INGENIERIA INDUSTRIAL
UNIVERSIDAD DE CHILE

Profesores: Richard Weber, Rodrigo Wolf

Auxiliares: Victor Bucarey, André Carboni, Nelson Devia, Diego Vergara

IN3701 – Modelamiento y Optimización
Un par de PPLs más para la colección :)
26 de Noviembre, 2010

Problema 1

Debido a los destrozos dejados por los terremotos y a la alta necesidad de materiales de construcción generada, la afamada empresa de concreto "Cementos Sandía" ha tenido ciertos problemas con la distribución de dichos materiales. Usted, como encargado de logística de dicha empresa, debe decidir cómo distribuir el material.

Cementos Sandía posee K camiones, los cuales se almacenan en unas pequeñas bodegas, una para cada camión, desde donde cada uno parte a las 7 de la mañana (considere que cada una de estas pequeñas bodegas se encuentra en puntos distintos de la ciudad). Una vez finalizado el recorrido, todos los camiones deben llegar a la bodega central de la empresa.

Considere que existen J clientes que usted debe visitar, los cuales corresponden a locales de importantes mayoristas de construcción. Para cada cliente j, el tiempo que demora descargar el material el sj. Debido a restricciones de ingreso de camiones de alto volumen a ciertas calles y a políticas de los mayoristas, cada cliente posee un tiempo mínimo y un tiempo máximo de llegada (una "ventana de tiempo" a la que puede llegar) denotados por $tmin_j$ y $tmax_j$ (Por ejemplo, un cliente j podría exigir que el camión llegue entre las 14 y las 17 horas. En tal caso, $tmin_j$=14 y $tmax_j$=17). Suponga además que cada camión puede visitar a lo más 5 clientes en un día. Los camiones no pueden llegar a la Bodega Central después de las 21 hrs.

Considerando que el tiempo de viaje entre cada origen-destino es t_{ij} a un costo de c_{ij} y que cada cliente debe ser visitado sólo una vez, formule un modelo de programación lineal que le permita decidir las rutas de sus camiones, minimizando los costos de viaje para un día.

Bonus track

Imagine que es posible llegar tarde donde algunos clientes, pero que dicho atraso cuesta R por minuto, independiente del cliente. Considerando lo anterior, modifique el PPL de forma que pueda aceptar ventanas de tiempo "relajadas", pero asumiendo los costos asociados.

¿Sabía usted que esta es la última clase del auxiliar después de 9 semestres en el equipo docente de optimiza?

Problema 2

Es un conocido problema combinatorial el intentar ubicar la mayor cantidad de reinas en un tablero de ajedrez (8x8) de modo que ningún par de ellas se ataque entre sí. Decimos que 2 reinas se atacan entre sí, si están en la misma fila, en la misma columna o en la misma diagonal.

1. Diseñe un modelo de programación lineal entera que permita encontrar la mayor cantidad de reinas que pueden colocarse en el tablero.
2. ¿Cómo modificaría el problema si le piden lo mismo, pero para torres?
3. ¿Cómo modificaría el problema si le piden lo mismo, pero para alfiles?
4. Explique (sin formular explicitamente todo el modelo) cómo resolvería el problema si le piden lo mismo, pero para caballos.

Nota: Los alfiles se mueven en el tablero solamente en diagonal, cualquier número de casillas; las torres en horizontal, cualquier número de casillas; y los caballos se mueven 2 casillas para adelante/atrás y 1 para la derecha/izquierda, o bien 1 casilla para adelante/atrás y 2 para la derecha/izquierda.

Solución

Problema 1

<u>Variables de decisión</u>:

$$X_{ijk} = \begin{cases} 1 & \text{Si camión k va de i a j} \\ 0 & \text{sino} \end{cases}$$

W_{kj} = Tiempo (en minutos) en que llega al nodo j el camión k

Nota: *Es el tiempo después que atiende al nodo j, porque es más cómodo para las restricciones trabajarlo así. Notar que asumimos tiempo discreto (variable natural, en minutos), pero es igual de válido poner una variable real, en horas. De hecho, para efectos de un modelo "de verdad", es mejor trabajar el tiempo con variables reales, se resuelve más rápido como ya vieron en el capítulo de programación entera.*

<u>Restricciones</u>:

