \in \mathbf{K}$  tiene asociado un nodo origen  $\mathbf{o}(\mathbf{k}) \in \mathbf{N}$ , un nodo destino  $\mathbf{d}(\mathbf{k}) \in \mathbf{N}$  y una demanda o nivel de requerimiento a enviar desde el nodo origen hacia el nodo destino denotada como  $\mathbf{r}^{\mathbf{k}}$ .

El emplear un arco  $(\mathbf{i}, \mathbf{j}) \in \mathbf{A}$  tiene un costo fijo  $\mathbf{f}_{\mathbf{ij}}$  independiente del tipo y cantidad de commodity que circule por él, y representa el costo de instalación del tramo.

Se define el conjunto de costos fijos  $\mathbf{F} = \{ \mathbf{f}_{\mathbf{ij}} \mid (\mathbf{i}, \mathbf{j}) \in \mathbf{A} \}$

En caso de que el arco  $(\mathbf{i}, \mathbf{j})$  sea ya existente, se modela tomando  $\mathbf{f}_{\mathbf{ij}} = 0$ .

El costo de transportar una unidad de flujo de la commodity  $\mathbf{k}$  en el arco  $(\mathbf{i}, \mathbf{j}) \in \mathbf{A}$  es  $\mathbf{c}_{\mathbf{ij}}^{\mathbf{k}}$ , y se supone que dicho costo se incrementa linealmente con la cantidad transportada.

Se define el conjunto de costos variables  $\mathbf{C} = \{ \mathbf{c}_{\mathbf{ij}}^{\mathbf{k}} \mid (\mathbf{i}, \mathbf{j}) \in \mathbf{A}, \mathbf{k} \in \mathbf{K} \}$

El problema de diseño de red es un problema de optimización que consiste en diseñar una red seleccionando nodos de  $\mathbf{N}$  y arcos de  $\mathbf{A}$  y eligiendo los flujos de las commodities en cada arco, de forma tal que se satisfaga la demanda de todas las commodities al menor costo total posible. Se formula como un problema de programación entera mixta (MIP, mixed-integer programming), en el que hay variables binarias asociadas a la decisión de usar ó no arcos potenciales y hay variables continuas que representan los flujos de las commodities en cada arco.

Se consideran dos modelos de trabajo, según existan ó no restricciones en la capacidad de transporte de los arcos de la red.

En el caso de no existir límite de capacidad para el tráfico de mercancías en cada arco, se tiene el problema de diseño de red sin restricciones de capacidad UNDP( $\mathbf{N}, \mathbf{A}, \mathbf{K}, \mathbf{F}, \mathbf{C}$ ) (Uncapacitated Network Design Problem).

Si cada arco  $(\mathbf{i}, \mathbf{j})$  tiene asociado un límite o capacidad máxima  $\mathbf{u}_{\mathbf{ij}}$  para el flujo total de mercancías, se define el conjunto de capacidades  $\mathbf{U} = \{ \mathbf{u}_{\mathbf{ij}} \mid (\mathbf{i}, \mathbf{j}) \in \mathbf{A} \}$

En este caso se plantea el problema de diseño de red con restricciones de capacidad CNDP( $\mathbf{N}, \mathbf{A}, \mathbf{K}, \mathbf{F}, \mathbf{C}, \mathbf{U}$ ) (Capacitated Network Design Problem).

## 2.1 Problema Determinístico Sin Restricciones de Capacidad

### 2.1.1 Formulación del Modelo Determinístico del Problema UNDP

Se definen las siguientes variables:

$y_{ij}^k$  = flujo de la commodity  $k$  en el arco  $(i, j)$

$$x_{ij} = \begin{cases} 1 & \text{si se usa el arco } (i, j) \\ 0 & \text{en otro caso} \end{cases}$$

El modelo matemático del problema queda:

$$[\text{UNDP}] z = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k$$

$$\text{sujeto a } \sum_{j:(i,j) \in A} y_{ij}^k - \sum_{j:(j,i) \in A} y_{ji}^k = b_i^k \quad \forall i \in N, \quad \forall k \in K \quad (1)$$

$$y_{ij}^k \leq r^k x_{ij} \quad \forall (i, j) \in A, \quad \forall k \in K \quad (2)$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4)$$

siendo

$$b_i^k = \begin{cases} r^k & \text{si } i = o(k) \\ -r^k & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Las restricciones (1) expresan el balance de flujo que debe existir para cada nodo y para cada commodity.

Las restricciones (2) imponen que si el flujo de una commodity cualquiera en un arco es distinto de cero, dicho arco debe estar activo pagando un costo fijo y además el flujo circulante no puede ser superior a la demanda de esa commodity.

Las restricciones (3) establecen la no negatividad de los flujos en cada arco.

Las restricciones (4) imponen el carácter binario de la variable de uso de los arcos.

### 2.1.2 Propiedades del Problema UNDP

- 1) Se tiene que UNDP es un problema de programación lineal entera mixta, con  $|A|$  variables binarias,  $|A||K|$  variables continuas,  $|N||K|$  restricciones de balance y  $|A||K|$  restricciones de activación de arcos.
- 2) El tamaño del modelo crece muy rápidamente con el tamaño del problema. Por ejemplo un problema con 100 nodos, 1000 arcos y 1000 commodities tiene 1 millón de variables.
- 3) Si todos los costos y las demandas son enteras, las variables de flujo  $y_{ij}^k$  y por lo tanto el valor de la función objetivo  $z$  toman valores enteros. Esto se cumple pues la matriz de restricciones para el conjunto de restricciones (1) es una matriz de incidencias de arcos en un grafo dirigido y por lo tanto es totalmente unimodular (TUM).

#### 4) Teorema 1 UNDP es NP-completo

Se puede demostrar que UNDP es NP-completo reduciendo polinomialmente el problema de Steiner en grafos (STPG) a él.

El STPG (Steiner Problem in Graphs) se define como sigue:

Sea  $G = (V, E)$  un grafo conexo no dirigido en donde  $V$  representa el conjunto de nodos y  $E$  el conjunto de las aristas de  $G$ . Se tiene una función de costos no negativos  $W$  definida sobre  $E$ , y un conjunto de nodos terminales  $T \mid T \subseteq V$ . El problema de Steiner en grafos  $STPG(V, E, W, T)$  consiste en hallar un subgrafo conexo de  $G$  de costo mínimo que cubra todos los nodos terminales en  $T$ . La solución del STPG es un árbol, llamado árbol de cubrimiento mínimo de Steiner, y a los nodos de dicho árbol que no pertenecen a  $T$  se les llama nodos de Steiner.

Formulados como problemas de decisión, se demostrará que toda instancia del STPG: dada una cierta cota  $B > 0$ , ¿es posible encontrar un árbol de Steiner de costo  $\leq B$ ? puede reducirse a una instancia del problema de decisión UNDP: dada una cota  $B > 0$ , ¿puede hallarse una selección de arcos de la red y una distribución de flujo en dichos arcos de manera de satisfacer la demanda con costo total  $\leq B$ ?

Como primer paso se observa que  $UNDP \in NP$ .

Dada una conjetura que establece una asignación de arcos  $x_{ij}$  y una distribución de flujos de cada commodity  $y_{ij}^k$ , se puede comprobar en tiempo polinómico si la conjetura proporciona una solución factible de costo  $\leq B$ , lo que muestra que  $UNDP \in NP$ .

Sea ahora una instancia dada  $STPG(V, E, W, T)$  del problema de Steiner. A partir de ella, se construye una instancia correspondiente del  $UNDP(N, A, K, F, C)$  como sigue:

- El conjunto de vértices  $N$  del UNDP es el mismo que el del STPG, es decir  $N \equiv V$ .
- El conjunto de arcos  $A$  se construye colocando dos arcos, uno en cada sentido, entre todo par de nodos de las aristas de  $E$ .
- Sea  $t \in T$  uno cualquiera de los nodos terminales de STPG. Se elige  $t$  como nodo origen de todas las commodities de  $K$ , y por cada nodo terminal distinto de  $t$  se tiene una commodity con ese destino, es decir  $|K| = |T| - 1$ ,  $t = o(k) \forall k \in K$  y  $d(k) \in T - \{t\} \forall k \in K$ .

- Se considera demanda unitaria para todas las commodities, es decir  $r^k = 1 \quad \forall k \in K$ .
- El conjunto  $F$  de costos fijos se forma asignándole a cada arco el costo de la arista correspondiente en el STPG, es decir que  $f_{ij} = f_{ji} = w_{ij} \quad \forall w_{ij} \in W$ .
- El conjunto de costos variables es vacío ( $C = \emptyset$ ) en la instancia del UNPD.

Es fácil ver que la transformación de una instancia dada del STPG en la instancia del UNPD así definida es computable en tiempo polinómico.

Hay que probar ahora que toda instancia SI del problema STPG implica una instancia SI del problema UNPD correspondiente y recíprocamente.

Directo: Sea un árbol de Steiner de costo  $C \leq B$  solución del STPG. Sobre la instancia correspondiente del UNPD, selecciono todos los arcos que forman parte de los caminos dirigidos desde el nodo origen  $t$  a cada uno de los nodos destino y cuya arista correspondiente en el STPG pertenece al cubrimiento de Steiner dado. Estos caminos dirigidos existen y son únicos por ser un árbol la solución del STPG y por incluir ésta a todos los nodos terminales. Se tiene que dicha elección de las variables de decisión  $x_{ij}$ , conjuntamente con la elección  $y_{ij}^k = 1$  para los arcos del camino dirigido de cada commodity  $k$ , verifica:

- a) Es solución del UNDP, pues se satisface la demanda de todas las commodities.
- b) Tiene exactamente el mismo costo  $C \leq B$  que la instancia del STPG que lo originó dado que los arcos del UNDP tienen asociados los mismos costos que las aristas del STPG.

De a) y b) se deduce que es una instancia SI del UNDP como se quería probar.

Recíproco: Dada una solución del UNDP de costo  $C \leq B$ , se considera el subgrafo no dirigido formado por las aristas del STPG que tienen una arista correspondiente formando parte de la solución del UNDP. Este subgrafo verifica:

- a) Incluye a todos los nodos terminales de  $T$ , pues la solución del UNDP incluye a todos los nodos destino y al origen y estos por la transformación definida son los terminales de  $T$ .
- b) Es conexo, pues en la solución del UNDP hay camino dirigido desde el nodo origen a todo nodo destino y por lo tanto en dicho subgrafo hay camino no dirigido entre todos los nodos del mismo.
- c) No tiene ciclos, pues la solución del UNDP ó no genera un subgrafo con ciclos ó si así fuera siempre es posible encontrar una solución del UNDP sin ciclos con costo  $< C$ . Esto implica que el subgrafo es un árbol.
- d) El costo del subgrafo es exactamente el de la solución de UNDP, es decir  $C \leq B$ , pues las aristas correspondientes en la solución del UNDP y en el subgrafo tienen igual costo asociado.

De a), b), c) y d) se obtiene que el subgrafo así obtenido es una instancia SI del STPG, lo que completa la demostración.

Como Karp [12] ha demostrado que el STPG es NP-completo, se concluye entonces que el UNDP es NP-completo. □

Como referencia del esquema formal de este tipo de demostración se tiene a Garey [5].

### 2.1.3 Formulación Alternativa del Problema UNDP

Teniendo en cuenta que no hay restricciones de capacidad de transporte en los arcos de la red, se pueden escalar las variables de flujo dividiéndolas entre la demanda correspondiente, de forma de que la tarea ahora es enviar *una unidad* de cada commodity entre su origen y destino. Para que el problema siga siendo equivalente deben multiplicarse los costos unitarios  $c_{ij}^k$  por las demandas  $r^k$  y trabajar entonces con costos totales de transporte por commodity y por arco.

Las variables  $y_{ij}^k$  representan ahora la *fracción* de la demanda total de la commodity  $k$  que circula por el arco  $(i, j)$ , por lo que su valor está acotado al intervalo  $[0, 1]$ .

La ventaja que tiene esta formulación consiste en que se disminuye el número de parámetros del modelo, el cual queda:

$$[\text{UNDP}] z = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k - \sum_{j:(j,i) \in A} y_{ji}^k = b_i^k \quad \forall i \in N, \quad \forall k \in K \quad (1)$$

$$y_{ij}^k \leq x_{ij} \quad \forall (i, j) \in A, \quad \forall k \in K \quad (2)$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4)$$

siendo

$$b_i^k = \begin{cases} 1 & \text{si } i = o(k) \\ -1 & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

y donde los  $c_{ij}^k$  representan ahora y como ya se dijo el producto de los costos unitarios originales por las demandas de la commodities, es decir los costos totales.

Observar que en esta formulación "normalizada" se dividen las restricciones (1) y (2) entre la demanda  $r^k$ , por lo que es aplicable siempre que las demandas sean no nulas.

### 2.1.4 Formulación del Problema Dual del UNDP

Se establecerá el problema dual del UNDP en base a la formulación alternativa precedente.

Se definen las variables duales  $\pi_i^k$  y  $\beta_{ij}^k$  asociadas a las restricciones (1) y (2)

respectivamente.

Haciendo la relajación lagrangeana, se tiene el problema relajado [UNDPLR]:

$$zI = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k + \sum_{k \in K} \sum_{i \in N} \pi_i^k \left( \sum_{j:(i,j) \in A} y_{ij}^k - \sum_{j:(j,i) \in A} y_{ji}^k - b_i^k \right) + \sum_{k \in K} \sum_{(i,j) \in A} \beta_{ij}^k (y_{ij}^k - x_{ij})$$

sujeto a

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (1)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (2)$$

siendo

$$b_i^k = \begin{cases} 1 & \text{si } i = o(k) \\ -1 & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Reagrupando términos, la función objetivo puede escribirse como:

$$zI = \min_{x,y} \sum_{k \in K} \sum_{(i,j) \in A} (\beta_{ij}^k + c_{ij}^k + \pi_i^k - \pi_j^k) y_{ij}^k + \sum_{(i,j) \in A} (f_{ij} - \sum_{k \in K} \beta_{ij}^k) x_{ij} - \sum_{k \in K} \sum_{i \in N} \pi_i^k b_i^k$$

El problema relajado tiene solución finita si los coeficientes de las variables  $y_{ij}^k$  y  $x_{ij}$  son no negativos, es decir si:

$$\beta_{ij}^k + c_{ij}^k + \pi_i^k - \pi_j^k \geq 0 \quad \text{y} \quad f_{ij} - \sum_{k \in K} \beta_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K$$

y la solución queda simplemente  $y_{ij}^k = x_{ij} = 0, \quad \forall (i, j) \in A, \quad \forall k \in K$

$$\text{con valor óptimo } zI_{\text{opt}} = - \sum_{k \in K} \sum_{i \in N} \pi_i^k b_i^k$$

El problema dual DUNDP puede entonces expresarse como:

$$[\text{DUNDP}] \quad z d_{\pi, \beta} = \max \sum_{k \in K} \pi_{d(k)}^k$$

sujeto a

$$\pi_j^k - \pi_i^k \leq \beta_{ij}^k + c_{ij}^k \quad \forall (i, j) \in A, \quad \forall k \in K \quad (1)$$

$$\sum_{k \in K} \beta_{ij}^k \leq f_{ij} \quad \forall (i, j) \in A \quad (2)$$

$$\beta_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (3)$$

En esta formulación se eligieron las  $\pi_{o(k)}^k = 0 \quad \forall k$ . Se tiene esta libertad pues las restricciones primales correspondientes son redundantes.

### **Propiedad del Problema Dual**

Una característica interesante del problema DUNPD es que, para valores fijos y factibles de las variables  $\beta_{ij}^k$  (es decir que satisfacen la restricción (2)), las variables  $\pi_{d(k)}^k$  representan el costo del camino mas corto entre  $\mathbf{o}(k)$  y  $\mathbf{d}(k)$ , empleando como costo de cada arco el valor  $\beta_{ij}^k + c_{ij}^k$ . Esto indica que para dichos valores fijos DUNPD se descompone en IKI problemas de longitud de camino mínima.

Esta propiedad se emplea en la técnica de resolución del problema primal llamada Método de Ascenso Dual.

### **2.1.5 Técnicas de Resolución del Problema UNDP**

Se describirán someramente una serie de técnicas aplicables para solucionar el UNDP.

#### **2.1.5.1 Branch & Bound estándar**

Se basa en resolver en cada paso un programa lineal relajado (PL), relajando las condiciones de integralidad de las variables  $x_{ij}$ .

En cada nodo del árbol binario se resuelve el problema lineal para las variables  $x_{ij}$  que todavía no han sido fijadas. Se tienen los siguientes casos:

- 1) La solución del subproblema relajado PL es entera. Si es menor que la menor solución entera hallada hasta el momento, se transforma en la solución de referencia (incumbente). El nodo queda vaciado (bounding).
- 2) No hay solución factible de PL. El nodo queda vaciado.
- 3) La solución de PL no es entera y es mayor o igual que la menor solución entera hallada hasta el momento. Esto implica que el problema no relajado no puede alcanzar una solución mejor a partir de dicho nodo. El nodo queda vaciado.
- 4) La solución de PL no es entera y es menor que la menor solución entera hallada hasta el momento. Se elige un arco  $x_{ij}$  para hacer la ramificación (branching) y de ese nodo sale una rama correspondiente a  $x_{ij} = 1$  y otro a  $x_{ij} = 0$ .

Ventajas:

- Es un método ampliamente implementado y disponible en paquetes estándar de optimización, por ejemplo CPLEX ó OSL.
- Proporciona una solución exacta del problema

Desventajas:

- No es un método eficiente para problemas UNDP grandes. Aparecen problemas de degeneración al resolver el problema PL en dichos casos.

Como referencia de este método puede citarse el texto de Nemhauser [15]

### 2.1.5.2 Branch & Bound basado en Heurística Lagrangeana

Se hará una síntesis de esta técnica según aparece en Holmberg y Hellstrand [8]

En este caso, en cada nodo del árbol binario del Branch & Bound, no se resuelve el problema lineal relajando las condiciones de integralidad sino que se hace una relación lagrangeana de las restricciones de balance de flujo del problema (restricciones (1) en el UNPD).

La idea básica es resolver el problema dual de esta relajación, empleando el método del subgradiente, el cual involucra una heurística para obtener soluciones primales factibles.

Llamando  $\omega_i^k$  a los multiplicadores de Lagrange de las restricciones de balance, el problema lagrangeano dual a resolver puede expresarse como

$$[\text{LD}] \quad z_D = \max_{\omega} \sum_{(i,j) \in A} g_{ij}(\omega) + \sum_{k \in K} \sum_{i \in N} \omega_i^k b_i^k$$

donde para cada arco  $(i, j)$  se tiene el subproblema dual

$$[\text{SD}] \quad g_{ij}(\omega) = \min_{x,y} f_{ij} x_{ij} + \sum_{k \in K} (c_{ij}^k + \omega_j^k - \omega_i^k) y_{ij}^k$$

$$\text{sujeto a} \quad y_{ij}^k \leq r^k x_{ij} \quad \forall k \in K$$

$$y_{ij}^k \geq 0 \quad \forall k \in K$$

$$x_{ij} \in \{0, 1\}$$

Es decir que cada subproblema [SD] contiene una variable binaria y  $|K|$  variables continuas.

El problema dual [LD] se optimiza con el método del subgradiente, en el cual, dada una solución  $\bar{y}_{ij}^k$  del problema [SD], se obtiene una dirección del subgradiente de la función dual como

$$\xi_i^k = b_i^k - \sum_{j: (i,j) \in A} \bar{y}_{ij}^k + \sum_{j: (j,i) \in A} \bar{y}_{ji}^k \quad \forall i \in N, \quad \forall k \in K$$

En cada iteración, los multiplicadores de Lagrange se actualizan según

$$\omega_i^{k(n+1)} = \omega_i^{k(n)} + t^{(n)} \xi_i^k \quad \forall i \in N \quad \forall k \in K$$

$$\text{siendo el tamaño del paso } t^{(n)} = \lambda \frac{\bar{z} - z_D(\omega^{(n)})}{\|\xi^{(n)}\|^2}$$

y donde  $0 < \lambda < 2$  y  $\bar{z}$  es una cota superior del problema primal.

Para hallar  $\bar{z}$  se considera una asignación fija de las variables binarias de decisión  $x_{ij}$  que define una subred dada por  $\bar{A} = \{(i,j) \in A \mid x_{ij} = 1\}$ .

Dicha asignación resulta de una solución  $\bar{x}$  de los subproblemas duales [DS], y se obtiene un

$$\text{valor óptimo del UNPD resultante como } z_P = \sum_{(i,j) \in \bar{A}} f_{ij} + \sum_{k \in K} z_{pk}(\bar{x})$$



donde para cada commodity  $k \in K$  se resuelve el siguiente subproblema primal SP:

$$[\text{SP}] \quad z_{PK}(\bar{X}) = \min_y \sum_{(i,j) \in \bar{A}} c_{ij}^k y_{ij}^k$$

sujeto a

$$\sum_{j:(i,j) \in \bar{A}} y_{ij}^k - \sum_{j:(j,i) \in \bar{A}} y_{ji}^k = b_i^k \quad \forall i \in N$$

$$y_{ij}^k \leq r^k \quad \forall (i, j) \in \bar{A}$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in \bar{A}$$

Puede verse que el hallar una solución factible del primal queda entonces equivalente a resolver  $|K|$  problemas de camino de costo mínimo.

En cada nodo del árbol del Branch & Bound, se resuelve el dual hasta alcanzar la solución óptima del mismo ó hasta que se supere un número límite de iteraciones ó no haya cambios significativos en los multiplicadores de Lagrange de una iteración a otra.

Ventajas:

- Es más eficiente que Branch & Bound estándar
- Proporciona una solución exacta del problema en un número finito de pasos

Desventajas:

- No disponible comercialmente, hay que implementarlo
- La eficiencia es muy dependiente de la correcta elección de una serie de parámetros heurísticos en la resolución de los subproblemas.

### 2.1.5.3 Método de Ascenso Dual

Esta es una técnica heurística para resolver el UNDP desarrollada por Balakrishnan, Magnanti y Wong [1].

Para describir el método se seguirá en general el desarrollo de Gendron [6], complementado con elementos de una presentación del método de ascenso dual de Balakrishnan [2].

La idea básica consiste en obtener una rápida solución al problema dual del UNDP, en su formulación alternativa, es decir el problema DUNDP ya presentado:

$$[\text{DUNDP}] \quad z_{d,\pi,\beta} = \max \sum_{k \in K} \pi_{o(k)}^k$$

sujeto a

$$\pi_j^k - \pi_i^k \leq \beta_{ij}^k + c_{ij}^k \quad \forall (i, j) \in A, \quad \forall k \in K \quad (1)$$

$$\sum_{k \in K} \beta_{ij}^k \leq f_{ij} \quad \forall (i, j) \in A \quad (2)$$

$$\beta_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (3)$$

donde se han elegido las  $\pi_{o(k)}^k = 0 \quad \forall k$ .

La propiedad estructural del problema que se explota es que, para valores fijos y factibles de las variables  $\beta_{ij}^k$  (es decir que satisfacen la restricción (2)), las variables  $\pi_{d(k)}^k$  representan el costo del camino de costo mínimo entre  $\mathbf{o}(k)$  y  $\mathbf{d}(k)$ , empleando como costo de cada arco el valor  $\beta_{ij}^k + c_{ij}^k$ .

Se definen:

$h_{ij}^k$  : Holgura para la commodity  $k$  en el arco  $(i, j)$

$$h_{ij}^k = \beta_{ij}^k + c_{ij}^k - \pi_j^k + \pi_i^k \quad \forall (i, j) \in A, \quad \forall k \in K$$

$r_{ij}$  : Costo Fijo Remanente en el arco  $(i, j)$

$$r_{ij} = f_{ij} - \sum_{k \in K} \beta_{ij}^k \quad \forall (i, j) \in A$$

Teniendo en cuenta las restricciones (1) y (2), deben ser  $h_{ij}^k \geq 0, r_{ij} \geq 0$

Un arco  $(i, j)$  de la red se dice que esta saturado para una cierta commodity  $k$  cuando su holgura correspondiente  $h_{ij}^k$  vale 0.

El procedimiento comienza asignando  $\beta_{ij}^k = 0$  y hallando para cada commodity  $k$  el camino de mínimo costo entre su origen y el resto de los nodos, inicializando con estos valores las variables duales  $\pi$ . A continuación se trata de incrementar en forma iterativa los valores de las  $\beta_{ij}^k$ , manteniendo la factibilidad del problema dual. El algoritmo utiliza un sistema de etiquetado de los nodos de la red para actualizar los valores de las  $\pi$  al variar las  $\beta$ , de forma tal que  $\pi_i^k$  representa en cada paso el camino de costo mínimo (con costos  $\beta_{ij}^k + c_{ij}^k$ ) desde  $\mathbf{o}(k)$  hasta el nodo  $i$ . Inicialmente se etiquetan solamente los nodos destinos de cada commodity.

En cada paso se incrementa el camino de largo mínimo con el mínimo de:

- costo fijo remanente mínimo entre arcos saturados con cola no etiquetada y cabeza etiquetada
- holgura mínima entre arcos no saturados con cola no etiquetada y cabeza etiquetada

Adicionalmente, en cada paso se incrementa la función objetivo dual y se decrementan los costos fijos remanentes de los arcos saturados por dicho valor mínimo.

Si quedan arcos con holgura 0 y costo fijo remanente 0 y cola no etiquetada, se etiqueta dicho nodo.

El algoritmo termina cuando todos los nodos orígenes de commodities quedan etiquetados.

Se definen las siguientes estructuras auxiliares:

$N_u(k)$  : Conjunto de nodos sin etiquetar para la commodity  $k$

$A(N_u(k))$  : Arcos con cola no etiquetada y cabeza etiquetada para la commodity  $k$  (cutset)

$$A(N_u(k)) = \{(i, j) \in A \mid i \in N_u(k), j \in N - N_u(k)\}$$

$K_u$  : Conjunto de commodities que son todavía candidatas para la iteración de ascenso dual

El algoritmo puede escribirse como sigue:

Paso 1 Inicialización

$$\beta_{ij}^k = 0 \quad \forall (i, j) \in A, \quad \forall k \in K$$

$\pi_i^k =$  largo de camino de costo mínimo desde  $o(k)$  a  $i$ ,  $\forall i \in N$ ,  $\forall k \in K$  (con costos  $c_{ij}^k$ )

$$r_{ij} = f_{ij} \quad \forall (i, j) \in A$$

$$h_{ij}^k = c_{ij}^k - \pi_j^k + \pi_i^k \quad \forall (i, j) \in A, \quad \forall k \in K$$

$$zd = \sum_{k \in K} \pi_{o(k)}^k$$

Paso 2 Inicialización de nodos no etiquetados

$$N_u(k) = N - \{d(k)\} \quad \forall k \in K$$

$$K_u = K$$

Paso 3 Selección

Elegir una commodity  $k \in K_u$

Paso 4 Iteración de ascenso dual

$$\delta_1 = \min\{r_{ij} / h_{ij}^k = 0, (i, j) \in A(N_u(k))\}$$

$$\delta_2 = \min\{h_{ij}^k / h_{ij}^k > 0, (i, j) \in A(N_u(k))\}$$

$$\delta = \min\{\delta_1, \delta_2\}$$

$$r_{ij} = r_{ij} - \delta \quad \forall (i, j) \in \{(i, j) \in A(N_u(k)) \mid h_{ij}^k = 0\}$$

$$\beta_{ij}^k = \beta_{ij}^k + \delta \quad \forall (i, j) \in \{(i, j) \in A(N_u(k)) \mid h_{ij}^k = 0\}$$

$$h_{ij}^k = h_{ij}^k - \delta \quad \forall (i, j) \in \{(i, j) \in A(N_u(k)) \mid h_{ij}^k > 0\}$$

$$\pi_i^k = \pi_i^k + \delta \quad \forall i \in N - N_u(k)$$

$$zd = zd + \delta$$

Paso 5 Etiquetado

Si  $\delta = \delta_1$ , seleccionar un arco  $(i^*, j^*) \in \{(i, j) \in A(N_u(k)) \mid h_{ij}^k = 0\}$  que verifique  $r_{i^*j^*} = 0$  (existe al menos uno) y establecer  $N_u(k) = N_u(k) - \{i^*\}$

Paso 6 Iteración sobre las commodities

$$K_u = K_u - \{k\}$$

Si  $K_u \neq \emptyset$ , ir al Paso 3.

Paso 7 Test de fin

$$K_u = \{k \in K \mid o(k) \in N_u(k)\}$$

Si  $K_u = \emptyset$ , FIN DEL ALGORITMO, sino ir al Paso 3.

Al terminar el procedimiento, se tiene que la red formada por los arcos con costo fijo remanente 0 es factible para el problema original, es decir que se puede obtener una solución del primal

resolviendo para cada commodity  $\mathbf{k}$  un problema de camino de costo mínimo (con costos  $\beta_{ij}^k + c_{ij}^k$ ) entre  $\mathbf{o}(\mathbf{k})$  y  $\mathbf{d}(\mathbf{k})$  sobre la subred  $A_0 = \{(i, j) \in A \mid r_{ij} = 0\}$ .

La solución del problema dual tiene las siguientes propiedades:

- Proporciona una *cota inferior* para la solución óptima del problema primal
- Da una solución primal factible a partir de la cual se puede aplicar una heurística de optimización local que mejore dicha solución

Ventajas del método:

- Es muy eficiente (algoritmo pseudo-polinómico), proporcionando rápidamente una solución factible cercana a la óptima. Esto es particularmente importante en problemas de diseño de gran escala.

Desventajas:

- No es un método exacto

En el *Anexo E.1* se muestra un ejemplo de aplicación de la heurística para una red sencilla.

#### **2.1.5.4 Método de Descomposición de Benders**

Consiste básicamente en realizar una generación retrasada de restricciones, se va resolviendo el problema de optimización con un subconjunto de las restricciones y se adicionan restricciones (cortes) si no se obtienen soluciones factibles ó si las soluciones halladas no son óptimas.

Se verá en detalle el algoritmo más adelante, en el contexto de los métodos para resolver el problema de diseño sin restricciones de capacidad en arcos cuando hay incertidumbre en la demanda inicial de las commodities, ya que puede verse como un caso particular de uno de dichos métodos.

Un trabajo de referencia sobre la aplicación de Benders al diseño de redes sin restricciones de capacidad es el de Magnanti [14].

## 2.2 Problema Determinístico Con Restricciones de Capacidad

### 2.2.1 Formulación del Modelo Determinístico del Problema CNDP

Se definen las siguientes variables:

$y_{ij}^k$  = flujo de la commodity  $k$  en el arco  $(i, j)$

$$x_{ij} = \begin{cases} 1 & \text{si se usa el arco } (i, j) \\ 0 & \text{en otro caso} \end{cases}$$

El modelo matemático del problema queda:

$$[\text{CNDP}] z = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k$$

$$\text{sujeto a } \sum_{j:(i,j) \in A} y_{ij}^k - \sum_{j:(j,i) \in A} y_{ji}^k = b_i^k \quad \forall i \in N, \quad \forall k \in K \quad (1)$$

$$\sum_{k \in K} y_{ij}^k \leq u_{ij} x_{ij} \quad \forall (i, j) \in A \quad (2)$$

$$y_{ij}^k \leq g_{ij}^k x_{ij} \quad \forall (i, j) \in A, \quad \forall k \in K \quad (3)$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K \quad (4)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (5)$$

siendo

$$g_{ij}^k = \min \{u_{ij}, r^k\} \quad \text{y} \quad b_i^k = \begin{cases} r^k & \text{si } i = \alpha(k) \\ -r^k & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Se asume que todos los coeficientes son enteros y no negativos.

Las restricciones (1) expresan el balance de flujo que debe existir para cada nodo y para cada commodity.

Las restricciones (2) limitan el flujo total por un arco a la capacidad del mismo.

Las restricciones (3) imponen que si el flujo de una commodity cualquiera en un arco es distinto de cero, dicho arco debe estar activo pagando un costo fijo y además el flujo circulante no puede ser superior al mínimo entre la demanda de esa commodity y la capacidad del arco.

Las restricciones (4) establecen la no negatividad de los flujos en cada arco.

Las restricciones (5) imponen el carácter binario de la variable de uso de los arcos.

Este modelo se conoce como formulación "fuerte" o desagregada, pues en rigor dadas las restricciones (2) y la forma de la función objetivo, las restricciones (3) son redundantes, pero las relajaciones del problema que se obtienen si no están presentes las restricciones (3) son más débiles, obteniéndose mejores resultados y eficiencia en el cómputo si están presentes.

## 2.2.2 Propiedades del Problema CNDP

1) Se tiene que CNDP es un problema de programación lineal entera mixta, con  $|A|$  variables binarias,  $|A||K|$  variables continuas,  $|N||K|$  restricciones de balance,  $|A||K|$  restricciones de activación de arcos y  $|A|$  restricciones de capacidad de flujo en arcos.

2) El tamaño del modelo crece muy rápidamente con el tamaño del problema. Por ejemplo un problema con 100 nodos 1000 arcos y 1000 commodities tiene 1 millón de variables.

3) CNDP es NP-completo

CNDP es claramente una generalización del problema UNDP, el cual se demostró que es NP-completo. Por ejemplo, toda instancia del UNDP puede transformarse en una instancia equivalente del CNDP considerando la misma red y asignando como capacidad a todos los arcos la suma de las demandas de las commodities del problema UNDP.

## 2.2.3 Técnicas de Resolución del Problema CNDP

Se describirán someramente una serie de técnicas aplicables para solucionar el CNDP.

### 2.2.3.1 Branch & Bound estándar

Técnica ya descrita para el caso del UNDP.

### 2.2.3.2 Branch & Bound basado en Heurística Lagrangeana

Se hará una síntesis de esta técnica según aparece en Holmberg y Yuan [9].

En este caso, al igual que en el caso anterior del problema UNDP, en cada nodo del árbol binario del Branch & Bound no se resuelve el problema lineal relajando las condiciones de integralidad sino que se hace una relación lagrangeana de las restricciones de balance de flujo del problema (restricciones (1) en el problema CNDP).

La idea básica es resolver el problema dual de esta relajación, empleando el método del subgradiente, el cual involucra una heurística para obtener soluciones primales factibles.

Llamando  $\omega_i^k$  a los multiplicadores de Lagrange de las restricciones de balance, el problema lagrangeano dual a resolver puede expresarse como

$$[\text{LD}] \quad z_D = \max_{\omega} \sum_{(i,j) \in A} h_{ij}(\omega) + \sum_{k \in K} \sum_{i \in N} \omega_i^k b_i^k$$

donde para cada arco  $(i, j)$  se tiene el subproblema dual

$$[\text{SD}] \quad h_{ij}(\omega) = \min_{x,y} f_{ij} x_{ij} + \sum_{k \in K} (c_{ij}^k + \omega_j^k - \omega_i^k) y_{ij}^k$$

$$\begin{aligned} \text{sujeto a} \quad & \sum_{k \in K} y_{ij}^k \leq u_{ij} x_{ij} \\ & y_{ij}^k \leq g_{ij}^k x_{ij} \quad \forall k \in K \\ & y_{ij}^k \geq 0 \quad \forall k \in K \\ & x_{ij} \in \{0, 1\} \end{aligned}$$

siendo 
$$\mathbf{g}_{ij}^k = \min \{ \mathbf{u}_{ij}, \mathbf{r}^k \} \quad \text{y} \quad \mathbf{b}_i^k = \begin{cases} \mathbf{r}^k & \text{si } i = \mathbf{o}(\mathbf{k}) \\ -\mathbf{r}^k & \text{si } i = \mathbf{d}(\mathbf{k}) \\ 0 & \text{en otro caso} \end{cases}$$

Es decir que cada subproblema [SD] contiene una variable binaria y  $|\mathbf{K}|$  variables continuas de flujo.

