Investigación de Operaciones I

Tarea 3: Modelamiento, Resolución y Análisis de Problemas de Programación Lineal entera

Cristian Maureira
2673030-9
cmaureir@inf.utfsm.cl

Gabriel Zamora
2673070-8
gzamora@inf.utfsm.cl

12 de noviembre de 2009

1. Desarrollo

Se tienen 4 salas, 5 bloques horarios, 9 asignaturas, dos profesores y 3 cursos. Se quiere obtener el horario más compacto posible utilizando programación lineal entera. La descripción de los ramos asignados a cada profesor y curso son los siguientes:

- El conjunto de asignaturas es $A = \{a_1, a_2, \ldots, a_9\}$
- El conjunto de asignaturas para el profesor 1 es $\Omega_1 = \{a_1, a_2, a_8, a_9\}$
- El conjunto de asignaturas para el profesor 2 es $\Omega_2 = \{a_3, a_4, a_5, a_6, a_7\}$
- El conjunto de asignaturas para el curso 1 es $\Delta_1 = \{a_1, a_2, a_3\}$
- El conjunto de asignaturas para el curso 2 es $\Delta_2 = \{a_4, a_5, a_6\}$
- El conjunto de asignaturas para el curso 3 es $\Delta_3 = \{a_7, a_8, a_9\}$
- Lógicamente en una sala no pueden haber dos cursos al mismo tiempo.

Defina claramente:

- Variables
  - X_{ijk} : Si se dicta la asignatura i en el bloque j, en la sala k.
    Con $i = \{1, \ldots, 9\}$, $j = \{1, \ldots, 5\}$ y $k = \{1, \ldots, 4\}$. $X_{ijk} = 1$ Si se dicta.
    $X_{ijk} = 0$ No se dicta.
- Restricciones
  - Cada profesor imparte todas sus asignaturas:
    - $\sum_{j=1}^{5} \sum_{k=1}^{4} X_{1jk} + X_{2jk} + X_{8jk} + X_{9jk} = 4$
    - $\sum_{j=1}^{5} \sum_{k=1}^{4} X_{3jk} + X_{4jk} + X_{5jk} + X_{6jk} + X_{7jk} = 5$
  - Cada profesor imparte como mucho una asignatura cada hora:
    - $\sum_{k=1}^{4} X_{1jk} + X_{2jk} + X_{8jk} + X_{9jk} \leq 1, \forall j$
    - $\sum_{k=1}^{4} X_{3jk} + X_{4jk} + X_{5jk} + X_{6jk} + X_{7jk} \leq 1, \forall j$

- Cada asignatura se imparte una sola vez:
    - $\sum_{j=1}^{5}\sum_{k=1}^{4} X_{ijk} = 1, \forall i$
- En cada bloque y sala se imparte como mucho una sola asignatura:
    - $\sum_{i=1}^{9} X_{ijk} \leq 1, \forall j, \forall k$
- En cada bloque, se enseña como mucho una asignatura de cada curso:
    - $\sum_{k=1}^{4} X_{1jk} + X_{2jk} + X_{3jk} \leq 1, \forall j$
    - $\sum_{k=1}^{4} X_{4jk} + X_{5jk} + X_{6jk} \leq 1, \forall j$
    - $\sum_{k=1}^{4} X_{7jk} + X_{8jk} + X_{9jk} \leq 1, \forall j$
- Naturaleza
    - $X_{ijk} \in \{0, 1\}, \forall i, \forall j, \forall k$

- Función objetivo
  Se pondera cada término de la función objetivo, por su posición al bloque 1 y sala 1 (factor j+k), para así compactar el horario a pocos bloques y pocas salas, además se da una prioridad a las asignaturas según su índice (factor i)
    - Min $z = \sum_{i=1}^{9}\sum_{j=1}^{5}\sum_{k=1}^{4} i(j+k)X_{ijk}$

2. Desarrollo en Lindo

- Resuelva el modelo desarrollado en el punto anterior utilizando el software Lindo. Si el problema no tuviera solución, haga las modificaciones necesarias para que si tenga una solución. De ser así adjunte los cambios y explique claramente que hizo para que funcionara.

