Programación con Restricciones

Tarea 3

Daniel San Martín

Departamento de Informática
Universidad Técnica Federico Santa María

21 de junio de 2017

El problema Set Covering

Dado un conjunto de elementos $\{1, 2, ..., m\}$ (llamado universo) y n conjuntos cuya unión comprende el universo, el *set cover problem* consiste en identificar el menor número de conjuntos cuya unión aun contiene todos los elementos del universo.

El problema Set Covering

Definición Formal

Sea el universo $\mathcal{U}$ y la familia $\mathcal{S}$ de subconjuntos de $\mathcal{U}$, una cobertura es una subfamilia $\mathcal{C} \subseteq \mathcal{S}$ de conjuntos cuya unión es $\mathcal{U}$.

El problema Set Covering

Este problema puede ser formulado como un problema de programación lineal entera.

$$\begin{aligned} \text{Minimizar } Z &= \sum_{S \in \mathcal{S}} x_S \\ \text{sujeto a } \sum_{S:\, e \in S} x_S &\leq 1, \ \forall e \in \mathcal{U} \\ x_S &\in \{0,1\}, \ \forall S \in \mathcal{S} \end{aligned}$$

Ejemplo: Ubicación de estaciones de bomberos

Hay seis ciudades en el condado de Kilroy. El condado debe determinar dónde construir las estaciones de bomberos. El condado quiere construir el número mínimo de estaciones de bomberos necesarias para asegurar que al menos una estación de bomberos se encuentre dentro de los 15 minutos (tiempo de conducción) de cada ciudad (Winston y Goldberg, 2004).

Ejemplo: Ubicación de estaciones de bomberos

	Hasta					
Desde	Ciudad 1	Ciudad 2	Ciudad 3	Ciudad 4	Ciudad 5	Ciudad 6
Ciudad 1	0	10	20	30	30	20
Ciudad 2	10	0	25	35	20	10
Ciudad 3	20	25	0	15	30	20
Ciudad 4	30	35	15	0	15	25
Ciudad 5	30	20	30	15	0	14
Ciudad 6	20	10	20	25	14	0

Cuadro 1: Tiempos entre ciudades

Ejemplo: Solución

Para cada ciudad donde se debe construir una estación se definen variables binarias x_i, $i = 1, ..., 6$, de forma que:

$$x_i = \begin{cases} 1 & \text{si hay que construir una estación en la ciudad } i \\ 0 & \text{en otro caso} \end{cases}$$

Ejemplo: Solución

De esta forma el número total de estaciones que se deben construir queda definido por $\sum_{i=1}^{6} x_i$, es decir, la función a minimizar es

$$z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$$

Ejemplo: Solución

Para construir las restricciones necesitamos conocer las ciudades que están dentro de 15 minutos de distancia. Esto se puede ver en el Cuadro 2

Ciudad	Dentro de 15 minutos
1	1, 2
2	1, 2, 6
3	3, 4
4	3, 4, 5
5	4, 5, 6
6	2, 5, 6

Cuadro 2: Ciudades dentro de los 15 minutos

Ejemplo: Solución

Con la información anterior, es posible modelar el problema de la siguiente forma:

$$\min z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$$

$$\begin{aligned} \text{s. a} \quad & x_1 + x_2 \geq 1 \\ & x_1 + x_2 + x_6 \geq 1 \\ & x_3 + x_4 \geq 1 \\ & x_3 + x_4 + x_5 \geq 1 \\ & x_4 + x_5 + x_6 \geq 1 \\ & x_2 + x_5 + x_6 \geq 1 \end{aligned}$$

Desarrollo

Para solucionar el problema, se realizaron 4 estrategias de *branching*:

1. Selección de variable de mayor dominio. Selección de valores mayores que la media de menor y mayor valor.
2. Selección de variable de mayor dominio. Selección de valores menores que la media de menor y mayor valor.
3. Selección de variable de menor dominio. Selección de valores mayores que la media de menor y mayor valor.
4. Selección de variable de menor dominio. Selección de valores menores que la media de menor y mayor valor.

Desarrollo: Resultados

Tanto para la segunda y cuarta (primera y tercera) estrategias se obtuvieron los mismos resultados. La solución del problema es $z = 2$, $x_2 = x_4 = 1$, $x_1 = x_3 = x_5 = x_5 = x_6 = 0$, esto significa ubicar las estaciones de bomberos en las ciudades 2 y 4. Las estadísticas de *Gecode* se muestran en el Cuadro 3

Desarrollo: Estadísticas *Gecode*

Estrategia		**2-4**	**1-3**
Initial	Propagators:	7	7
	Branchers:	2	2
Summary	Runtime:	~0.23 ms	~0.33 ms
	Solutions:	1	5
	Propagations:	37	85
	Nodes:	7	17
	Failures:	3	4
	Restarts:	0	0
	No-goods:	0	0
	Peak depth:	4	6

Cuadro 3: Resumen resultados

Conclusiones

Del desarrollo de la tarea se obtuvieron las siguientes conclusiones:

- En este caso particular, para las estrategias que seleccionan el menor valor de variable, encuentran el resultado casi instantáneamente. Esto tiene lógica, dado que lo que buscamos en el problema es minimizar una función, por lo que sería adecuado comenzar seleccionando las variables igual a 0.
- Las estrategias que seleccionen los mayores valores de variable, son las que más procesos debe realizar. Al contrario del caso anterior, las variables comenzaría con el valor 1 por lo tanto mostraría cada una de las posibles soluciones (y es lo que muestra *Gecode* al ejecutar el código).

Referencias

Schulte, C., Lagerkvist, M., y Tack, G. (2017). Gecode. *Software download and online material at the website: http://www.gecode.org.*

Winston, W. L., y Goldberg, J. B. (2004). *Operations research: applications and algorithms* (Vol. 3). Duxbury press Boston.