Asimismo, se tiene que cada uno de los subproblemas [SD] se resuelve fácilmente mediante una técnica Greedy.

El problema dual [LD] se optimiza con el método del subgradiente, en el cual, dada una solución  $\bar{\mathbf{y}}_{ij}^k$  del problema [SD], se obtiene una dirección del subgradiente de la función dual como

$$\xi_i^k = \mathbf{b}_i^k - \sum_{j:(i,j) \in \mathbf{A}} \bar{\mathbf{y}}_{ij}^k + \sum_{j:(j,i) \in \mathbf{A}} \bar{\mathbf{y}}_{ji}^k \quad \forall i \in \mathbf{N}, \quad \forall k \in \mathbf{K}$$

En cada iteración, los multiplicadores de Lagrange se actualizan según

$$\omega_i^{k(n+1)} = \omega_i^{k(n)} + \mathbf{t}^{(n)} \xi_i^k \quad \forall i \in \mathbf{N} \quad \forall k \in \mathbf{K}$$

siendo el tamaño del paso 
$$\mathbf{t}^{(n)} = \lambda \frac{\bar{\mathbf{z}} - \mathbf{z}_D(\omega^{(n)})}{\|\xi^{(n)}\|^2}$$

y donde  $0 < \lambda < 2$  y  $\bar{\mathbf{z}}$  es una cota superior del problema primal.

Para asignar valores iniciales a los multiplicadores  $\omega_i^k$ , se consideran los valores duales correspondientes a la solución del problema de hallar los caminos de mínimo costo en la red desde origen a destino para cada commodity.

Como criterios prácticos del fin de la iteración de subgradiente se emplean:

- 1)  $\|\xi^{(n)}\| < \varepsilon$  ó  $\mathbf{t}^{(n)} < \varepsilon$
- 2)  $\mathbf{n} > \mathbf{M}$  (Límite máximo de iteraciones)
- 3)  $\|\bar{\mathbf{z}} - \mathbf{z}_D(\omega^{(n)})\| < \varepsilon$

Para hallar  $\bar{\mathbf{z}}$  se considera una asignación fija de las variables binarias de decisión  $\mathbf{x}_{ij}$  que define una subred dada por  $\bar{\mathbf{A}} = \{ (i,j) \in \mathbf{A} \mid \mathbf{x}_{ij} = 1 \}$ .

Dicha asignación resulta de una solución  $\bar{\mathbf{x}}$  de los subproblemas duales [DS], y se obtiene un valor óptimo del CNDP como resultado del siguiente subproblema primal SP:

[SP] 
$$\mathbf{z}_P(\bar{\mathbf{x}}) = \min_{\mathbf{y}} \sum_{(i,j) \in \bar{\mathbf{A}}} \mathbf{c}_{ij}^k \mathbf{y}_{ij}^k + \sum_{(i,j) \in \bar{\mathbf{A}}} \mathbf{f}_{ij}$$

sujeto a

$$\begin{aligned} \sum_{j:(i,j) \in \bar{\mathbf{A}}} \mathbf{y}_{ij}^k - \sum_{j:(j,i) \in \bar{\mathbf{A}}} \mathbf{y}_{ji}^k &= \mathbf{b}_i^k \quad \forall i \in \mathbf{N}, \quad \forall k \in \mathbf{K} \\ \sum_{k \in \mathbf{K}} \mathbf{y}_{ij}^k &\leq \mathbf{u}_{ij} \mathbf{x}_{ij} \quad \forall (i,j) \in \bar{\mathbf{A}} \\ \mathbf{y}_{ij}^k &\geq 0 \quad \forall (i,j) \in \bar{\mathbf{A}}, \quad \forall k \in \mathbf{K} \end{aligned}$$

No se incluyen en el subproblema las restricciones de activación de arcos por ser redundantes y no influir ventajosamente en los métodos de resolución del mismo.

Puede verse que el hallar una solución factible del primal queda entonces equivalente a resolver un problema de flujo múltiple de mercancías en una red (multicommodity network flow problem). Este problema se soluciona a través de una heurística ó empleando un solver estándar de programación lineal, como CPLEX.

En cada nodo del árbol del Branch & Bound, las ramificaciones se hacen sobre las variables de diseño binarias de la red  $x_{ij}$ , generándose una rama con  $x_{ij} = 1$  y otra con  $x_{ij} = 0$ .

Se aplica en cada nodo la heurística lagrangeana descrita, generándose cotas inferiores y superiores para la función objetivo en ese punto. Se comienza obteniendo una solución primal factible resolviendo el problema [SP] con todos los arcos de la red que no se hayan fijado con valor 0. Luego se aplica el método del subgradiente para resolver [SD], y periódicamente se utiliza la heurística primal ó CPLEX para mejorar la cota superior en base a información proporcionada por la solución de dual.

En cada nodo puede ocurrir lo siguiente:

- Se poda el árbol debido a que se satisface algún criterio de poda.
- Se ramifica el árbol cuando se da alguna de las condiciones de fin de la iteración de subgradiente

En adición a los tres criterios de poda estándar (solución óptima, inexistencia de soluciones factibles, solución del dual mayor o igual que el incumbente), para este problema en particular se tienen criterios adicionales basados en la relación entre los costos fijos y los variables y en la información dada por la solución primal en cada nodo.

### 2.2.3.3 Búsqueda Tabú con Vecindades basadas en Ciclos

Es un método heurístico desarrollado por Ghamlouche, Crainic y Gendreau [7].

Se parte de la formulación del modelo sin redundancias para el problema:

$$z = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k - \sum_{j:(j,i) \in A} y_{ji}^k = b_i^k \quad \forall i \in N, \quad \forall k \in K$$

$$\sum_{k \in K} y_{ij}^k \leq u_{ij} x_{ij} \quad \forall (i, j) \in A$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A, \quad \forall k \in K$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A$$

siendo

$$b_i^k = \begin{cases} r^k & \text{si } i = \alpha(k) \\ -r^k & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$



La idea básica es explorar el espacio de las variables de diseño de arco  $x_{ij}$  redireccionando el flujo de commodities alrededor de ciclos, y abriendo (activando) y cerrando arcos según corresponda. Se entiende por ciclo al conjunto de arcos pertenecientes a dos caminos dirigidos que tienen el mismo origen y el mismo destino.

Sea el vector  $\bar{x}$  una solución factible del problema, se define  $A(\bar{x})$  como el conjunto de arcos correspondiente a dicha solución. Es decir que  $A(\bar{x})$  contiene los mismo arcos que  $A$  pero cada arco tiene además el atributo de estar abierto o cerrado según la solución  $\bar{x}$ .

La vecindad (neighbourhood) de una solución  $\bar{x}$  se define como:

$$v(\bar{x}) = \{ x \text{ obtenido a partir de } \bar{x} \text{ complementando el estado de cierto número de arcos en base a la desviación de flujo en un ciclo dado de } A(\bar{x}) \}$$

El movimiento de una solución a otra vecina se hace de la siguiente forma:

- 1) Se identifican dos nodos de la red con dos caminos dirigidos que los conectan, formando un ciclo.
- 2) Se desvía el flujo total desde un camino hacia el otro de forma tal que al menos un arco abierto quede sin flujo (vacío).
- 3) Se cierran todos los arcos que luego de la desviación quedan vacíos y se abren los previamente cerrados que ahora tienen flujo.

Los movimientos que define una vecindad toman en cuenta el impacto en el costo total del diseño de los cambios en los flujos de varias commodities a la vez. Se buscan movimientos que cierren al menos un arco y abran nuevos caminos para un grupo de commodities.

Para cerrar un arco, se debe poder desviar todo su flujo a un camino alternativo. Se define capacidad residual de un ciclo al máximo flujo que se puede desviar alrededor del ciclo.

Esta capacidad residual de un ciclo que contenga al arco  $(i, j)$  debe ser por lo menos igual al monto del flujo total en dicho arco dado por  $\sum_{k \in K} y_{ij}^k$ . Esto implica que los ciclos de interés sean aquellos que presenten una capacidad residual igual a alguno de los valores del conjunto de

flujos totales en los arcos de la solución, dado por  $\Gamma(\bar{x}) = \left\{ \sum_{k \in K} y_{ij}^k > 0 \mid (i, j) \in A(\bar{x}) \right\}$ .

Dada una solución  $\bar{x}$ , para cada valor  $\gamma \in \Gamma(\bar{x})$  se deben identificar ciclos dirigidos en la llamada *red  $\gamma$ -residual* de  $A(\bar{x})$ . El ciclo que lleve a la mejora mas grande de la función objetivo del problema corresponde al mejor movimiento a realizar en la vecindad de  $\bar{x}$ .

La red  $\gamma$ -residual de  $A(\bar{x})$  se obtiene sustituyendo cada arco de  $A(\bar{x})$  por a lo sumo dos arcos  $(i, j)^+$  y  $(j, i)^-$ .

Un arco  $(i, j)^+$  es incluido en la red  $\gamma$ -residual de  $A(\bar{x})$  solamente si el arco soporta un flujo adicional de valor  $\gamma$ , es decir que cuenta con una capacidad residual  $u_{ij} - \sum_{k \in K} y_{ij}^k \geq \gamma$ . Se le asigna un costo  $c_{ij}^+$  que aproxima el costo de transporte de las  $\gamma$  unidades adicionales de flujo por el arco  $(i, j)$ , y toma en cuenta el que dicho arco estuviese originalmente cerrado ó no:

$$c_{ij}^+ = \frac{\sum_{k \in K} c_{ij}^k}{|K|} \gamma + f_{ij}(1 - \min(1, \bar{x}_{ij}))$$

Un arco  $(j, i)^-$  no es incluido en la red  $\gamma$ -residual de  $A(\bar{x})$  si su flujo actual es menor que  $\gamma$ , es decir si  $\sum_{k \in K} y_{ij}^k < \gamma$ . En caso contrario se incluye dicho arco y se le asigna un costo  $c_{ji}^-$  que es el promedio ponderado de los costos de transporte de las commodities que fluyen por el arco  $(i, j)$  en la solución  $\bar{x}$ :

$$c_{ji}^- = \begin{cases} -\frac{\sum_{k \in K} c_{ij}^k y_{ij}^k}{\sum_{k \in K} y_{ij}^k} \gamma - f_{ij} & \text{si } \sum_{k \in K} y_{ij}^k = \gamma \\ -\frac{\sum_{k \in K} c_{ij}^k y_{ij}^k}{\sum_{k \in K} y_{ij}^k} \gamma & \text{si } \sum_{k \in K} y_{ij}^k > \gamma \end{cases}$$

Para identificar el ciclo dirigido de mínimo costo que pasa por un cierto arco  $(i, j)$  en la red  $\gamma$ -residual, se aplica una heurística que trata de hallar el camino dirigido de menor costo entre el nodo  $j$  y el nodo  $i$ . Este método etiqueta a cada nodo  $u$  con la distancia mínima  $d(u)$  desde el nodo origen, y en cada iteración elige un nodo  $u$  y para todos los nodos cabeza de arco  $v$  que tienen a  $u$  como cola, les corrige su distancia al origen si  $d(v) > d(u) + c_{uv}$ .

Se mantiene para cada nodo su predecesor en el camino de mínimo costo desde el origen, y se evita también que el procedimiento entre en loop debido a la existencia de ciclos con costo negativo.

Definiendo  $N^+(i) = \{j \mid (i, j) \in \text{a la red } \gamma\text{-residual de } A(\bar{x})\}$ , y siendo  $s$  el origen, el esquema del algoritmo queda:

$$d(s) = 0; \text{ pred}(s) = 0, d(j) = \infty \forall j \neq s; \text{ LISTA} = \{s\}$$

Mientras  $\text{LISTA} \neq \emptyset$

Sacar un elemento  $i$  de  $\text{LISTA}$

Para cada nodo  $j \in N^+(i)$

Si  $(d(j) > d(i) + c_{ij})$  y  $(j$  no está en el camino desde  $s$  a  $i)$

$d(j) = d(i) + c_{ij}$ ;

$\text{pred}(j) = i$ ;

si  $j \notin \text{LISTA}$  entonces  $\text{LISTA} = \text{LISTA} \cup \{j\}$ ;

Fin Si

Fin Para

Fin Mientras

La Búsqueda Tabú investiga el espacio de las variables de diseño de arco  $x_{ij}$  realizando una *búsqueda local* basada en el concepto de vecindad. Para evitar ciclos, se implementa una *lista tabú* utilizando una memoria de corta duración en donde se registran las características de las soluciones recientemente visitadas. En cada iteración, se realiza el mejor movimiento que no esté en la lista tabú, independientemente de si mejora el valor objetivo global ó no.

Cuando se encuentra una solución particularmente buena, se realiza una *intensificación* de la búsqueda, en la cual se considera la distribución de flujo de una sola commodity a la vez.

Una solución se considera particularmente buena cuando da un valor de la función objetivo inferior al de la mejor solución hallada hasta el momento, o difiere de este último menos que un porcentaje predeterminado.

Si la solución hallada por un movimiento de búsqueda local resulta no factible, se realiza una etapa de *restauración* (restoration) para volver a una solución factible.

Se da por terminado el procedimiento cuando se alcanza un criterio predeterminado de finalización, que puede ser por ejemplo un número máximo de iteraciones ó un tope de tiempo de cómputo.

Dada una solución factible  $\bar{x}$ , se define a  $C(\bar{x}) \subseteq A(\bar{x})$  como el conjunto de arcos "candidatos", donde cada arco  $(i, j) \in C$  se considera como posible arco de arranque de ciclos. Para reducir el volumen de cómputo el método restringe  $C$  a un subconjunto de arcos *cerrados* de  $A(\bar{x})$ , generados en forma randómica.

Asimismo, se define a  $C(\gamma) \subseteq C(\bar{x})$  como el subconjunto de arcos de  $C$  que puede soportar un movimiento de flujo de valor  $\gamma$ . Si un arco  $(i, j)$  está cerrado, para pertenecer a  $C(\gamma)$  debe ser  $u_{ij} \geq \gamma$ . Si el arco está abierto, debe tener un flujo total mayor que  $\gamma$  (caso de disminución de flujo) ó una capacidad residual superior a  $\gamma$  (caso de aumento de flujo).

Con estas definiciones, el esquema general del algoritmo de Búsqueda Tabú queda:

### Paso 1 Inicialización

Generar una solución factible. Esto se hace abriendo todos los arcos de la red original, y resolviendo para ese caso un problema de flujo óptimo de red multicommodity MNFP (Multicommodity Network Flow Problem), el cual para una solución dada  $\bar{x}$  se expresa como:

$$[\text{MNFP}] \quad z = \min_y \sum_{k \in K} \sum_{(i,j) \in A(\bar{x})} c_{ij}^k y_{ij}^k$$

$$\text{sujeto a} \quad \sum_{j:(i,j) \in A(\bar{x})} y_{ij}^k - \sum_{j:(j,i) \in A(\bar{x})} y_{ji}^k = b_i^k \quad \forall i \in N, \forall k \in K$$

$$\sum_{k \in K} y_{ij}^k \leq u_{ij} \bar{x}_{ij} \quad \forall (i, j) \in A(\bar{x})$$

$$y_{ij}^k \geq 0 \quad \forall (i, j) \in A(\bar{x}), \forall k \in K$$

siendo

$$b_i^k = \begin{cases} r^k & \text{si } i = o(k) \\ -r^k & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Inicializar los parámetros *SoluciónActual* y *MejorSolución* al valor solución del MNFP.

### Paso 2 Test de Fin

Si se cumple el criterio de parada → FIN DEL ALGORITMO

### Paso 3

Dada la solución actual  $\bar{x}_1$ , construir los conjuntos  $C(\bar{x}_1)$  y  $\Gamma(\bar{x}_1)$ .

#### Paso 4 Determinación del mejor movimiento (ciclo de mínimo costo)

Para cada  $\gamma \in \Gamma(\bar{x}_1)$

Determinar el conjunto  $C(\gamma)$

Construir la red  $\gamma$ -residual

Para cada arco  $(i, j) \in C(\gamma)$  hallar el ciclo dirigido de mínimo costo empleando la técnica heurística en base a etiquetado

Fin Para

Seleccionar el ciclo de menor costo

#### Paso 5 Implementación del movimiento

Moverse desde la solución  $\bar{x}_1$  a una nueva solución  $\bar{x}_2$ , abriendo y cerrando los arcos apropiados según el ciclo de mínimo costo. La solución  $\bar{x}_2$  pasa a ser la solución actual.

Determinar el valor exacto de la solución  $\bar{x}_2$  resolviendo el problema MNFP para la misma.

Actualizar el parámetro *SoluciónActual* al valor solución del MNFP.

#### Paso 6 Actualización de la Lista Tabú

Colocar todos los arcos que en los que difieren  $\bar{x}_1$  y  $\bar{x}_2$  en la Lista Tabú.

Actualizar los tiempos de permanencia de los arcos ya existentes en la Lista Tabú.

#### Paso 7 Restauración

Si la solución  $\bar{x}_2$  no es factible, efectuar una fase de *restauración*

#### Paso 8 Intensificación

Dado el parámetro *IntensGap* que establece el límite para que una solución dada se considere "buena", si  $\frac{\text{SoluciónActual} - \text{MejorSolución}}{\text{MejorSolución}} \leq \text{IntensGap}$  se realiza una fase de *intensificación* de la búsqueda local.

#### Paso 9

Si **SoluciónActual < MejorSolución**  $\rightarrow$  **MejorSolución = SoluciónActual**

Ir al Paso 2.

La implementación del método requiere la calibración adecuada de los siguientes parámetros:

- El tamaño de la vecindad explorada en cada iteración, dado por la dimensión del conjunto  $C$ , o sea el porcentaje de arcos cerrados que se eligen en forma aleatoria.
- El número de iteraciones que los arcos permanecen en la Lista Tabú (tabu tenure)
- El valor de *IntensGap*

### 3 Conceptos Básicos de Programación Estocástica

Para la elaboración de este capítulo se sigue en general el texto de Birge y Louveaux [3] y las Notas del Curso de Optimización bajo Incertidumbre del profesor Carlos Testuri [17].

#### 3.1 Problema Lineal en Dos Etapas con Recurso Fijo

Se denomina problema con recurso a aquel en que ciertas decisiones o acciones correctivas (recurso) pueden tomarse *después* que se dilucida la incertidumbre.

Sea un experimento aleatorio  $\omega$ , que fija los valores del vector de variables aleatorias  $\xi(\omega)$  que modela la incertidumbre del sistema bajo estudio.

Se tiene un conjunto de decisiones  $\mathbf{x}$  que deben tomarse *antes* que ocurra el experimento, llamadas *decisiones de primera etapa*, y el período de tiempo en que se toman se denomina *primera etapa*.

Asimismo hay presentes otro conjunto de decisiones  $\mathbf{y}(\mathbf{x}, \omega)$  que se toman *luego* del experimento, llamadas *decisiones de segunda etapa* y tomadas en el período conocido como *segunda etapa*.

La secuencia en el tiempo de los eventos y las decisiones que se toman es entonces:

$$\mathbf{x} \rightarrow \xi(\omega) \rightarrow \mathbf{y}(\mathbf{x}, \omega)$$

El problema estocástico lineal en dos etapas con recurso fijo puede formularse genéricamente como:

$$\min_{\mathbf{x}} \quad \mathbf{c}^T \mathbf{x} + E_{\xi} Q(\mathbf{x}, \xi) \quad \{\text{Problema de Primera Etapa}\}$$

sujeto a

$$\mathbf{A} \mathbf{x} = \mathbf{b} \quad \{\text{Restricciones determinísticas o de Primera Etapa}\}$$

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

donde

$\mathbf{c}$  es el *vector de costos de primera etapa*

$E_{\xi}$  es la *esperanza matemática con respecto a  $\xi$* .

$$\text{y } Q(\mathbf{x}, \xi(\omega)) = \min_{\mathbf{y}} \quad \mathbf{q}^T(\omega) \mathbf{y} \quad \{\text{Problema de Segunda Etapa}\}$$

sujeto a

$$\mathbf{W} \mathbf{y} = \mathbf{h}(\omega) - \mathbf{T}(\omega) \mathbf{x}$$

$$\mathbf{y} \geq 0$$

y donde dados los eventos aleatorios  $\omega$ , se define  $\xi^T(\omega) = [\mathbf{q}^T(\omega), \mathbf{h}^T(\omega), T_1(\omega), \dots, T_m(\omega)]$  siendo  $T_i(\omega)$  la fila  $i$ -ésima de la matriz  $\mathbf{T}$ .

La matriz  $\mathbf{W}$  es fija y se denomina *matriz de recurso*.

La matriz  $\mathbf{T}$  es llamada *matriz tecnológica*.

El vector  $\mathbf{q}$  es llamado *vector de costos o precios de segunda etapa*

El vector  $\mathbf{h}$  es el *vector de requerimientos*

Llamando función de recurso a  $Q(\mathbf{x}) = E_{\xi}Q(\mathbf{x}, \xi)$  (valor esperado de segunda etapa con decisión de primera etapa  $\mathbf{x}$ ), se obtiene el problema equivalente determinístico (PED):

$$\begin{aligned} \text{[PED]} \quad & \min_{\mathbf{x}} \quad \mathbf{c}^T \mathbf{x} + Q(\mathbf{x}) \\ & \text{sujeto a} \\ & \quad \mathbf{Ax} = \mathbf{b} \\ & \quad \mathbf{x} \geq 0 \end{aligned}$$

### 3.2 Conjuntos Factibles

Se define  $K_1$  como el conjunto de vectores de primera etapa que satisfacen las restricciones determinísticas de primera etapa, esto es  $K_1 = \{\mathbf{x} \mid \mathbf{Ax} = \mathbf{b}\}$  y se lo denomina conjunto factible de primera etapa.

Las decisiones de primera etapa  $\mathbf{x}$  que interesan son no sólo las que satisfacen las restricciones de primera etapa, sino las que además dan un valor finito al problema de segunda etapa para todos los valores que puede tomar la variable aleatoria  $\xi$ . Es así entonces que se define  $K_2 = \{\mathbf{x} \mid Q(\mathbf{x}) < +\infty\}$  y se lo conoce como conjunto factible de segunda etapa.

Para  $\mathbf{x}$  y  $\xi$  fijos, si el problema de segunda etapa  $Q(\mathbf{x}, \xi)$  es no acotado o no factible se lo denota como  $Q(\mathbf{x}, \xi) = +\infty$ .

Se definen los conjuntos factibles elementales como  $K_2(\xi) = \{\mathbf{x} \mid Q(\mathbf{x}, \xi) < +\infty\}$

Como lo que interesa saber es si para un  $\mathbf{x}$  dado el valor de segunda etapa es finito, una definición alternativa a  $K_2$  es el conjunto

$$K_2^P = \{\mathbf{x} \mid \forall \xi \in \Xi, \exists \mathbf{y} \geq 0 \text{ s.a. } \mathbf{Wy} = \mathbf{h}(\xi) - \mathbf{T}(\xi)\mathbf{x}\} = \bigcap_{\xi \in \Xi} K_2(\xi)$$

Si el soporte  $\Xi$  de  $\xi$  es finito, se verifica que  $K_2^P = K_2$

### 3.3 Formulación Extensiva del Modelo Estocástico

En el caso de que haya un número finito de realizaciones del vector aleatorio  $\xi$ , es decir que el mismo tenga soporte  $\Xi$  finito, y al ser lineal la función objetivo y las restricciones, puede plantearse un modelo en el que aparecen en forma explícita las variables de decisión de segunda etapa para todos los escenarios, denominado *forma extensiva* del problema estocástico.

Sea un conjunto de  $S$  escenarios con realizaciones  $\xi_s^T = [\mathbf{q}_s^T, \mathbf{h}_s^T, T_s]$ , con probabilidad  $\mathbf{p}_s$ ,  $s = 1 \dots S$ .

Definiendo las variables de segunda etapa como  $\mathbf{y}_s$ , el modelo extensivo se formula como

$$\min_{\mathbf{x}, \mathbf{y}} \quad \mathbf{c}^T \mathbf{x} + \sum_{s=1}^S \mathbf{p}_s \mathbf{q}_s^T \mathbf{y}_s$$

sujeto a

$$\begin{aligned} & \mathbf{Ax} = \mathbf{b}, \\ & \mathbf{T}_s \mathbf{x} + \mathbf{Wy}_s = \mathbf{h}_s, \quad s = 1, \dots, S, \\ & \mathbf{x} \geq 0, \mathbf{y}_s \geq 0, \quad s = 1, \dots, S \end{aligned}$$

Ventajas del modelo extensivo:

- Es fácil de formular e implementar

Desventajas del modelo extensivo:

- Queda un modelo con gran número de variables y restricciones, lo que puede volverlo ineficiente
- No toma en cuenta la estructura particular del problema, por lo que se desaprovechan oportunidades de mejorar la eficiencia de la resolución

### 3.4 Valor del Modelo Estocástico

Se presentan en esta sección las definiciones y resultados que cuantifican el valor de la información y de la solución estocástica en particular.

#### 3.4.1 Problema de Distribución

Dado un escenario particular  $\xi$ , se define el problema paramétrico asociado al mismo:

$$\min_{\mathbf{x}} z(\mathbf{x}, \xi) = \mathbf{c}^T \mathbf{x} + \min_{\mathbf{y}} \{ \mathbf{q}^T \mathbf{y} \mid \mathbf{W}\mathbf{y} = \mathbf{h} - \mathbf{T}\mathbf{x}, \mathbf{y} \geq 0 \}$$

s.a

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq 0$$

Se asume que para cada escenario  $\xi$  existe una solución factible del problema, y por lo tanto una solución óptima  $\mathbf{x}^*(\xi)$  con valor óptimo  $z(\mathbf{x}^*(\xi), \xi) \quad \forall \xi \in \Xi$ .

La búsqueda para todos los escenarios de  $\mathbf{x}^*(\xi)$  y  $z(\mathbf{x}^*(\xi), \xi)$  se denomina *Problema de Distribución*, pues justamente lo que hace es investigar la distribución de dichos valores en función del parámetro estocástico  $\xi$ .

Tomando el valor esperado del valor óptimo de dicho problema se obtiene la llamada solución de esperar y ver (wait and see, **WS**):

$$\mathbf{WS} = E_{\xi} [z(\mathbf{x}^*(\xi), \xi)]$$

#### 3.4.2 Valor Esperado de la Información Perfecta

Dado el problema de programación estocástica de dos etapas con recurso, su valor óptimo **RP** (Recourse Problem) se puede expresar como

$$\mathbf{RP} = \min_{\mathbf{x}} E_{\xi} [z(\mathbf{x}, \xi)]$$

que tiene como solución óptima a  $\mathbf{x}^*$  y como valor óptimo  $z(\mathbf{x}^*)$ .

El valor esperado de la información perfecta **EVPI** (Expected Value of Perfect Information), se define como

$$\mathbf{EVPI} = \mathbf{RP} - \mathbf{WS}$$

**EVPI** se puede interpretar como el beneficio máximo que se obtiene si se conoce por anticipado el valor que va a tomar la realización aleatoria, es decir si se anticipa el futuro, medido con respecto al valor óptimo esperado de la solución estocástica.

Otra forma de verlo es como la pérdida de beneficio debido a la presencia de incertidumbre, representando la cantidad que se estaría dispuesto a pagar por un conocimiento perfecto de los eventos futuros.

### 3.4.3. Problema de Valor Esperado

Se obtiene reemplazando el valor de las variables aleatorias con sus valores esperados.

Sea  $\bar{\xi} = \mathbf{E}(\xi)$  el valor esperado del vector aleatorio  $\xi$ .

El Problema de Valor Esperado se plantea como

$$\mathbf{EV} = \min_{\mathbf{x}} \mathbf{z}(\mathbf{x}, \bar{\xi}) = \mathbf{c}^T \mathbf{x} + \min_{\mathbf{y}} \left\{ \mathbf{q}^T(\bar{\xi}) \mathbf{y} \mid \mathbf{W} \mathbf{y} = \mathbf{h}(\bar{\xi}) - \mathbf{T}(\bar{\xi}) \mathbf{x}, \mathbf{y} \geq 0 \right\}$$

s.a

$$\mathbf{A} \mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq 0$$

La solución a este problema se llama solución de valor esperado  $\mathbf{x}^*(\bar{\xi})$ .

El resultado esperado de usar la solución de valor esperado de **EV** se define como

$$\mathbf{EEV} = \mathbf{E}_{\xi} \left[ \mathbf{z}(\mathbf{x}^*(\bar{\xi}), \xi) \right]$$

Es decir que para calcular **EEV** se resuelve para cada escenario  $\xi$  un problema de minimización en las variables de segunda etapa  $\mathbf{y}$ , con valores de las variables de primera etapa fijos y dados por la solución óptima  $\mathbf{x}^*(\bar{\xi})$  de **EV**.

#### **Valor de la solución estocástica**

El Valor de la Solución Estocástica **VSS** (value of stochastic solution) se define como

$$\mathbf{VSS} = \mathbf{EEV} - \mathbf{RP}$$

Representa el beneficio de utilizar el enfoque estocástico frente a emplear la solución de valor esperado, o equivalentemente el costo de ignorar la incertidumbre a la hora de tomar las decisiones.



### 3.4.4 Relaciones de Valor Básicas

- Para todo problema estocástico se cumple

$$WS \leq RP \leq EEV$$

- Dado un problema estocástico con  $q$ ,  $T$  y  $W$  fijos, se verifica

$$EV \leq WS$$

La justificación de estas dos propiedades puede verse en Birge y Louveaux [3].

Se verifican adicionalmente los siguientes resultados:

- Para todo problema estocástico se cumple

$$EVPI \geq 0, \quad VSS \geq 0$$

- Dado un problema estocástico con  $q$  y  $W$  fijos, se cumple

$$EVPI \leq EEV - EV, \quad VSS \leq EEV - EV$$

### 3.4 Técnicas de Resolución del Problema Estocástico

En el caso general del problema de dos etapas con recurso y con un número discreto de realizaciones  $S$ , se tiene el problema maestro

$$\min_x c^T x + \sum_{s=1}^S p_s Q(x, \xi_s)$$

sujeto a  $Ax = b$ ,

$$x \geq 0$$

donde para cada decisión de primera etapa  $x$ , las decisiones de segunda etapa se obtienen como resultado de:

$$Q(x, \xi_s) = \min_y q_s^T y$$

sujeto a

$$Wy = h_s - T_s x,$$

$$y \geq 0$$

Cuando el número de realizaciones del vector aleatorio es finito, y tanto la función objetivo como las restricciones son lineales, el problema se puede formular en forma extensiva como un programa determinístico lineal equivalente, según lo expresado en 3.3 :

$$\min_x c^T x + \sum_{s=1}^S p_s q_s^T y_s$$

sujeto a  $Ax = b$ ,

$$T_s x + W y_s = h_s, \quad s = 1, \dots, S,$$

$$x \geq 0, y_s \geq 0, \quad s = 1, \dots, S$$

Con un número grande de realizaciones, el modelo puede quedar complejo y difícil de resolver. Un enfoque alternativo consiste en realizar una linearización exterior de la función de recurso  $Q(x)$ , y en cada paso resolver una aproximación al problema maestro original, adicionando en

forma secuencial cortes o restricciones de factibilidad y de optimalidad hasta llegar a la solución óptima. Esta es la base de funcionamiento del método de resolución L y del método de Cortes Múltiples.

### 3.4.1 Dualización del Problema de Segunda Etapa

Tanto en el método L como en el método de cortes múltiples, para establecer la forma de los cortes de optimalidad es necesario conocer la solución del problema dual de segunda etapa para cada realización del vector aleatorio.

Dado el problema primal de segunda etapa correspondiente al escenario  $s$ :

$$\begin{aligned} Q(\mathbf{x}, s) = \min_{\mathbf{y}} \quad & \mathbf{q}_s^T \mathbf{y} \\ \text{s.a.} \quad & \mathbf{W}\mathbf{y} = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}, \\ & \mathbf{y} \geq 0 \end{aligned}$$

Se hace la relajación lagrangena con multiplicadores  $\pi$ , quedando el problema relajado

$$\begin{aligned} \min_{\mathbf{y}} \quad & \mathbf{q}_s^T \mathbf{y} + \pi^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x} - \mathbf{W}\mathbf{y}) = \min_{\mathbf{y}} (\mathbf{q}_s^T - \pi^T \mathbf{W}) \mathbf{y} + \pi^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}) \\ \text{s.a.} \quad & \mathbf{y} \geq 0 \qquad \qquad \qquad \text{s.a.} \quad \mathbf{y} \geq 0 \end{aligned}$$

Imponiendo que la solución sea acotada, se tiene que debe cumplirse la condición  $\mathbf{q}_s^T - \pi^T \mathbf{W} \geq 0$ , siendo la solución  $\mathbf{y} = 0$  con valor óptimo  $\pi^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x})$ .

El problema dual queda entonces

$$\begin{aligned} \max_{\pi} \quad & \pi^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}) \\ \text{s.a.} \quad & \pi^T \mathbf{W} \leq \mathbf{q}_s^T \end{aligned}$$

Sea  $\pi_s$  la solución del problema dual, el valor de la función de recurso para el escenario  $s$  puede expresarse entonces como  $Q(\mathbf{x}, \xi_s) = \pi_s^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x})$ .

### 3.4.2 Método L

Se describirá en detalle el método L en el caso general de variables continuas de primera etapa, y se verán luego las modificaciones que tiene cuando dichas variables son enteras ó binarias, el llamado "método L entero".

El método L es una técnica de planos cortantes en la que se realiza una linearización exterior de la función de recurso  $Q(\mathbf{x})$ , y en cada paso se resuelve una aproximación al problema maestro original, adicionando en forma secuencial cortes o restricciones de factibilidad y de optimalidad hasta llegar a la solución óptima. Se caracteriza por el hecho de que cada uno de los cortes de optimalidad se definen teniendo en cuenta y combinando todas las realizaciones del vector aleatorio. Debe su nombre a la forma de "L" que adquiere la estructura de la matriz

global de restricciones cuando el problema se formula en su forma extensiva, la cual para un problema con S escenarios discretos tiene la forma:

$$\begin{array}{ll}
 \mathbf{A} & \\
 \mathbf{T}_1 \mathbf{W} & \\
 \mathbf{T}_2 & \mathbf{W} \\
 \mathbf{T}_3 & \mathbf{W} \\
 \cdot & \\
 \cdot & \\
 \cdot & \\
 \mathbf{T}_S & \dots \dots \mathbf{W}
 \end{array}$$

El método de resolución L se basa en la equivalencia en términos de su solución del problema determinístico equivalente PED y el problema resuelto por el método L, PML:

$$\begin{array}{ll}
 \text{[PED]} \min_{\mathbf{x}} & \mathbf{c}^T \mathbf{x} + \mathbf{Q}(\mathbf{x}) \quad \sim \quad \text{[PML]} \min_{\mathbf{x}, \theta} \quad \mathbf{c}^T \mathbf{x} + \theta \\
 \text{s. a.} & \mathbf{x} \in \mathbf{K}_1 \cap \mathbf{K}_2 \quad \quad \quad \text{s.a.} \quad \mathbf{Q}(\mathbf{x}) \leq \theta, \\
 & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \mathbf{x} \in \mathbf{K}_1 \cap \mathbf{K}_2, \theta \in \mathbf{R}
 \end{array}$$

Se presentará primero el algoritmo del método, dando posteriormente la justificación de los cortes de optimalidad y factibilidad empleados.

**Algoritmo (Método L)**

Paso 0 Inicializar variables  $\mathbf{v} := \mathbf{r} := \mathbf{t} := 0$ .