1. Naturaleza de las variables

$$X_{ijk} \in \{0,1\}, W_{kj} \in \mathrm{N}$$

2. Conservación de flujo:

$$\sum_{\substack{j \in J \cup K \cup \{BC\} \\ k \in K}} X_{ijk} - \sum_{\substack{j \in J \cup K \cup \{BC\} \\ k \in K}} X_{jik} = \begin{cases} 1 & \text{Si i pertenece a K} \\ 0 & \text{Si i pertenece a J} \\ -K & \text{Si i=\{BC\}} \end{cases}$$

Nota: *Fíjense que el flujo si i pertenece a K es 1 porque desde esos nodos solo salen camiones. Asimismo, en el nodo BC el flujo es –K porque a ese nodo solo llegan K camiones y no sale ninguno. Es –K porque estoy sumando sobre k, si no hubiera sumado sobre k entonces el flujo sería -1 porque la restricción quedaría para todo k (es decir es el flujo de 1 camión, al nodo BC el camión llega 1 vez). Como les comenté en la clase yo había puesto -1 cuando lo resolví pero en la clase por la forma que lo expliqué vimos que era –K, al final dependía de si ponían o no la suma sobre k, el efecto es el mismo para el modelo.*

3. Todo cliente debe ser atendido por exactamente 1 camión

$$\sum_{j \in J} X_{ijk} = 1 \qquad \forall i,k$$

Nota: *Esta es una de las restricciones que hace que la restricción 4 funcione (y no pase que el tiempo W_{kj} se vaya siempre a cero, como preguntó alguien en la clase). Esta restricción obliga a pasar por todos los nodos cliente, y por ende el modelo no puede hacer que X_{ijk} valga siempre cero. Como no habíamos escrito aún esta restricción surgió esa duda. Fijense también que esta restricción es necesaria, porque en la restricción 2 dijimos que el flujo en los nodos cliente era cero, pero*

eso puede significar que a un nodo llega y sale 1 arco, asi como podrían llegar 2 arcos y salir 2 arcos, etc. Como en el enunciado nos dicen que los camiones pasan 1 vez por cliente, hay que decirle que a los nodos en J entre (o salga) solo 1 arco.

4. Restricción de tiempos de llegada

$$W_{kj} \geq W_{ki} + t_{ij} + s_j - M(1 - X_{ijk}) \qquad \forall i, j, k; \qquad M >> 0$$

Nota: *El tiempo solo se actualiza si efectivamente el camión k va de i a j, sino el tiempo en ese nodo puede "valer lo que sea", por eso restamos una M grande para no restringirlo. Noten que como estamos minimizando los costos por tiempo de viaje, eso afecta directamente a W_{kj}, por lo que es muy probable que el programa trate de hacer que W_{kj} sea lo más chico posible, y por ende se cumpla la restricción con igualdad. Después de darle unas vueltas veo que aún así es posible que el modelo decida que no se cumpla la igualdad (i.e. defina un W_{kj} tal que es mayor que la expresión que la derecha), pero esto en verdad no nos afecta porque nosotros queremos decidir la ruta óptima, la hora de llegada a los nodos no nos importan tanto en la medida que se respeten las ventanas de tiempo (eso es lo único que nos interesa, llegar en el intervalo solicitado). Los tiempos de llegada a los nodos podrían calcularse "a mano" una vez teniendo la ruta óptima asi que no es tan grave.*

5. Restricción de Ventanas de Tiempo

$$t\min_j \leq W_{kj} \leq t\max_j \qquad \forall j, k$$

6. Camión k visita máximo 5 clientes:

$$\sum_{\substack{i \in K \cup J \\ j \in J}} X_{ijk} \leq 5 \qquad \forall\ k$$

7. Condición inicial y final:

$$W_{kk} = 7 \qquad W_{BC,k} \leq 21$$

<u>Función Objetivo</u>

$$\min\{\sum_{i,j,k} c_{ij} * x_{ijk}\}$$

Bonus

Basta sólo con transformar la función objetivo y la restricción de Ventanas de tiempo. El PPL queda

<u>Variable de decisión adicional:</u>

d_{kj} = Retraso (minutos) con que llego al nodo j con camión k

Restricción modificada:

Restricción de Ventanas de Tiempo

$$t\min_j \le W_{kj} \le t\max_j - d_{kj} \qquad \forall j,k$$

Función Objetivo:

$$\min\{\sum_{i,j,k}(c_{ij} * x_{ijk} + d_{kj} * R\}$$

Problema 2

- **Problema para reinas (3.5 ptos)**

Conjunto de casilleros $N = 1..,8$
$NxN = (i, j)$ casillero en la fila i, columna j del tablero

