INGENIERIA INDUSTRIAL
UNIVERSIDAD DE CHILE

Profesora: Leonardo López, Ximena Schultz, Richard Weber

Auxiliares: Fernanda Bravo. André Carboni, Nelson Devia, Ignacio Escobar, Rodrigo Wolf

IN34A – Optimización
Auxiliar Extra Examen
20 de Noviembre de 2008

Problema 1

Considere que se cuenta con un camión a llenar de capacidad K, y que debemos escoger entre N diferentes ítems a colocar, cada uno tiene un beneficio unitario de c_i y un peso w_i, y una disponibilidad máxima de di (es decir, disponemos a lo más de di unidades del ítem i para colocar en nuestro camión).

Asuma que los ítems no pueden fragmentarse (es decir, sólo se admiten ítems completos en el camión), y que la capacidad no usada en el camión no reporta beneficios.

1) Formule el problema anterior como un problema de programación dinámica.

2) Usando la formulación anterior, resuelva la siguiente instancia: K = 100, N = 4, y los datos en la tabla 1.

Item	c_i	w_i	d_i
1	47	55	1
2	23	27	2
3	11	13	2
4	6	7	3

Cuadro 1: Beneficio, peso y disponibilidad para cada ítem.

Problema 2

1) ¿Conoce algún problema en P que esté también en NP? ¿Conoce algún problema en P que esté también en NP-Completo? Explique el significado en términos de teoría de la complejidad de sus 2 respuestas.

2) Explique qué es un algoritmo exacto y qué es una heurística para resolver un determinado problema. ¿En qué casos se debería usar cada uno? Relacione esta respuesta con lo estudiado sobre teoría de la complejidad computacional.

3) Muestre como se deduce cuál es el siguiente punto x^{k+1} de las iteraciones del Método de Newton para optimizar funciones no lineales. Explique conceptual y geométricamente el funcionamiento del mismo.

4) ¿Es cierto que todo óptimo de una función lineal está en un vértice del poliedro factible? ¿Es cierto que el algoritmo SIMPLEX es un algoritmo polinomial para programación lineal? ¿Es cierto que el problema de programación lineal es un problema polinomial? Justifique en cada caso.

5) Mencione al menos 2 algoritmos para programación lineal de complejidad polinomial. ¿Qué ventajas o desventajas tienen con respecto a SIMPLEX?

6) Explique en qué casos un problema de programación entera puede ser resuelto en forma exacta relajando la condición de integralidad de las variables.

7) Suponga que tiene un problema de programación entera puro (todas sus variables deben ser enteras) de minimización. ¿Qué puede afirmar si la solución de la relajación lineal es entera? Si la solución no es entera, ¿qué información le da el valor del óptimo de la relajación lineal?

8) Formule el problema de obtener un flujo máximo en una red con capacidades inferiores y superiores en cada arco, como un problema lineal.

9) Muestre en un ejemplo por qué Dijkstra podría funcionar mal si hay arcos de longitud negativa. Justifique.

10) Dado un problema de programación dinámica, explique el significado de las variables de decisión, de las variables exógenas (parámetros) y de las variables de estado.

Problema 3

La familia Simpson va a salir de vacaciones desde su ciudad natal Springfield. La familia desea visitar n ciudades y dispone de un total de M días para hacerlo, con $M \geq n$. La familia desea saber cuantos días permanecer en cada ciudad de modo de maximizar la satisfacción total de sus vacaciones sabiendo que para cada ciudad i existe una función de satisfacción g_i que es función del número de días de permanencia.

1. Plantee un modelo de programación dinámica para resolver la planificación de las vacaciones de los Simpsons.
2. Suponga que n = 3 y M = 5 y que las funciones de beneficio $g_k(x_k)$ vienen dadas por:

	$g_1(x_1)$	$g_2(x_2)$	$g_3(x_3)$
$x_k = 0$	0	0	0
$x_k = 1$	1	1	1
$x_k = 2$	2	4	3
$x_k = 3$	3	6	3
$x_k = 4$	4	8	2
$x_k = 5$	5	8	1

Problema 4

Las eliminatorias sudamericanas para Sudáfrica 2010, que comienzan dentro de algunos meses, son jugadas por 10 equipos. Los 4 primeros clasifican para el Mundial y el sistema es de todos contra todos con partido y revancha. La asignación de puntaje es de 3 puntos por partido ganado y 1 punto por empate. Ante igualdad en las posiciones se define la ubicación por diferencia de goles.