$\mathbf{v}$ : número de iteraciones

$\mathbf{r}$ : número de cortes de factibilidad

$\mathbf{t}$ : número de cortes de optimalidad

Paso 1 Establecer  $\mathbf{v} := \mathbf{v} + 1$ . Resolver el problema lineal maestro

$$\begin{array}{ll}
 \min_{\mathbf{x}, \theta} & \mathbf{c}^T \mathbf{x} + \theta \\
 \text{s.a.} & \mathbf{A} \mathbf{x} = \mathbf{b}, \\
 & \mathbf{D}_l \mathbf{x} \geq \mathbf{d}_l, \quad l = 1, \dots, r \quad (\text{cortes de factibilidad}) \quad (1) \\
 & \mathbf{E}_l \mathbf{x} + \theta \geq \mathbf{e}_l, \quad l = 1, \dots, t \quad (\text{cortes de optimalidad}) \quad (2) \\
 & \mathbf{x} \geq 0, \quad \theta \in \mathbf{R}
 \end{array}$$

Sea  $(\mathbf{x}^{\mathbf{v}}, \theta^{\mathbf{v}})$  la solución óptima. Si no hay restricciones de optimalidad ( $\mathbf{t} = 0$ ) se establece  $\theta^{\mathbf{v}} = -\infty$  y no se considera en el cálculo de  $\mathbf{x}^{\mathbf{v}}$ .

Paso 2 Para cada escenario  $\mathbf{s} = 1, \dots, \mathbf{S}$  resolver el problema de factibilidad

$$\begin{array}{ll}
 \min_{\mathbf{y}, \mathbf{v}^+, \mathbf{v}^-} & \mathbf{e}^T \mathbf{v}^+ + \mathbf{e}^T \mathbf{v}^- \\
 \text{s.a.} & \mathbf{W} \mathbf{y} + \mathbf{I} \mathbf{v}^+ - \mathbf{I} \mathbf{v}^- = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}^{\mathbf{v}}, \\
 & \mathbf{y} \geq 0, \quad \mathbf{v}^+ \geq 0, \quad \mathbf{v}^- \geq 0, \\
 \text{donde} & \mathbf{e}^T = (1, \dots, 1), \text{ hasta que para algún } \mathbf{s} \text{ se verifique que el valor óptimo } \mathbf{u} > 0.
 \end{array}$$

Esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero ambas variables artificiales introducidas.

En este caso, siendo  $\sigma^y$  la solución óptima del problema dual asociado (multiplicadores simplex óptimos del problema primal), se definen  $\mathbf{D}_{r+1} := (\sigma^y)^T \mathbf{T}_s$  y  $\mathbf{d}_{r+1} := (\sigma^y)^T \mathbf{h}_s$

Agregar la restricción  $\mathbf{D}_{r+1} \mathbf{x} \geq \mathbf{d}_{r+1}$  al problema maestro (corte de factibilidad).

Poner  $r := r + 1$  e ir al Paso 1.

Si  $\forall s, \mathbf{u} = 0$ , ir al Paso 3.

**Paso 3** Para  $s = 1, \dots, S$  resolver el problema lineal

$$\begin{aligned} \min_{\mathbf{y}} \quad & \mathbf{q}_s^T \mathbf{y} \\ \text{s.a.} \quad & \mathbf{W} \mathbf{y} = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}^y, \\ & \mathbf{y} \geq 0 \end{aligned}$$

Sea  $\pi_s^y$  la solución óptima del dual de este problema para el escenario  $s$ .

Se definen  $\mathbf{E}_{t+1} := \sum_{s=1}^S \mathbf{p}_s (\pi_s^y)^T \mathbf{T}_s$  y  $\mathbf{e}_{t+1} := \sum_{s=1}^S \mathbf{p}_s (\pi_s^y)^T \mathbf{h}_s$

Si  $\theta^y \geq \mathbf{e}_{t+1} - \mathbf{E}_{t+1} \mathbf{x}^y$ , entonces  $\mathbf{x}^y$  es una solución óptima. FIN del algoritmo.

En caso contrario, agregar la restricción  $\mathbf{E}_{t+1} \mathbf{x} + \theta \geq \mathbf{e}_{t+1}$  al problema maestro (corte de optimalidad).

Poner  $t := t + 1$  e ir al Paso 1.

### **Justificación de los cortes de optimalidad**

Los cortes de optimalidad son aproximaciones lineales a la función de recurso  $\mathbf{Q}(\mathbf{x})$  en su región de dominio finito.

De la dualización del problema de segunda etapa se tiene que, dada una decisión de primera etapa  $\mathbf{x}^y$ , y siendo  $\pi_s^y$  la solución del problema dual de segunda etapa en el escenario  $s$ , se cumple

$$\mathbf{Q}(\mathbf{x}^y, \xi_s) = (\pi_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}^y)$$

Como se tiene que  $\mathbf{Q}(\mathbf{x}, \xi_s)$  es convexa respecto a  $\mathbf{x}$ , por el teorema del subgradiente se verifica que  $\mathbf{Q}(\mathbf{x}, \xi_s) \geq \mathbf{Q}(\mathbf{x}^y, \xi_s) + (\mathbf{x} - \mathbf{x}^y)^T \boldsymbol{\eta}$ , donde  $\boldsymbol{\eta} = \frac{\partial \mathbf{Q}(\mathbf{x}, \xi_s)}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{x}^y} = -\pi_s^y \mathbf{T}_s$

Sustituyendo en esta desigualdad los valores de  $\mathbf{Q}(\mathbf{x}^y, \xi_s)$  y de  $\boldsymbol{\eta}$  se obtiene

$$\mathbf{Q}(\mathbf{x}, \xi_s) \geq (\pi_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x})$$

Calculando la esperanza a ambos lados de la desigualdad se llega a

$$\mathbf{Q}(\mathbf{x}) \geq \sum_{s=1}^S \mathbf{p}_s (\pi_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x})$$

Se cumple la igualdad solamente para  $\mathbf{x} = \mathbf{x}^y$ , esto es  $\mathbf{Q}(\mathbf{x}^y) = \sum_{s=1}^S \mathbf{p}_s (\pi_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}^y)$

Sea  $(\mathbf{x}^y, \theta^y)$  la solución óptima del problema maestro obtenida en el Paso 1. En el Paso 3 del algoritmo, se calcula  $Q(\mathbf{x}^y)$ .

Como en el problema equivalente PML debe cumplirse la restricción  $Q(\mathbf{x}) \leq \theta$ , si  $\theta^y \geq Q(\mathbf{x}^y)$  se tiene que  $(\mathbf{x}^y, \theta^y)$  es solución óptima del problema PML.

En cambio, si  $\theta^y < Q(\mathbf{x}^y)$ , para asegurar  $Q(\mathbf{x}) \leq \theta$  debe imponerse la restricción

$$\theta \geq \sum_{s=1}^S \mathbf{p}_s (\boldsymbol{\pi}_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}),$$

que es justamente el corte de optimalidad definido en el Paso (3) del algoritmo e introducido en el problema maestro para la iteración siguiente.

### **Justificación de los cortes de factibilidad**

Si en el Paso 2 del algoritmo, se verifica que para algún escenario  $s$  el valor óptimo del problema de factibilidad es estrictamente mayor que 0, considerando la solución óptima del problema dual  $\boldsymbol{\sigma}^y$ , que verifica  $\{ (\boldsymbol{\sigma}^y)^T \mathbf{W} \leq 0, -\mathbf{e}^T \leq (\boldsymbol{\sigma}^y)^T \leq \mathbf{e}^T \}$ , se cumple necesariamente que  $(\boldsymbol{\sigma}^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}^y) > 0$ .

Esto implica que si  $\mathbf{x} \in \mathbf{K}_2$ , el valor óptimo del problema de factibilidad del Paso 2 es 0, por lo que se cumple que  $(\boldsymbol{\sigma}^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}) \leq 0$ . Este es justamente el corte de factibilidad que se introduce en el Paso 2 cuando la solución del problema maestro  $\mathbf{x}^y$  del Paso 1 resulta no factible para el problema de segunda etapa en algún escenario  $s$ .

### **3.4.3 Método de Cortes Múltiples**

En este método se hace una linearización exterior de la función de recurso  $Q(\mathbf{x})$  agregando en cada iteración cortes de factibilidad similares a los del método L y cortes de optimalidad definidos para cada una de las realizaciones del vector aleatorio.

Esto implica que en lugar de definir un único corte por iteración combinando las realizaciones tal cómo hace el método L, en el método de cortes múltiples se identifica en cada iteración las realizaciones con solución no óptima y se incluye selectivamente un corte de optimalidad por cada una de ellas. De esta forma, al mandar al problema maestro cortes desagregados, se le pasa información mas detallada, lo que tiende a disminuir el número de iteraciones necesarias frente al método L. Esto es a costa de una mayor complejidad del modelo (mayor número de cortes).

El método de cortes múltiples se basa en la equivalencia en términos de su solución del problema determinístico equivalente PED y el problema del método de cortes múltiples PMCM:

$$\begin{aligned} \text{[PED]} \quad \min_{\mathbf{x}} \quad & \mathbf{c}^T \mathbf{x} + \sum_{s=1}^S \mathbf{p}_s Q(\mathbf{x}, \xi_s) \quad \sim \quad \text{[PMCM]} \quad \min_{\mathbf{x}, \theta_s} \quad \mathbf{c}^T \mathbf{x} + \sum_{s=1}^S \theta_s \\ \text{s. a.} \quad & \mathbf{x} \in \mathbf{K}_1 \cap \mathbf{K}_2 \quad \quad \quad \text{s.a.} \quad \theta_s \geq \mathbf{p}_s Q(\mathbf{x}, \xi_s), \quad s = 1, \dots, S \\ & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \mathbf{x} \in \mathbf{K}_1 \cap \mathbf{K}_2, \quad \theta_1, \dots, \theta_S \in \mathbf{R} \end{aligned}$$

### Algoritmo (Método de Cortes Múltiples)

Paso 0 Inicializar variables  $v := r := 0$  y  $t(s) := 0, s = 1, \dots, S$

$v$ : número de iteraciones

$r$ : número de cortes de factibilidad

$t(s)$ : número de cortes de optimalidad en el escenario  $s$

Paso 1 Establecer  $v := v + 1$ . Resolver el problema lineal maestro

$$\min_{\mathbf{x}, \theta_s} \mathbf{c}^T \mathbf{x} + \sum_{s=1}^S \theta_s$$

s.a.

$$\mathbf{A} \mathbf{x} = \mathbf{b},$$

$$\mathbf{D}_l \mathbf{x} \geq \mathbf{d}_l, \quad l = 1, \dots, r \quad (\text{cortes de factibilidad}) \quad (1)$$

$$\mathbf{E}_{l(s)} \mathbf{x} + \theta_s \geq \mathbf{e}_{l(s)}, \quad l(s) = 1, \dots, t(s), \quad s = 1, \dots, S \quad (\text{cortes de optimalidad}) \quad (2)$$

$$\mathbf{x} \geq 0, \quad \theta_1, \dots, \theta_S \in \mathbf{R}$$

Sea  $(\mathbf{x}^v, \theta_1^v, \dots, \theta_S^v)$  la solución óptima. Si para algún escenario  $s$  no hay restricciones de optimalidad ( $t(s) = 0$ ) se establece  $\theta_s^v = -\infty$  y no se considera en el cálculo de  $\mathbf{x}^v$ .

Paso 2 Idéntico al Paso 2 del método L. Se resuelve un problema de factibilidad por escenario.

Paso 3 Para  $s = 1, \dots, S$  resolver el problema lineal

$$\min_{\mathbf{y}} \mathbf{q}_s^T \mathbf{y}$$

$$\text{s.a. } \mathbf{W} \mathbf{y} = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}^v,$$

$$\mathbf{y} \geq 0$$

Sea  $\pi_s^v$  la solución óptima del dual de este problema para el escenario  $s$ .

Se definen  $\mathbf{E}_{t(s)+1} := \mathbf{p}_s (\pi_s^v)^T \mathbf{T}_s$  y  $\mathbf{e}_{t(s)+1} := \mathbf{p}_s (\pi_s^v)^T \mathbf{h}_s$

Si  $\theta_s^v < \mathbf{e}_{t(s)+1} - \mathbf{E}_{t(s)+1} \mathbf{x}^v$ , agregar la restricción  $\mathbf{E}_{t(s)+1} \mathbf{x} + \theta_s \geq \mathbf{e}_{t(s)+1}$  al problema maestro (corte de optimalidad) y fijar  $t(s) := t(s) + 1$ .

Si  $\theta_s^v \geq \mathbf{e}_{t(s)+1} - \mathbf{E}_{t(s)+1} \mathbf{x}^v \quad \forall s = 1, \dots, S$ , entonces  $\mathbf{x}^v$  es una solución óptima. FIN del algoritmo.

En caso contrario, ir al Paso 1.

En general, el método de cortes múltiples resulta mas eficiente que el método L cuando el número de realizaciones  $S$  no es mucho mayor que el número de restricciones de primera etapa, dado por el número de filas  $m_1$  de la matriz determinística de primera etapa  $\mathbf{A}$ .

### 3.4.4 Método L Entero

Se considerarán ahora las modificaciones del método L anterior cuando las variables de primera etapa son enteras, y en particular binarias. Se asume que las variables de segunda etapa siguen siendo continuas. La idea principal es relajar las restricciones de integralidad de las variables de primera etapa, y manejarlas a través de un árbol de ramificación y acotamiento, empleando una técnica de Branch & Cut.

**Algoritmo (Método L Entero)**

Paso 0 Inicializar variables  $v := r := t := 0, \hat{z} := +\infty$ .

$v$ : número de iteraciones

$r$ : número de cortes de factibilidad

$t$ : número de cortes de optimalidad

$\hat{z}$  : valor de la mejor solución entera hallada hasta el momento

Se crea una lista  $L$  de problemas (nodos) pendientes, con un único problema inicial:

$$\begin{aligned} \min_{\mathbf{x}, \theta} \quad & \mathbf{c}^T \mathbf{x} + \theta \\ \text{s.a.} \quad & \mathbf{A} \mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq 0, \theta \in \mathbf{R} \end{aligned}$$

Paso 1 Si  $L = \emptyset$ , FIN, la solución correspondiente a  $\hat{z}$  es óptima. En caso contrario, seleccionar un nodo pendiente  $P$  de  $L$  y removerlo de la lista.

Paso 2 Establecer  $v := v + 1$ . Resolver el problema lineal maestro correspondiente a  $P$  :

$$\begin{aligned} \min_{\mathbf{x}_p, \theta_p} \quad & \mathbf{c}^T \mathbf{x}_p + \theta_p \\ \text{s.a.} \quad & \mathbf{A} \mathbf{x}_p = \mathbf{b}, \\ & \mathbf{D}_l \mathbf{x}_p \geq \mathbf{d}_l, \quad l = 1, \dots, r \quad (\text{cortes de factibilidad}) \quad (1) \\ & \mathbf{E}_l \mathbf{x}_p + \theta_p \geq \mathbf{e}_l, \quad l = 1, \dots, t \quad (\text{cortes de optimalidad}) \quad (2) \\ & \mathbf{x}_p \geq 0, \theta_p \in \mathbf{R} \end{aligned}$$

donde  $\mathbf{x}_p$  tiene componentes fijos dados por los valores de las aristas del árbol del Branch&Cut desde la raíz hasta el nodo  $P$ , y componentes variables sobre los que se hace la minimización.

Si el problema es no factible, vaciar el nodo e ir al Paso 1. Sino, sea  $(\mathbf{x}_p^v, \theta_p^v)$  la solución óptima. Si no hay restricciones de optimalidad ( $t = 0$ ) se considera  $\theta_p^v = -\infty$  y no se considera en el cálculo de  $\mathbf{x}_p^v$ .

Si  $\mathbf{c}^T \mathbf{x}_p^v + \theta_p^v \geq \hat{z}$ , vaciar el nodo  $P$  e ir al Paso 1.

Paso 3 Para cada escenario  $s = 1, \dots, S$  resolver el problema de factibilidad

$$\begin{aligned} \min_{\mathbf{y}, \mathbf{v}^+, \mathbf{v}^-} \quad & \mathbf{e}^T \mathbf{v}^+ + \mathbf{e}^T \mathbf{v}^- \\ \text{s.a.} \quad & \mathbf{W} \mathbf{y} + \mathbf{I} \mathbf{v}^+ - \mathbf{I} \mathbf{v}^- = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}_p^v, \\ & \mathbf{y} \geq 0, \mathbf{v}^+ \geq 0, \mathbf{v}^- \geq 0, \end{aligned}$$

donde  $\mathbf{e}^T = (1, \dots, 1)$ , hasta que para algún  $s$  se verifique que el valor óptimo  $\mathbf{u} > 0$ .

Esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero todas las variables artificiales introducidas.

En este caso, siendo  $\sigma^y$  la solución óptima del problema dual asociado (multiplicadores simplex óptimos del problema primal), se definen  $\mathbf{D}_{r+1} := (\sigma^y)^T \mathbf{T}_s$  y  $\mathbf{d}_{r+1} := (\sigma^y)^T \mathbf{h}_s$

Agregar la restricción  $\mathbf{D}_{r+1} \mathbf{x}_p \geq \mathbf{d}_{r+1}$  al problema maestro en el nodo  $\mathbf{P}$ , (corte de factibilidad).  
Poner  $r := r + 1$  e ir al Paso 2.

Si  $\forall s, \mathbf{u} = 0$ , ir al Paso 4.

**Paso 4** Verificar las restricciones de integralidad de  $\mathbf{x}_p^y$ . Si se viola una restricción, ramificar en la componente relevante de  $\mathbf{x}_p^y$  en dos nuevas ramas, creando dos nuevos subproblemas que se agregan a la lista de nodos pendientes  $\mathbf{L}$ , e ir al Paso 1.

**Paso 5** Para  $s = 1, \dots, \mathbf{S}$  resolver el problema lineal

$$\begin{aligned} \min_y \quad & \mathbf{q}_s^T \mathbf{y} \\ \text{s.a.} \quad & \mathbf{W} \mathbf{y} = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}_p^y, \\ & \mathbf{y} \geq 0 \end{aligned}$$

Sea  $\pi_s^y$  la solución óptima del dual de este problema para el escenario  $s$ .

Se definen  $\mathbf{E}_{t+1} := \sum_{s=1}^{\mathbf{S}} \mathbf{p}_s (\pi_s^y)^T \mathbf{T}_s$  y  $\mathbf{e}_{t+1} := \sum_{s=1}^{\mathbf{S}} \mathbf{p}_s (\pi_s^y)^T \mathbf{h}_s$

Calcular  $\mathbf{Q}(\mathbf{x}_p^y) = \sum_{s=1}^{\mathbf{S}} \mathbf{p}_s (\pi_s^y)^T (\mathbf{h}_s - \mathbf{T}_s \mathbf{x}_p^y) = \mathbf{e}_{t+1} - \mathbf{E}_{t+1} \mathbf{x}_p^y$

Actualizar la cota superior  $\bar{\mathbf{z}} := \min \{ \bar{\mathbf{z}}, \mathbf{c}^T \mathbf{x}_p^y + \mathbf{Q}(\mathbf{x}_p^y) \}$ . Esto es posible pues en este punto ya se tiene una solución factible entera.

**Paso 6** Si  $\theta_p^y \geq \mathbf{Q}(\mathbf{x}_p^y)$ , vaciar el nodo actual e ir al Paso 1. La solución  $\mathbf{x}_p^y$  es óptima para el problema del nodo  $\mathbf{P}$ .

En caso contrario, agregar la restricción  $\mathbf{E}_{t+1} \mathbf{x}_p + \theta_p \geq \mathbf{e}_{t+1}$  al problema maestro en  $\mathbf{P}$  (corte de optimalidad).

Poner  $t := t + 1$  e ir al Paso 2.

Una aplicación del Método L Entero a la resolución de problemas estocásticos de ruteo puede verse en Laporte y Louveaux [13].

### 3.4.5 Método de Cortes Múltiples Entero

Se considerarán ahora las modificaciones del Método de Cortes Múltiples cuando las variables de primera etapa son enteras, y en particular binarias. Se asume que las variables de segunda etapa siguen siendo continuas. La idea básica es relajar las restricciones de integralidad de las variables de primera etapa, y manejarlas a través de un árbol de ramificación y acotamiento, empleando una técnica de Branch & Cut.



**Algoritmo (Método de Cortes Múltiples Entero)**

Paso 0 Inicializar variables  $\mathbf{v} := \mathbf{r} := \mathbf{t}(\mathbf{s}) := 0$ ,  $\mathbf{s} = 1, \dots, \mathbf{S}$ ,  $\hat{\mathbf{z}} := +\infty$ .

$\mathbf{v}$ : número de iteraciones

$\mathbf{r}$ : número de cortes de factibilidad

$\mathbf{t}(\mathbf{s})$ : número de cortes de optimalidad en el escenario  $\mathbf{s}$

$\hat{\mathbf{z}}$ : valor de la mejor solución entera hallada hasta el momento

Se crea una lista  $\mathbf{L}$  de problemas (nodos) pendientes, con un único problema inicial:

$$\min_{\mathbf{x}, \theta_s} \mathbf{c}^T \mathbf{x} + \sum_{s=1}^{\mathbf{S}} \theta_s$$

$$\text{s.a. } \mathbf{x} \geq 0, \quad \theta_1, \dots, \theta_s \in \mathbf{R}$$

Paso 1 Si  $\mathbf{L} = \emptyset$ , FIN, la solución correspondiente a  $\hat{\mathbf{z}}$  es óptima. En caso contrario, seleccionar un nodo pendiente  $\mathbf{P}$  de  $\mathbf{L}$  y removerlo de la lista.

Paso 2 Establecer  $\mathbf{v} := \mathbf{v} + 1$ . Resolver el problema lineal maestro correspondiente a  $\mathbf{P}$ :

$$\min_{\mathbf{x}^{\mathbf{P}}, \theta_s^{\mathbf{P}}} \mathbf{c}^T \mathbf{x}^{\mathbf{P}} + \sum_{s=1}^{\mathbf{S}} \theta_s^{\mathbf{P}}$$

$$\text{s.a. } \mathbf{D}^{\mathbf{l}} \mathbf{x}^{\mathbf{P}} \geq \mathbf{d}^{\mathbf{l}}, \quad \mathbf{l} = 1, \dots, \mathbf{r} \quad (\text{cortes de factibilidad}) \quad (1)$$

$$\mathbf{E}^{\mathbf{l}(\mathbf{s})} \mathbf{x}^{\mathbf{P}} + \theta_s^{\mathbf{P}} \geq \mathbf{e}^{\mathbf{l}(\mathbf{s})}, \quad \mathbf{l}(\mathbf{s}) = 1, \dots, \mathbf{t}(\mathbf{s}) \quad (\text{cortes de optimalidad}) \quad (2)$$

$$\mathbf{x}^{\mathbf{P}} \geq 0, \quad \theta_1^{\mathbf{P}}, \dots, \theta_s^{\mathbf{P}} \in \mathbf{R}$$

donde  $\mathbf{x}^{\mathbf{P}}$  tiene componentes fijos dados por los valores de las aristas del árbol del Branch&Cut desde la raíz hasta el nodo  $\mathbf{P}$ , y componentes variables sobre los que se hace la minimización.

Si el problema es no factible, vaciar el nodo e ir al Paso 1. Sino, sea  $(\mathbf{x}^{\mathbf{Pv}}, \theta_1^{\mathbf{Pv}}, \dots, \theta_s^{\mathbf{Pv}})$  la solución óptima. Si no hay restricciones de optimalidad ( $\mathbf{t}(\mathbf{s}) = 0 \forall \mathbf{s}$ ) se considera  $\theta_s^{\mathbf{Pv}} = -\infty \forall \mathbf{s}$  y no se toman en cuenta en el cálculo de  $\mathbf{x}^{\mathbf{Pv}}$ .

Si  $\mathbf{c}^T \mathbf{x}^{\mathbf{Pv}} + \sum_{s=1}^{\mathbf{S}} \theta_s^{\mathbf{Pv}} \geq \hat{\mathbf{z}}$ , vaciar el nodo  $\mathbf{P}$  e ir al Paso 1.

Paso 3 Para cada escenario  $\mathbf{s} = 1, \dots, \mathbf{S}$  resolver el problema de factibilidad

$$\min_{\mathbf{y}, \mathbf{v}^+, \mathbf{v}^-} \mathbf{u} = \mathbf{e}^T \mathbf{v}^+ + \mathbf{e}^T \mathbf{v}^-$$

$$\text{s.a. } \mathbf{W}\mathbf{y} + \mathbf{l}\mathbf{v}^+ - \mathbf{l}\mathbf{v}^- = \mathbf{h}_s - \mathbf{T}_s \mathbf{x}^{\mathbf{Pv}},$$

$$\mathbf{y} \geq 0, \quad \mathbf{v}^+ \geq 0, \quad \mathbf{v}^- \geq 0,$$

donde  $\mathbf{e}^T = (1, \dots, 1)$ , hasta que para algún  $\mathbf{s}$  se verifique que el valor óptimo  $\mathbf{u} > 0$ .

Esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero todas las variables artificiales introducidas.

En este caso, siendo  $\sigma^y$  la solución óptima del problema dual asociado (multiplicadores simplex óptimos del problema primal), se definen  $D^{r+1} := (\sigma^y)^T T_s$  y  $d^{r+1} := (\sigma^y)^T h_s$

Agregar la restricción  $D^{r+1} x^P \geq d^{r+1}$  al problema maestro en el nodo  $P$  (corte de factibilidad). Poner  $r := r + 1$  e ir al Paso 2.

Si  $\forall s, u = 0$ , ir al Paso 4.

Paso 4 Verificar las restricciones de integralidad de  $x^{Pv}$ . Si se viola una restricción, ramificar en la componente relevante  $x_{ij}^{Pv}$  en dos nuevas ramas, una con  $x_{ij}^{Pv} \leq \lfloor x_{ij}^{Pv} \rfloor$  y la otra con  $x_{ij}^{Pv} \geq \lceil x_{ij}^{Pv} \rceil$ , creando dos nuevos subproblemas que se agregan a la lista de nodos pendientes  $L$ , e ir al Paso 1.

Paso 5 Para  $s = 1, \dots, S$  resolver el problema lineal

$$\begin{aligned} \min_y \quad & q_s^T y \\ \text{s.a.} \quad & W y = h_s - T_s x^{Pv}, \\ & y \geq 0 \end{aligned}$$

Sea  $\pi_s^y$  la solución óptima del dual de este problema para el escenario  $s$ .

Se definen  $E^{t(s)+1} := p_s (\pi_s^y)^T T_s$  y  $e_{t+1} := p_s (\pi_s^y)^T h_s$

$$\text{Calcular } Q(x^{Pv}) = \sum_{s=1}^S p_s (\pi_s^y)^T (h_s - T_s x^{Pv}) = \sum_{s=1}^S (e^{t(s)+1} - E^{t(s)+1} x^{Pv})$$

Actualizar la cota superior  $\hat{z} := \min \{ \hat{z}, c^T x^{Pv} + Q(x^{Pv}) \}$ . Esto es posible pues en este punto ya se tiene una solución factible entera.

Paso 6 Para  $s = 1, \dots, S$ :

Si  $\theta_s^{Pv} < e^{t(s)+1} - E^{t(s)+1} x^{Pv}$ , agregar la restricción  $E^{t(s)+1} x^P + \theta_s^P \geq e^{t(s)+1}$  al problema maestro en  $P$  (corte de optimalidad) y fijar  $t(s) := t(s) + 1$ .

Si  $\theta_s^{Pv} \geq e^{t(s)+1} - E^{t(s)+1} x^{Pv} \quad \forall s = 1, \dots, S$ , entonces la solución  $x^{Pv}$  es óptima para el problema del nodo  $P$ . Vaciar el nodo actual  $P$  e ir al Paso 1. En caso contrario, ir al Paso 2.

## 4 Problema Estocástico de Diseño de Red

Se considera ahora el problema de diseño de red pero introduciendo incertidumbre en la demanda inicial  $\mathbf{r}^k$  de las commodities, manteniendo fijo los orígenes  $\mathbf{o}(\mathbf{k})$  y destinos  $\mathbf{d}(\mathbf{k})$ .

Se supondrá la existencia de un conjunto discreto de escenarios de demanda  $\mathbf{S}$ , en donde para cada realización o escenario  $s \in \mathbf{S}$  se tiene una demanda  $\mathbf{r}^{ks}$  para la commodity  $\mathbf{k}$  en dicho escenario.

Cada escenario  $s$  tiene asociada una probabilidad  $p_s$  de forma que  $\sum_{s \in \mathbf{S}} p_s = 1$ .

El problema de diseño óptimo con incertidumbre puede plantearse entonces como un problema de programación estocástica lineal en dos etapas con recurso fijo, en el cual:

1) Las variables  $x_{ij}$  que indican cuales arcos van a formar parte de la red constituyen las *variables de decisión de primera etapa*. Se denota al vector de variables de primera etapa como  $\mathbf{x} := [x_{ij}]$ ,  $(i, j) \in \mathbf{A}$

2) Los flujos de las commodities  $y_{ij}^k$  en cada arco de la red se identifican con las *variables de decisión de segunda etapa ó variables de recurso*, y representan la acción correctiva a tomar (“lo mejor que se puede hacer”) una vez elegidas las variables de primera etapa y resuelta la incertidumbre.

Para determinar el valor óptimo de las variables de primera etapa  $x_{ij}$  se tienen en cuenta en forma simultánea y ponderada todos los posibles escenarios de demanda  $s$  y los consiguientes flujos  $y_{ij}^k$  resultantes.

Una vez elegidos los arcos que van a formar parte del diseño final de la red y resuelta la incertidumbre, se elige la distribución de flujos  $y_{ij}^k$  de las commodities con el criterio de minimizar los costos de transporte por la red.

### 4.1 Problema Estocástico Sin Restricciones de Capacidad

El caso del del problema de diseño óptimo de red sin restricciones de capacidad en los arcos y en el contexto de incertidumbre se denomina SUNDP (Stochastic Uncapacitated Network Design Problem) .

#### 4.1.1 Formulación Extensiva del Problema SUNDP

Sean las variables

$$x_{ij} = \begin{cases} 1 & \text{si se usa el arco } (i, j) \\ 0 & \text{en otro caso} \end{cases}$$

$y_{ij}^{ks}$  = flujo de la commodity  $\mathbf{k}$  en el arco  $(i, j)$  y en el escenario  $s$  de demandas

El modelo extensivo del problema SUNDP queda

$$[\text{SUNDP}] z = \min_{x,y} \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{s \in S} \sum_{k \in K} \sum_{(i,j) \in A} p_s c_{ij}^k y_{ij}^{ks}$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^{ks} - \sum_{j:(j,i) \in A} y_{ji}^{ks} = b_i^{ks} \quad \forall i \in N, \forall k \in K, \forall s \in S \quad (1)$$

$$y_{ij}^{ks} \leq r^{ks} x_{ij} \quad \forall (i, j) \in A, \forall k \in K, \forall s \in S \quad (2)$$

$$y_{ij}^{ks} \geq 0 \quad \forall (i, j) \in A, \forall k \in K, \forall s \in S \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (4)$$

siendo

$$b_i^{ks} = \begin{cases} r^{ks} & \text{si } i = o(k) \\ -r^{ks} & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

### **Propiedades del Modelo Extensivo**

Siguen valiendo las mismas propiedades establecidas para el caso determinístico, pero se tiene ahora una mayor complejidad en el modelo por el aumento tanto en el número de variables como en el de restricciones.

Por ejemplo, el número de variables de flujo pasa de  $|A|IKI$  a  $|A|IKIS|$ .

Una diferencia es que en el caso estocástico, si las demandas son enteras la función objetivo no lo es necesariamente, pues en su determinación intervienen las probabilidades de cada escenario, que toman valores reales.

Para solucionar el problema estocástico SUNDP expresado en su formulación extensiva se puede aplicar cualquiera de las técnicas descritas para resolver el problema del caso determinístico UNDP (por ej. Branch & Bound, Ascenso Dual).

### **4.1.2 Formulación del Modelo en Dos Etapas del Problema SUNDP**

En el caso del SUNDP, el modelo en dos etapas queda

$$[\text{SUNDP}] z = \min_x \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{s=1}^S p_s Q(x, \xi(s))$$

sujeto a

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (1)$$

siendo el problema de segunda etapa

$$Q(x, \xi(s)) = \min_y \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k(s)$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k(s) - \sum_{j:(j,i) \in A} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K \quad (2)$$

$$y_{ij}^k(s) \leq r^k(s) x_{ij} \quad \forall (i, j) \in A, \forall k \in K \quad (3)$$

$$y_{ij}^k(s) \geq 0 \quad \forall (i, j) \in A, \forall k \in K \quad (4)$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Observaciones:

Identificando términos con relación a la formulación general en 3.1, para el *problema de primera etapa* se verifica que:

- No hay restricciones determinísticas que no dependan de la incertidumbre en la demanda (matriz **A** y vector **b** nulos)
- Las variables de primera etapa son binarias

Para el *problema de segunda etapa*, se tiene que:

- La matriz de recurso **W** es fija (determinística), dado que todos los coeficientes de las variables de segunda etapa son fijos en todas las restricciones
- El vector **q** de costos no tiene incertidumbre asociada
- Los componentes no nulos de la matriz tecnológica **T** y del vector de requerimientos **h** son las demandas aleatorias  $r^k(\omega)$ , a menos del signo.

### 4.1.3 Propiedades del Modelo en Dos Etapas del Problema SUNDP

Como para el SUNDP la matriz de recurso **W** es fija y el vector aleatorio  $\xi$  es discreto y tiene soporte  $\Xi$  finito, y como las restricciones de integralidad se dan solamente en las variables de primera etapa, se verifica que:

- El conjunto factible de primera etapa es  $K_1 = \{x \mid x_{ij} \in \{0, 1\}\}$
- $Q(x, \xi)$  es una función lineal a trozos y convexa de **x** en  $K_2$
- $Q(x)$  es la suma ponderada con coeficientes positivos de las realizaciones de  $\xi$

- $K_2$  es poliédrico, cerrado y convexo
- $Q(\mathbf{x})$  es una función lineal a trozos y convexa en  $K_2$

Para  $\mathbf{x}$  y  $\xi$  fijos, el problema de segunda etapa queda desacoplado en  $|K|$  subproblemas de camino de costo mínimo, uno por cada commodity  $\mathbf{k}$ , que se resuelven en forma independiente y están definidos sobre la subred formada por los arcos  $\{(i, j) \in A \mid x_{ij} = 1\}$ .

#### 4.1.4 Factibilidad del Modelo en Dos Etapas del Problema SUNDP

El Problema Equivalente Determinístico PED para el SUNDP se puede expresar como

$$\min_{\mathbf{x}} \sum_{(i,j) \in A} f_{ij} x_{ij} + Q(\mathbf{x})$$

sujeto a

$$\mathbf{x} \in K_1 \cap K_2$$

La factibilidad del problema de segunda etapa no depende exclusivamente de los valores de  $\mathbf{x}$ , sino que requiere que en la red de base original haya camino dirigido entre todo nodo origen y todo nodo destino de una misma commodity. Esto está expresado por las restricciones (2) del SUNDP.

Esto implica que el SUNDP no es lo que se denomina un *problema con recurso relativamente completo*, pues no se satisface la condición  $K_1 \subset K_2$ , es decir no toda solución factible para el problema de primera etapa es factible para el problema de segunda etapa.

El efecto de  $\mathbf{x}$  sobre la factibilidad del problema de segunda etapa se da a través de las restricciones (3) del SUNDP.