Solución

Var	Valor	Var	Valor	Var	Valor	Var	Valor	Var	Valor	Var	Valor
X111	0	X112	0	X113	0	X114	0	X121	0	X122	0
X123	0	X124	0	X131	0	X132	0	X133	0	X134	0
X141	0	X142	1	X143	0	X144	0	X151	0	X152	0
X153	0	X154	0	X211	0	X212	0	X213	0	X214	0
X221	0	X222	0	X223	0	X224	0	X231	0	X232	1
X233	0	X234	0	X241	0	X242	0	X243	0	X244	0
X251	0	X252	0	X253	0	X254	0	X311	0	X312	0
X313	0	X314	0	X321	0	X322	0	X323	0	X324	0
X331	0	X332	0	X333	0	X334	0	X341	0	X342	0
X343	0	X344	0	X351	1	X352	0	X353	0	X354	0
X411	0	X412	0	X413	0	X414	0	X421	0	X422	0
X423	0	X424	0	X431	0	X432	0	X433	0	X434	0
X441	1	X442	0	X443	0	X444	0	X451	0	X452	0
X453	0	X454	0	X511	0	X512	0	X513	0	X514	0
X521	0	X522	1	X523	0	X524	0	X531	0	X532	0
X533	0	X534	0	X541	0	X542	0	X543	0	X544	0
X551	0	X552	0	X553	0	X554	0	X611	0	X612	1
X613	0	X614	0	X621	0	X622	0	X623	0	X624	0
X631	0	X632	0	X633	0	X634	0	X641	0	X642	0
X643	0	X644	0	X651	0	X652	0	X653	0	X654	0
X711	0	X712	0	X713	0	X714	0	X721	0	X722	0
X723	0	X724	0	X731	1	X732	0	X733	0	X734	0
X741	0	X742	0	X743	0	X744	0	X751	0	X752	0
X753	0	X754	0	X811	0	X812	0	X813	0	X814	0
X821	1	X822	0	X823	0	X824	0	X831	0	X832	0
X833	0	X834	0	X841	0	X842	0	X843	0	X844	0
X851	0	X852	0	X853	0	X854	0	X911	1	X912	0
X913	0	X914	0	X921	0	X922	0	X923	0	X924	0
X931	0	X932	0	X933	0	X934	0	X941	0	X942	0
X943	0	X944	0	X951	0	X952	0	X953	0	X954	0

Visualización gráfica

Bloque / Sala	1	2	3	4
1	a_9	a_6		
2	a_8	a_5		
3	a_7	a_2		
4	a_4	a_1		
5	a_3			

- Haga una modificación a la función haga una modificación a la función objetivo y explique claramente que representa y si tiene solución. Por ejemplo: agregue un costo de ”flojera”. los primeros bloques o que una sala no pueda tener clases con el mismo curso en dos bloques consecutivos.

Modificación

Se hizo un pequeño cambio en los coeficientes de la función objetivo, omitiendo los términos donde se encontraran asignaturas en un bloque i, en una sala i. Asumiendo que en ese bloque horario, la iluminación de dicha sala es mala.