1) Diseñe un modelo de programación lineal entera que (asignando todos los resultados posibles) antes de realizada la primer fecha indique cuál es la máxima cantidad de puntos con que Chile puede no clasificar para Sudáfrica 2010.

2) Si una vez finalizada la primera fecha, Chile (equipo 1) le ganó su partido a la poderosa selección trasandina (equipo 2), ¿cómo se incluye este resultado en el modelo? ¿Y si fue un empate? ¿Y si fue derrota?

3) Una vez resuelto el modelo para una fecha determinada de la eliminatoria, ¿qué le contestaría al técnico Bielsa si éste le pregunta con cuántos puntos tiene Chile garantizada su clasificación?

Sugerencias: utilice al menos una familia de variables enteras que definan la cantidad de partidos que el equipo i le gana al equipo j, y una variable entera para cada equipo rival de Chile que valga 1 si el conjunto tiene igual o más puntos que Chile, y 0 en caso contrario.

IN34A – Optimización
Pauta Auxiliar Extra Examen
20 de Noviembre de 2008

Problema 1

1)

a) Etapas: Cada uno de los ítems, i = 1, 2, ...,N.

b) Variables de Decisión:

Xi: Cantidad de unidades del ítem i que se carga al camión.

c) Variables de Estado:

Si: Capacidad disponible en el camión al inicio de la etapa i.

d) Recurrencia de Estados:

$$S_{i+1} = S_i - w_i X_i$$

e) Función de Beneficios:

$$V_i(S_i, X_i) = c_i X_i + V^*_{i+1}(S_{i+1})$$

Con:

$$V_i^*(S_i) = \max_{0 \le X_i \le \min\{d_i, \frac{S_i}{w_i}\}} \{V_i(S_i, X_i)\}$$

ó bien:

$$V_i^*(S_i) = \max_{\substack{0 \le X_i \le d_i \\ 0 \le X_i \le \frac{S_i}{w_i}}} \{V_i(S_i, X_i)\}$$

f) Condiciones de Borde:

$$S_1 = K$$

$$V_{N+1} = 0$$

2)

Etapa 4						
S4\X4	0	1	2	3	X*4	V*4
100	0	6	12	18	3	18
87	0	6	12	18	3	18
74	0	6	12	18	3	18
73	0	6	12	18	3	18
60	0	6	12	18	3	18
47	0	6	12	18	3	18
46	0	6	12	18	3	18
45	0	6	12	18	3	18
33	0	6	12	18	3	18
32	0	6	12	18	3	18
20	0	6	12	-	2	12
19	0	6	12	-	2	12
18	0	6	12	-	2	12
5	0	-	-	-	0	0

Etapa 3					
S3\X3	0	1	2	X*3	V*3
100	18	29	40	2	40
73	18	29	40	2	40
46	18	29	34	2	34
45	18	29	34	2	34
18	12	11	-	0	12

Etapa 2					
S2\X2	0	1	2	X*2	V*2
100	40	63	80	2	80
45	34	35	-	1	35

Etapa 1				
S1\X1	0	1	X*1	V*1
100	80	82	1	82

Politica Optima	
X*1	1
X*2	1
X*3	0
X*4	2
V*1	82
Cap Usada	96

Problema 2

1)

¿Conoce algún problema en P que esté también en NP?

Como $P \subseteq NP$, todo problema en P está en NP.

¿Conoce algún problema en P que esté también en NP-Completo?

No. Si encontráramos un problema en P que está también en NP-completo entonces todos los problemas pertenecientes a este conjunto podrían ser resueltos con complejidad polinomial. Es un problema abierto saber si existe algún algoritmo polinomial para cualquier problema de NP-completo.

2)

Un **algoritmo** es una secuencia finita de pasos que garantiza el encontrar una solución de un problema para una instancia dada. La complejidad de un algoritmo varía, pero puede llegar a ser muy complejo e impracticable de resolver computacionalmente. Es por ello que para resolver problemas muy complejos es mejor utilizar una **heurística**, que si bien no asegura llegar al óptimo, obtiene buenas soluciones en un tiempo razonable. En general en problemas polinomiales se suele usar algoritmos exactos y en problemas NP-completos (sobre todo de gran dimensión) se suelen usar heurísticas.