Intuitivamente se observa que los valores de  $\mathbf{x}$  que hacen factible el problema de segunda etapa son aquellos que no eliminan la existencia de al menos un camino dirigido entre el origen y el destino de cada commodity  $\mathbf{k}$ .

#### 4.1.5 Existencia de la Solución Óptima del Problema SUNDP

Como el vector aleatorio  $\xi$  tiene momentos de segundo orden finitos y la región factible es acotada, dado que el conjunto  $K_1$  lo es al ser finito el número de arcos de la red base y tomar estos valores binarios, se asegura que si  $K_1 \cap K_2$  es no vacío, el PED tiene solución óptima finita.

La condición para que haya óptimo finito pasa a ser entonces la que asegura la factibilidad del problema de segunda etapa para al menos algún valor de las variables de primera etapa, y esto es que en la red base exista al menos un camino dirigido entre el origen  $\mathbf{o}(\mathbf{k})$  y el destino  $\mathbf{d}(\mathbf{k})$  de toda commodity  $\mathbf{k}$  con demanda no nula.

La condición de existencia de solución es entonces equivalente a que el siguiente problema lineal tenga solución óptima nula ( $\mathbf{u} = 0$ ) para todo escenario  $\mathbf{s}$ :

$$\min_{\mathbf{y}, \mathbf{v}, \mathbf{h}} \quad \mathbf{u} = \sum_{\mathbf{k} \in \mathbf{K}} \sum_{i \in \mathbf{N}} (\mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) + \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}))$$

s.a.

$$\sum_{j: (i,j) \in \mathbf{A}} \mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) - \sum_{j: (j,i) \in \mathbf{A}} \mathbf{y}_{ji}^{\mathbf{k}}(\mathbf{s}) + \mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) - \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}) = \mathbf{b}_i^{\mathbf{k}}(\mathbf{s}) \quad \forall i \in \mathbf{N}, \forall \mathbf{k} \in \mathbf{K}$$

$$\mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) + \mathbf{h}_{ij}^{\mathbf{k}}(\mathbf{s}) = \mathbf{r}^{\mathbf{k}}(\mathbf{s}) \quad \forall (i, j) \in \mathbf{A}, \forall \mathbf{k} \in \mathbf{K}$$

$$\mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) \geq 0, \mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) \geq 0, \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}) \geq 0, \mathbf{h}_{ij}^{\mathbf{k}}(\mathbf{s}) \geq 0 \quad \forall (i, j) \in \mathbf{A}, \forall \mathbf{k} \in \mathbf{K}$$

siendo

$$\mathbf{b}_i^{\mathbf{k}}(\mathbf{s}) = \begin{cases} \mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } i = \mathbf{o}(\mathbf{k}) \\ -\mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } i = \mathbf{d}(\mathbf{k}) \\ 0 & \text{en otro caso} \end{cases}$$

Este problema considera que todos los arcos de la red están activos y siempre tiene la solución factible

$$\mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) = 0, \mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) = \begin{cases} \mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } i = \mathbf{o}(\mathbf{k}) \\ 0 & \text{en otro caso} \end{cases}, \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}) = \begin{cases} \mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } i = \mathbf{d}(\mathbf{k}) \\ 0 & \text{en otro caso} \end{cases}, \mathbf{h}_{ij}^{\mathbf{k}}(\mathbf{s}) = \mathbf{r}^{\mathbf{k}}(\mathbf{s})$$

Si para algún escenario  $\mathbf{s}$  se cumple  $\mathbf{u} > 0$ , esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero simultáneamente todas las variables artificiales  $\mathbf{v}_i^{\mathbf{k}+}(\mathbf{s})$  y  $\mathbf{v}_i^{\mathbf{k}-}(\mathbf{s})$  introducidas.

En este caso se cumple que  $\mathbf{K}_2 = \emptyset$  y el problema SUNDP no tiene solución.

#### 4.1.6 Aplicación del Método L Entero al Problema SUNDP

En el caso del SUNDP, el algoritmo queda:

Paso 0 Inicializar variables  $\mathbf{v} := \mathbf{r} := \mathbf{t} := 0, \widehat{\mathbf{z}} := +\infty$ .

$\mathbf{v}$ : número de iteraciones

$\mathbf{r}$ : número de cortes de factibilidad

$\mathbf{t}$ : número de cortes de optimalidad

$\widehat{\mathbf{z}}$ : valor de la mejor solución entera hallada hasta el momento

Se crea una lista  $\mathbf{L}$  de problemas (nodos) pendientes, con un único problema inicial:

$$\min_{\mathbf{x}, \theta} \quad \sum_{(i,j) \in \mathbf{A}} \mathbf{f}_{ij} \mathbf{x}_{ij} + \theta$$

s.a.  $0 \leq \mathbf{x} \leq 1, \theta \in \mathbf{R}$

Paso 1 Si  $\mathbf{L} = \emptyset$ , FIN, la solución que correspondiente a  $\widehat{\mathbf{z}}$  es óptima. En caso contrario, seleccionar un nodo pendiente  $\mathbf{P}$  de  $\mathbf{L}$  y removerlo de la lista.

Paso 2 Establecer  $v := v + 1$ . Resolver el problema lineal maestro correspondiente a  $\mathbf{P}$  :

$$\min_{\mathbf{x}^P, \theta^P} \sum_{(i,j) \in A} f_{ij} x_{ij}^P + \theta^P$$

$$\text{s.a. } \mathbf{D}^l \mathbf{x}^P \geq \mathbf{d}^l, \quad l = 1, \dots, r \quad (\text{cortes de factibilidad}) \quad (1)$$

$$\mathbf{E}^l \mathbf{x}^P + \theta^P \geq \mathbf{e}^l, \quad l = 1, \dots, t \quad (\text{cortes de optimalidad}) \quad (2)$$

$$0 \leq \mathbf{x}^P \leq 1, \theta^P \in \mathbf{R}$$

donde  $\mathbf{x}^P$  tiene componentes fijos dados por los valores de las aristas del árbol del Branch&Cut desde la raíz hasta el nodo  $\mathbf{P}$ , y componentes variables sobre los que se hace la minimización.

Si el problema es no factible, vaciar el nodo e ir al Paso 1. Sino, sea  $(\mathbf{x}^{Pv}, \theta^{Pv})$  la solución óptima. Si no hay restricciones de optimalidad ( $t = 0$ ) se considera  $\theta^{Pv} = -\infty$  y no se toma en cuenta en el cálculo de  $\mathbf{x}^{Pv}$ .

Si  $\sum_{(i,j) \in A} f_{ij} x_{ij}^{Pv} + \theta^{Pv} \geq \hat{z}$ , vaciar el nodo  $\mathbf{P}$  e ir al Paso 1.

Paso 3 Para cada escenario  $s = 1, \dots, S$  resolver el problema de factibilidad hasta que para algún  $s$  se verifique que el valor óptimo  $\mathbf{u} > 0$  ó se tenga  $\mathbf{u} = 0 \forall s$ .

$$\min_{\mathbf{y}, \mathbf{v}, \mathbf{h}} \mathbf{u} = \sum_{k \in K} \sum_{i \in N} (v_i^{k+} + v_i^{k-}) + \sum_{k \in K} \sum_{(i,j) \in A} (v_{ij}^{k-} + v_{ij}^{k+})$$

s.a.

$$\sum_{j: (i,j) \in A} y_{ij}^k - \sum_{j: (j,i) \in A} y_{ji}^k + v_i^{k+} - v_i^{k-} = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$y_{ij}^k + h_{ij}^k + v_{ij}^{k+} - v_{ij}^{k-} = r^k(s) x_{ij}^{Pv} \quad \forall (i, j) \in A, \forall k \in K$$

$$y_{ij}^k \geq 0, v_i^{k+} \geq 0, v_i^{k-} \geq 0, h_{ij}^k \geq 0, v_{ij}^{k+} \geq 0, v_{ij}^{k-} \geq 0, \quad \forall (i, j) \in A, \forall k \in K$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Este problema siempre tiene la solución factible

$$y_{ij}^k = 0, v_i^{k+} = \begin{cases} r^k(s) & \text{si } i = o(k) \\ 0 & \text{en otro caso} \end{cases}, v_i^{k-} = \begin{cases} r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}, h_{ij}^k = r^k(s) x_{ij}^{Pv}, v_{ij}^{k+} = v_{ij}^{k-} = 0$$

Si para algún escenario  $s$  se cumple  $\mathbf{u} > 0$ , esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero todas las variables artificiales introducidas.



En este caso, sean  $\sigma_i^k$  y  $\sigma_{ij}^k$  los multiplicadores simplex óptimos del problema primal, correspondientes a las restricciones de balance y de activación de arcos respectivamente (solución óptima del problema dual asociado).

Se definen el vector  $D^{r+1} / D_{ij}^{r+1} := -\sum_k \sigma_{ij}^k r^k(s)$ , y  $d^{r+1} := \sum_{i \in N} \sum_{k \in K} \sigma_i^k b_i^k(s)$

Agregar la restricción  $D^{r+1} x^P \geq d^{r+1}$  al problema maestro en el nodo  $P$ , (corte de factibilidad).

Poner  $r := r + 1$  e ir al Paso 2.

Si  $\forall s, u = 0$ , ir al Paso 4.

**Paso 4** Verificar las restricciones de integralidad de  $x^{PV}$ . Si se viola una restricción, ramificar en la componente relevante de  $x^{PV}$  en dos nuevas ramas, una con  $x_{ij}^{PV} = 1$  y la otra con  $x_{ij}^{PV} = 0$ , creando dos nuevos subproblemas que se agregan a la lista de nodos pendientes  $L$ , e ir al Paso 1.

**Paso 5** Para  $s = 1, \dots, S$  resolver el problema lineal

$$\min_y \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k(s)$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k(s) - \sum_{j:(j,i) \in A} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$y_{ij}^k(s) + h_{ij}^k(s) = r^k(s) x_{ij}^{PV} \quad \forall (i, j) \in A, \forall k \in K$$

$$y_{ij}^k(s) \geq 0, h_{ij}^k(s) \geq 0 \quad \forall (i, j) \in A, \forall k \in K$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Sean sean  $\pi_i^{ks}$  y  $\pi_{ij}^{ks}$  los multiplicadores simplex óptimos del problema primal para el escenario  $s$  (solución óptima del problema dual asociado), correspondientes a las restricciones de balance y de activación de arcos respectivamente.

Se definen el vector  $E^{t+1} / E_{ij}^{t+1} := -\sum_{s=1}^S \sum_k p_s \pi_{ij}^{ks} r^k(s)$  y  $e^{t+1} := \sum_{s=1}^S \sum_{i \in N} \sum_{k \in K} p_s \pi_i^{ks} b_i^k(s)$

Calcular  $Q(x^{PV}) = e^{t+1} - E^{t+1} x^{PV} = \sum_{s=1}^S \sum_{i \in N} \sum_{k \in K} p_s \pi_i^{ks} b_i^k(s) + \sum_{s=1}^S \sum_{(i,j) \in A} \sum_{k \in K} p_s \pi_{ij}^{ks} r^k(s) x_{ij}^{PV}$

Actualizar la cota superior  $\hat{z} = \min \{ \hat{z}, \sum_{(i,j) \in A} f_{ij} x_{ij}^{PV} + Q(x^{PV}) \}$ . Esto es posible pues en este punto ya se tiene una solución factible entera.

**Paso 6** Si  $\theta^{pv} \geq Q(\mathbf{x}^{pv})$ , vaciar el nodo actual e ir al Paso 1. La solución  $\mathbf{x}^{pv}$  es óptima para el problema del nodo  $P$ .

En caso contrario, agregar la restricción  $E^{t+1}\mathbf{x}^P + \theta^P \geq e^{t+1}$  al problema maestro en  $P$  (corte de optimalidad).

Poner  $t := t + 1$  e ir al Paso 2.

En el *Anexo A* se presenta un ejemplo de aplicación del método al problema SUNDP.

### **Observación sobre el caso particular $ISI = 1$**

El caso del problema determinístico UNDP puede resolverse como una aplicación particular de este algoritmo con un único escenario  $s$  con probabilidad 1, siendo conocida esta técnica en el caso determinístico con el nombre de *Algoritmo de Descomposición de Benders*.

### **4.1.7 Valor de la Solución Estocástica en el Caso del Problema SUNDP**

Para el SUNDP, y considerando la definición de la función objetivo  $\mathbf{z}$  del problema paramétrico en  $\xi$  de la sección 3.4.1 y la formulación del modelo estocástico en dos etapas de 4.1.2, se tiene

$$\mathbf{z}(\mathbf{x}, \xi) = \sum_{(i,j) \in A} f_{ij} x_{ij} + Q(\mathbf{x}, \xi) \quad , \quad x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A$$

con

$$Q(\mathbf{x}, \xi(s)) = \min_y \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k(s)$$

sujeto a

$$\sum_{j:(0,j) \in A} y_{ij}^k(s) - \sum_{j:(j,0) \in A} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$y_{ij}^k(s) \leq r^k(s) x_{ij} \quad \forall (i, j) \in A, \forall k \in K$$

$$y_{ij}^k(s) \geq 0 \quad \forall (i, j) \in A, \forall k \in K$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

**Lema 1** Para el SUNDP, se verifica que  $EEV = EV$ .

Sean  $\mathbf{x}^*(\bar{\xi})$  y  $\mathbf{z}(\mathbf{x}^*(\bar{\xi}), \bar{\xi})$  la solución óptima y el valor óptimo respectivamente del problema de valor esperado para el SUNDP, es decir  $EV = \mathbf{z}(\mathbf{x}^*(\bar{\xi}), \bar{\xi})$ , donde a cada commodity  $k$  se le asigna una demanda media  $\bar{r}^k = \sum_{s=1}^S p_s r^k(s)$ .

Sea la subred solución del problema  $EV$  dada por  $\bar{A} = \{(i,j) \in A / x_{ij} = 1 \text{ en } \mathbf{x}^*(\bar{\xi})\}$ .

Para cada commodity  $k$ , se define  $C^k$  como el camino que sigue el flujo de dicha commodity

en la solución de valor esperado, es decir  $C^k = \{(i,j) \in \bar{A} / y_{ij}^k = \bar{r}^k \text{ en } z(x^*(\bar{\xi}), \bar{\xi})\}$

Puede escribirse entonces  $EV = z(x^*(\bar{\xi}), \bar{\xi}) = \sum_{(i,j) \in \bar{A}} f_{ij} + \sum_{k \in K} \sum_{(i,j) \in C^k} c_{ij}^k \bar{r}^k$

Se considera ahora el cálculo de **EEV**, valor esperado de la solución de valor esperado, que puede plantearse como

$$EEV = E_{\xi} [z(x^*(\xi), \xi)] = \sum_{(i,j) \in \bar{A}} f_{ij} + E_{\xi} [Q(x^*(\xi), \xi(s))] = \sum_{(i,j) \in \bar{A}} f_{ij} + \sum_{s=1}^S p_s Q(x^*(\xi), \xi(s))$$

con  $Q(x, \xi(s)) = \min_y \sum_{k \in K} \sum_{(i,j) \in \bar{A}} c_{ij}^k y_{ij}^k(s)$

sujeto a

$$\sum_{j:(i,j) \in \bar{A}} y_{ij}^k(s) - \sum_{j:(j,i) \in \bar{A}} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$y_{ij}^k(s) \leq r^k(s) \quad \forall (i, j) \in \bar{A}, \forall k \in K$$

$$y_{ij}^k(s) \geq 0 \quad \forall (i, j) \in \bar{A}, \forall k \in K$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

De esta formulación del problema **EEV** puede observarse que:

- La componente de costos fijos de **EEV** es la misma que la del problema **EV**, pues en los dos casos se tienen los mismos arcos activos de la red  $\bar{A}$  solución de **EV**.
- La componente de costos variables de **EEV** implica que para cada escenario  $s$  hay que resolver  $|K|$  problemas independientes, en donde para cada commodity  $k$  con demanda  $r^k(s)$  hay que hallar el camino de costo mínimo entre su origen y su destino sobre la red fija  $\bar{A}$  solución del problema **EV**. El resultado básico que se verifica es que este camino es el mismo camino óptimo  $C^k$  para dicha commodity  $k$  en el problema **EV**, en donde se trabaja con su valor medio de demanda  $\bar{r}^k$ . Esto es así debido a que la red es fija y la misma en ambos casos, con iguales costos variables en cada arco activo, siendo la demanda de la commodity a transportar lo único que varía de un caso al otro; y se sabe que, para una red fija sin restricciones de capacidad en los arcos, si un camino es óptimo para una cierta demanda de una commodity  $k$ , entonces dicho camino es óptimo para cualquier otro valor no nulo de dicha commodity.

Esto implica que el problema **EEV** pueda escribirse como

$$EEV = \sum_{(i,j) \in \bar{A}} f_{ij} + \sum_{s=1}^S p_s \sum_{k \in K} \sum_{(i,j) \in C^k} c_{ij}^k r^k(s) = \sum_{(i,j) \in \bar{A}} f_{ij} + \sum_{k \in K} \sum_{(i,j) \in C^k} c_{ij}^k \sum_{s=1}^S p_s r^k(s) = EV$$

que es el resultado que se quería demostrar. □

**Teorema 2** Para el SUNDP, se verifica que  $VSS = 0$

Se considera ahora la red formada por la solución óptima del problema estocástico  $\mathbf{x}^*$ , se tiene que dicha solución es factible para el problema de valor esperado  $EV$ , por lo que se verifica necesariamente que  $EV = \mathbf{z}(\mathbf{x}^*(\bar{\xi}), \bar{\xi}) \leq \mathbf{z}(\mathbf{x}^*, \bar{\xi})$ .

Asimismo, con idéntico argumento al empleado en la demostración de la igualdad  $EEV = EV$ , se infiere que  $\mathbf{z}(\mathbf{x}^*, \bar{\xi}) = E_{\xi} [\mathbf{z}(\mathbf{x}^*, \xi)]$

Empleando estos dos resultados conjuntamente con el Lema 1 se obtiene

$$EEV = EV = \mathbf{z}(\mathbf{x}^*(\bar{\xi}), \bar{\xi}) \leq \mathbf{z}(\mathbf{x}^*, \bar{\xi}) = E_{\xi} [\mathbf{z}(\mathbf{x}^*, \xi)] = \mathbf{z}(\mathbf{x}^*) = RP$$

Se tiene entonces que para el SUNDP se verifica la propiedad  $EEV \leq RP$ .

Como para todo problema estocástico se cumple  $RP \leq EEV$ , se concluye que para el SUNDP se cumple  $RP = EEV$ .

En consecuencia, para el SUNDP se tiene que  $VSS_{SUNDP} = EEV - RP = 0$  □

La importante propiedad de que el valor de la solución estocástica sea nulo para el SUNDP, implica que para dicho problema no tiene valor agregado resolver el problema estocástico, resultando suficiente resolver el problema determinístico correspondiente a las demandas medias de las commodities para obtener el diseño óptimo de la red.

## 4.2 Problema Estocástico Con Restricciones de Capacidad

El caso del problema de diseño óptimo de red con restricciones de capacidad en los arcos y en el contexto de incertidumbre se denomina SCNDP (Stochastic Capacitated Network Design Problem)

### 4.2.1 Formulación Extensiva del Problema SCNDP

Sean las variables

$$\mathbf{x}_{ij} = \begin{cases} 1 & \text{si se usa el arco } (i, j) \\ 0 & \text{en otro caso} \end{cases}$$

$y_{ij}^{ks}$  = flujo de la commodity  $k$  en el arco  $(i, j)$  y en el escenario  $s$  de demandas

El modelo extensivo del problema SCNDP puede formularse como

$$[SCNDP] \mathbf{z} = \min_{\mathbf{x}, \mathbf{y}} \sum_{(i,j) \in A} f_{ij} \mathbf{x}_{ij} + \sum_{s \in S} \sum_{k \in K} \sum_{(i,j) \in A} p_s c_{ij}^k y_{ij}^{ks}$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^{ks} - \sum_{j:(j,i) \in A} y_{ji}^{ks} = b_i^{ks} \quad \forall i \in N, \forall k \in K, \forall s \in S \quad (1)$$

$$\sum_{k \in K} y_{ij}^{ks} \leq u_{ij} x_{ij} \quad \forall (i, j) \in A, \forall s \in S \quad (2)$$

$$y_{ij}^{ks} \leq g_{ij}^{ks} x_{ij} \quad \forall (i, j) \in A, \forall k \in K, \forall s \in S \quad (3)$$

$$y_{ij}^{ks} \geq 0 \quad \forall (i, j) \in A, \forall k \in K, \forall s \in S \quad (4)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (5)$$

siendo

$$g_{ij}^{ks} = \min \{u_{ij}, r^{ks}\}, \quad b_i^{ks} = \begin{cases} r^{ks} & \text{si } i = o(k) \\ -r^{ks} & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

#### 4.2.2 Formulación del Modelo en Dos Etapas del Problema SCNDP

En el caso del SCNDP, en el modelo en dos etapas se emplea la formulación sin las restricciones redundantes de activación de arcos (formulación "débil"), pues en este esquema dichas restricciones no suponen ventajas desde el punto de vista del método de resolución.

El modelo queda:

$$[\text{SCNDP}] \quad z = \min_x \sum_{(i,j) \in A} f_{ij} x_{ij} + \sum_{s=1}^S p_s Q(x, \xi(s))$$

sujeto a

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (1)$$

siendo el problema de segunda etapa

$$Q(x, \xi(s)) = \min_y \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k(s)$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k(s) - \sum_{j:(j,i) \in A} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K \quad (2)$$

$$\sum_{k \in K} y_{ij}^k(s) \leq u_{ij} x_{ij} \quad \forall (i, j) \in A \quad (3)$$

$$y_{ij}^k(s) \geq 0 \quad \forall (i, j) \in A, \forall k \in K \quad (4)$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

### Observaciones:

Identificando términos con relación a la formulación general en 3.1, para el problema de *primera etapa* se verifica que:

- No hay restricciones determinísticas que no dependan de la incertidumbre en la demanda (matriz  $\mathbf{A}$  y vector  $\mathbf{b}$  nulos)
- Las variables de primera etapa son binarias

Para el problema de *segunda etapa*, se tiene que:

- La matriz de recurso  $\mathbf{W}$  es fija, dado que todos los coeficientes de las variables de segunda etapa son fijos en todas las restricciones, con valores de 0, 1 ó -1.
- El vector  $\mathbf{q}$  de costos no tiene incertidumbre asociada
- Los componentes no nulos de la matriz tecnológica  $\mathbf{T}$  son las capacidades de los arcos  $\mathbf{u}_{ij}$  y las demandas aleatorias  $\mathbf{r}^k(\omega)$ .
- Los componentes no nulos del vector de requerimientos  $\mathbf{h}$  son las demandas aleatorias  $\mathbf{r}^k(\omega)$ , a menos del signo.

### 4.2.3 Propiedades del Modelo en Dos Etapas del Problema SCNDP

Como para el SCNDP la matriz de recurso  $\mathbf{W}$  es fija y el vector aleatorio  $\xi$  es discreto y tiene soporte  $\Xi$  finito, y como las restricciones de integralidad se dan solamente en las variables de primera etapa, se verifica que:

- $Q(\mathbf{x}, \xi)$  es una función lineal a trozos y convexa de  $\mathbf{x}$  en  $\mathbf{K}_2$
- $Q(\mathbf{x})$  es la suma ponderada con coeficientes positivos de las realizaciones de  $\xi$
- $\mathbf{K}_2$  es poliédrico, cerrado y convexo
- $Q(\mathbf{x})$  es una función lineal a trozos y convexa en  $\mathbf{K}_2$

Para  $\mathbf{x}$  y  $\xi$  fijos, el problema de segunda etapa es un problema de flujo óptimo en red de un conjunto de commodities (multicommodity network flow problem), definido sobre la subred formada por los arcos  $\{(\mathbf{i}, \mathbf{j}) \in \mathbf{A} / \mathbf{x}_{ij} = 1\}$

### 4.2.4 Factibilidad del Modelo en Dos Etapas del Problema SCNDP

El PED para el SCNDP se puede expresar como

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{(\mathbf{i}, \mathbf{j}) \in \mathbf{A}} \mathbf{f}_{ij} \mathbf{x}_{ij} + Q(\mathbf{x}) \\ \text{sujeto a} \quad & \mathbf{x} \in \mathbf{K}_1 \cap \mathbf{K}_2 \end{aligned}$$

La factibilidad del problema de segunda etapa no depende exclusivamente de los valores de las variables de primera etapa  $\mathbf{x}$ , sino que requiere que se cumpla la siguiente condición:

Debe poder existir en la red base original una distribución de flujos de las commodities que asegure

- a) Se satisfaga la demanda ente todo par (origen, destino) de cada commodity
- b) El flujo total por cada arco debe ser menor o igual a su capacidad de transporte

Esta condición implica la existencia en la red de base original de camino dirigido entre todo nodo origen y destino de una misma commodity, condición que en el caso del SUNDP era suficiente para garantizar factibilidad, y en el caso del SCNDP es sólo condición necesaria.

Se verifica que el problema SCNDP no es un problema con *recurso relativamente completo*, pues no se satisface la condición  $\mathbf{K}_1 \subset \mathbf{K}_2$ . Es decir que no se cumple que toda decisión de primera etapa sea factible para la segunda etapa.

El efecto de  $\mathbf{x}$  sobre la factibilidad del problema de segunda etapa se da a través de las restricción (3) del SCNDP.

#### 4.2.5 Existencia de la Solución Óptima del Problema SCNDP

Como el vector aleatorio  $\xi$  tiene momentos de segundo orden finitos y la región factible es acotada, dado que el conjunto  $\mathbf{K}_1$  lo es al ser finito el número de arcos de la red base y tomar estos valores binarios, se asegura que si  $\mathbf{K}_1 \cap \mathbf{K}_2$  es no vacío, el PED tiene solución óptima finita.

La condición para que haya óptimo finito pasa a ser entonces la que asegura la factibilidad del problema de segunda etapa para al menos algún valor de las variables de primera etapa, lo que es equivalente a que el siguiente problema lineal tenga solución óptima nula ( $\mathbf{u} = 0$ ) para todo escenario  $\mathbf{s}$ :

$$\begin{aligned} \min_{\mathbf{y}, \mathbf{v}, \mathbf{u}} \quad & \sum_{\mathbf{k} \in \mathbf{K}} \sum_{\mathbf{i} \in \mathbf{N}} (\mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) + \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s})) \\ \text{s.a.} \quad & \sum_{\mathbf{j}: (\mathbf{i}, \mathbf{j}) \in \mathbf{A}} \mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) - \sum_{\mathbf{j}: (\mathbf{j}, \mathbf{i}) \in \mathbf{A}} \mathbf{y}_{ji}^{\mathbf{k}}(\mathbf{s}) + \mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) - \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}) = \mathbf{b}_i^{\mathbf{k}}(\mathbf{s}) \quad \forall \mathbf{i} \in \mathbf{N}, \forall \mathbf{k} \in \mathbf{K} \\ & \sum_{\mathbf{k} \in \mathbf{K}} \mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) + \mathbf{h}_{ij}(\mathbf{s}) = \mathbf{u}_{ij} \quad \forall (\mathbf{i}, \mathbf{j}) \in \mathbf{A} \\ & \mathbf{y}_{ij}^{\mathbf{k}}(\mathbf{s}) \geq 0, \mathbf{v}_i^{\mathbf{k}+}(\mathbf{s}) \geq 0, \mathbf{v}_i^{\mathbf{k}-}(\mathbf{s}) \geq 0, \mathbf{h}_{ij}(\mathbf{s}) \geq 0 \quad \forall (\mathbf{i}, \mathbf{j}) \in \mathbf{A}, \forall \mathbf{k} \in \mathbf{K} \end{aligned}$$

siendo

$$\mathbf{b}_i^{\mathbf{k}}(\mathbf{s}) = \begin{cases} \mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } \mathbf{i} = \mathbf{o}(\mathbf{k}) \\ -\mathbf{r}^{\mathbf{k}}(\mathbf{s}) & \text{si } \mathbf{i} = \mathbf{d}(\mathbf{k}) \\ 0 & \text{en otro caso} \end{cases}$$

Este problema considera que todos los arcos de la red están activos y siempre tiene la solución factible

$$\mathbf{y}_{ij}^k(s)=0, \mathbf{v}_i^{k+}(s)= \begin{cases} \mathbf{r}^k(s) & \text{si } \mathbf{i}=\mathbf{o}(k) \\ 0 & \text{en otro caso} \end{cases}, \mathbf{v}_i^{k-}(s)= \begin{cases} \mathbf{r}^k(s) & \text{si } \mathbf{i}=\mathbf{d}(k) \\ 0 & \text{en otro caso} \end{cases}, \mathbf{h}_{ij}(s)=\mathbf{u}_{ij}$$

Si para algún escenario  $s$  se cumple  $\mathbf{u} > 0$ , esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero simultáneamente todas las variables artificiales  $\mathbf{v}_i^{k+}(s)$  y  $\mathbf{v}_i^{k-}(s)$  introducidas.

En este caso se cumple que  $\mathbf{K}_2 = \emptyset$  y el problema SCNDP no tiene solución.

#### 4.2.6 Aplicación del Método de Cortes Múltiples Entero al SCNDP

En el caso del SCNDP, el algoritmo de Cortes Múltiples queda:

Paso 0 Inicializar variables  $\mathbf{v} := \mathbf{r} := \mathbf{t}(s) := 0, s = 1, \dots, \mathbf{S}, \widehat{\mathbf{z}} := +\infty$ .

$\mathbf{v}$ : número de iteraciones

$\mathbf{r}$ : número de cortes de factibilidad

$\mathbf{t}(s)$ : número de cortes de optimalidad en el escenario  $s$

$\widehat{\mathbf{z}}$ : valor de la mejor solución entera hallada hasta el momento

Se crea una lista  $\mathbf{L}$  de problemas (nodos) pendientes, con un único problema inicial:

$$\min_{\mathbf{x}, \theta_s} \sum_{(i,j) \in \mathbf{A}} \mathbf{f}_{ij} \mathbf{x}_{ij} + \sum_{s=1}^{\mathbf{S}} \theta_s$$

s.a.  $0 \leq \mathbf{x} \leq 1, \theta_1, \dots, \theta_s \in \mathbf{R}$

Paso 1 Si  $\mathbf{L} = \emptyset$ , FIN, la solución que correspondiente a  $\widehat{\mathbf{z}}$  es óptima. En caso contrario, seleccionar un nodo pendiente  $\mathbf{P}$  de  $\mathbf{L}$  y removerlo de la lista.

Paso 2 Establecer  $\mathbf{v} := \mathbf{v} + 1$ . Resolver el problema lineal maestro correspondiente a  $\mathbf{P}$ :

$$\min_{\mathbf{x}^P, \theta_s^P} \sum_{(i,j) \in \mathbf{A}} \mathbf{f}_{ij} \mathbf{x}_{ij}^P + \sum_{s=1}^{\mathbf{S}} \theta_s^P$$

s.a.  $\mathbf{D}^l \mathbf{x}^P \geq \mathbf{d}^l, \quad \mathbf{l} = 1, \dots, \mathbf{r} \quad (\text{cortes de factibilidad}) \quad (1)$

$\mathbf{E}^{l(s)} \mathbf{x}^P + \theta_s^P \geq \mathbf{e}^{l(s)}, \quad \mathbf{l}(s) = 1, \dots, \mathbf{t}(s) \quad (\text{cortes de optimalidad}) \quad (2)$

$0 \leq \mathbf{x}^P \leq 1, \theta_1^P, \dots, \theta_s^P \in \mathbf{R}$

donde  $\mathbf{x}^P$  tiene componentes fijos dados por los valores de las aristas del árbol del Branch&Cut desde la raíz hasta el nodo  $\mathbf{P}$ , y componentes variables sobre los que se hace la minimización.

Si el problema es no factible, vaciar el nodo e ir al Paso 1. Sino, sea  $(\mathbf{x}^{Pv}, \theta_1^{Pv}, \dots, \theta_s^{Pv})$  la solución óptima. Si no hay restricciones de optimalidad ( $\mathbf{t}(s) = 0 \forall s$ ) se considera  $\theta_s^{Pv} = -\infty \forall s$  y no se toman en cuenta en el cálculo de  $\mathbf{x}^{Pv}$ .

Si  $\sum_{(i,j) \in \mathbf{A}} \mathbf{f}_{ij} \mathbf{x}_{ij}^{Pv} + \sum_{s=1}^{\mathbf{S}} \theta_s^{Pv} \geq \widehat{\mathbf{z}}$ , vaciar el nodo  $\mathbf{P}$  e ir al Paso 1.



**Paso 3** Para cada escenario  $s = 1, \dots, S$  resolver el problema de factibilidad hasta que para algún  $s$  se verifique que el valor óptimo  $f > 0$  ó se tenga  $f = 0 \forall s$ .

$$\min_{y, v, h} f = \sum_{k \in K} \sum_{i \in N} (v_i^{k+} + v_i^{k-}) + \sum_{(i,j) \in A} (v_{ij}^+ + v_{ij}^-)$$

s.a.

$$\sum_{j: (i,j) \in A} y_{ij}^k - \sum_{j: (j,i) \in A} y_{ji}^k + v_i^{k+} - v_i^{k-} = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$\sum_{k \in K} y_{ij}^k + h_{ij} + v_{ij}^+ - v_{ij}^- = u_{ij} x_{ij}^{pv} \quad \forall (i, j) \in A$$

$$y_{ij}^k \geq 0, v_i^{k+} \geq 0, v_i^{k-} \geq 0, h_{ij} \geq 0, v_{ij}^+ \geq 0, v_{ij}^- \geq 0, \quad \forall (i, j) \in A$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Este problema siempre tiene la solución factible

$$y_{ij}^k = 0, v_i^{k+} = \begin{cases} r^k(s) & \text{si } i = o(k) \\ 0 & \text{en otro caso} \end{cases}, v_i^{k-} = \begin{cases} r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}, h_{ij} = u_{ij} x_{ij}^{pv}, v_{ij}^+ = v_{ij}^- = 0$$

Si para algún escenario  $s$  se cumple  $f > 0$ , esto implica que el problema de segunda etapa correspondiente a dicho escenario no es factible, al no ser cero todas las variables artificiales introducidas.

En este caso, sean  $\sigma_i^k$  y  $\sigma_{ij}$  los multiplicadores simplex óptimos del problema primal, correspondientes a las restricciones de balance y de capacidad de arcos respectivamente (solución óptima del problema dual asociado).

Se definen el vector  $D^{r+1} / D_{ij}^{r+1} := -\sigma_{ij} u_{ij}$  y  $d^{r+1} := \sum_{i \in N} \sum_{k \in K} \sigma_i^k b_i^k(s)$

Agregar la restricción  $D^{r+1} x^p \geq d^{r+1}$  al problema maestro en el nodo  $P$ , (corte de factibilidad).

Poner  $r := r + 1$  e ir al Paso 2.

Si  $\forall s, f = 0$ , ir al Paso 4.

**Paso 4** Verificar las restricciones de integralidad de  $x^{pv}$ . Si se viola una restricción, ramificar en la componente relevante de  $x^{pv}$  en dos nuevas ramas, una con  $x_{ij}^{pv} = 1$  y la otra con  $x_{ij}^{pv} = 0$ , creando dos nuevos subproblemas que se agregan a la lista de nodos pendientes  $L$ , e ir al Paso 1.

