Complejidad y optimización

Reducción SAT a Programación Entera

Carlos Andrés Delgado S.

Facultad de Ingeniería. Universidad del Valle

Febrero 2017

Contenido

Programación entera

Definición

Se tiene conjunto A de v variables enteras, un conjunto de desigualdades entre estas variables y una función $f(v)$ de variables a maximizar y un entero B.

Problema de decisión

¿Existe una asignación de enteros de v que satisfaga todas las desigualdades y $f(v) \geq B$?. **Recuerda:** Un problema de decisión tiene como respuesta **SI** o **NO**

Ejemplo

Una instancia del problema de programación entera es:

$$A = \{v_1, v_2\}$$
$$f(v) = 2v_2, B = 3$$
$$v_1 \geq 1, v_2 \geq 0$$
$$v_1 + v_2 \leq 3$$

¿Que valores de v_1 y v_2 satisfacen este problema?

Ejemplo

Una instancia del problema de programación entera es:

$$\begin{aligned} A &= \{v_1, v_2\} \\ f(v) &= 2v_2, B = 3 \\ v_1 &\geq 1, v_2 \geq 0 \\ v_1 + v_2 &\leq 3 \end{aligned}$$

¿Que valores de v_1 y v_2 satisfacen este problema?.
Respuesta: $v_1 = 1$ y $v_2 = 2$

Ejemplo

Otro problema:

$$
\begin{aligned}
A &= \{v_1, v_2\} \\
f(v) &= 2v_2, B = 5 \\
v_1 &\geq 1, v_2 \geq 0 \\
v_1 + v_2 &\leq 3
\end{aligned}
$$

No hay solución, debido a que la restricción $v_1 + v_2 \leq 3$ y $v_1 \geq 1$ impiden que v_2 tome un valor mayor que 2, y debe cumplirse $f(v) \geq B$.

Programación entera

Programación entera es NP

El problema e programación entera es NP, ya que si tomamos un conjunto A de v_i variables enteras de tamaño n, se necesitará generar un conjunto de combinaciones d^n de las variables v_i para solucionar el problema, donde d son los valores posibles que puede tomar v_i.

Ejemplo

Supongamos que $d = 3$ y $n = 4$. entonces los valores que pueden tomar v_i son $\{1, 2, 3\}$, las combinatorias de los valores son ~~$\{(0, 0, 0, 0)(0, 0, 0, 1), ..., (3, 3, 3, 3)\}$~~ El tamaño de ese conjunto es 4^3.

3^4

(1,1,1,1) (1,1,1,2) ... (3,3,3,3)

3 3 3 3 =

__ __ __ __

Contenido

¿Programación entera es NPC?

Demostración.

Postulado Sabemos que SAT es NP-Completo, entonces reduciremos desde una instancia de SAT a una instancia de programación entera. Denotaremos Programación Entera como **IP** *(Integer Programming)* □

Importante

$$SAT \leq_p IP.$$

¿Programación entera es NPC?

Procedimiento de reducción

Se sabe que SAT, es un conjunto de v_i variables y un conjunto de clausulas c_i en forma normal conjuntiva. Para realizar la reducción se crean las siguientes restricciones:

- $0 \leq v_i \leq 1$ y $0 \leq \neg v_i \leq 1$ Ambas variables están restringidas por valores 0 y 1, equivalentes a verdadero o falso.
- $1 \leq v_i + \neg v_i \leq 1$ Si una de las variables es 1, su negado debe ser 0 y viceversa.
- Por cada clausula $c_i = \{v_i, v_j, ... v_k\}$ se crea una restricción $v_i + v_j + ... + v_k \geq 1$. Esto garantiza que si la clausula es satisfecha en SAT debe al menos existir una variable que sea verdadera.

¿Programación entera es NPC?

Procedimiento de reducción

Continuando:

- La función de maximización es relativamente poco importante, basta con: $f(v) = v_1$ y $B = 0$.

Como se puede observar esta reducción es en tiempo polinomial

1. Si se tienen n variables en SAT, se crean $2n$ variables y $3n$ restricciones en PI
2. Si se tienen y clausulas en SAT, se crean $2y$ restricciones en PI

¿Programación entera es NPC?

Instancias positivas en SAT = instancias positivas en PI

Si todos las cláusulas en SAT son verdaderas $c_i = \{v_i, v_j, \dots v_k\}$, si cumplirá $v_i + v_j + \dots + v_k \geq 1$. Las restricciones $0 \leq v_i \leq 1$, $0 \leq \neg v_i \leq 1$ y $1 \leq v_i + \neg v_i \leq 1$ **siempre se cumplen** ya que v_i toma valores 0 o 1.

Instancias negativas en SAT = instancias negativas en PI

Si alguna de las clausulas $c_i = \{v_i, v_j, \dots v_k\}$ no se cumple, entonces $v_i + v_j + \dots + v_k \geq 1$ no se puede cumplir debido a que todas las $v_i = 0$.

¿Programación entera es NPC?

Ejemplo

Transformar esta instancia de SAT a PI

$$C = \{v_1, \neg v_2, \neg v_3\}, \{v_2, v_3\}$$

Como se puede observar este SAT se puede satisfacer con $v_1 = V, v_2 = V, v_3 = F$

¿Programación entera es NPC?

Ejemplo

$$C = \{v_1, \neg v_2, \neg v_3\}, \{v_2, v_3\}$$

v1+¬v2+¬v3 >= 1 , v2 + v3 >= 1

- Generamos las variables $\{v_1, v_2, v_3, \neg v_1, \neg v_2, \neg v_3\}$
- Se agregan las restricciones $0 \leq v_1 \leq 1, 0 \leq v_2 \leq 1, 0 \leq v_3 \leq 1, 0 \leq \neg v_1 \leq 1, 0 \leq \neg v_2 \leq 1, 0 \leq \neg v_3 \leq 1$
- Agregamos las restricciones $1 \leq v_1 + \neg v_1 \leq 1, 1 \leq v_2 + \neg v_2 \leq 1, 1 \leq v_3 + \neg v_3 \leq 1$
- Finalmente $f(v) = v_1$ y $B = 0$

Si comprobamos efectivamente se cumple PI, con $v_1 = 1, v_2 = 1, v_3 = 0, \neg v_1 = 0, \neg v_2 = 0, \neg v_3 =$ 1

¿Programación entera es NPC?

Ejercicio 1

$$V = \{v_1, v_2, v_3\}$$
$$C = \{\{\neg v_1, \neg v_2, \neg v_3\}, \{v_2, v_3\}, \{\neg v_1 . v_2\}, \{v_2, \neg v_3\}\}$$

Ejercicio 2

$$V = \{v_1, v_2, v_3, v_4\}$$
$$C = \{\{\neg v_1, \neg v_3\}, \{v_2, v_3\}, \{\neg v_4\}, \{\neg v_1 . v_2\}, \{v_2, \neg v_3\}, \{v_4\}\}$$

En ambos casos piense siempre ¿Esta instancia de SAT se puede satisfacer?

$v1>=0, v2>=0, v3>=0, \neg v1>=0, \neg v2>=0, \neg v3>=0$

$1>=v1+\neg v1>=1, 1>=v2+\neg v2>=1, 1>=v3+\neg v3>=1$

$\neg v1+\neg v2+\neg v3 >= 1$
$v2 + v3 >= 1$
$\neg v1+v2 >= 1$
$v2+\neg v3 >= 1$

v1=F, v2=V, v3=F SAT
$v1=0, v2=1, v3=0, \neg v1=1, \neg v2=0, \neg v3=1$ PI

Preguntas