3)
El método de Newton se basa en aproximar la función f por una función cuadrática (en cada punto x^k se aproxima por la expansión de Taylor de orden 2) y esta aproximación se minimiza exactamente, generando un nuevo punto x^{k+1}.

$$q(x) = f(x^k) + \nabla f(x^k)^T (x - x^k) + \frac{1}{2}(x - x^k)^T H f(x^k)(x - x^k)$$

$$\nabla q(x) = \nabla f(x^k) + H f(x^k)(x - x^k) = 0$$

Luego, despejando x obtenemos la solución de este problema, la que determina el punto x^{k+1}.
Entonces:

$$x^{k+1} = x^k - [H f(x^k)]^{-1} \nabla f(x^k)$$

4)
En un problema lineal la solución es o bien un vértice del poliedro o una de sus aristas. El algoritmo SIMPLEX puede resultar ser exponencial si es que el número de vértices es exponencial y los recorre todos. Es cierto que el problema de programación lineal es un problema polinomial pues siempre se puede encontrar un algoritmo que lo resuelva en tiempo polinomial (Khachyan y Karmarkar son dos de ellos).

5)
- El de **Khachiyan** (1979) conocido como Método de las Elipsoides, de complejidad $O(Ln^5)$ donde L es el número de bits necesarios para almacenar los datos del problema (asumiendo que $n >> m$). En el sentido del análisis del peor caso el Método de las Elipsoides es teóricamente mejor que SIMPLEX. En la práctica, en la gran mayoría de los casos, SIMPLEX funciona mucho mejor.
- El de **Karmarkar** (1984) conocido como un algoritmo de punto interior. La característica central de este algoritmo es que se mueve por el interior de la región factible (y no por los vértices, como SIMPLEX). Este algoritmo además funciona mejor que SIMPLEX en la práctica (Complejidad Teórica $O(Ln^{3\text{-}5})$). Nesterov y Nemirovskii (1994) desarrollan otro algoritmo de punto interior de mejor complejidad teórica, $O(Lm^3)$. Si bien estos pueden ser mejores que SIMPLEX, especialmente para problemas de gran tamaño, la forma en que son implementados computacionalmente resulta un factor crítico y una desventaja con respecto a SIMPLEX.

6)
En el caso de que todos los vértices del poliedro factible sean enteros. Otra posibilidad es relajar, aplicar Simplex y si el óptimo da entero quiere decir que se podía relajar para obtener el óptimo del problema entero.

7)
Si la solución óptima de la relajación lineal es entera, esta también es la solución óptima del problema entero original. Si no es entera, como el problema es de minimización, el valor óptimo de la función objetivo de la relajación lineal es una cota inferior del valor óptimo de la función objetivo del problema entero original $z^*_{RL} \leq z^*_{PE}$.
Esto pues el conjunto de soluciones factibles del problema entero es un subconjunto del conjunto de soluciones factibles del problema relajado.

8)
Consideremos el grafo G = [N;A]. En cada arco $(i,j) \in A$ se tiene una capacidad mínima l_{ij} y una máxima u_{ij} de flujo que puede pasar por este. Se desea llevar la máxima cantidad de flujo desde el nodo inicial $s \in N$ hasta el nodo final $t \in N$, el que se decide a través de la variable $F \geq 0$. Luego el modelo lineal queda:

$$\text{máx} \quad F \tag{6}$$

$$s.a. \quad \sum_{\{j/(s,j)\in A\}} f_{sj} - \sum_{\{k/(k,s)\in A\}} f_{ks} = F \tag{7}$$

$$\sum_{\{j/(i,j)\in A\}} f_{ij} - \sum_{\{k/(k,i)\in A\}} f_{ki} = 0 \qquad \forall i \in N \setminus \{s,t\} \tag{8}$$

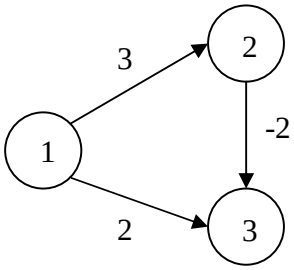
$$\sum_{\{j/(t,j)\in A\}} f_{tj} - \sum_{\{k/(k,t)\in A\}} f_{kt} = -F \tag{9}$$

$$l_{ij} \leq f_{ij} \leq u_{ij} \qquad \forall (i,j) \in A \tag{10}$$

$$F \geq 0 \tag{11}$$

Es válido suponer que no existen arcos que llegan a $s \in N$, ni arcos que salen de $t \in N$. Bajo este supuesto se tendría que $\sum_{\{k/(k,s)\in A\}} f_{ks} = 0$ y $\sum_{\{j/(t,j)\in A\}} f_{tj} = 0$ en las restricciones (7) y (9) respectivamente.