Paso 5 Para  $s = 1, \dots, S$  resolver el problema lineal

$$\min_y \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k y_{ij}^k(s)$$

sujeto a

$$\sum_{j:(i,j) \in A} y_{ij}^k(s) - \sum_{j:(j,i) \in A} y_{ji}^k(s) = b_i^k(s) \quad \forall i \in N, \forall k \in K$$

$$\sum_{k \in K} y_{ij}^k(s) + h_{ij}(s) = u_{ij} x_{ij}^{pv} \quad \forall (i, j) \in A$$

$$y_{ij}^k(s) \geq 0, h_{ij}(s) \geq 0 \quad \forall (i, j) \in A, \forall k \in K$$

siendo

$$b_i^k(s) = \begin{cases} r^k(s) & \text{si } i = o(k) \\ -r^k(s) & \text{si } i = d(k) \\ 0 & \text{en otro caso} \end{cases}$$

Sean sean  $\pi_i^{ks}$  y  $\pi_{ij}^s$  los multiplicadores simplex óptimos del problema primal para el escenario  $s$  (solución óptima del problema dual asociado), correspondientes a las restricciones de balance y de capacidad de arcos respectivamente.

Se definen  $E^{t(s)+1} \mid E_{ij}^{t(s)+1} := -p_s \pi_{ij}^s u_{ij}$  y  $e^{t(s)+1} := \sum_{i \in N} \sum_{k \in K} p_s \pi_i^{ks} b_i^k(s)$

$$\text{Calcular } Q(x^{pv}) = \sum_{s=1}^S (e^{t(s)+1} - E^{t(s)+1} x^{pv}) = \sum_{s=1}^S \sum_{i \in N} \sum_{k \in K} p_s \pi_i^{ks} b_i^k(s) + \sum_{s=1}^S \sum_{(i,j) \in A} p_s \pi_{ij}^s u_{ij} x_{ij}^{pv}$$

Actualizar la cota superior  $\bar{z} = \min \{ \bar{z}, \sum_{(i,j) \in A} f_{ij} x_{ij}^{pv} + Q(x^{pv}) \}$ . Esto es posible pues en este punto ya se tiene una solución factible entera.

Paso 6 Para  $s = 1, \dots, S$ :

Si  $\theta_s^{pv} < e^{t(s)+1} - E^{t(s)+1} x^{pv}$ , agregar la restricción  $E^{t(s)+1} x^p + \theta_s^p \geq e^{t(s)+1}$  al problema maestro en  $P$  (corte de optimalidad) y establecer  $t(s) := t(s) + 1$ .

Si  $\theta_s^{pv} \geq e^{t(s)+1} - E^{t(s)+1} x^{pv} \quad \forall s = 1, \dots, S$ , entonces la solución  $x^{pv}$  es óptima para el problema del nodo  $P$ . Vaciar el nodo actual  $P$  e ir al Paso 1. En caso contrario, ir al Paso 2.

En la tesis doctoral de Morten Riis [16], puede encontrarse un algoritmo de resolución del problema de diseño de red con restricciones de capacidad basado en el Método de Cortes Múltiples Entero.

#### **4.2.7 Valor de la Solución Estocástica en el Caso del Problema SCNDP**

En el caso del problema de diseño de red con restricciones de capacidad de transporte en los arcos, no hay un resultado general en cuanto al valor de la solución estocástica  $VSS$ , debiéndose investigar caso a caso el valor agregado de emplear la técnica de programación estocástica respecto a resolver el problema determinístico de valor esperado  $EV$ .

Asumiendo que problema estocástico  $RP$  tiene solución óptima, pueden verificarse los siguientes casos:

- ❑ La solución del problema de valor esperado  $EV$  no es factible para el problema estocástico. Esto puede considerarse como que  $VSS = +\infty$
- ❑ La solución del problema de valor esperado  $EV$  es factible pero no óptima para el problema  $RP$ , por lo que se tiene  $VSS > 0$  finito.
- ❑ La solución óptima del problema de valor esperado  $EV$  coincide con la solución óptima del problema de programación estocástica  $RP$ , y se tiene  $VSS = 0$ .

En el *Anexo B* se muestran ejemplos de cada uno de estos casos.

## 5 Implementación de Casos de Prueba de Diseño

### 5.1 Implementación en GAMS – OSL

El problema estocástico se formula en su forma extensiva, y se resuelve utilizando software comercial estándar.

La formulación se hace en GAMS [4] y luego se convierte a formato MPS, que se utiliza como entrada del solver MIP de la versión 3.0 del paquete OSL [11], corriendo en una plataforma SPARC/SOLARIS.

GAMS (General Algebraic Modeling System) es un potente lenguaje de alto nivel de modelación algebraica de uso general.

MPS (Mathematical Programming System) es un formato matricial de especificación de problemas de optimización con nivel de estándar de la industria.

OSL (Optimization Solutions Guide) es un software de optimización matemática desarrollado por IBM.

La formulación del problema y sus resultados se almacenan en archivos, y su proceso de generación es como indica la Figura 5.1.



Figura 5.1 Esquema de Resolución en GAMS-OSL

Las estructuras de datos en GAMS empleadas para representar tanto el problema SUNDP como el SCNDP se presentan en el *Anexo C.1*. Un ejemplo concreto de formulación del modelo se describe en el *Anexo C.2*.

El problema de factibilidad tanto del SUNDP como del SCNDP se codifica en GAMS en forma iterativa (se plantea un problema de factibilidad para cada escenario por separado), quedando problemas lineales que son resueltos por el solver BDMPL de GAMS, El esquema correspondiente aparece en el *Anexo C.3*.

Puede también plantearse el problema de factibilidad en forma extensiva, en cuyo caso se resuelve un problema único donde todos los escenarios se consideran en forma conjunta.

### 5.1.1 Caso de Diseño 1

El primer ejemplo esta basado en el caso denominado *cexa2*, presentado por Holmberg [10].

#### **Topología de la red origen**

Consiste en el diseño de una red con 7 nodos, con 42 arcos potenciales.

En este caso la red de arcos potenciales es un grafo completo, con arco dirigido entre todo par de nodos.

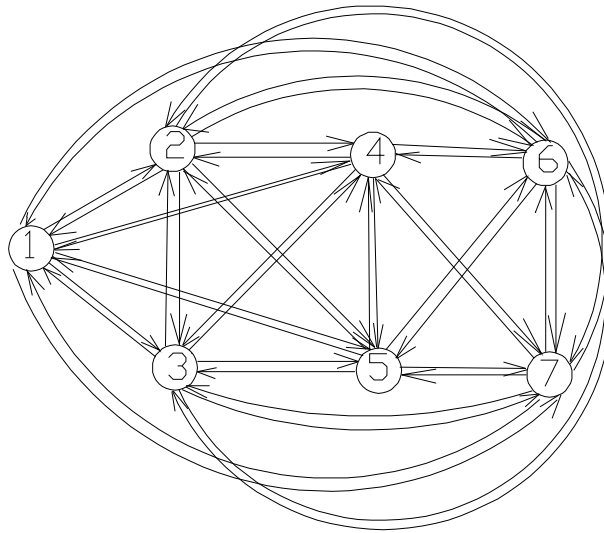


Figura 5.2 Red Potencial del Caso de Prueba 1

#### **Costos**

Los costos fijos son iguales para todos los arcos y de valor 4000.

Los costos variables unitarios en un mismo arco son iguales para todas las commodities, y varían entre 0 y 19, con un promedio de 10.

Este caso de prueba tiene una relación costo fijo/costo variable alta, del orden de 80.

En el cálculo de este último indicador se hace el promedio del costo fijo de cada arco dividido su costo variable unitario multiplicado por la demanda media de todas las commodities.

#### **Capacidades de los arcos**

Varían entre 60 y 90, con valor medio 75.

### Modelo de demanda incierta

Cada nodo tiene asociadas 6 commodities con origen en ese nodo, cada una de ellas con destinos distintos en los restantes 6 nodos. Es decir que  $|K| = 42$ .

Para modelar la demanda incierta se supone una distribución gaussiana de las cargas idéntica en cada nodo, que se aproxima mediante una distribución binomial discreta con parámetros  $n = 20$  y  $p = 0.5$ . Esta distribución tiene valor medio 10 y varianza 5.

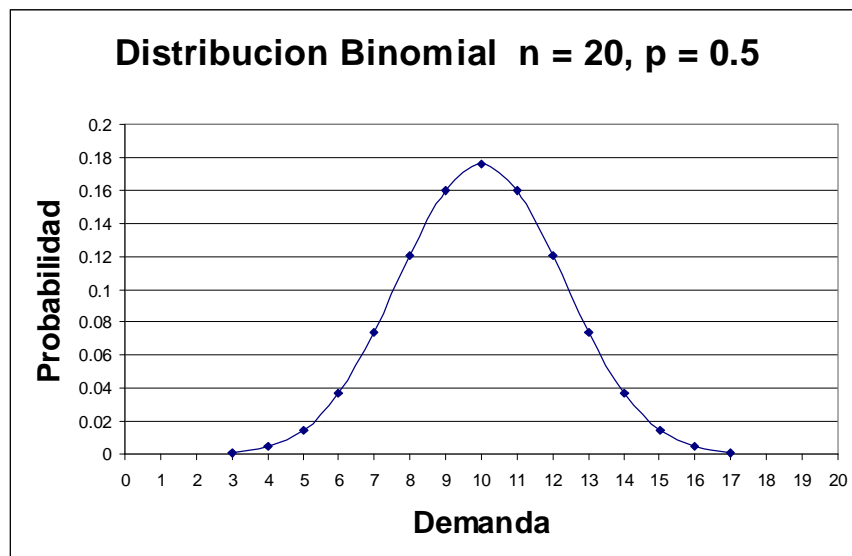


Figura 5.3 Distribución de Probabilidad del Caso 1

Las probabilidades asociadas a cada demanda  $r$  en la distribución binomial se calculan como

$$P(r) = \binom{n}{r} p^r (1 - p)^{n-r}$$

Se supone una distribución conjunta de las demandas de las commodities que hace que todas tengan simultáneamente el mismo valor.

Para armar el caso de prueba, se consideran 11 escenarios de demanda entera, correlativos desde el valor 5 hasta 15.

En el ejemplo *cexa2* de Holmberg se utiliza una demanda unitaria para todas las commodities y un costo fijo de arcos de 400. En este caso, se toma una demanda media de 10, y para mantener la relación entre costos fijos y variables del caso de prueba original, se escalan por 10 los costos fijos de los arcos, así como las capacidades de los mismos.

La codificación GAMS con los datos de base de este caso y los resultados obtenidos por el solver MIP de OSL se muestran en el *Anexo D*.

## **Análisis de resultados del Caso 1 - Problema SUNDP**

### **Solución óptima estocástica**

En el caso sin restricciones de capacidad se llega a una solución óptima del problema estocástico con la selección de variables de primera etapa  $x_{ij}$  de la Figura 5.4.

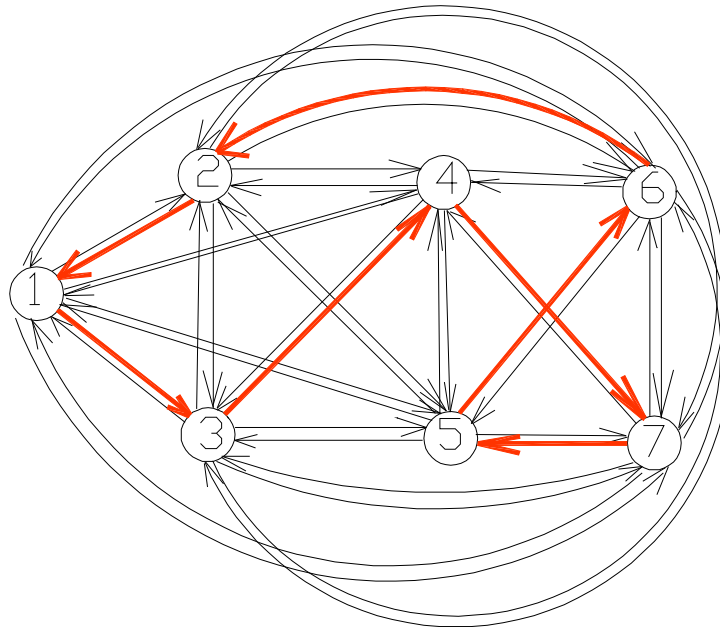


Figura 5.4 Solución óptima del SUNDP para el Caso 1

La solución óptima es el ciclo  $(1,3) \rightarrow (3,4) \rightarrow (4,7) \rightarrow (7,5) \rightarrow (5,6) \rightarrow (6,2) \rightarrow (2,1)$

El que la solución sea un ciclo es explicable pues como cada nodo es origen y a la vez destino de commodities, debe tener al menos un arco saliente y otro incidente a él, por lo que necesariamente cualquier solución tiene como mínimo tantos arcos como nodos, lo que se cumple para el ciclo de la figura.

Se observa que la pérdida de cualquier arco o link del ciclo implica en este caso que se pierda exactamente el 50% de las conexiones entre las parejas (origen, destino) de cada commodity  $k$ .

Se tiene en este caso que los costos fijos representan un 87% del costo de la solución óptima, siendo el 13% restante imputable a los costos esperados de transporte.

### **Solución óptima del problema relajado**

Se calcula la relajación lineal del problema, obteniéndose una solución óptima con valor óptimo 0.7% inferior al valor óptimo del caso entero. Se tiene que para el caso relajado las variables de primera etapa  $x_{ij}$  toman valores no enteros, como se indica en en la Figura 5.5

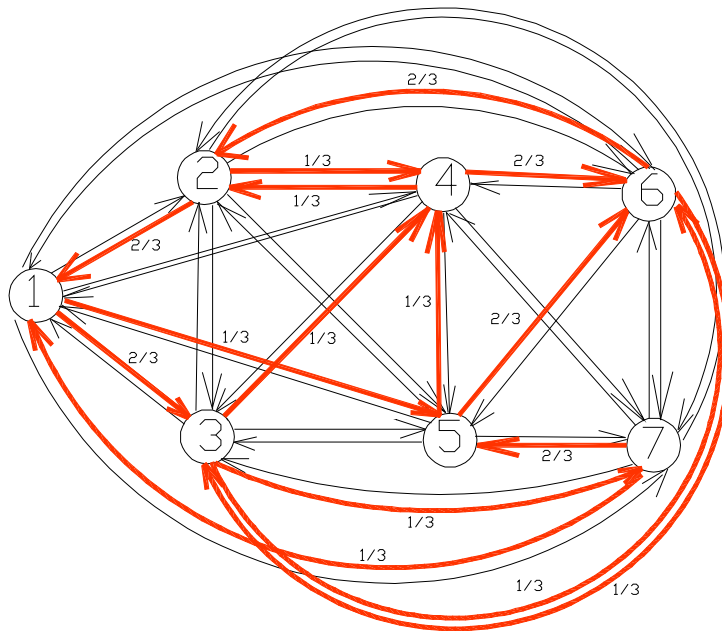


Figura 5.5 Solución óptima de la Relajación del SUNDP del Caso 1

Se observa que la suma de los pesos de los arcos salientes de cada nodo es igual a la unidad, y como los costos fijos son iguales para todos los arcos se tiene que el costo fijo de los arcos de esta solución es igual al costo fijo de los arcos de la solución estocástica, donde se tiene un único arco saliente por nodo.

### **Solución de valor esperado**

Se corrió el problema determinístico para el valor de demanda media para todas las commodities (problema **EV**), resultando la solución óptima de valor esperado coincidente con el diseño óptimo estocástico, como era de esperar para el problema SUNDP de acuerdo al Teorema 2 de 4.1.7.



### **Sensibilidad a la relación costo fijo/ costo variable de la solución**

Se modela el mismo caso de prueba pero con costos fijos de valor 400, 10 veces menor que en caso original, lo que implica bajar la relación costo fijo/costo variable en un factor de 10.

Se obtiene una solución óptima del problema estocástico ligeramente distinta, que se muestra en la Figura 5.6.

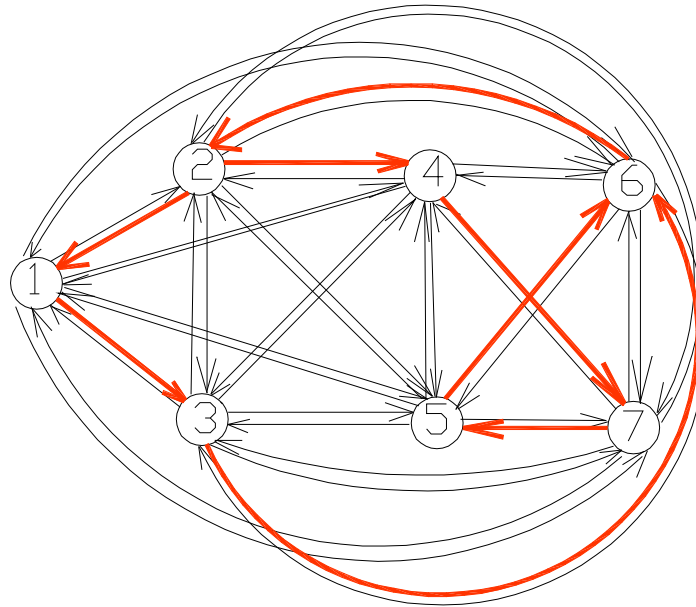


Figura 5.6 Solución del SUNDP con costos fijos reducidos

Se observa que esta solución tiene un arco más que la anterior, existiendo dos arcos salientes del nodo 2. Es decir que hay un arco más que los estrictamente necesarios para asegurar la conectividad entre todo par (origen,destino) de cada commodity, cuyo costo fijo adicional es compensado por las economías que se obtienen al tener menores costos de transporte.

Se comprueba además que esta solución es más robusta que la anterior frente a una contingencia de arco único, pues se tienen los ciclos  $(2,4) \rightarrow (4,7) \rightarrow (7,5) \rightarrow (5,6) \rightarrow (6,2)$  y  $(2,1) \rightarrow (1,3) \rightarrow (3,6) \rightarrow (6,2)$ , que aseguran la conectividad entre todos los nodos del ciclo frente a fallas en cualquier arco no perteneciente al ciclo.

Aquí los costos fijos representan el 52% del costo total, siendo 48% el peso de los costos del tráfico, es decir que las proporciones están equilibradas.

De nuevo en este caso la solución de valor esperado (problema **EV**) coincide con la solución óptima estocástica.

## **Análisis de resultados del Caso 1 - Problema SCNDP**

### **Solución estocástica**

En el caso con restricciones de capacidad, el solver MIP de OSL con los criterios de parada fijados por defecto (número máximo de iteraciones, tiempo de CPU), no llega en este caso a una solución óptima del problema estocástico, proporcionando la solución de la Figura 5.7.

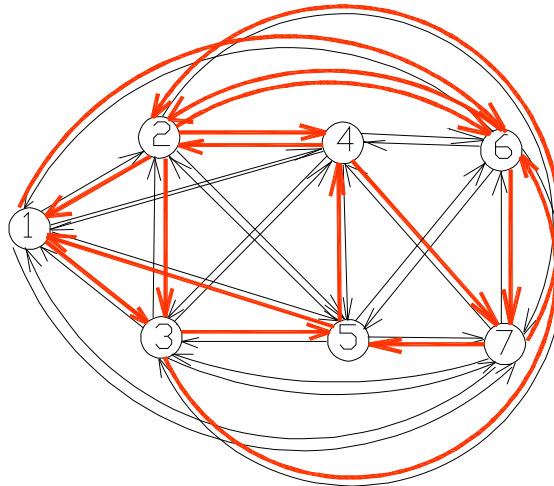


Figura 5.7 Solución OSL del SCNDP para el Caso1

Esta solución del SCNDP tiene 16 arcos, el doble que la solución óptima del SUNDP. Se tiene en este caso que los costos fijos representan un 94% del costo total de la solución, siendo el 6% restante imputable a los costos esperados de transporte.

### **Solución de valor esperado**

Considerando un solo escenario con demanda media, OSL llega a la siguiente solución:

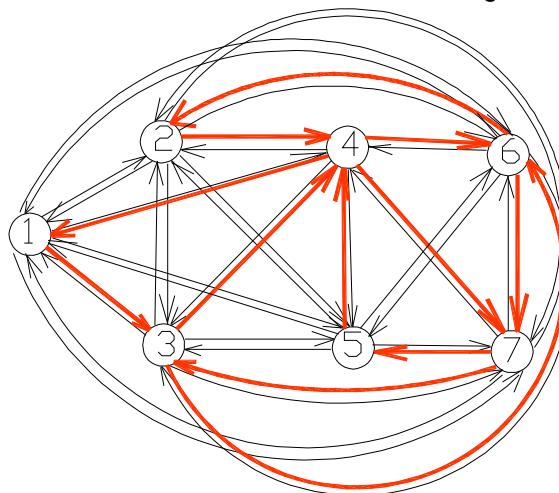


Figura 5.8 Solución de Valor Esperado para el SCNDP en el Caso1

Esta solución consta de 12 arcos activos, y se verifica que no es factible para ninguno de los escenarios con demanda superior a la media.

### 5.1.2 Caso de Diseño 2

Este segundo caso de prueba se basa en el caso denominado *cexd2*, de Holmberg [10]. Es un ejemplo mas general que el Caso 1.

#### **Topología de la red origen**

Consiste en el diseño de una red con 9 nodos y 35 arcos potenciales, sin ninguna estructura especial de la red potencial.

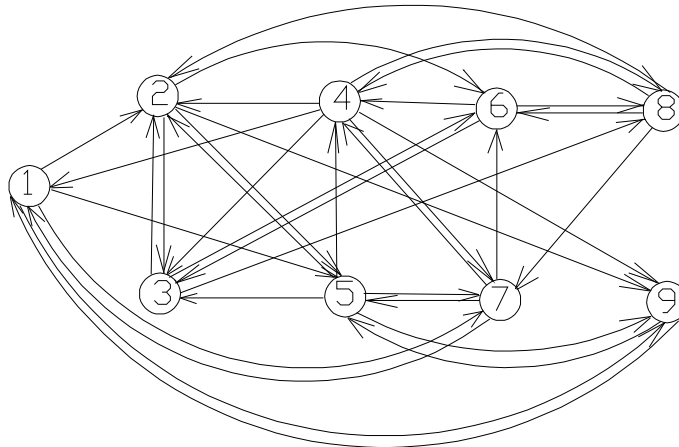


Figura 5.9 Red Potencial del Caso de Prueba 2

#### **Costos**

Los costos fijos varían de arco a arco.

Los costos variables en un mismo arco son iguales para todas las commodities.

Este caso de prueba tiene una relación media costo fijo/costo variable de valor 2, es decir que puede considerarse como de valor bajo.

#### **Capacidades**

Todos los arcos tienen la misma capacidad. Se aumentaron en un 20% las capacidades del caso original para hacer factible el problema estocástico.

#### **Modelo de demanda incierta**

Análogamente al Caso 1, cada nodo tiene asociadas 8 commodities con origen en ese nodo, cada una de ellas con destinos distintos en los restantes 8 nodos. Es decir que  $IKI = 72$ , existiendo demanda recíproca entre todo par de nodos.

Para modelar la demanda incierta se suponen tres escenarios de demanda:

- ❑ Escenario de demanda baja, con probabilidad 0.2
- ❑ Escenario de demanda media, con probabilidad 0.5
- ❑ Escenario de demanda alta, con probabilidad 0.3

Las demandas en los distintos nodos no son independientes, sino que están fuertemente correlacionadas, asumiendo para los escenarios con demanda baja y alta un valor 20% inferior y 20% superior respectivamente al del escenario de demanda media.

La codificación GAMS con los datos de base de este caso y los resultados obtenidos por el solver MIP de OSL se muestran en el *Anexo D*.

## **Análisis de resultados del Caso 2 - Problema SUNDP**

### **Solución óptima estocástica**

Para este caso se llega a una solución óptima del problema estocástico con la selección de variables de primera etapa  $x_{ij}$  que muestra la Figura 5.10.

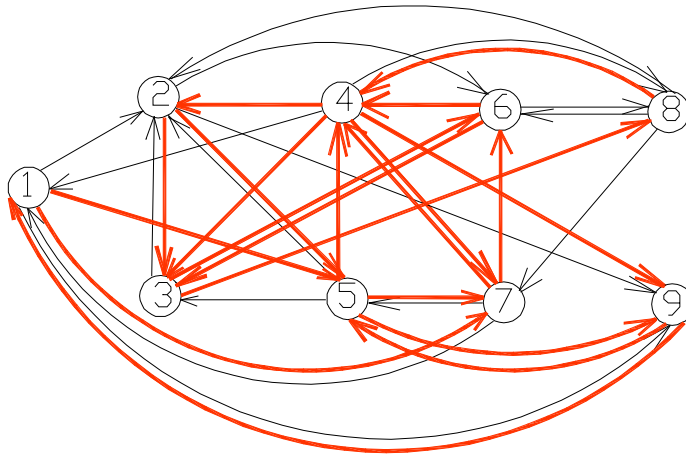


Figura 5.10 Solución óptima del SUNDP para el Caso 2

Se observa que la solución óptima tiene 20 arcos, 12 más de los estrictamente necesarios para asegurar conectividad entre todo par de nodos, pues por ejemplo el siguiente ciclo formado con un subconjunto de 8 arcos de la solución óptima es una solución factible:

$(1,7) \rightarrow (7,6) \rightarrow (6,3) \rightarrow (3,8) \rightarrow (8,4) \rightarrow (4,2) \rightarrow (2,5) \rightarrow (5,9) \rightarrow (9,1)$

El sobrecosto de los costos fijos adicionales es absorbido por los menores costos de tráfico de la red.

Se tiene que este diseño es mucho más robusto que el del Caso 1, siendo de sólo 2% el valor esperado de pérdidas de conexiones entre parejas (origen, destino) de commodities frente a una falla única de arco.

En este caso los costos fijos representan un 23% del costo de la solución óptima, siendo el 77% restante correspondiente a los costos esperados de transporte.

### **Solución de valor esperado del problema del Caso 2**

Coincide con el diseño óptimo estocástico.

## **Análisis de resultados del Caso 2 - Problema SCNDP**

### **Solución estocástica**

En el caso con restricciones de capacidad, OSL encuentra una solución óptima del problema con la selección de variables de primera etapa  $x_{ij}$  de la Figura 5.11.

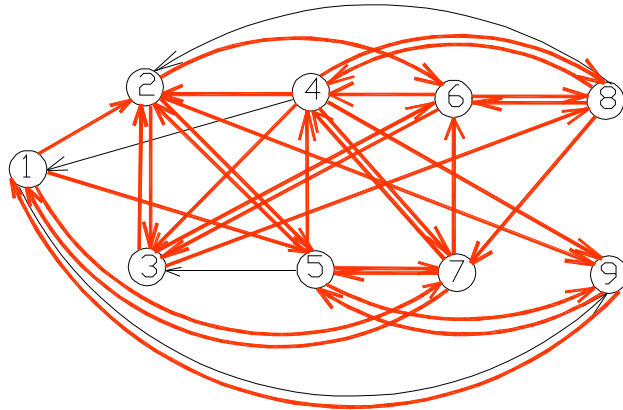


Figura 5.11 Solución óptima del SCNDP para el Caso 2

Esta solución del SCNDP tiene 31 arcos, es decir que se emplea el 89% de los arcos de la red base original y un 50% más de arcos que en el caso del SUNDP.

Resulta ser un 14% más costosa que la solución correspondiente al SUNDP, teniéndose en este caso que los costos fijos son responsables del 36% del costo total de la solución, mientras que el 64% restante es imputable a los costos esperados de transporte.

### **Solución de valor esperado**

Considerando un solo escenario con la demanda media de las commodities, OSL llega a la siguiente solución óptima:

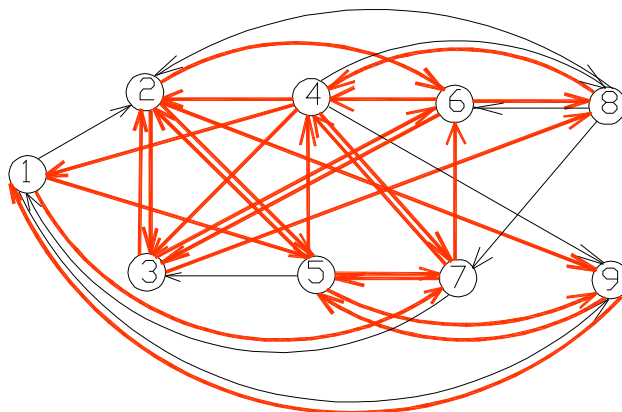


Figura 5.12 Solución de Valor Esperado del SCNDP para el Caso 2

Este diseño consta de 26 arcos activos y no es factible para el escenario de máxima demanda.

### 5.1.3 Caso de Diseño 3

Esta instancia de diseño está elaborada en base al problema de diseño denominado *cexd6*, propuesto por Holmberg [10].

#### **Topología de la red origen**

Consiste en el diseño de una red con 14 nodos y 56 arcos potenciales, sin ninguna estructura especial de la red.

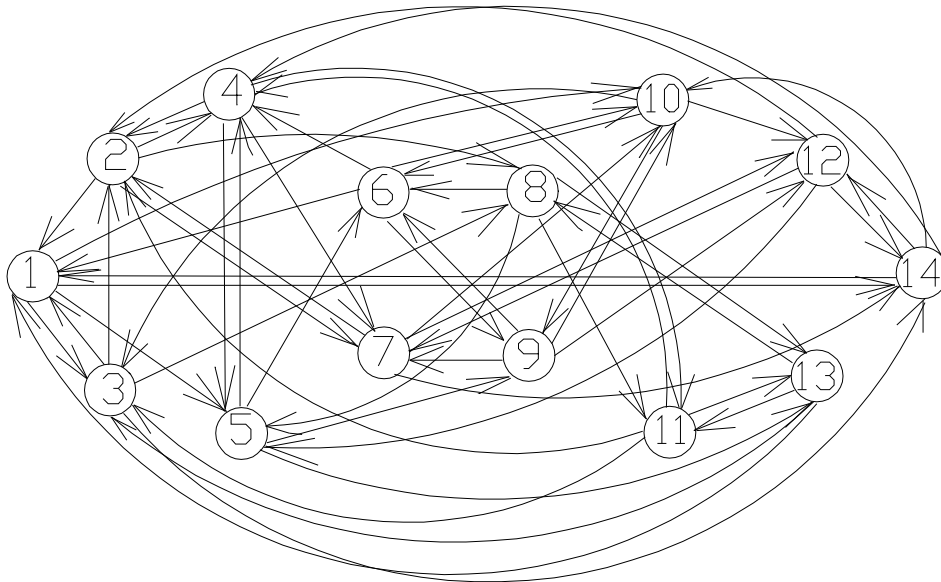


Figura 5.13 Red Potencial del Caso de Prueba 3

#### **Costos**

Los costos fijos varían de arco a arco.

Los costos variables varían de arco a arco y en un mismo arco son iguales para todas las commodities.

Este caso de prueba tiene una relación media costo fijo/costo variable de valor 2, es decir puede considerarse como de valor bajo.

#### **Capacidades**

Todos los arcos tienen la misma capacidad.

### Modelo de demanda incierta

Se considera que cada nodo de la red pertenece a uno de tres grupos de nodos: el Grupo 1, formado por los nodos 1,2,3,4 y 5, el Grupo 2, integrado por los nodos 6,7,8 y 9 y el Grupo 3, formado por los nodos 10,11,12, 13 y 14 como muestra la Figura 5.15.

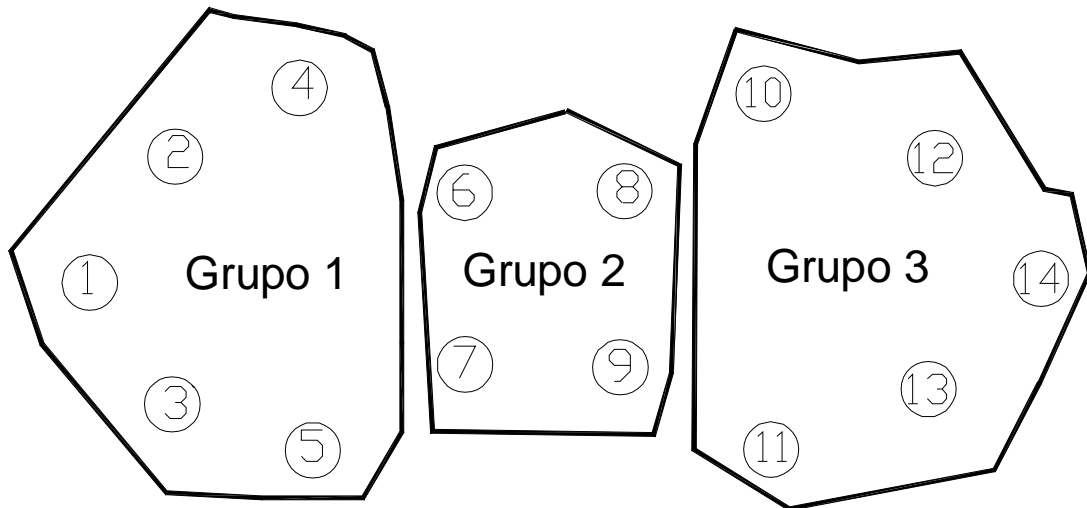


Figura 5.14 Grupos de Demanda de la Red Potencial del Caso de Prueba 3

Los nodos de los grupos 1 y 3 son los únicos que son origen y destino de commodities, existiendo para cada par de nodos del conjunto formado por dichos dos grupos una commodity con origen en cada uno de dichos nodos y destino en el otro (demanda recíproca). Es decir que se tienen en total  $|K| = 90$  commodities.

Los cuatro nodos del grupo 2 son nodos de tránsito.

Para modelar la demanda incierta se suponen tres escenarios de demanda:

- ❑ Un escenario cuya demanda coincide con la demanda del problema determinístico original, al que se le asigna probabilidad 0.5
- ❑ Un escenario en que la demanda entre nodos de un mismo grupo es un 30% mayor que la demanda determinística, y a su vez la demanda entre nodos de distintos grupos es un 30% inferior a la determinística. Este escenario tiene una probabilidad de 0.35
- ❑ Un escenario inverso al anterior, en que la demanda entre nodos de un mismo grupo es un 30% inferior que la demanda determinística, y la demanda entre nodos de distintos grupos es un 30% superior a la determinística. A este escenario se le asigna una probabilidad de 0.15

El modelado de la demanda de este caso de prueba, implica que a diferencia de los casos anteriores no exista un escenario claramente definido de demanda “máxima” por la red.

La codificación GAMS con los datos de base del caso y los resultados obtenidos por el solver MIP de OSL se muestran en el Anexo D.

## **Análisis de resultados del Caso 3 - Problema SUNDP**

### **Solución óptima estocástica**

Para este caso en que no juegan las restricciones de capacidad el solver MIP de OSL llega a una solución óptima del problema estocástico con la selección de variables de primera etapa  $x_{ij}$  que muestra la Figura 5.15.