Var	Valor	Var	Valor	Var	Valor	Var	Valor	Var	Valor	Var	Valor
X111	0	X112	0	X113	0	X114	0	X121	0	X122	0
X123	0	X124	0	X131	0	X132	1	X133	0	X134	0
X141	0	X142	0	X143	0	X144	0	X151	0	X152	0
X153	0	X154	0	X211	0	X212	0	X213	0	X214	0
X221	0	X222	0	X223	0	X224	1	X231	0	X232	0
X233	0	X234	0	X241	0	X242	0	X243	0	X244	0
X251	0	X252	0	X253	0	X254	0	X311	0	X312	0
X313	0	X314	0	X321	0	X322	0	X323	0	X324	0
X331	0	X332	0	X333	0	X334	0	X341	1	X342	0
X343	0	X344	0	X351	0	X352	0	X353	0	X354	0
X411	0	X412	0	X413	0	X414	0	X421	0	X422	0
X423	0	X424	0	X431	0	X432	0	X433	0	X434	0
X441	0	X442	0	X443	0	X444	0	X451	1	X452	0
X453	0	X454	0	X511	0	X512	0	X513	0	X514	0
X521	1	X522	0	X523	0	X524	0	X531	0	X532	0
X533	0	X534	0	X541	0	X542	0	X543	0	X544	0
X551	0	X552	0	X553	0	X554	0	X611	0	X612	0
X613	1	X614	0	X621	0	X622	0	X623	0	X624	0
X631	0	X632	0	X633	0	X634	0	X641	0	X642	0
X643	0	X644	0	X651	0	X652	0	X653	0	X654	0
X711	0	X712	0	X713	0	X714	0	X721	0	X722	0
X723	0	X724	0	X731	1	X732	0	X733	0	X734	0
X741	0	X742	0	X743	0	X744	0	X751	0	X752	0
X753	0	X754	0	X811	0	X812	0	X813	0	X814	0
X821	0	X822	0	X823	0	X824	0	X831	0	X832	0
X833	0	X834	0	X841	0	X842	0	X843	0	X844	0
X851	0	X852	1	X853	0	X854	0	X911	0	X912	1
X913	0	X914	0	X921	0	X922	0	X923	0	X924	0
X931	0	X932	0	X933	0	X934	0	X941	0	X942	0
X943	0	X944	0	X951	0	X952	0	X953	0	X954	0

Visualización gráfica

Bloque / Sala	1	2	3	4
1		a_9	a_6	
2	a_5			a_2
3	a_7	a_1		
4	a_3			
5	a_4	a_8		

3. Preguntas

1. Investigue y explique brevemente en que consiste el problema de la mochila. Busque y comente algún caso especial.
   R:
   El problema de la mochila

   - Definición:
     Es un problema típico de programación entera, el presente problema responde a la siguiente situación:

     Imaginémosnos hacer una excursión a la que solo podemos llevar una mochila con capacidad limitada (obviedad). Cada objeto que introducimos ocupa un volumen dentro de la misma y durante el viaje nos proporcionará un beneficio o utilidad (cantimplora, brújula, etc), el problema surge cuando debemos elegir *qué objetos seleccionar* para llevar en la mochila de forma que nuestro *beneficio sea máximo* (tener todo lo necesario) sin exceder su capacidad.

     Esta situación se presenta con cierta frecuencia en los ámbitos económico e industrial, donde la mochila suele representar la restricción presupuestaria (cantidad máxima de recursos económicos de los que se dispone) y donde la utilidad de los objetos seleccionados se equipara a un beneficio económico por adquirir o llevar a cabo ciertas acciones.

   - Caso Especial:
     El algoritmo de Merkle-Hellman
     Merkle-Hellman es un criptosistema asimétrico, ésto significa que para la comunicación, se necesitan dos llaves: una llave pública y una privada. Otra diferencia con RSA, es que sirve sólo para cifrado, es decir, la llave pública es usada sólo para cifrar (no para verificar firma) y la llave privada es usada sólo para descifrar (no para firmar). De este modo, no se puede usar para tareas de autentificación por firma electrónica.

     El algoritmo de Merkle-Hellman está basado en el problema de la mochila de decisión (un caso especial del problema de la mochila de optimización): dados una secuencia de números y un número, determinar si existe un subconjunto de la secuencia cuya suma dé dicho número.

     En general, es sabido que este problema es de clase NP-completo. Sin embargo, si la secuencia de números es supercreciente - ésto es, si cada elemento de la secuencia es mayor que la suma de todos los anteriores - el problema es "fácil", y es posible resolverlo en tiempo polinomial con un simple algoritmo voraz.