9)
Si hay arcos de longitud negativa, el algoritmo podría no encontrar la ruta más corta. Por ejemplo, en una red de 3 nodos, donde hay un camino de 1 al 3 de longitud 2, de 1 al 2 de longitud 3 y otro de 2 al 3 de longitud -2, Dijkstra no encuentra la ruta mas corta de 1 a 3. Encuentra el camino de longitud 2 y no el de longitud 1.



10)
Las **variables de decisión** son decisiones cuantificables, cuyos valores se intenta determinar por medio de la resolución del modelo. Su valor determina el valor de las variables de estado de las etapas futuras. Las **variables de estado** son variables que caracterizan la situación en la que se encuentra el sistema en una etapa dada. Estas variables dan la independencia a la etapa actual de las etapas anteriores, por lo que deben existir tantas variables de estado como las que permitan establecer en que condiciones comienza (o finaliza) una etapa para su posterior optimización. Las **variables exógenas o parámetros** representan los valores conocidos del sistema y se diferencian de las variables de decisión en que no son controlables.

Problema 3
1) Consideremos que la familia ya definió cual será el orden en que visitará las ciudades (si efectivamente decide visitarlas). En dicho caso, una etapa estará en relación uníivoca con una ciudad (Cada ciudad es una etapa y se pasará a la siguiente etapa cuando se pase a la siguiente ciudad). Además, el estado vendrá dado por el número de días que le restan a la familia para completar el total de días disponibles.

Así, podemos definir: x_i = Número de días en la ciudad i (variable de decisión de la etapa i). y_i = Número de días sobrantes después de visitar la ciudad i − 1 ó justo antes de visitar la ciudad i (variable de estado).
Analicemos las ecuaciones recursivas:

Condición de borde (ultima etapa):

En este caso habremos visitado las ciudades 1, 2, . . . , n−1 y tendremos y_n días disponibles para usar (dependiendo de cuantos días hayamos decidido quedarnos en las ciudades anteriores, y_n puede tener varios valores posibles. Luego, el problema a resolver viene dado por:

$$\begin{aligned} f_n(y_n) &= \text{máx } g_n(x_n) \\ & \quad s.a \quad 0 \leq x_n \leq y_n \\ & \quad x_n \text{ entero} \\ &= \text{valor de la política óptima de estadía en la ciudad n si la familia} \\ & \quad \text{ya ha visitado n-1 ciudades y aun dispone de } y_n \text{ días disponibles} \end{aligned}$$

Recursión genérica k:

En este caso habremos visitados las ciudades 1, 2, . . . , k−1 y nos quedan por visitar las ciudades k, k+1, . . . , n siendo y_k el numero de días que aun nos quedan disponibles. Luego nuestro problema a resolver será encontrar el número de días a permanecer en la ciudad k, de modo de maximizar el beneficio de actual más el beneficio de visitar las próximas ciudades suponiendo que de ahora en adelante tomaremos las decisiones óptimas dado los días que nos quedarán luego de tomar nuestra decisión hoy:

$$\begin{aligned} f_k(y_k) &= \text{máx}\{g_k(x_k) + f_{k+1}(y_k - x_k)\} \\ & \quad s.a \quad 0 \leq x_k \leq y_k \\ & \quad x_n \text{ entero} \\ &- \text{satisfacción total óptima por visitar a las ciudades } k, k+1, \ldots, n. \end{aligned}$$

Finalmente, el óptimo de la satisfacción de las vacaciones de la familia Simpson viene dado por

$$f = f_1(M)$$

2) Debemos resolver los problemas asociados a cada etapa partiendo desde la última, para cada uno de los posibles estados en que puede llegar el problema a la etapa en cuestión. Para resolver cada uno de estos problemas haremos una enumeración explícita de los casos posibles para cada etapa y seleccionaremos la mejor.