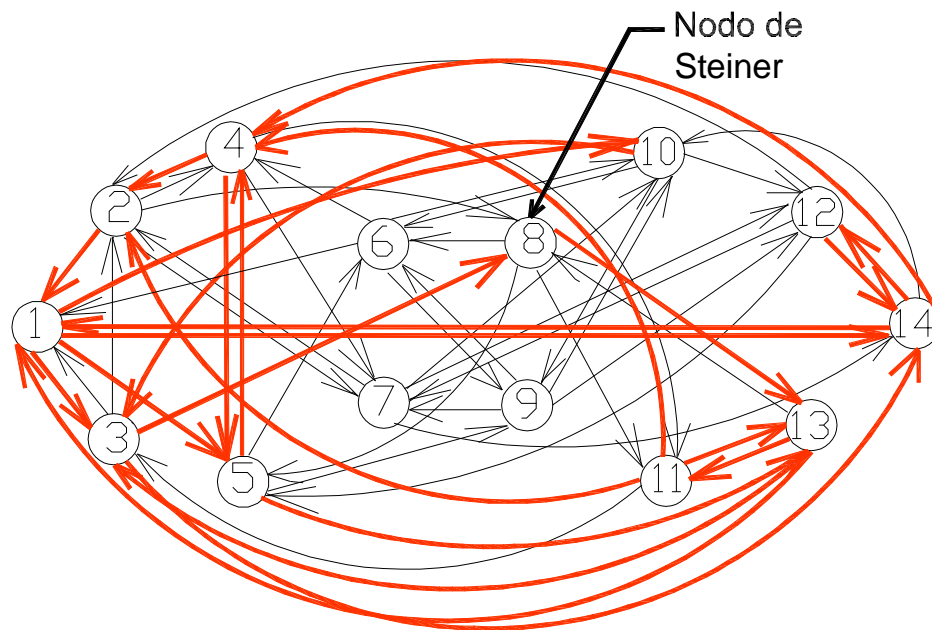


Figura 5.15 Solución óptima del SUNDP para el Caso 3

Se observa que la solución óptima tiene 23 arcos, es decir que emplea el 41% de los 56 arcos potenciales disponibles.

Como característica interesante de la solución se tiene que en ella aparece un Nodo de Steiner, el nodo 8 perteneciente al Grupo 2 y que no es un nodo terminal, origen o destino de commodities.

El análisis de robustez de la solución muestra que el valor esperado de pérdidas de conexiones entre parejas (origen, destino) de commodities frente a una falla única de arco es de 3% del total de conexiones existentes.

En este caso los costos fijos representan un 21% del costo de la solución óptima, siendo el 79% restante correspondiente a los costos esperados de transporte.

El tiempo de ejecución de este caso de prueba es del orden de decenas de segundos.

### **Solución del problema de valor esperado**

Coincide con el diseño óptimo estocástico como lo prevé el Teorema 2 de 4.1.7.



## Análisis de resultados del Caso 3 - Problema SCNDP

### Solución estocástica

En el caso con restricciones de capacidad, OSL arriba a una solución óptima del problema estocástico con la selección de variables de primera etapa  $x_{ij}$  de la Figura 5.16

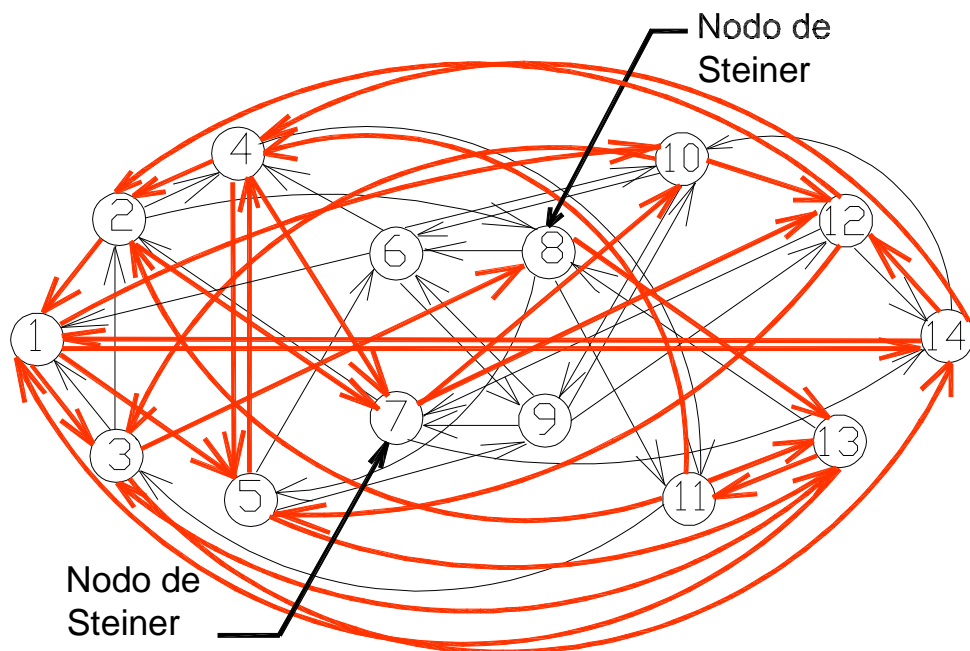


Figura 5.16 Solución óptima del SCNDP para el Caso 3

Esta solución del SCNDP tiene 29 arcos, es decir que se emplea el 52% de los arcos de la red base original. Tiene 6 arcos mas que la solución correspondiente del SUNDP y es un 6% mas costosa.

Se tiene en este caso que los costos fijos representan un 27% del costo total de la solución, siendo el 73% restante imputable a los costos esperados de transporte.

La solución tiene la peculiaridad de emplear dos Nodos de Steiner, los nodos 7 y 8.

El tiempo de ejecución de este caso con capacidad finita de arcos es del orden de 90 minutos.

### Solución de valor esperado

En este caso, la solución del problema de valor esperado *EV* coincide con la solución del problema *RP*, por lo que **el valor de la solución estocástica VSS es cero.**

## 5.2 Implementación de la Heurística de Ascenso Dual

Teniendo en cuenta que en el caso de diseño de redes sin restricciones de capacidad SUNDP el valor de la solución estocástica es nulo, siendo dicho problema equivalente a resolver el caso determinístico empleando las demandas medias de las commodities, se desarrolla una implementación en C++ de la técnica heurística de Ascenso Dual descrita en 2.1.5, como una alternativa al empleo de un solver MIP estándar como el de OSL para resolver el SUNDP.

Como características relevantes de la implementación se destacan:

- ❑ Para encontrar los caminos de mínimo costo entre pares de nodos se implementa el algoritmo de Dijkstra, siguiendo el esquema de Weiss [18].
- ❑ Se considera que los costos de transporte unitarios de cada arco  $c_{ij}^k$  son iguales para todas las commodities. Esta hipótesis se verifica en un gran número de casos reales en redes de telecomunicación.
- ❑ No se realiza una optimización local de la solución primal factible hallada por la heurística.
- ❑ Se tienen dos parámetros de entrada al programa, uno con el nombre del archivo de texto de datos de la red potencial y commodities, y el otro con el nombre del archivo de texto de salida donde se escriben los resultados.

El archivo de entrada de datos al programa contiene, por su orden y a razón de uno por línea:

- ❑ El número de nodos de la red, de arcos y de commodities
- ❑ Para cada arco, su origen, destino, costo unitario de transporte y costo fijo
- ❑ Para cada commodity, número de commodity, origen, destino y demanda media

En el archivo de salida se presenta:

- ❑ El costo de la solución del problema primal de diseño a la que llega la heurística
- ❑ Los costos totales de transporte y fijos de dicha solución
- ❑ El costo de la solución del problema dual correspondiente, cota inferior de la solución óptima.
- ❑ El listado de arcos que forman parte de la solución encontrada

En el *Anexo E.2* se muestra el código fuente y en el *Anexo E.3* los archivos de entrada y salida correspondientes a los casos de estudio presentados.

### 5.2.1 Aplicación de la Heurística de Ascenso Dual a los Casos de Diseño

Se compara el diseño proporcionado por la heurística contra el diseño óptimo de la red calculado por el solver MIP de OSL.

En los grafos de red se marcan con color rojo los arcos que son comunes a la solución óptima y a la heurística, con color verde los arcos que aparecen sólo en la solución óptima y con color azul los arcos que aparecen sólo en la solución de la heurística.

#### Caso 1

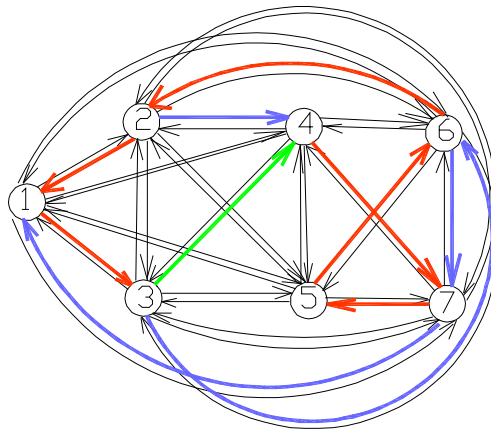


Figura 5.17 Solución de Ascenso Dual para el Caso 1

En este caso, la solución que proporciona la heurística es bastante pobre, teniendo 3 arcos más que la óptima y un sobre costo de 32% respecto a ésta.

La cota inferior que da la heurística como solución del dual es sin embargo bastante buena, siendo solamente un 4% inferior al valor óptimo.

#### Caso 1b

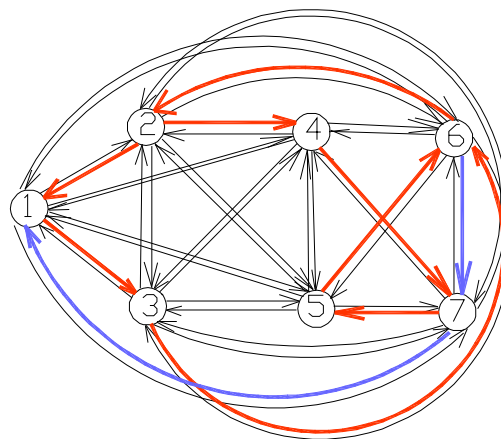


Figura 5.18 Solución de Ascenso Dual para el Caso 1.b

En este caso en que los costos fijos de la red se reducen a la décima parte, la heurística da una solución que contiene a todos los arcos de la solución óptima más dos arcos adicionales.

El costo de la solución heurística es un 5% mayor que el valor óptimo.

La cota inferior que proporciona la heurística está a un 8% del valor óptimo.

### Caso 2

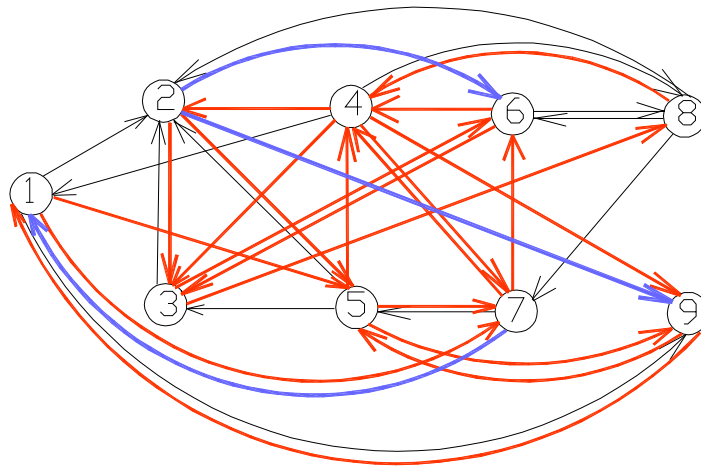


Figura 5.19 Solución de Ascenso Dual para el Caso 2

En esta instancia de diseño, la solución heurística contiene todos los arcos de la solución óptima mas un 10% adicional de arcos.

El costo de la solución heurística es sólo 2% mayor que el valor óptimo.

La cota inferior que proporciona la heurística está un 2% debajo del valor óptimo.

### Caso 3

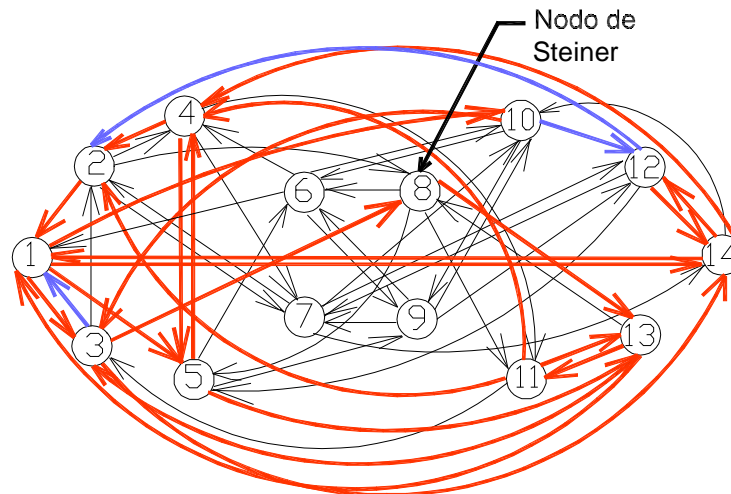


Figura 5.20 Solución de Ascenso Dual para el Caso 3

En este caso la solución de la heurística tiene también un 10% más arcos que la solución óptima. Se mantiene el mismo nodo de Steiner de la solución óptima.

El tiempo de ejecución de la heurística en este caso es inferior a 1 segundo.

El costo adicional de la solución heurística es apenas de 1.5% del valor óptimo, y la cota inferior que proporciona la heurística está un 0.7% debajo del valor óptimo, por lo que puede considerarse que el diseño heurístico es excelente para esta instancia.

## 6 Conclusiones

El problema de diseño óptimo de red con flujo multicommodity y costos fijos y de transporte asociados a los arcos, tanto en la versión sin restricciones de capacidad como con ellas, es un problema NP-Completo ya en el caso determinístico, al cual la incorporación de aleatoriedad en la demanda inicial de las commodities le agrega mas complejidad aún.

El caso determinístico ha sido abordado y estudiado a fondo por diversos investigadores, que en general han desarrollado técnicas heurísticas de solución que buscan obtener soluciones muy próximas a las óptimas con rápidos tiempos de cómputo, particularmente apropiadas para casos de redes con muchos nodos, enlaces y pares origen-destino, como por ejemplo redes de telefonía.

El caso de diseño con incertidumbre en la demanda inicial de las commodities puede plantearse en forma natural como un problema de programación estocástica lineal en dos etapas con recurso fijo, con variables de primera etapa binarias que indican la pertenencia ó no de los arcos de la red al diseño final y con variables de segunda etapa continuas representando los flujos de las commodities en cada arco y para cada escenario de demanda.

Como opciones para resolver el problema estocástico de diseño se tienen:

- Formular el problema en forma extensiva, reduciéndolo al caso determinístico, y utilizar luego alguna de las técnicas aplicables en dicho caso.
- Emplear alguna técnica específica de programación estocástica, como el Método L Entero ó el Método de Cortes Múltiples en su versión entera, basados en la formulación en dos etapas del problema. Sobre esta alternativa de resolución aplicada al diseño de redes hay escasas referencias y publicaciones.

Tanto en el caso determinístico como en el estocástico, la existencia de soluciones factibles y por consiguiente la de una solución óptima no está asegurada a priori para cualquier combinación de red potencial y conjunto de commodities, debiéndose resolver un problema de factibilidad previo para establecerla.

El resultado teórico mas interesante que se presenta en este trabajo es el hecho de que, en el caso del problema de diseño estocástico sin restricciones de capacidad (SUNDP), el valor de la solución estocástica es nulo y dicho problema es completamente equivalente a resolver el caso determinístico considerando las demandas medias de las commodities sobre todos los escenarios de realización definidos.

En el caso de diseño con restricciones (SCNDP), no existe un resultado general análogo para el valor de la solución estocástica y hay que hacer el estudio caso a caso.

Considerando las instancias de diseño formuladas en forma extensiva y resueltas con el solver MIP de OSL, se constata una gran diferencia en tiempo de cómputo entre los casos sin y con restricciones de capacidad en arcos, verificándose que el caso con restricciones es mucho mas "duro " de resolver desde el punto de vista computacional (en uno de los ejemplos estudiados el solver no llega a la solución óptima).

Se observa que el diseño solución de los casos de prueba tiende a tener relativamente pocos arcos en los casos en que se tiene una relación costo fijo/costo de transporte alta por arco, incrementándose notablemente el porcentaje de arcos de la red potencial original presentes en la solución cuando dicha relación es baja. Esto implica que con una relación de costos alta el diseño es más vulnerable frente a la contingencia de pérdida de un único enlace entre un par cualquiera de nodos, mientras que para una relación de costos relativamente baja los diseños obtenidos resultan ser muy robustos, empleando como medida de robustez la cantidad de conexiones origen-destino de commodities que se pierden al no tener camino de flujo alternativo.

Se identificaron los siguientes parámetros que influyen fuertemente en el diseño final de la red:

- ❑ Para los dos casos SUNDP y SCNDP, la relación entre los costos fijos de instalación y los costos variables de transporte de los arcos.
- ❑ Para el caso SCNDP, la relación entre las capacidades finitas de los arcos y las demandas.

De la implementación de la Heurística de Ascenso Dual para resolver el problema SUNDP es posible concluir que:

- ❑ Los resultados son muy satisfactorios, lográndose aproximaciones muy buenas a la optimalidad.
- ❑ Es una heurística eficiente, con tiempos de ejecución sensiblemente inferiores que el solver MIP de OSL.

## 7 Referencias

- [1] Balakrishnan A., Magnanti T.L., Wong R.T. A Dual Ascent procedure for large scale uncapacitated network design. *Operations Research* 37, 1989, 716-740.
- [2] Balakrishnan A. Network design models for Distribution Systems. Mc Combs School of Business, 2003. [http:// www.aladdin.cs.cmu.edu/workshops/integrated\\_wshp/slides2/anb.pdf](http://www.aladdin.cs.cmu.edu/workshops/integrated_wshp/slides2/anb.pdf)
- [3] Birge J.R., Louveaux F. Introduction to Stochastic Programming. Springer Series on Operations Research, 1997
- [4] GAMS, A User's Guide. Gams Developement Corporation, 1998.
- [5] Garey M., Johnson D. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H.Freeman & Co.,1979.
- [6] Gendron B. A note on a Dual-Ascent Approach to the Fixed-Charge Capacitated Network Design Problem.*European Journal of Operational Research* 138, 671- 675, 2002.
- [7] Ghamlouche I.,Cairnic T.G., Gendreau M. Cycle-Based Neighbourhoods for Fixed-Charge Capacitated Multicommodity Network Design. Publication CRT-2001-01. Centre de recherche sur les transports, Université de Montréal, Montréal,QC,Canada.
- [8] Holmberg K., Hellstrand J. Solving the uncapacitated network design problem by a Lagrangean heuristic and Branch and Bound. *Operations Research* 46, 1998, 247-259.
- [9] Holmberg K., Yuan D. A Lagrangean heuristic based Branch-and-Bound approach for the capacitated network design problem. *Operations Research*, 48, 2000, 461-481.
- [10] Holmberg K. Capacitated Network Design Data <http://www.mai.liu.se/~kahol/problemdata/cnd/>
- [11] IBM Optimization Solutions and Library- MIP Solutions User's Guide, 1997 .
- [12] Karp R.M. Reducibility among Combinatorial Problems. *Complexity of Computer Computations*. R.E.Miller and J.W.Thatcher, editors. Plenum Press, NY, 1972, 85-104.
- [13] Laporte G., Louveaux F. Solving Stochastic Routing Problems with the Integer L-Shaped Method. *Fleet Management and Logistics*.Kluwer Academics Publishers,1998.
- [14] MagnantiT.L., Mireault P., Wong R.T. Tailoring Benders decomposition for uncapacitated network design. *Mathematical Programming Study* 26, 1986, 112-152.
- [15] Nemhauser G., Wolsey L. *Integer and Combinatorial Optimization*. Wiley Interscience Series on Discrete Mathematics and Optimization. 1988.
- [16] Riis M. Classical and Non-Classical Stochastic Recourse Programs with Applications in Telecommunications. PhD Thesis. University of Aarhus, 2003, 91-107.
- [17] Testuri C. Notas del Curso de Optimización Bajo Incertidumbre. INCO, Facultad de Ingeniería, Universidad de la República, Mayo 2003.
- [18] Weiss M.A. *Data Structures and Algorithm Analysis*. The Benjamin/Cummings Publishing Company, 1992.

## Anexo A Ejemplo de Aplicación del Método L Entero al SUNDP

Se considera el problema de diseño sobre la red potencial de la Figura A.1, en la que sobre cada arco se indica su costo fijo y su costo de transporte unitario, en ese orden:

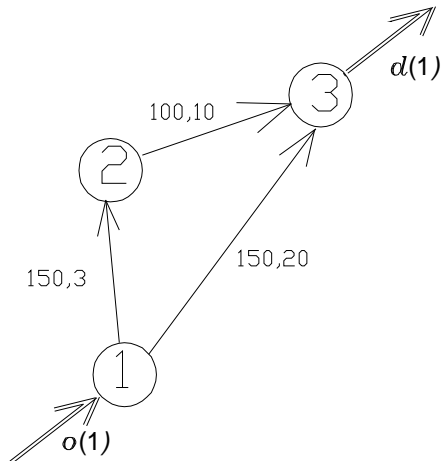


Figura A.1 Red Potencial para ejemplo del Método L Entero

Por ejemplo, el arco  $1 \rightarrow 2$  tiene costo fijo 150 y costo de transporte unitario de valor 3.

Hay una única commodity, con origen en el nodo 1 y destino en el nodo 3.

Se consideran 3 escenarios de realización futura de demanda, con valores 5, 10 y 15, y probabilidades respectivas de 0.25, 0.5 y 0.25.

### Inicialización

$$v := r := t := 0, \hat{z} := +\infty.$$

$$x = \{x_{ij} / (i, j) \in A\} = \{x_{12}, x_{13}, x_{23}\}$$

$$F = \{f_{ij} / (i, j) \in A\} = \{150, 150, 100\}$$

$$C = \{c_{ij}^k / (i, j) \in A, k \in K\} = \{3, 20, 10\}$$

Se crea lista L con un único problema maestro:

$$\min_{x, \theta} 150x_{12} + 150x_{13} + 100x_{23} + \theta$$

$$\text{s.a. } 0 \leq x \leq 1, \theta \in \mathbf{R}$$

### Problema Maestro

$$v := 1$$

Se toma  $\theta^1 = -\infty$  pues no hay restricciones de optimalidad

El problema maestro tiene la solución  $x^1 = (0, 0, 0)$



### Problema de Factibilidad

Se considera el escenario  $s = 1$  con demanda 5.

El problema no es factible para este escenario, siendo los valores solución del dual:

$$\sigma_1^1 = 0, \sigma_2^1 = -1, \sigma_3^1 = -1, \sigma_{12}^1 = -1, \sigma_{13}^1 = -1, \sigma_{23}^1 = 0$$

Se tiene  $D^1 = (5, 5, 0)$  y  $d^1 = 5$

Se agrega al problema maestro la restricción de factibilidad  $x_{12} + x_{13} \geq 1$

$$r := 1$$

### Problema Maestro

$$v := 2$$

$$\theta^2 = -\infty$$

Se resuelve el problema

$$\min_{x,\theta} \quad 150x_{12} + 150x_{13} + 100x_{23}$$

$$\text{s.a.} \quad x_{12} + x_{13} \geq 1$$

$$0 \leq x \leq 1,$$

que da como solución óptima  $x^2 = (1,0,0)$

### Problema de Factibilidad

Se considera el escenario  $s = 1$  con demanda 5.

El problema no es factible para este escenario, siendo los valores solución del dual:

$$\sigma_1^1 = 0, \sigma_2^1 = 0, \sigma_3^1 = -1, \sigma_{12}^1 = 0, \sigma_{13}^1 = -1, \sigma_{23}^1 = -1$$

Se tiene  $D^2 = (0, 5, 5)$  y  $d^2 = 5$

Se agrega al problema maestro la restricción de factibilidad  $x_{13} + x_{23} \geq 1$

$$r := 2$$

### Problema Maestro

$$v := 3$$

$$\theta^3 = -\infty$$

Se resuelve el problema

$$\min_{x,\theta} \quad 150x_{12} + 150x_{13} + 100x_{23}$$

$$\text{s.a.} \quad x_{12} + x_{13} \geq 1$$

$$x_{13} + x_{23} \geq 1$$

$$0 \leq x \leq 1,$$

que da como solución óptima  $x^3 = (0,1,0)$

### Problema de Factibilidad

El problema es factible para los tres escenarios de demanda.

Restricciones de integralidad

Se cumplen.

Problema de Optimalidad

Se resuelve el problema de optimalidad dando los siguientes valores de los multiplicadores simplex, iguales para los tres escenarios:

$$\pi_1^1 = 10, \pi_2^1 = 0, \pi_3^1 = -10, \pi_{12}^1 = -7, \pi_{13}^1 = 0, \pi_{23}^1 = 0$$

Se tiene  $E^1 = (70, 0, 0)$  y  $e^1 = 200$

La función de recurso vale  $Q(x^3) = e^1 - E^1 x^3 = e^1 = 200$

La cota superior se actualiza a  $\bar{z} = 350$

Se agrega al problema maestro la restricción de optimalidad  $70x_{12} + \theta \geq 200$

$t := 1$

Problema Maestro

$v := 4$

Se resuelve el problema

$$\begin{aligned} \min_{x, \theta} \quad & 150x_{12} + 150x_{13} + 100x_{23} + \theta \\ \text{s.a.} \quad & x_{12} + x_{13} \geq 1 \\ & x_{13} + x_{23} \geq 1 \\ & 70x_{12} + \theta \geq 200 \\ & 0 \leq x \leq 1, \theta \in \mathbf{R} \end{aligned}$$

que da como solución óptima  $x^4 = (0, 1, 0)$  y  $\theta^4 = 200$

Se verifica que  $\sum_{(i,j) \in A} f_{ij} x_{ij}^4 + \theta^4 \geq \bar{z}$ , por lo que se vacía el nodo y como  $L = \emptyset$ ,  $\rightarrow$ FIN.

La solución óptima es  $x = (0, 1, 0)$ , con valor óptimo  $\bar{z} = 350$

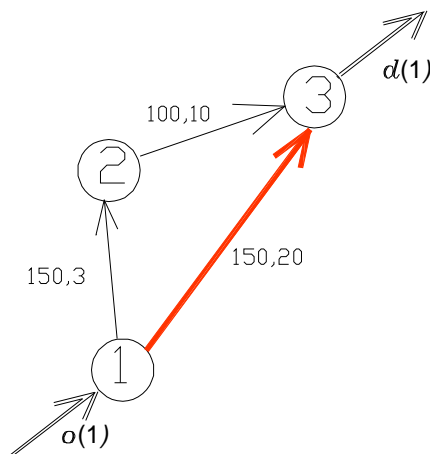


Figura A.2 Solución ejemplo Método L Entero

## Anexo B Valor de la Solución Estocástica para el SCNDP

En el caso de diseño con restricciones de capacidad en la red, no hay resultados generales respecto al valor de la solución estocástica VSS, pudiendo darse tres casos distintos:

Caso 1) La solución del problema de valor esperado EV no es factible para el problema estocástico.

Se considera el problema de diseño óptimo sobre la red de la Figura B.1, en la que sobre cada arco se indica capacidad, costo fijo y costo variable unitario del arco, en ese orden:

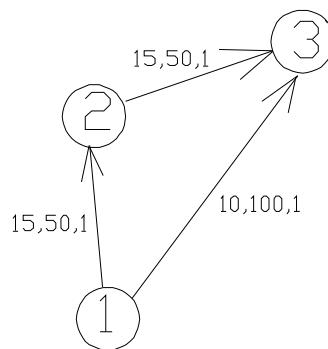


Figura B.1 Red Potencial para Caso 1 de VSS

Por ejemplo, el arco  $1 \rightarrow 2$  tiene capacidad 15, costo fijo 50 y costo variable unitario 1.

Se tiene una sola commodity con origen en nodo 1 y destino en nodo 3, y hay tres escenarios de demanda equiprobables, bajo de valor 5, medio de valor 10 y alto de valor 15.

En las Figuras B.2 y B.3 se muestran la solución óptima estocástica del problema con recurso RP y la solución óptima del problema de valor esperado EV, planteado este último para el escenario de demanda media de valor 10.

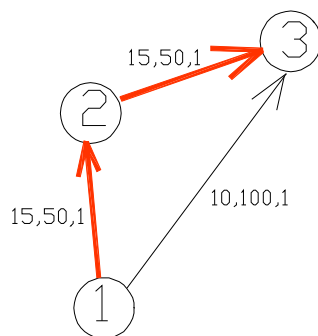


Figura B.2 Solución óptima RP Caso 1

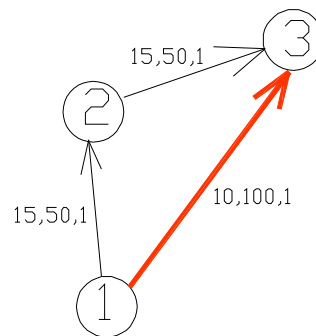


Figura B.3 Solución óptima EV Caso 1

Se observa que ambas soluciones difieren, debido a que la solución de valor esperado no es factible para el escenario de demanda alta, con valor 15.

Caso 2) La solución del problema *EV* es factible pero no óptima para el problema *RP*, por lo que se tiene  $VSS > 0$  finito.

Se considera el problema de diseño óptimo sobre la red de la Figura B.4.

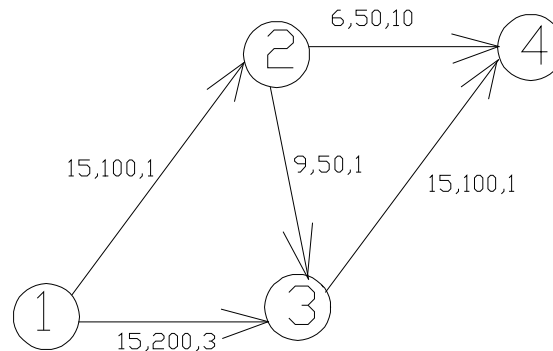


Figura B.4 Red Potencial para Caso 2 de VSS

Se tiene una sola commodity con origen en el nodo 1 y destino en el nodo 4, y hay tres escenarios de demanda, bajo de valor 5, medio de valor 10 y alto de valor 15, con probabilidades 0.25, 0.5 y 0.25 respectivamente, siendo 10 entonces la demanda media.

Las Figuras B.5 y B.6 ilustran las soluciones de los problemas *RP* y *EV* respectivamente.

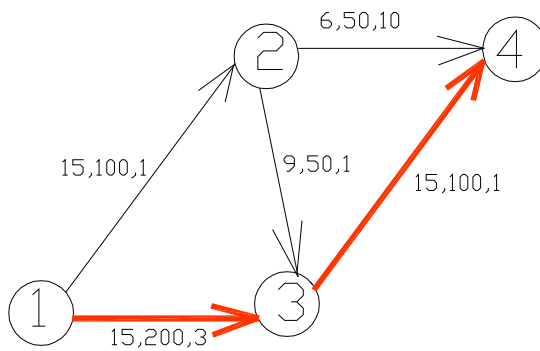


Figura B.5 Solución óptima *RP* Caso 2

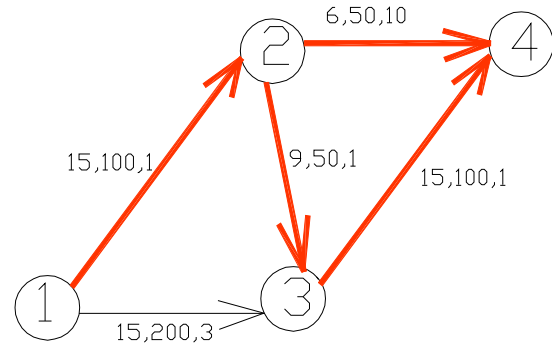


Figura B.6 Solución óptima *EV* Caso 2

Se observa que la solución del problema de valor esperado *EV* es factible para el problema estocástico con recurso *RP*

Se hará el análisis del problema *EEV* sobre la red solución del problema *EV* y se calculará  $VSS$

Para cada escenario se ilustra la distribución de flujo de la demanda en los arcos de la red (variables  $y_{ij}^k(s)$ ).

Escenario 1- Demanda 5

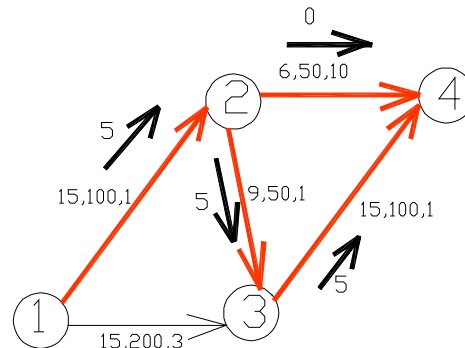


Figura B.7 Flujos en escenario de demanda 5

El costo variable de transporte es 15.

Escenario 2-Demanda 10

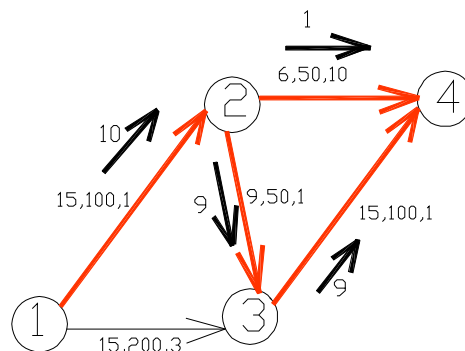


Figura B.8 Flujos en escenario de demanda 10

El costo variable de transporte vale 38.

Escenario 3 – Demanda 15

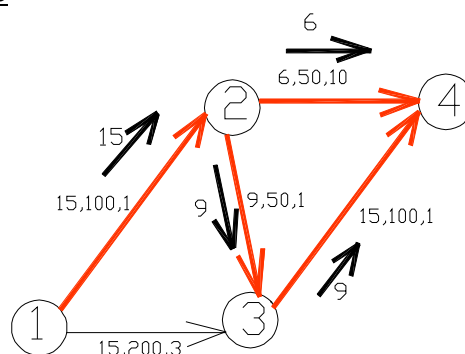


Figura B.9 Flujos en escenario de demanda 15

El costo variable de transporte es en este caso vale 93.

Ponderando con las probabilidades de cada escenario, se tiene un costo variable esperado de valor  $0.25 \cdot 15 + 0.5 \cdot 38 + 0.25 \cdot 93 = 46$

Como el costo fijo de la red solución de *EV* es de 300, se tiene que  $EEV = 300 + 46 = 346$

En el caso del problema *RP*, se tiene que el costo de transporte esperado es

$$0.25 \cdot 20 + 0.5 \cdot 40 + 0.25 \cdot 60 = 40$$

Como el costo fijo de la solución *RP* es 300, se tiene  $RP = 300 + 40 = 340$

Se cumple entonces que  $VSS = EEV - RP = 346 - 340 = 6 = 2\% \text{ del valor de } EEV$

Caso 3) La solución óptima del problema de valor esperado *EV* coincide con la solución óptima del problema estocástico *RP*, y se tiene  $VSS = 0$ .

Se considera el mismo problema del Caso 1), pero ahora cambiando la capacidad del arco  $1 \rightarrow 3$  de 10 a 15.

Se tiene que tanto el problema *RP* como *EV* tienen la misma solución óptima de la Figura B.10.

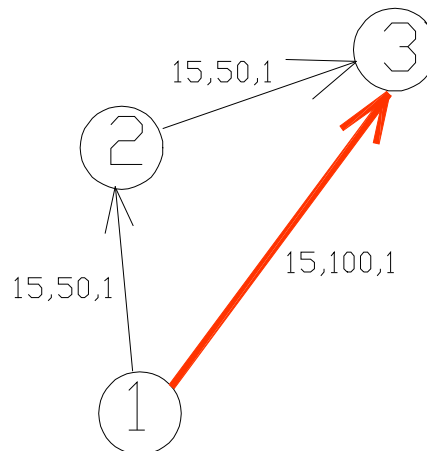


Figura B.10 Solución óptima *RP* y *EV* del Caso 3

Se verifica por lo tanto que  $RP = EEV$ , y de ahí  $VSS = 0$

## Anexo C Formulación GAMS del Modelo Estocástico Extensivo

### C.1 Estructuras de Datos en GAMS

En la Figura C.1 se muestran las estructuras de datos que intervienen en la formulación extensiva del problema estocástico de diseño de red con y sin restricciones de capacidad y con escenarios discretos de demanda inicial.