2. Investigue y explique brevemente algún método para resolver problemas de programación lineal entera, binaria y mixta.
   R:
   Algunos algoritmos que sirven para resolver los tipos de ejercicios planteados son:

   - Programación Lineal Entera Pura:
     - Método de Plano de Corte.
     - Algoritmo Gomory.
     - Método de Ramificación y Acotamiento.
   - Programación Lineal Entera Binaria:
     - Método de Ramificación y Acotamiento.
     - Distancia de Hamming y Retículos.
     - Método de Trubin.
   - Programación Lineal Entera Mixta
     - Algoritmo Entero Mixto de Gomory.

- Método de Benders.
- Método de Ramificación y Acotamiento.

Pasaremos a definir entonces un método para poder resolver los tres tipos de problemas planteados:

Ramificación y Poda (Branch and Bound)

Es un algoritmo que surge como una variante del Backtracking mejorado sustancialmente. Se aplica mayoritariamente para resolver problemas de optimización.

La técnica de Ramificación y poda se suele interpretar como un árbol de soluciones, donde cada rama nos lleva a una posible solución posterior a la actual. La característica de esta técnica con respecto a otras anteriores es que el algoritmo se encarga de detectar en qué ramificación las soluciones dadas ya no están siendo óptimas, para "podar" esa rama del árbol y no continuar malgastando recursos y procesos en casos que se alejan de la solución óptima.

Nuestra meta será encontrar el valor mínimo de una función $f(x)$ donde fijamos x rangos sobre un determinado conjunto S de posibles soluciones. Un procedimiento de ramificación y poda requiere dos herramientas.

La primera es la de un procedimiento de expansión, que dado un conjunto fijo S de candidatos, devuelve dos o más conjuntos más pequeños $S_1, S_2, \ldots, S_n$ cuya unión cubre S. Nótese que el mínimo de $f(x)$ sobre S es $minV1, V2, \ldots$ donde cada v_i es el mínimo de $f(x)$ sin S_i.

Este paso es llamado ramificación; como su aplicación es recursiva, esta definirá una estructura de árbol cuyos nodos serán subconjuntos de S.

La idea clave del algoritmo de ramificación y poda es: si la menor rama para algún árbol nodo(conjunto de candidatos) A es mayor que la rama padre para otro nodo B, entonces A debe ser descartada con seguridad de la búsqueda. Este paso es llamado "poda", y usualmente es implementado manteniendo una variable global m que graba el mínimo nodo padre visto entre todas las subregiones examinadas hasta entonces. Cualquier nodo cuyo nodo hijo es mayor que m puede ser descartado.

3. ¿Qué tipo de problemas son más difíciles de solucionar? ¿Por qué?
   R:

   Los problemas de programación entera pura son casos en lo que se requerí que todas las variables tengan valores enteros, por lo cual lo hacen mucho mas complejo que el problema de programación entera mixta, pues en esos casos solo se requiere que algunas pero no todas, las variables de decisivo tengan valores enteros; y que el problema de programación entera binaria, que es un caso especial cuando todas las variables de decisión deben tener valores enteros 1 o 0.

   La solución de un problema de programación entera es mucho más deifico de resolver que un problema de PL. El tiempo requerido para resolver algunos de éstos problemas anteriormente vistos, puede ser extenso incluso con computadores de última generación.

4. ¿Qué impide que la UTFSM implemente un horario compacto, qué restricciones impiden esto?
   R:
   El problema de poder obtener un horario compacto, va a depender de muchos factores los cuales serán señalados a continuación.

   Si bien es cierto nuestra universidad posee una gran *cantidad de salas*, esta no es posible acomodar cada año debido a que la *cantidad de alumnos* varía cada año de una forma impredecible.

   De la misma forma es fácil darnos cuenta que *todas las salas* no tienen la misma capacidad, lo cual impide que se pueda impartir cualquier curso, con cualquier cantidad de alumnos, en cualquier sala.

Otro punto importante es que los profesores en la universidad, dictan más de una asignatura, lo cual no permite a un profesor en más de una instancia en un bloque.

Finalmente tenemos que notas que existen muchos *bloques Libres* en el horario de cada alumno, ya sea para almuerzos, recreos, y sobre todo las ventanas que existen en cada uno de nuestros horarios.

Lo anteriormente señalado, es un factor clave para no poder implementar un horario compacto en nuestra universidad.