- Ciudad 3:

y_3	$g_3(0)$	$g_3(1)$	$g_3(2)$	$g_3(3)$	$g_3(4)$	$g_3(5)$	$f_3(y_3)$	x_3^*
0	0	i	i	i	i	i	0	0
1	0	1	i	i	i	i	1	1
2	0	1	3	i	i	i	3	2
3	0	1	3	3	i	i	3	2,3
4	0	1	3	3	2	i	3	2,3
5	0	1	3	3	2	1	3	2,3

i: infactible, es decir, la familia no puede quedarse esa cantidad de días porque ya no le quedan tantos.

- Ciudad 2:

y_2	$x_2=0$	$x_2=1$	$x_2=2$	$x_2=3$	$x_2=4$	$x_2=5$	$f_2(y_2)$	x_2^*
	$g_2(x_2)+f_3(y_2-x_2)$							
0	0+0	i	i	i	i	i	0	0
1	0+1	1+0	i	i	i	i	**1**	0,1
2	0+3	1+1	4+0	i	i	i	4	2
3	0+3	1+3	4+1	6+0	i	i	6	4
4	0+3	1+3	4+3	6+1	8+0	i	8	4
5	0+3	1+3	4+3	6+3	8+1	8+0	9	4,**3**

i: infactible, es decir, la familia no puede quedarse esa cantidad de días porque ya no le quedan tantos.

- Ciudad 1:

En este caso, como sabemos que inicialmente (antes de visitar la primera ciudad), la familia dispone de 5 días para sus vacaciones, solo analizamos el caso $y_1 = 5$

y_1	$x_1=0$	$x_1=1$	$x_1=2$	$x_1=3$	$x_1=4$	$x_1=5$	$f_1(y_1)$	x_1^*
	$g_1(x_1)+f_2(y_1-x_1)$							
5	0+9	1+8	2+6	3+4	4+1	5+0	9	0,1

Finalmente, existen 3 soluciones óptimas, todas con beneficio óptimo =9, cuyas estadías en cada ciudad viene dadas por:

Solución	Ciudad 1	Ciudad 2	Ciudad 3
Itinerario 1	0	3	2
Itinerario 2	0	4	1
Itinerario 3	1	4	0

Problema 4

1. (4 ptos.)

Variables de decisión

x_{ij} : cantidad de partidos que el equipo i le gana al equipo j.
y_i : 1 si el equipo i tiene los mismos o más puntos que la roja.
P_i : cantidad de puntos que tiene el equipo i.

Restricciones

Llamemos a Chile $i = 1$ y a Argentina $i = 2$.

a) Naturaleza de las variables.

$$x_{ij} \in \{0, 1, 2\} \quad \forall i, j.$$
$$y_i \in \{0, 1\} \quad \forall i.$$
$$P_i \geq 0 \quad \forall i.$$

b) Combinación de resultados posibles.

$$x_{ij} + x_{ji} \leq 2 \quad \forall i, j \text{ tal que } i \neq j.$$

c) Calculo de los puntos de cada equipo.

$$P_i = 3 \sum_{j \neq i} x_{ij} + \sum_{j \neq i} (2 - x_{ij} - x_{ji}) \quad \forall i.$$

d) Condicion para quedar eliminado.

$$\sum_{i \neq 1} y_i \geq 4$$

e) Obligar a la variable y_i a tomar valor cero cuando corresponda.

$$P_i \geq P_1 - M(1 - y_i) \quad \forall i$$

f) Obligar a la variable y_i a tomar valor uno cuando corresponda.

$$P_i + 1 \leq P_1 + M y_i \quad \forall i$$

Función Objetivo

$$\text{máx} \quad Z = P_1$$

2. (1 pto.) Para ello es necesario agregar como restricciones el resultado.

   a) **Triunfo:** $x_{12} \geq 1$.

   b) **Derrota:** $x_{21} \geq 1$.

   c) **Empate:** $x_{12} + x_{21} \leq 1$.

3. (1 pto.) Como se ha visto, el modelo entrega la mayor cantidad de puntos que puede obtener un equipo y aún así tener posibilidades de no clasificar para el mundial. Dado esto, al resultado obtenido (Z^*) se le debe agregar un punto más, con lo cual se obtiene la cantidad de puntos que asegura la clasificación.

Dudas y/o comentarios a:
Nelson Devia
ndevia@ing.uchile.cl