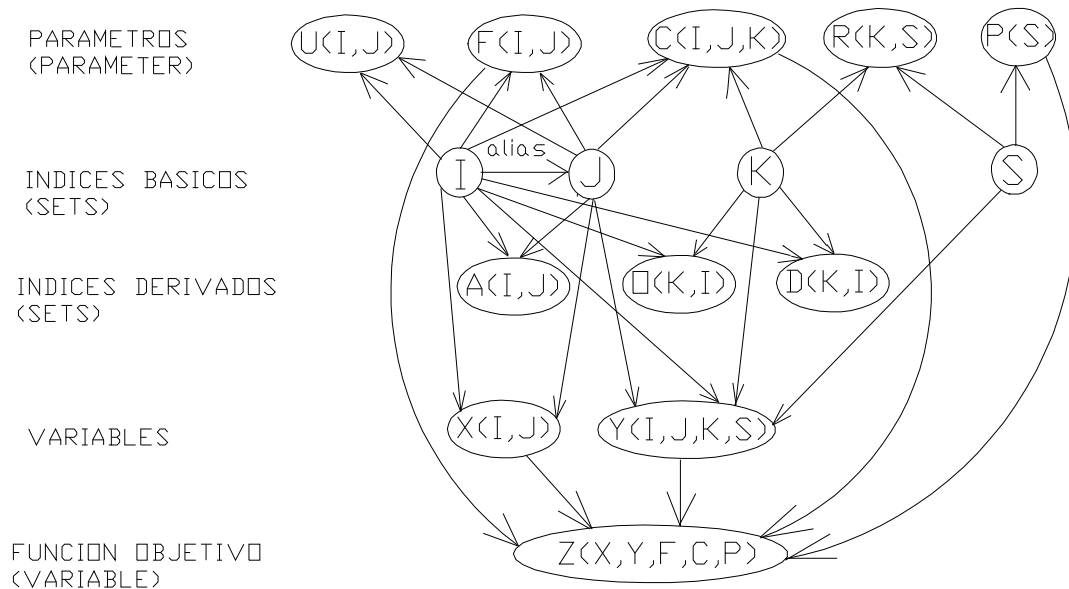


Figura C.1 Estructuras de datos del Modelo GAMS de SUNDP y SCNDP

I,J: índices que representan los nodos de la red

K: índice de las commodities

S: índice de los escenarios discretos de demanda

A(I,J): arco (i,j) potencial de la red

O(K,I): nodo origen de la commodity k

D(K,I): nodo destino de la commodity k

F(I,J): costo de instalación del arco (i,j)

U(I,J): capacidad total del arco (i,j)

C(I,J,K): costo variable unitario de la commodity k en el arco (i,j)

R(K,S): demanda de la commodity k en el escenario s

P(S): probabilidad del escenario s

X(I,J) variable binaria de primera etapa, indica la inclusión o no del arco (i,j) en el diseño de la red

Y(I,J,K,S): variable de segunda etapa, representa el flujo de la commodity k en el arco (i,j) en el escenario s

## C.2 Ejemplo de Formulación del Modelo en GAMS

A los efectos de ilustrar la codificación en GAMS del modelo extensivo, tanto en el caso del SUNDP como en el del SCNDP, se presentará un ejemplo sencillo de diseño de una red con las siguientes características:

4 nodos, 1..4

Arco potencial entre todo par de nodos (grafo dirigido completo), con 12 arcos potenciales

12 commodities, siendo cada nodo el origen de una commodity con destino a cada uno de los restantes nodos

Demandas iguales para todas las commodities, en tres escenarios equiprobables, uno central con demanda unitaria, uno de baja demanda con valor 0.7 y uno de alta demanda con valor 1.3

Costos fijos iguales para todos los arcos, de valor 200.

Capacidades iguales para todos los arcos y de valor 4.

Costos variables según Figura C.2, iguales en un mismo arco para todas las commodities

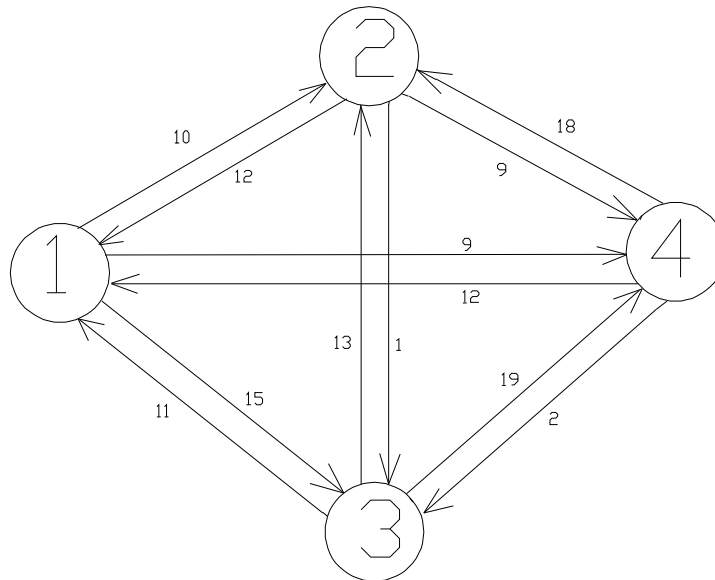


Figura C.2 Red potencial para ejemplo de formulación GAMS

Este problema tiene 12 variables de decisión de primera etapa  $x_{ij}$  y 432 variables de decisión de segunda etapa  $y_{ij}^{ks}$ .

Para el SUNDP hay 144 restricciones de balance y 432 restricciones de activación de arcos.

Para el SCNDP hay 144 restricciones de balance, 432 restricciones de activación de arcos y 36 restricciones de capacidad de arcos.



La codificación en GAMS del problema ejemplo en el caso SUNDP queda:

\$TITLE EJEMPLO DE DISEÑO DE RED CON INCERTIDUMBRE EN LA DEMANDA  
\$TITLE CASO SIN RESTRICCIONES DE CAPACIDAD EN ARCOS

SETS

I nodos de la red /1\*4/  
K mercancías /1\*12/  
S escenarios de demanda inicial /1\*3/  
O(K,I) origen de K /1.1,2.1,3.1,4.2,5.2,6.2,7.3,8.3,9.3,10.4,11.4,12.4/  
D(K,I) destino de K /1.2,2.3,3.4,4.1,5.3,6.4,7.1,8.2,9.4,10.1,11.2,12.3/;

ALIAS (I,J);

SET

A(I,J) arcos potenciales /1.2,1.3,1.4,2.1,2.3,2.4,3.1,3.2,3.4,4.1,4.2,4.3/;

PARAMETER

P(S) probabilidad del escenario S  
/1 0.333, 2 0.333, 3 0.333/

F(I,J) costo fijo de establecer el arco I->J  
/ 1.2\*4 200, 2.(1,3\*4) 200, 3.(1\*2,4) 200, 4.1\*3 200/

C(I,J,K) costo variable de la mercancía K en el arco I->J  
/1.2.1\*12 10, 1.3.1\*12 15, 1.4.1\*12 9, 2.1.1\*12 12, 2.3.1\*12 1, 2.4.1\*12 9, 3.1.1\*12 11,  
3.2.1\*12 13, 3.4.1\*12 19, 4.1.1\*12 12, 4.2.1\*12 18, 4.3.1\*12 2/

R(K,S) demanda de la mercancía K en el escenario S  
/1\*12.1 0.7, 1\*12.2 1, 1\*12.3 1.3/;

VARIABLES

X(I,J) activación del arco I->J  
Y(I,J,K,S) flujo en el arco I->J para la mercancía K en escenario S  
Z costo esperado de la red;

BINARY VARIABLE X;  
POSITIVE VARIABLE Y;

EQUATIONS

COSTO función objetivo a minimizar  
BALANCE(I,K,S) balance de flujo en el nodo I para la mercancía K y el escenario S  
ACTIVACION(I,J,K,S) activación del arco I->J para la mercancía K y el escenario S;

COSTO.. Z =E=SUM((I,J)\$A(I,J)),F(I,J)\*X(I,J))+SUM((I,J,K,S)\$A(I,J)),P(S)\*C(I,J,K)\*Y(I,J,K,S));

BALANCE(I,K,S).. SUM(J\$A(I,J)),Y(I,J,K,S))-SUM(J\$A(J,I)),Y(J,I,K,S)) =E= R(K,S)\$O(K,I))-  
R(K,S)\$D(K,I));

ACTIVACION(I,J,K,S)\$A(I,J).. Y(I,J,K,S) =L= R(K,S)\*X(I,J);

MODEL EJEMPLO /ALL/;  
SOLVE EJEMPLO USING MIP MINIMIZING Z;

La codificación en GAMS del problema ejemplo en el caso SCNDP queda:

```
$TITLE EJEMPLO DE DISEÑO DE RED CON INCERTIDUMBRE EN LA DEMANDA  
$TITLE CASO CON RESTRICCIONES DE CAPACIDAD EN ARCOS
```

#### SETS

```
  I nodos de la red /1*4/  
  K mercancías /1*12/  
  S escenarios de demanda inicial /1*3/  
  O(K,I) origen de K /1.1,2.1,3.1,4.2,5.2,6.2,7.3,8.3,9.3,10.4,11.4,12.4/  
  D(K,I) destino de K /1.2,2.3,3.4,4.1,5.3,6.4,7.1,8.2,9.4,10.1,11.2,12.3/;
```

```
ALIAS (I,J);
```

#### SET

```
  A(I,J) arcos potenciales /1.2,1.3,1.4,2.1,2.3,2.4,3.1,3.2,3.4,4.1,4.2,4.3/;
```

#### PARAMETERS

```
  P(S) probabilidad del escenario discreto S  
  /1 0.333, 2 0.333, 3 0.333/
```

```
  F(I,J) costo fijo de establecer el arco I-->J  
  /1.2*4 200, 2.(1,3*4) 200, 3.(1*2,4) 200, 4.1*3 200/
```

```
  U(I,J) capacidad de transporte del arco I-->J  
  /1.2*4 4, 2.(1,3*4) 4, 3.(1*2,4) 4, 4.1*3 4/
```

```
  C(I,J,K) costo variable de la mercancía K en el arco I-->J  
  /1.2.1*12 10, 1.3.1*12 15, 1.4.1*12 9, 2.1.1*12 12, 2.3.1*12 1,2.4.1*12 9,  
  3.1.1*12 11, 3.2.1*12 13, 3.4.1*12 19, 4.1.1*12 12, 4.2.1*12 18, 4.3.1*12 2/
```

```
  R(K,S) demanda de la mercancía K es el escenario S  
  /1*12.1 0.7, 1*12.2 1, 1*12.3 1.3/
```

```
  G(I,J,K,S) parametro de activación de arco ;  
  G(I,J,K,S) = MIN(U(I,J),R(K,S));
```

#### VARIABLES

```
  X(I,J) activación del arco I->J  
  Y(I,J,K,S) flujo en el arco I->J para la mercancía K en el escenario S  
  Z costo esperado de la red;
```

```
BINARY VARIABLE X;
```

```
POSITIVE VARIABLE Y;
```

#### EQUATIONS

```
  COSTO funcion objetivo a minimizar  
  BALANCE(I,K,S) balance de flujo en el nodo I para la mercancía K y el escenario S  
  CAPACIDAD(I,J,S) límite de la capacidad de transporte del arco I-->J en escenario S;  
  ACTIVACION(I,J,K,S) activación del arco I->J para la mercancía K y el escenario S  
  COSTO.. Z =E=SUM((I,J)$A(I,J)),F(I,J)*X(I,J))+SUM((I,J,K,S)$A(I,J)),P(S)*C(I,J,K)*Y(I,J,K,S));  
  BALANCE(I,K,S).. SUM(J$(A(I,J)),Y(I,J,K,S))-SUM(J$(A(J,I)),Y(J,I,K,S)) =E= R(K,S)$O(K,I)-  
  R(K,S)$D(K,I));  
  CAPACIDAD(I,J,S)$A(I,J).. SUM(K,Y(I,J,K,S)) =L= U(I,J)*X(I,J);  
  ACTIVACION(I,J,K,S)$A(I,J).. Y(I,J,K,S) =L= G(I,J,K,S)*X(I,J);
```

```
MODEL EJEMPLO /ALL/;
```

```
SOLVE EJEMPLO USING MIP MINIMIZING Z;
```

### **C.3 Codificación GAMS del Problema de Factibilidad de Diseño de Red**

#### **Codificación iterativa en GAMS del esquema del problema de factibilidad del SUNDP**

\$TITLE ESQUEMA DE FACTIBILIDAD DEL PROBLEMA SUNDP

SETS

I nodos de la red //  
K mercancías //  
S escenarios de demanda inicial //  
ST(S)

O(K,I) origen de K //

D(K,I) destino de K //

ALIAS (I,J);

SET

A(I,J) arcos potenciales //;

PARAMETERS

R(K,S) demanda de la mercancía K es el escenario S //;

VARIABLES

VMAS(I,K) variable artificial restricciones de balance  
VMEN(I,K) variable artificial restricciones de balance  
HACT(I,J,K) variable holgura restricciones de activación  
Y(I,J,K) flujo en el arco I->J para la mercancía K  
Z ;

POSITIVE VARIABLE VMAS,VMEN,HACT,Y;

EQUATIONS

COSTO función objetivo a minimizar  
BALANCE(I,K,S) balance de flujo en el nodo I para la mercancía K y el escenario S  
ACTIVACION(I,J,K,S) activación del arco A para la mercancía K y el escenario S;

COSTO.. Z =E= SUM((I,K),VMAS(I,K)+VMEN(I,K));

BALANCE(I,K,ST).. SUM(J\$(A(I,J)),Y(I,J,K))-SUM(J\$(A(J,I)),Y(J,I,K))+VMAS(I,K)-VMEN(I,K)  
=E= R(K,ST)\$O(K,I))-R(K,ST)\$D(K,I));

ACTIVACION(I,J,K,ST)\$A(I,J).. Y(I,J,K)+HACT(I,J,K) =E= R(K,ST);

MODEL FACT /ALL/;

ST(S) = NO;

LOOP(S,

ST(S) = YES;

SOLVE FACT USING LP MINIMIZING Z;

DISPLAY ST,Z,L;

ABORT \$(Z.L > EPS) 'PROBLEMA NO FACTIBLE PARA ESTE ESTADO';

ST(S) = NO;

### **Codificación iterativa en GAMS del esquema del problema de factibilidad del SCNDP**

\$TITLE ESQUEMA DE PROBLEMA DE FACTIBILIDAD DEL SCNDP

SETS

I nodos de la red //  
K mercancías //  
S escenarios de demanda inicial //  
ST(S) escenario actual  
O(K,I) origen de K //  
D(K,I) destino de K //;

ALIAS (I,J);

SET

A(I,J) arcos potenciales //;

PARAMETERS

U(I,J) capacidad de transporte del arco I-->J //

R(K,S) demanda de la mercancía K es el escenario S //

G(I,J,K,S) parametro de activacion de arco ;

G(I,J,K,S) = MIN(U(I,J),R(K,S));

VARIABLES

VMAS(I,K) variable artificial restricciones de balance  
VMEN(I,K) variable artificial restricciones de balance  
HCAP(I,J) variable de holgura restricciones de capacidad  
Y(I,J,K) flujo en el arco I->J para la mercancía K  
Z ;

POSITIVE VARIABLE VMAS,VMEN,HCAP,Y;

EQUATIONS

COSTO funcion objetivo a minimizar

BALANCE(I,K,ST) balance de flujo en el nodo I para la mercancía K y el escenario S

CAPACIDAD(I,J) limite de la capacidad de transporte del arco I-->J en escenario S;

COSTO.. Z =E= SUM((I,K),VMAS(I,K)+VMEN(I,K));

BALANCE(I,K,ST).. SUM(J\$(A(I,J)),Y(I,J,K))-SUM(J\$(A(J,I)),Y(J,I,K))+VMAS(I,K)-VMEN(I,K)  
=E= R(K,ST)\$O(K,I)-R(K,ST)\$D(K,I);

CAPACIDAD(I,J)\$A(I,J).. SUM(K,Y(I,J,K))+HCAP(I,J) =E= U(I,J);

MODEL FACT /ALL/;

ST(S) = NO;

LOOP(S,

ST(S) = YES;

SOLVE FACT USING LP MINIMIZING Z;

DISPLAY ST,Z,L;

ABORT \$(Z.L > EPS) 'PROBLEMA NO FACTIBLE PARA ESTE ESTADO';

ST(S) = NO);

## **Anexo D Codificación GAMS y Soluciones OSL de Casos**

### **D.1 Codificación GAMS de los Casos de Prueba**

#### **Caso 1**

\$TITLE CASO1 DE DISEÑO DE RED CON INCERTIDUMBRE EN LA DEMANDA

SETS

I nodos de la red /1\*7/

K mercancías /1\*42/

S escenarios de demanda inicial /1\*11/

O(K,I) origen de K

/1.1,2.1,3.1,4.1,5.1,6.1,7.2,8.2,9.2,10.2,11.2,12.2,13.3,14.3,15.3,16.3,17.3,18.3,19.4,20.4,21.4,22.4,23.4,24.4,25.5,26.5,27.5,28.5, 29.5,30.5,31.6,32.6,33.6,34.6,35.6,36.6,37.7, .7,39.7,40.7,41.7,42.7/

D(K,I) destino de K

/1.2,2.3,3.4,4.5,5.6,6.7,7.1,8.3,9.4,10.5,11.6,12.7,13.1,14.2,15.4,16.5,17.6,18.7,19.1,20.2,21.3,22.5, 23.6,24.7,25.1,26.2,27.3, 28.4, 29.6,30.7,31.1,32.2,33.3,34.4,35.5,36.7,37.1, .2,39.3,40.4,41.5,42.6/

ALIAS (I,J);

SET

A(I,J) arcos potenciales

/1.2,1.3,1.4,1.5,1.6,1.7,2.1,2.3,2.4,2.5,2.6,2.7,3.1,3.2,3.4,3.5,3.6,3.7,4.1,4.2,4.3,4.5,4.6,4.7,5.1,5.2,5.3,5.4,5.6,5.7,6.1,6.2,6.3,6.4,6.5,6.7,7.1,7.2,7.3,7.4,7.5,7.6/;

PARAMETERS

P(S) probabilidad del escenario S

/1 0.0148,2 0.037,3 0.074,4 0.12,5 0.16,6 0.176,7 0.16,8 0.12,9 0.074,10 0.037,11 0.0148/

C(I,J,K) costo variable de la mercancía K en el arco I-->J

/1.2.1\*42 18,1.3.1\*42 3,1.4.1\*42 8,1.5.1\*42 10,1.6.1\*42 17,1.7.1\*42 16, 2.1.1\*42 1,2.3.1\*42 14,2.4.1\*42 1,2.5.1\*42 9,2.6.1\*42 17,2.7.1\*42 9, 3.1.1\*42 11,3.2.1\*42 13,3.4.1\*42 6,3.5.1\*42 14,3.6.1\*42 5,3.7.1\*42 7, 4.1.1\*42 11,4.2.1\*42 8,4.3.1\*42 8,4.5.1\*42 14,4.6.1\*42 17,4.7.1\*42 1, 5.1.1\*42 13,5.2.1\*42 19,5.3.1\*42 16,5.4.1\*42 7,5.6.1\*42 4,5.7.1\*42 19, 6.1.1\*42 19,6.2.1\*42 2,6.3.1\*42 2,6.4.1\*42 7,6.5.1\*42 12,6.7.1\*42 0, 7.1.1\*42 4,7.2.1\*42 14,7.3.1\*42 14,7.4.1\*42 5,7.5.1\*42 3,7.6.1\*42 11/

R(K,S) demanda de la mercancía K es el escenario S

/1\*42.1 5,1\*42.2 6,1\*42.3 7,1\*42.4 8,1\*42.5 9,1\*42.6 10,1\*42.7 11,1\*42.8 12,1\*42.9 13,1\*42.10 14,1\*42.11 15/

F(I,J) costo fijo de establecer el arco I-->J

/1.2\*7 4000, 2.(1,3\*7) 4000, 3.(1\*2,4\*7) 4000, 4.(1\*3,5\*7) 4000, 5.(1\*4,6\*7) 4000, 6.(1\*5,7) 4000, 7.1\*6 4000/

U(I,J) capacidad de transporte del arco I-->J

/1.2 70,1.3\*7 80,2.1 60,2.3 80,2.(4\*5,7) 70, 2.6 90,3.1 70,3.(2,4,6\*7) 80,3.5 90, 4.1\*2 80, 4.(3,5\*7) 70,5.1 80,5.2\*3 70,5.4 90,5.6\*7 60,6.(1,7) 90,6.(2,4\*5) 80,6.3 70, 7.(1,4,6) 60,7.(2,5) 70,7.3 80/;

## Caso 2

### \$TITLE CASO 2 DE DISEÑO DE RED CON INCERTIDUMBRE EN LA DEMANDA

#### SETS

I nodos de la red /1\*9/  
K mercancías /1\*72/  
S escenarios de demanda inicial /1\*3/

#### O(K,I) origen de K

/1.1,2.1,3.1,4.1,5.1,6.1,7.1,8.1,9.2,10.2,11.2,12.2,13.2,14.2,15.2,16.2,17.3,18.3,19.3,20.3,21.3,22.3,23.3,24.3,25.4,26.4,27.4,28.4,29.4,30.4,31.4,32.4,33.5,34.5,35.5,36.5,37.5,38.5,39.5,40.5,41.6,42.6,43.6,44.6,45.6,46.6,47.6,48.6,49.7,50.7,51.7,52.7,53.7,54.7,55.7,56.7,57.8,58.8,59.8,60.8,61.8,62.8,63.8,64.8, 65.9,66.9,67.9,68.9,69.9,70.9,71.9,72.9/

#### D(K,I) destino de K

/1.2,2.3,3.4,4.5,5.6,6.7,7.8,8.9,9.1,10.3,11.4,12.5,13.6,14.7,15.8,16.9,17.1,18.2,19.4,20.5,21.6,22.7,23.8,24.9,25.1,26.2,27.3,28.5,29.6,30.7,31.8,32.9,33.1,34.2,35.3,36.4,37.6,38.7,39.8,40.9,41.1,42.2,43.3,44.4,45.5,46.7,47.8,48.9,49.1,50.2,51.3,52.4,53.5,54.6,55.8,56.9,57.1,58.2,59.3,60.4,61.5,62.6,63.7,64.9,65.1,66.2, 67.3,68.4,69.5,70.6,71.7,72.8/

#### ALIAS (I,J);

#### SET

#### A(I,J) arcos potenciales

/1.2,1.5,1.7,1.9,2.3,2.5,2.6,2.9,3.2,3.6,3.8,4.1,4.2,4.3,4.7,4.8,4.9,5.2,5.3,5.4,5.7,5.9,6.3,6.4,6.8,7.1,7.4,7.5,7.6,8.2,8.4,8.6,8.7,9.1,9.5/;

#### PARAMETERS

#### P(S) probabilidad del escenario S

/1 0.2, 2 0.5, 3 0.3/

#### C(I,J,K) costo variable de la mercancía K en el arco I-->J

/1.2.1\*72 84,1.5.1\*72 22,1.7.1\*72 48,1.9.1\*72 84,2.3.1\*72 41,2.5.1\*72 68, 2.6.1\*72 71,2.9.1\*72 73,3.2.1\*72 87,3.6.1\*72 69,3.8.1\*72 52,4.1.1\*72 77, 4.2.1\*72 50,4.3.1\*72 44,4.7.1\*72 56,4.8.1\*72 82,4.9.1\*72 60,5.2.1\*72 70, 5.3.1\*72 87,5.4.1\*72 62,5.7.1\*72 70,5.9.1\*72 21,6.3.1\*72 36,6.4.1\*72 52, 6.8.1\*72 43,7.1.1\*72 67,7.4.1\*72 50,7.5.1\*72 78,7.6.1\*72 76,8.2.1\*72 83, 8.4.1\*72 57,8.6.1\*72 51,8.7.1\*72 81,9.1.1\*72 51,9.5.1\*72 18/

#### F(I,J) costo fijo de establecer el arco I-->J

/1.2 8448,1.5 2318,1.7 4915,1.9 8574,2.3 4132,2.5 6868, 2.6 7159,2.9 7415, 3.2 8783,3.6 6940,3.8 5284,4.1 7791,4.2 5077,4.3 4421,4.7 5666,4.8 8273, 4.9 6146,5.2 7047,5.3 8741,5.4 6342,5.7 7095,5.9 2184,6.3 3704,6.4 5357, 6.8 4396,7.1 6826,7.4 5135,7.5 7930,7.6 7656,8.2 8457,8.4 5805,8.6 5270, 8.7 8315,9.1 5191,9.5 1944/

#### U(I,J) capacidad de transporte del arco I-->J

/1.(2,5,7,9) 255,2.(3,5,6,9) 255,3.(2,6,8) 255,4.(1\*3,7\*9) 255,5.(2\*4,7,9) 255,6.(3,4,8) 255,7.(1,4\*6) 255,8.(2,4,6,7) 255,9.(1,5) 255/

#### R(K,S) demanda de la mercancía K es el escenario S

/1.1 14,1.2 17,1.3 20,2.1 34,2.2 43,2.3 52,3.1 35, 3.2 44, 3.3 53,4.1 36, 4.2 45,4.3 54,5.1 38,5.2 48, 5.3 58, 6.1 73, 6.2 91, 6.3 109,7.1 3,7.2 4, 7.3 5,8.1 71,8.2 89,8.3 107,9.1 78,9.2 98,9.3 118,10.1 60,10.2 75,10.3 90, 11.1 62,11.2 77,11.3 92,12.1 66,12.2 83,12.3 100,13.1 54,13.2

67,13.3 80, 14.1 19,14.2 24,14.3 29,15.1 53,15.2 66,15.3 79,16.1 34,16.2 42,16.3 50, 17.1  
 56,17.2 70,17.3 84,18.1 11,18.2 14,18.3 17,19.1 74,19.2 93,19.3 112, 20.1 67,20.2 84,20.3  
 101,21.1 79,21.2 99,21.3 119,22.1 13,22.2 16,22.3 19, 23.1 50,23.2 63,23.3 76,24.1 68,24.2  
 85,24.3 102,25.1 9,25.2 11,25.3 13, 26.1 34,26.2 43,26.3 52,27.1 25,27.2 31,27.3 37,28.1  
 62,28.2 78,28.3 94, 29.1 48,29.2 60,29.3 72,30.1 59,30.2 74,30.3 89,31.1 18,31.2 22,31.3 26,  
 32.1 5,32.2 6,32.3 7,33.1 10,33.2 13,33.3 16,34.1 6,34.2 8,34.3 10, 35.1 10,35.2 13,35.3  
 16,36.1 8,36.2 10,36.3 12,37.1 5,37.2 6,37.3 7, 38.1 70,38.2 88,38.3 106,39.1 70,39.2 88,39.3  
 106,40.1 72,40.2 90,40.3 108, 41.1 10,41.2 12,41.3 14,42.1 46,42.2 57,42.3 68,43.1 78,43.2  
 98,43.3 118, 44.1 18,44.2 22,44.3 26,45.1 22,45.2 27,45.3 32,46.1 10,46.2 12,46.3 14, 47.1  
 13,47.2 16,47.3 19,48.1 10,48.2 12,48.3 14,49.1 26,49.2 33,49.3 40, 50.1 60,50.2 75,50.3  
 0,51.1 78,51.2 98,51.3 118,52.1 75,52.2 94,52.3 113, 53.1 69,53.2 86,53.3 103,54.1 33,54.2  
 41,54.3 49,55.1 20,55.2 25,55.3 30, 56.1 52,56.2 65,56.3 78,57.1 2,57.2 3,57.3 4,58.1 70,58.2  
 87,58.3 104, 59.1 6,59.2 8,59.3 10,60.1 18,60.2 23,60.3 28,61.1 4,61.2 5,61.3 6, 62.1 14,62.2  
 17,62.3 20,63.1 29,63.2 36,63.3 43,64.1 10,64.2 12,64.3 14, 65.1 19,65.2 24,65.3 29,66.1 ,66.2  
 6,66.3 7,67.1 11,67.2 14,67.3 17, 68.1 64,68.2 80,68.3 96,69.1 10,69.2 13,69.3 16,70.1 58,70.2  
 72,70.3 86, 71.1 63,71.2 79,71.3 95,72.1 29,72.2 36,72.3 43/;

### Caso 3

\$TITLE CASO 3 DE DISEÑO DE RED CON INCERTIDUMBRE EN LA DEMANDA

#### SETS

I nodos de la red /1\*14/  
 K mercancías /1\*90/  
 S escenarios de demanda inicial /1\*3/

#### O(K,I) origen de K

/1.1,2.1,3.1,4.1,5.1,6.1,7.1,8.1,9.1,10.2,11.2,12.2,13.2,14.2,15.2,16.2,17.2,  
 18.2,19.3,20.3,21.3,22.3,23.3,24.3,25.3,26.3,27.3,28.4,29.4,30.4,31.4,32.4,  
 33.4,34.4,35.4,36.4,37.5,38.5,39.5,40.5,41.5,42.5,43.5,44.5,45.5,46.10,47.10,48.10,  
 49.10,50.10,51.10,52.10,53.10,54.10,55.11,56.11,57.11,58.11,59.11,60.11,61.11,62.11,  
 63.11,64.12,65.12,66.12,67.12,68.12,69.12,70.12,71.12,72.12,73.13,74.13,75.13,76.13,  
 77.13,78.13,79.13,80.13,81.13,82.14,83.14,84.14,85.14,86.14,87.14,88.14,89.14,90.14/

#### D(K,I) destino de K

/1.2,2.3,3.4,4.5,5.10,6.11,7.12,8.13,9.14,10.1,11.3,12.4,13.5,14.10,15.11,16.12,17.13,  
 18.14,19.1,20.2,21.4,22.5,23.10,24.11,25.12,26.13,27.14,28.1,29.2,30.3,31.5,32.10,  
 33.11,34.12,35.13,36.14,37.1,38.2,39.3,40.4,41.10,42.11,43.12,44.13,45.14,46.1,47.2,48.3,  
 49.4,50.5,51.11,52.12,53.13,54.14,55.1,56.2,57.3,58.4,59.5,60.10,61.12,62.13,63.14,64.1,  
 65.2,66.3,67.4,68.5,69.10,70.11,71.13,72.14,73.1,74.2,75.3,76.4,77.5,78.10,79.11,80.12,  
 81.14,82.1,83.2,84.3,85.4,86.5,87.10,88.11,89.12,90.13/

ALIAS (I,J);

#### SET

A(I,J) arcos potenciales  
 /1.3,1.5,1.10,1.14,2.1,2.4,2.7,2.8,3.1,3.2,3.8,3.14,4.2,4.5,4.7,4.11,5.4,5.6,5.9,5.13,  
 6.1,6.4,6.9,6.10,7.2,7.10,7.12,7.14,8.5,8.6,8.11,8.13,9.6,9.7,9.10,9.12,10.3,10.6,10.9,  
 10.12,11.2,11.3,11.4,11.13,12.2,12.5,12.7,12.14,13.1,13.3,13.8,13.11,14.1,14.4,14.10,14.12/;

#### PARAMETERS

P(S) probabilidad del escenario S  
 /1 0.35, 2 0.5, 3 0.15/

C(I,J,K) costo variable de la mercancía K en el arco I-->J

/1.3.1\*90 35,1.5.1\*90 63,1.10.1\*90 35,1.14.1\*90 33,2.1.1\*90 48,2.4.1\*90 72,2.7.1\*90  
44,2.8.1\*90 84, 3.1.1\*90 53,3.2.1\*90 89,3.8.1\*90 47,3.14.1\*90 58,4.2.1\*90 52,4.5.1\*90  
79,4.7.1\*90 91,4.11.1\*90 94, 5.4.1\*90 31,5.6.1\*90 84,5.9.1\*90 64,5.13.1\*90 43,6.1.1\*90  
87,6.4.1\*90 98,6.9.1\*90 68,6.10.1\*90 12, 7.2.1\*90 81,7.10.1\*90 36,7.12.1\*90 53,7.14.1\*90  
95,8.5.1\*90 70,8.6.1\*90 86,8.11.1\*90 60,8.13.1\*90 16, 9.6.1\*90 95,9.7.1\*90 48,9.10.1\*90  
66,9.12.1\*90 78,10.3.1\*90 53,10.6.1\*90 67,10.9.1\*90 54,10.12.1\*90 84, 11.2.1\*90 42,11.3.1\*90  
66,11.4.1\*90 36,11.13.1\*90 38,12.2.1\*90 84,12.5.1\*90 88,12.7.1\*90 42,12.14.1\*90 80,  
13.1.1\*90 61,13.3.1\*90 38,13.8.1\*90 16,13.11.1\*90 37,14.1.1\*90 19,14.4.1\*90 52,14.10.1\*90  
83,14.12.1\*90 42/

F(I,J) costo fijo de establecer el arco I-->J

/1.3 3679,1.5 6454,1.10 3591,1.14 3398,2.1 4893,2.4 7317,2.7 4472,2.8 8477,3.1 5482,3.2  
8980,3.8 4781, 3.14 5955,4.2 5345,4.5 8011,4.7 9278,4.11 9484,5.4 3227,5.6 8557,5.9  
6535,5.13 4432,6.1 8752,6.4 9844, 6.9 6943,6.10 1331,7.2 8163,7.10 3726,7.12 5340,7.14  
9584,8.5 7127,8.6 8730,8.11 6118,8.13 1690, 9.6 9580,9.7 4849,9.10 6638,9.12 7852,10.3  
5428,10.6 6841,10.9 5536,10.12 8496,11.2 4377,11.3 6649, 11.4 3745,11.13 3875,12.2  
8518,12.5 8877,12.7 4368,12.14 8057,13.1 6262,13.3 3903,13.8 1675,13.11 3830,  
14.1 2038,14.4 5335,14.10 8395,14.12 4318/

U(I,J) capacidad de transporte del arco I-->J

/1.(3,5,10,14) 400,2.(1,4,7,8) 400,3.(1,2,8,14) 400,4.(2,5,7,11) 400,5.(4,6,9,13) 400,6.(1,4,9,10)  
400, 7.(2,10,12,14) 400,8.(5,6,11,13) 400,9.(6,7,10,12) 400,10.(3,6,9,12) 400,11.(2\*4,13)  
400,12.(2,5,7,14) 400, 13.(1,3,8,11) 400,14.(1,4,10,12) 400/