Variables de decisión:

$$X_{ij} = \begin{cases} 1 & \text{Si se pone reina en posición (i,j)} \\ 0 & \sim \end{cases}$$

Restricciones:

1. Una reina a lo más en cada fila

$$\sum_j X_{ij} \le 1 \quad \forall i \in N$$

2. Una reina a lo más en cada columna

$$\sum_i X_{ij} \le 1 \quad \forall j \in N$$

3. En diagonales a lo más una reina

diagonal continua:

$$\sum_{k=1}^{n-j} X_{k,k+j} \le 1 \quad \forall j \in 0,\ldots,n-1$$

diagonal cortada:

$$\sum_{k=1}^{n-j} X_{k+j,k} \le 1 \quad \forall j \in 0,\ldots,n-1$$

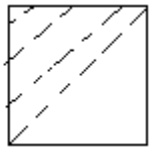
(perdonen lo rasca del dibujo). La idea es que la restricción representa las diagonales que se muestran al lado).

diagonal continua:

$$\sum_{k=1}^{n-j} X_{n+1-k,k+j} \leq 1 \quad \forall j \in 0,\ldots,n-1$$

diagonal cortada:

$$\sum_{k=1}^{n-j} X_{n+1-k-j,k} \leq 1 \quad \forall j \in 0,\ldots,n-1$$



4. Naturaleza de las variables

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

Función Objetivo:

$$Max \quad \sum_{i,j} X_{ij}$$

- **Problema para Torres (0.75 ptos.)**

$$X_{ij} = \begin{cases} 1 & \text{Si se pone torre en posición (i,j)} \\ 0 & \sim \end{cases}$$

1. Una torre a lo más en cada fila

$$\sum_{j} X_{ij} \leq 1 \quad \forall i \in N$$

2. Una torre a lo más en cada columna

$$\sum_{i} X_{ij} \leq 1 \quad \forall j \in N$$

3. Naturaleza de las variables

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

Función Objetivo:

$$Max \quad \sum_{i,j} X_{ij}$$

Problema para los alfiles:

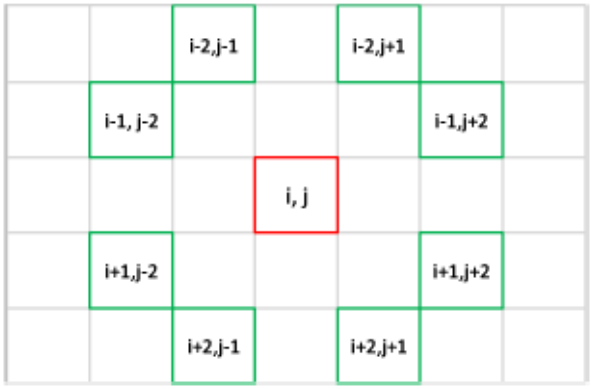
Es igual al de las 4 diagonales de la reina (misma variable, misma función objetivo y set de restricciones 3).

- **Problema para caballos (1 ptos)**

Para el movimiento de los caballos, basta con explicar los casos que deben considerarse.

En la siguiente figura se esquematiza para una posición (i, j) del tablero, los posibles caballos que podrian atacar. Sea la variable:

$$X_{ij} = \begin{cases} 1 & \text{Si se pone caballo en posición (i,j)} \\ 0 & \sim \end{cases}$$



Se deben considerar 8 restricccciones. Tal que si pongo un caballo en la posición (i, j), entonces de no debe haber otro caballo en las otras o casillas señaladas.

$X_{i-1,j+2} \leq 1 - X_{ij}$
$X_{i-2,j+1} \leq 1 - X_{ij}$
$X_{i+1,j+2} \leq 1 - X_{ij}$
$X_{i-2,j+1} \leq 1 - X_{ij}$
$X_{i-1,j-2} \leq 1 - X_{ij}$
$X_{i-2,j-1} \leq 1 - X_{ij}$
$X_{i+1,j-2} \leq 1 - X_{ij}$
$X_{i-2,j-1} \leq 1 - X_{ij}$

Se deben incluir las condiciones de borde para no salirse del tablero.

Para la corrección no es necesario que escriban las 8 restricciones, sólo que noten cuales son las combinaciones que deben considerar para cada casilla del tablero.

Dudas y/o comentarios a:
André Carboni
andre@carboni.cl