R(K,S) demanda de la mercancía K es el escenario S

/1.1 121,1.2 93,1.3 65,2.1 125,2.2 96,2.3 67,3.1 1, 3.2 1, 3.3 1,4.1 83,4.2 64,4.3 45,  
5.1 101,5.2 78,5.3 55,6.1 61,6.2 47,6.3 33,7.1 8,7.2 11,7.3 14,8.1 25,8.2 36,8.3 47,  
9.1 50,9.2 72,9.3 94,10.1 38,10.2 54,10.3 70,11.1 47,11.2 67,11.3 87,12.1 12,12.2 17,12.3 22,  
13.1 13,13.2 19,13.3 25,14.1 101,14.2 78,14.3 55,15.1 78,15.2 60,15.3 42,16.1 107,16.2  
82,16.3 57, 17.1 33,17.2 25,17.3 18,18.1 3,18.2 2,18.3 1,19.1 23,19.2 18,19.3 13,20.1 15,20.2  
22,20.3 29, 21.1 42,21.2 60,21.3 78,22.1 31,22.2 44,22.3 57,23.1 32,23.2 45,23.3 59,24.1  
16,24.2 23,24.3 30, 25.1 39,25.2 56,25.3 73,26.1 38,26.2 54,26.3 70,27.1 56,27.2 43,27.3  
30,28.1 114,28.2 88,28.3 62, 29.1 91,29.2 70,29.3 49,30.1 47,30.2 36,30.3 25,31.1 111,31.2  
85,31.3 60,32.1 92,32.2 71,32.3 50, 33.1 1,33.2 1,33.3 1,34.1 44,34.2 63,34.3 82,35.1 13,35.2  
18,35.3 23,36.1 8,36.2 11,36.3 14, 37.1 69,37.2 99,37.3 129,38.1 64,38.2 91,38.3 118,39.1  
46,39.2 66,39.3 86,40.1 10,40.2 8,40.3 6, 41.1 111,41.2 85,41.3 60,42.1 86,42.2 66,42.3  
46,43.1 113,43.2 87,43.3 61,44.1 60,44.2 46,44.3 32, 45.1 62,45.2 48,45.3 34,46.1 8,46.2  
12,46.3 16,47.1 34,47.2 48,47.3 62,48.1 47,48.2 67,48.3 87, 49.1 25,49.2 35,49.3 46,50.1  
6,50.2 9,50.3 12,51.1 8,51.2 11,51.3 14,52.1 57,52.2 81,52.3 105, 53.1 88,53.2 68,53.3 48,54.1  
46,54.2 35,54.3 25,55.1 85,55.2 65,55.3 46,56.1 14,56.2 11,56.3 8, 57.1 30,57.2 23,57.3  
16,58.1 46,58.2 35,58.3 25,59.1 33,59.2 47,59.3 61,60.1 6,60.2 8,60.3 10, 61.1 5,61.2 7,61.3  
9,62.1 33,62.2 47,62.3 61,63.1 50,63.2 72,63.3 94,64.1 18,64.2 26,64.3 34, 65.1 41,65.2  
59,65.3 77,66.1 22,66.2 17,66.3 12,67.1 34,67.2 26,67.3 18,68.1 52,68.2 40,68.3 28,  
69.1 34,69.2 26,69.3 18,70.1 16,70.2 12,70.3 8,71.1 8,71.2 6,71.3 4,72.1 9,72.2 13,72.3 17,  
73.1 41,73.2 58,73.3 75,74.1 39,74.2 55,74.3 72,75.1 18,75.2 25,75.3 33,76.1 5,76.2 7,76.3 9,  
77.1 15,77.2 22,77.3 29,78.1 42,78.2 60,78.3 78,79.1 43,79.2 33,79.3 23,80.1 53,80.2 41,80.3  
29, 81.1 65,81.2 50,81.3 35,82.1 1,82.2 1,82.3 1,83.1 99,83.2 76,83.3 53,84.1 20,84.2 15,84.3  
11, 85.1 9,85.2 13,85.3 17,86.1 1,86.2 1,86.3 1,87.1 36,87.2 51,87.3 66,88.1 42,88.2 60,88.3  
78, 89.1 6,89.2 9,89.3 12,90.1 41,90.2 59,90.3 77/



## ***D.2 Soluciones del Solver OSL de los Casos de Prueba***

### ***Caso de Diseño 1***

Problema SUNDP

Costo óptimo de la red : 32148

Costos de transporte: 4148

Costos fijos: 28000

Arcos de la solución óptima:

origen	destino
1	3
3	4
4	7
7	5
5	6
6	2
2	1

Problema SCNDP

Costo de la red (subóptimo) : 68273

Costos de transporte: 4273

Costos fijos: 64000

Arcos de la solución óptima:

origen	destino
1	3
1	6
2	1
2	3
2	4
2	6
3	5
3	6
4	2
4	7
5	1
5	4
6	2
6	7
7	1
7	5

### **Caso de Diseño 2**

Problema SUNDP

Costo óptimo de la red : 438667

Costos de transporte: 336487

Costos fijos: 102180

Arcos de la solución óptima:

origen	destino
1	5
1	7
2	3
2	5
3	6
3	8
4	2
4	3
4	7
4	9
5	4
5	7
5	9
6	3
6	4
7	4
7	6
8	4
9	1
9	5

Problema SCNDP

Costo óptimo de la red : 498814

Costos de transporte: 316952

Costos fijos: 181862

Arcos de la solución óptima:

origen	destino
1	2
1	5
1	7
2	3
2	5
2	6
2	9
3	2
3	6
3	8
4	2
4	3
4	7
4	8
4	9
5	2
5	4
5	7
5	9
6	3
6	4
6	8
7	1
7	4
7	5
7	6
8	4
8	6
8	7
9	1
9	5

### Caso de Diseño 3

Problema SUNDP

Costo óptimo de la red : 503779

Costos de transporte: 397155

Costos fijos: 106624

Arcos de la solución óptima:

origen	destino
1	3
1	5
1	10
1	14
2	1
3	8
3	14
4	2
4	5
5	4
5	13
8	13
10	3
11	2
11	4
11	13
12	14
13	1
13	3
13	11
14	1
14	4
14	12

Problema SCNDP

Costo óptimo de la red : 535885

Costos de transporte: 388611

Costos fijos: 147274

Arcos de la solución óptima:

origen	destino
1	3
1	5
1	10
1	14
2	1
2	7
3	8
3	14
4	2
4	5
4	7
5	4
5	13
7	10
7	12
8	13
10	3
10	12
11	2
11	4
11	13
12	2
12	5
13	1
13	3
13	11
14	1
14	4
14	12

## Anexo E Heurística de Ascenso Dual

### E.1 Ejemplo de Aplicación de la Heurística de Ascenso Dual al UNDP

Se considera el problema de diseño sobre la red potencial de la Figura E.1, en la que sobre cada arco se indica su costo fijo y su costo total de transporte, en ese orden:

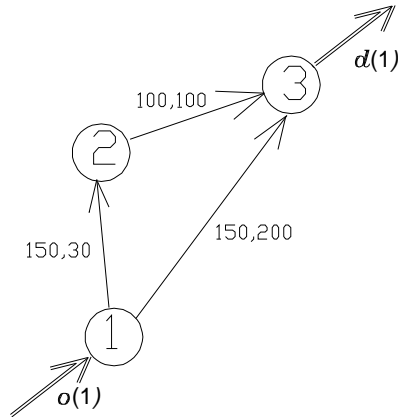


Figura E.1 Red Potencial para ejemplo de Ascenso Dual

Por ejemplo, el arco 1→2 tiene costo fijo 150 y costo de transporte 30.

Hay una única commodity, con origen en el nodo 1 y destino en el nodo 3.

#### Inicialización

Considerando sólo los costos de transporte, el camino de largo mínimo desde (1) a (3) es (1)→(2)→(3), por lo que se tiene:

$$\beta_{12}^1 = \beta_{23}^1 = \beta_{13}^1 = 0$$

$$\pi_1^1 = 0, \pi_2^1 = 30, \pi_3^1 = 130$$

$$r_{12} = 150, r_{23} = 100, r_{13} = 150$$

$$h_{12}^1 = 0, h_{23}^1 = 0, h_{13}^1 = 70$$

$$z\mathbf{d} = \pi_3^1 = 130$$

$$\mathbf{N}_u(1) = \{1, 2\} \text{ nodos no etiquetados de la commodity 1 (unlabeled)}$$

$$\mathbf{K}_u = \{1\} \text{ conjunto de commodities candidatas para la iteración}$$

#### Iteración de ascenso 1

$$\mathbf{A}(\mathbf{N}_u(1)) = \{(2, 3), (1, 3)\}$$

$$\{(i, j) \in \mathbf{A}(\mathbf{N}_u(k)) / h_{ij}^k = 0\} = \{(2, 3)\}$$

$$\{(i, j) \in \mathbf{A}(\mathbf{N}_u(k)) / h_{ij}^k > 0\} = \{(1, 3)\}$$

$$\delta_1 = 100, \delta_2 = 70$$

$$\delta = \min\{100, 70\} = 70$$

$$r_{12} = 150, r_{23} = 30, r_{13} = 150$$

$$\begin{aligned}h_{12}^1 &= 0, h_{23}^1 = 0, h_{13}^1 = 0 \\ \pi_1^1 &= 0, \pi_2^1 = 30, \pi_3^1 = 200 \\ \mathbf{zd} &= \pi_3^1 = 200 \\ \beta_{12}^1 &= 0, \beta_{23}^1 = 70, \beta_{13}^1 = 0\end{aligned}$$

Iteración sobre commodities

$$K_u = \emptyset$$

Test de Fin

$$K_u = \{1\}$$

¿ $K_u = \emptyset$ ? NO  $\Rightarrow$  SEGUIR

Iteración 2

$$\begin{aligned}A(N_u(1)) &= \{(2, 3), (1, 3)\} \\ \{(i, j) \in A(N_u(k)) / h_{ij}^k = 0\} &= \{(2, 3), (1, 3)\} \\ \{(i, j) \in A(N_u(k)) / h_{ij}^k > 0\} &= \emptyset \\ \delta_1 &= 30, \delta_2 = +\infty \\ \delta &= 30 \\ r_{12} &= 150, r_{23} = 0, r_{13} = 120 \\ h_{12}^1 &= 0, h_{23}^1 = 0, h_{13}^1 = 0 \\ \pi_1^1 &= 0, \pi_2^1 = 30, \pi_3^1 = 230 \\ \mathbf{zd} &= \pi_3^1 = 230 \\ \beta_{12}^1 &= 0, \beta_{23}^1 = 100, \beta_{13}^1 = 30\end{aligned}$$

Etiquetado

$$N_u(1) = \{1\}$$

Iteración sobre commodities

$$K_u = \emptyset$$

Test de Fin

$$K_u = \{1\}$$

¿ $K_u = \emptyset$ ? NO  $\Rightarrow$  SEGUIR

Iteración 3

$$\begin{aligned}A(N_u(1)) &= \{(1, 2), (1, 3)\} \\ \{(i, j) \in A(N_u(k)) / h_{ij}^k = 0\} &= \{(1, 2), (1, 3)\}\end{aligned}$$

$$\{(i, j) \in A(N_u(k)) / h_{ij}^k > 0\} = \emptyset$$

$$\delta_1 = \min\{150, 120\} = 120, \quad \delta_2 = +\infty$$

$$\delta = 120$$

$$r_{12} = 30, r_{23} = 0, r_{13} = 0$$

$$h_{12}^1 = 0, h_{23}^1 = 0, h_{13}^1 = 0$$

$$\pi_1^1 = 0, \pi_2^1 = 150, \pi_3^1 = 350$$

$$zd = \pi_3^1 = 350$$

$$\beta_{12}^1 = 120, \beta_{23}^1 = 100, \beta_{13}^1 = 150$$

### Etiquetado

$$N_u(1) = \emptyset$$

### Iteración sobre commodities

$$K_u = \emptyset$$

### Test de Fin

$$K_u = \emptyset$$

¿ $K_u = \emptyset$ ? SI  $\Rightarrow$  FIN DEL ALGORITMO

Se tiene al final del algoritmo que  $A_0 = \{(i, j) \in A / r_{ij} = 0\} = \{(2, 3), (1, 3)\}$

En la Figura E.2 se ilustra esta subred indicando en los arcos los valores  $(\beta_{ij}^k, c_{ij}^k)$ .

Sobre esta subred se halla el camino de costo mínimo para la commodity 1 desde su origen en el nodo 1 hasta su destino en el nodo 3, que resulta ser el arco (1,3), el cual es el diseño solución proporcionado por la heurística y se muestra en la Figura E.3. En este caso la solución heurística coincide con la solución óptima del problema.

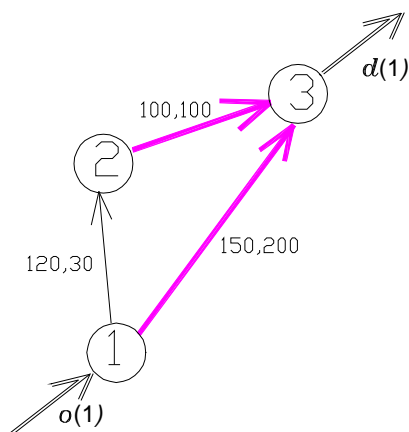


Figura E.2 Arcos con  $r_{ij} = 0$

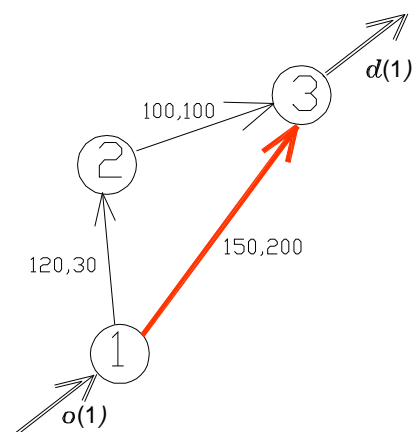


Figura E.3 Solución Heurística

## ***E.2 Código Fuente C++ del Programa que Implementa la Heurística***

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream.h>
#include <limits.h>

#define LARGO_LINEA 80
#define BLANCO ' '
#define INF LONG_MAX

typedef struct {
    int orig;
    int dest;
    long dem;
    int es_cand;
    int *Nu;
}comm;

typedef comm *conj_comm;

typedef struct {
    long h;
    long b;
} holg;

typedef struct {
    long c_unit;
    long c_fijo;
    long costo;
    int r;
    int es_sol;
    holg *conj_holg;
}arco;

typedef arco *p_arco;

typedef p_arco **grafo;

typedef long **min_cost;

typedef struct {
    long dist;
    int pred;
    int conocido;
}nodo_dijk;

typedef nodo_dijk *conj_dijk;
```

```

void cargar_datos(char *arch_datos, grafo &G, conj_comm &CK, int &N, int &K)
{
    FILE *fdatos;
    char *buff_datos = new char[LARGO_LINEA];
    char *p;
    int e, i, j, v, a, c, k;

    if((fdatos=fopen(arch_datos, "r"))==NULL){
        cerr << "Error: no se pudo abrir el archivo de datos \n";
        exit(1);
    }
    fgets(buff_datos, LARGO_LINEA, fdatos);
    p = buff_datos;
    N = atoi(p);
    while(*p++!= BLANCO);
    e = atoi(p);
    while(*p++!= BLANCO);
    K = atoi(p);
    G = new p_arco *[N+1];
    for(v=1; v<=N; v++)
        G[v] = new p_arco[N+1];
    for(i=1; i<=N; i++)
        for(j=1; j<=N; j++)
            G[i][j]=NULL;
    // loop de cargado de los arcos de la red
    for(a=1; a<=e; a++){
        if(fgets(buff_datos, LARGO_LINEA, fdatos)==NULL){
            cerr << "Error: el total de arcos no coincide con los presentes \n";
            exit(1);
        }
    }
    else{
        p = buff_datos;
        i = atoi(p);
        while(*p++!= BLANCO);
        j = atoi(p);
        G[i][j] = new arco;
        while(*p++!= BLANCO);
        G[i][j] -> c_unit = atol(p);
        while(*p++!= BLANCO);
        G[i][j] -> c_fijo = atol(p);
        G[i][j] -> conj_holg = new holg[K+1];
    }
    }
    CK = new comm[K+1];
    // loop de cargado de las commodities
    for(c=1; c<=K; c++){
        if(fgets(buff_datos, LARGO_LINEA, fdatos)==NULL){
            cerr << "Error: el total de commodities no coincide con las presentes \n";
            exit(1);
        }
    }
    else{
        p = buff_datos;
        k = atoi(p);
        while(*p++!= BLANCO);
        CK[k].orig = atoi(p);
    }
}

```



```

while(*p++!= BLANCO);
CK[k].dest = atoi(p);
while(*p++!= BLANCO);
CK[k].dem = atol(p);
CK[k].Nu = new int[N+1];
}
}
fclose(fdatos);
}

void listar_solucion(char *arch_sal, grafo &G, int N, long z, long zt, long zd)
{
int i, j;
FILE *arch;
arch = fopen(arch_sal, "w");
fprintf(arch, "el costo total de la solucion de ascenso dual es %ld \n", z);
fprintf(arch, "el costo de transporte de la solucion es %ld \n", zt);
fprintf(arch, "el costo fijo de la solucion es %ld \n", z-zt);
fprintf(arch, "la cota inferior solucion del dual vale %ld \n", zd);
fprintf(arch, "el conjunto de arcos solucion es: \n");
cout << "el costo total de la solucion de ascenso dual es " << z << "\n";
cout << "el costo de transporte de la solucion es " << zt << "\n";
cout << "el costo fijo de la solucion es " << z-zt << "\n";
cout << "la cota inferior solucion del dual vale " << zd << "\n";
cout << "el conjunto de arcos solucion es:" << "\n";
for(i=1; i<=N; i++)
for(j=1; j<=N; j++)
if(G[i][j]!=NULL){
fprintf(arch, "%d --> %d\n", i, j);
cout << i << " --> " << j << "\n";
}
fclose(arch);
}

void Dijkstra(grafo &G, int N, int origen, conj_dijk &CD)
{
int v, i, v_min;
long d_min;

for(v=1; v<=N; v++){
CD[v].dist = INF;
CD[v].pred = 0;
CD[v].conocido = 0;
}
CD[origen].dist = 0;
CD[origen].conocido = 1;
for(v=1; v<=N; v++){
if(G[origen][v]!=NULL){
CD[v].dist = G[origen][v]->costo;
CD[v].pred = origen;
}
}
}

```

```

for(i=1;i<N;i++){
    d_min = INF;
    for(v=1;v<=N;v++){
        if(CD[v].conocido==0 && CD[v].dist<d_min){
            v_min = v;
            d_min = CD[v].dist;
        }
    }
    if(d_min<INF){
        CD[v_min].conocido=1;
        for(v=1;v<=N;v++){
            if(G[v_min][v]!=NULL && CD[v].conocido==0)
                if((CD[v_min].dist+G[v_min][v]->costo) < CD[v].dist){
                    CD[v].dist = CD[v_min].dist+G[v_min][v]->costo;
                    CD[v].pred = v_min;
                }
        }
    }
    else break;
}
}

void cargar_costo_k(grafo &G,int N,conj_comm &CK,int k)
{
    int i,j;
    for(i=1;i<=N;i++)
        for(j=1;j<=N;j++)
            if(G[i][j]!=NULL)
                G[i][j]->costo = G[i][j]->c_unit*CK[k].dem;
}

void cargar_costo_modif_k(grafo &G,int N,conj_comm &CK,int k)
{
    int i,j;
    for(i=1;i<=N;i++)
        for(j=1;j<=N;j++)
            if(G[i][j]!=NULL)
                G[i][j]->costo=G[i][j]->c_unit*CK[k].dem+G[i][j]->conj_holg[k].b;
}

int vacio_Ku(conj_comm &CK, int K)
{
    int k;
    for(k=1;k<=K;k++)
        if(CK[k].es_cand) break;
    return (k<=K?0:1);
}

// precondition: no vacio Ku
int elegir_commodity(conj_comm &CK, int K)
{
    int k;
    for(k=1;k<=K;k++)
        if(CK[k].es_cand) break;
    return (k);
}

```

```

void inicializar(grafo &G,int N,conj_comm &CK,int K,conj_dijk &CD,min_cost &PI)
{
    int i,j,v,k;

    // Definicion de estructuras auxiliares
    CD = new nodo_dijk[N+1];
    PI = new long *[K+1];
    for(k=1;k<=K;k++)
        PI[k] = new long[N+1];
    // Inicializacion
    for(k=1;k<=K;k++){ // loop de calculo de distancias mas cortas
        cargar_costo_k(G,N,CK,k);
        Dijkstra(G,N,CK[k].orig,CD);
        for(v=1;v<=N;v++)
            PI[k][v] = CD[v].dist;
    }
    for(i=1;i<=N;i++)
        for(j=1;j<=N;j++)
            if(G[i][j]!=NULL){
                G[i][j]->r = G[i][j]->c_fijo;
                G[i][j]->es_sol = 0;
                for(k=1;k<=K;k++)
                    if(PI[k][i]!=INF){
                        G[i][j]->conj_holg[k].h=G[i][j]->c_unit*CK[k].dem+PI[k][i]-PI[k][j];
                        G[i][j]->conj_holg[k].b=0;
                    }
                else G[i][j]->conj_holg[k].h = INF;
            }
    for(k=1;k<=K;k++){
        CK[k].es_cand=1;
        for(v=1;v<=N;v++)
            CK[k].Nu[v]=1;
        CK[k].Nu[CK[k].dest]=0;
    }
}

long delta1(grafo &G,int N,conj_comm &CK,int k)
{
    int i,j;
    long min = INF;

    for(i=1;i<=N;i++)
        for(j=1;j<=N;j++)
            if(G[i][j]!=NULL) //si hay arco
                if(CK[k].Nu[i]==1&&CK[k].Nu[j]==0) //si el arco es de A(Nu(k))
                    if(G[i][j]->conj_holg[k].h==0) //si el arco esta saturado
                        if(G[i][j]->r < min)
                            min = G[i][j]->r;
    return min;
}

```

```

long delta2(grafo &G,int N,conj_comm &CK,int k)
{
int i,j;
long min = INF;

for(i=1;i<=N;i++)
for(j=1;j<=N;j++)
if(G[i][j]!=NULL) //si hay arco
if(CK[k].Nu[i]==1&&CK[k].Nu[j]==0) //si el arco es de A(Nu(k))
if(G[i][j]->conj_holg[k].h>0) // si el arco no esta saturado
if(G[i][j]->conj_holg[k].h < min)
min = G[i][j]->conj_holg[k].h;
return min;
}

void actualizar_holguras(grafo &G,int N,conj_comm &CK,int k,long delta)
{
int i,j;

for(i=1;i<=N;i++)
for(j=1;j<=N;j++)
if(G[i][j]!=NULL)
if(CK[k].Nu[i]==1&&CK[k].Nu[j]==0)
if(G[i][j]->conj_holg[k].h<INF)
if(G[i][j]->conj_holg[k].h==0){
G[i][j]->r -= delta;
G[i][j]->conj_holg[k].b += delta;
}
else G[i][j]->conj_holg[k].h -= delta;
}

void etiquetar_nodo(grafo &G,int N,conj_comm &CK,int k)
{
int i,j;
int encuentre=0;
for(i=1;i<=N&&!encuentre;i++)
for(j=1;j<=N&&!encuentre;j++)
if(G[i][j]!=NULL) //si hay arco
if(CK[k].Nu[i]==1&&CK[k].Nu[j]==0) //si el arco es de A(Nu(k))
if(G[i][j]->conj_holg[k].h==0) //si el arco esta saturado
if(G[i][j]->r==0){ //si el costo fijo remanente del arco es 0
CK[k].Nu[i] = 0; //etiquetar nodo cabeza del arco
encuentre=1;
}
}

void actualizar_Ku(conj_comm &CK,int K)
{
int k;
for(k=1;k<=K;k++)
if(CK[k].Nu[CK[k].orig])
CK[k].es_cand = 1;
}

```

```

void armar_solucion(grafo &G,int N,conj_comm &CK,int K,conj_dijk &CD,long &z,long &zt,long
&zd)
{
int i,j,k,v,w;
z = zt = zd = 0;
//borrado de arcos con costo fijo remanente r != 0
for(i=1;i<=N;i++)
for(j=1;j<=N;j++)
if(G[i][j]!=NULL)
if(G[i][j]->r>0){
delete G[i][j];
G[i][j] = NULL;
}
//calculo de caminos mas cortos con costos modificados y marcaje de arcos
for(k=1;k<=K;k++){
cargar_costo_modif_k(G,N,CK,k);
Dijkstra(G,N,CK[k].orig,CD);
w = CK[k].dest;
zd += CD[w].dist;
v=CD[w].pred;
while(v!=0){
G[v][w]->es_sol=1;
w = v;
v = CD[w].pred;
}
}
// borrado de arcos que no forman parte de la solucion
for(i=1;i<=N;i++)
for(j=1;j<=N;j++)
if(G[i][j]!=NULL)
if(G[i][j]->es_sol==0){
delete G[i][j];
G[i][j] = NULL;
}
//calculo de distancias mas cortas con costos originales y con red solucion
for(k=1;k<=K;k++){
cargar_costo_k(G,N,CK,k);
Dijkstra(G,N,CK[k].orig,CD);
zt += CD[CK[k].dest].dist;
}
z = zt;
//añadido de costos fijos
for(i=1;i<=N;i++)
for(j=1;j<=N;j++)
if(G[i][j]!=NULL)
z += G[i][j]->c_fijo;
}

```

```

void ascenso_dual(grafo &G,int N,conj_comm &CK,int K,long &z,long &zt,long &zd)
{
int k;
long d1,d2,delta;
conj_dijk CD;
min_cost PI;
int fin=0;

//inicializacion
inicializar(G,N,CK,K,CD,PI);

//iteracion de ascenso dual
while(!fin){
k = elegir_commodity(CK,K);
d1 = delta1(G,N,CK,k);
d2 = delta2(G,N,CK,k);
delta = (d1<d2)?d1:d2;
actualizar_holguras(G,N,CK,k,delta);
if(delta==d1)
etiquetar_nodo(G,N,CK,k);
CK[k].es_cand = 0;
if(vacio_Ku(CK,K)){
actualizar_Ku(CK,K);
if(vacio_Ku(CK,K))
fin = 1;
}
}
//armado de la solucion primal a partir del resultado de la iteracion dual
armar_solucion(G,N,CK,K,CD,z,zt,zd);
}

main(int argc, char *argv[])
{
grafo G;
conj_comm CK;
int N,K;
char *arch_datos, *arch_sal;
long z,zt,zd;

if(--argc != 2){
cerr << "Error en los parametros de entrada al programa \n";
exit(1);
}
arch_datos = *++argv;
arch_sal = *++argv;
cargar_datos(arch_datos,G,CK,N,K);
ascenso_dual(G,N,CK,K,z,zt,zd);
listar_solucion(arch_sal,G,N,z,zt,zd);
return 0;
}

```

### **E.3 Archivos de Entrada y Salida de la Heurística en los Casos de Prueba**

#### **Archivo de salida de la heurística de ascenso dual aplicada al Caso 1 del SUNDP**

el costo total de la solución de ascenso dual es 42450

el costo de transporte de la solución es 2450

el costo fijo de la solución es 40000

la cota inferior solución del dual vale 30820

el conjunto de arcos solución es:

1 --> 3

2 --> 1

2 --> 4

3 --> 6

4 --> 7

5 --> 6

6 --> 2

6 --> 7

7 --> 1

7 --> 5

#### **Archivo de salida de la heurística de ascenso dual aplicada al Caso 1b del SUNDP**

el costo total de la solución de ascenso dual es 6450

el costo de transporte de la solución es 2450

el costo fijo de la solución es 4000

la cota inferior solución del dual vale 5660

el conjunto de arcos solución es:

1 --> 3

2 --> 1

2 --> 4

3 --> 6

4 --> 7

5 --> 6

6 --> 2

6 --> 7

7 --> 1

7 --> 5

#### **Archivo de salida de la heurística de ascenso dual aplicada al Caso 2 del SUNDP**

el costo total de la solución de ascenso dual es 448040

el costo de transporte de la solución es 324460

el costo fijo de la solución es 123580

la cota inferior solución del dual vale 430317

el conjunto de arcos solución es:

1 --> 5

1 --> 7

2 --> 3

2 --> 5

2 --> 6

2 --> 9

3 --> 6

3 --> 8

4 --> 2

4 --> 3

4 --> 7  
4 --> 9  
5 --> 4  
5 --> 7  
5 --> 9  
6 --> 3  
6 --> 4  
7 --> 1  
7 --> 4  
7 --> 6  
8 --> 4  
9 --> 1  
9 --> 5

**Archivo de salida de la heurística de ascenso dual aplicada al Caso 3 del SUNDP**

el costo total de la solución de ascenso dual es 511327

el costo de transporte de la solución es 388162

el costo fijo de la solución es 123165

la cota inferior solución del dual vale 500216

el conjunto de arcos solución es:

1 --> 3  
1 --> 5  
1 --> 10  
1 --> 14  
2 --> 1  
3 --> 1  
3 --> 8  
4 --> 2  
4 --> 5  
5 --> 4  
5 --> 13  
8 --> 13  
10 --> 3  
10 --> 12  
11 --> 2  
11 --> 4  
11 --> 13  
12 --> 2  
12 --> 14  
13 --> 1  
13 --> 3  
13 --> 11  
14 --> 1  
14 --> 4  
14 --> 12



**Archivo de entrada de la heurística de ascenso dual aplicada al Caso 1 del SUNDP**

7 42 42	10 2 5 10
1 2 18 4000	11 2 6 10
1 3 3 4000	12 2 7 10
1 4 8 4000	13 3 1 10
1 5 10 4000	14 3 2 10
1 6 17 4000	15 3 4 10
1 7 16 4000	16 3 5 10
2 1 1 4000	17 3 6 10
2 3 14 4000	18 3 7 10
2 4 1 4000	19 4 1 10
2 5 9 4000	20 4 2 10
2 6 17 4000	21 4 3 10
2 7 9 4000	22 4 5 10
3 1 11 4000	23 4 6 10
3 2 13 4000	24 4 7 10
3 4 6 4000	25 5 1 10
3 5 14 4000	26 5 2 10
3 6 5 4000	27 5 3 10
3 7 7 4000	28 5 4 10
4 1 11 4000	29 5 6 10
4 2 8 4000	30 5 7 10
4 3 8 4000	31 6 1 10
4 5 14 4000	32 6 2 10
4 6 17 4000	33 6 3 10
4 7 1 4000	34 6 4 10
5 1 13 4000	35 6 5 10
5 2 19 4000	36 6 7 10
5 3 16 4000	37 7 1 10
5 4 7 4000	38 7 2 10
5 6 4 4000	39 7 3 10
5 7 19 4000	40 7 4 10
6 1 19 4000	41 7 5 10
6 2 2 4000	42 7 6 10
6 3 2 4000	
6 4 7 4000	
6 5 12 4000	
6 7 0 4000	
7 1 4 4000	
7 2 14 4000	
7 3 14 4000	
7 4 5 4000	
7 5 3 4000	
7 6 11 4000	
1 1 2 10	
2 1 3 10	
3 1 4 10	
4 1 5 10	
5 1 6 10	
6 1 7 10	
7 2 1 10	
8 2 3 10	
9 2 4 10	

**Archivo de entrada de la heurística de ascenso dual aplicada al Caso 2 del SUNDP**

9 35 72	14 2 7 25	63 8 7 37
1 2 84 8448	15 2 8 67	64 8 9 12
1 5 22 2318	16 2 9 43	65 9 1 25
1 7 48 4915	17 3 1 71	66 9 2 6
1 9 84 8574	18 3 2 14	67 9 3 14
2 3 41 4132	19 3 4 95	68 9 4 82
2 5 68 6868	20 3 5 86	69 9 5 13
2 6 71 7159	21 3 6 101	70 9 6 73
2 9 73 7415	22 3 7 16	71 9 7 81
3 2 87 8783	23 3 8 64	72 9 8 37
3 6 69 6940	24 3 9 87	
3 8 52 5284	25 4 1 11	
4 1 77 7791	26 4 2 44	
4 2 50 5077	27 4 3 32	
4 3 44 4421	28 4 5 80	
4 7 56 5666	29 4 6 61	
4 8 82 8273	30 4 7 76	
4 9 60 6146	31 4 8 22	
5 2 70 7047	32 4 9 6	
5 3 87 8741	33 5 1 13	
5 4 62 6342	34 5 2 8	
5 7 70 7095	35 5 3 13	
5 9 21 2184	36 5 4 10	
6 3 36 3704	37 5 6 6	
6 4 52 5357	38 5 7 90	
6 8 43 4396	39 5 8 90	
7 1 67 6826	40 5 9 92	
7 4 50 5135	41 6 1 12	
7 5 78 7930	42 6 2 58	
7 6 76 7656	43 6 3 100	
8 2 83 8457	44 6 4 22	
8 4 57 5805	45 6 5 28	
8 6 51 5270	46 6 7 12	
8 7 81 8135	47 6 8 16	
9 1 51 5191	48 6 9 12	
9 5 18 1944	49 7 1 34	
1 1 2 17	50 7 2 77	
2 1 3 44	51 7 3 100	
3 1 4 45	52 7 4 96	
4 1 5 46	53 7 5 88	
5 1 6 49	54 7 6 42	
6 1 7 93	55 7 8 26	
7 1 8 4	56 7 9 66	
8 1 9 91	57 8 1 3	
9 2 1 100	58 8 2 89	
10 2 3 77	59 8 3 8	
11 2 4 79	60 8 4 24	
12 2 5 85	61 8 5 5	
13 2 6 68	62 8 6 17	

**Archivo de entrada de la heurística de ascenso dual aplicada al Caso 3 del SUNDP**

14 56 90	13 3 38 3903	44 5 13 49
1 3 35 3679	13 8 16 1675	45 5 14 51
1 5 63 6454	13 11 37 3830	46 10 1 11
1 10 35 3591	14 1 19 2038	47 10 2 45
1 14 33 3398	14 4 52 5335	48 10 3 63
2 1 48 4893	14 10 83 8395	49 10 4 33
2 4 72 7317	14 12 42 4318	50 10 5 8
2 7 44 4472	1 1 2 99	51 10 11 10
2 8 84 8477	2 1 3 102	52 10 12 76
3 1 53 5482	3 1 4 1	53 10 13 72
3 2 89 8980	4 1 5 68	54 10 14 37
3 8 47 4781	5 1 10 83	55 11 1 69
3 14 58 5955	6 1 11 50	56 11 2 12
4 2 52 5345	7 1 12 10	57 11 3 24
4 5 79 8011	8 1 13 34	58 11 4 37
4 7 91 9278	9 1 14 68	59 11 5 44
4 11 94 9484	10 2 1 51	60 11 10 8
5 4 31 3227	11 2 3 63	61 11 12 7
5 6 84 8557	12 2 4 16	62 11 13 44
5 9 64 6535	13 2 5 18	63 11 14 68
5 13 43 4432	14 2 10 83	64 12 1 24
6 1 87 8752	15 2 11 64	65 12 2 55
6 4 98 9844	16 2 12 87	66 12 3 18
6 9 68 6943	17 2 13 27	67 12 4 28
6 10 12 1331	18 2 14 2	68 12 5 42
7 2 81 8163	19 3 1 19	69 12 10 28
7 10 36 3726	20 3 2 21	70 12 11 13
7 12 53 5340	21 3 4 56	71 12 13 6
7 14 95 9584	22 3 5 41	72 12 14 12
8 5 70 7127	23 3 10 43	73 13 1 55
8 6 86 8730	24 3 11 22	74 13 2 52
8 11 60 6118	25 3 12 53	75 13 3 24
8 13 16 1690	26 3 13 51	76 13 4 7
9 6 95 9580	27 3 14 46	77 13 5 21
9 7 48 4849	28 4 1 93	78 13 10 56
9 10 66 6638	29 4 2 74	79 13 11 35
9 12 78 7852	30 4 3 38	80 13 12 43
10 3 53 5428	31 4 5 90	81 13 14 53
10 6 67 6841	32 4 10 75	82 14 1 1
10 9 54 5536	33 4 11 1	83 14 2 81
10 12 84 8496	34 4 12 59	84 14 3 16
11 2 42 4377	35 4 13 17	85 14 4 12
11 3 66 6649	36 4 14 10	86 14 5 1
11 4 36 3745	37 5 1 93	87 14 10 48
11 13 38 3875	38 5 2 86	88 14 11 56
12 2 84 8518	39 5 3 62	89 14 12 8
12 5 88 8877	40 5 4 8	90 14 13 55
12 7 42 4368	41 5 10 90	
12 14 80 8057	42 5 11 70	
13 1 61 6262	43 5 12 92	