

Universidad de La Habana

FACULTAD DE MATEMÁTICA Y COMPUTACIÓN

PROBLEMA DE OPTIMIZACIÓN LIBRE DEL CONTEXTO

Daniel Abad Fundora C-411

Anabel Benítez González C-411

Juan Carlos Espinoza Delgado C-411

Raudel Alejandro Gómez Molina C-411

Alex Sierra Alcalá C-411

Introducción

Los problemas de optimización en enteros son una rama de la optimización matemática que se ocupa de encontrar soluciones óptimas a problemas donde las variables de decisión solo pueden tomar valores enteros. Estos problemas se encuentran en una amplia variedad de áreas, como:

- Ingeniería: diseño de estructuras, planificación de producción, gestión de proyectos.
- Finanzas: selección de inversiones, planificación financiera, gestión de riesgos.
- Ciencias de la computación: criptografía, diseño de algoritmos, aprendizaje automático.

Los POE son generalmente más difíciles de resolver que los problemas de optimización continuos. Esto se debe a que la discreción de las variables enteras añade complejidad al problema.

En el presente trabajo abordaremos una propuesta de solución a problemas de optimización en enteros, con variables binarias (que cumplen ciertas restricciones) usando herramientas de teoría de lenguajes formales y autómatas.

Antecedentes

Este trabajo se basa en los resultados obtenidos en [1], en el cual se plantea un algoritmo polinomial para resolver el problema de la satisfacibilidad booleana libre del contexto, el cual se basa en la teoría de lenguajes formales y autómatas. En este trabajo se propone una adaptación a este problema para resolver problemas de optimización en enteros.

En [1], se transforma una fórmula booleana en una lista de instancias de variables, donde se asume que 2 instancias de una variable no tienen por qué tener el mismo valor de verdad. Luego se define un autómata que dada una cadena de 0 y 1 y una fórmula booleana, determina si se obtiene un valor de verdad para la fórmula booleana donde cada instancia de una variable toma el valor de verdad que se corresponde con la cadena de 0 y 1. Entonces para verificar que 2 instancias de una variable tengan el mismo valor de verdad se intercepta dicho autómata con una gramática libre del contexto obteniendo un autómata de pila. Después de esto se plantea un algoritmo para dado este autómata de pila, generar todas las posibles cadenas de 0 y 1 que se corresponden con una asignación de valores de verdad y como consecuencia se obtiene un generador de todas las posibles asignaciones de valores de verdad para una fórmula booleana. Luego para determinar si la fórmula es satisfacible solo queda verificar si este conjunto de soluciones es no vacío.

Una fórmula booleana se considera libre del contexto si para cualquier par de instancias de una variable x_i y x_j con $i < j$ se cumple que si existe otra variable con instancia x_k con $i < k < j$ entonces todas las instancias de esta nueva variable ocurren entre x_i y x_j .

El problema antes planteado ha sido generalizado en [2] y [3] donde en vez de una gramática libre del contexto se usa una gramática de concatenación de rango y gramáticas matriciales simples respectivamente, para ser interceptadas con el autómata booleano y al igual que en [1] obtener un mecanismo generador de soluciones que satisfacen la fórmula booleana.

Planteamiento del problema

Para utilizar el algoritmo propuesto en [1] para resolver problemas de optimización en enteros, una fórmula booleana a una restricción de un problema de optimización en enteros. Por ejemplo sea la fórmula booleana:

$$x_1 \vee x_2 \vee x_3 \vee x_4 \tag{1}$$

La restricción para que sea satisfacible puede ser interpretada de la siguiente manera:

$$x_1 + x_2 + x_3 + x_4 \geq 1 \tag{2}$$

donde los x_i son variables binarias.

Problema de optimización libre del contexto

Sea x_1, x_2, \dots, x_n un conjunto de variables binarias. Se desea minimizar la función objetivo:

$$f(x_1, x_2, \dots, x_n) = c_1x_1 + c_2x_2 + \dots + c_nx_n \quad (3)$$

Sujeto a las restricciones:

$$\sum_{i=1}^n b_{j,i}x_i \geq 1, \quad j = 1, 2, \dots, m \quad (4)$$

donde los $b_{j,i}$ son 0 o 1 y además se cumple para si todo k entre 1 y n existen i y j tal que $b_{i,k} = 1$ y $b_{j,k} = 1$, $p \neq k$, z tal que $i < k < j$ y $b_{z,p} = 1$ entonces no existe q con $q < i \vee j > q$ tal que $b_{q,k} = 1$.

Las restricciones de dicho problema de optimización se pueden plantear con un SAT libre del contexto (resultado obtenido en [1]) de la siguiente manera:

$$x_{a_{j,1}} \vee x_{a_{j,2}} \vee \dots \vee x_{a_{j,k}}, \quad \forall i : b_{1,i} \neq 0, \quad j = 1, 2, \dots, m \quad (5)$$

para este SAT se obtiene una gramática libre del contexto que genera todas las posibles asignaciones de valores de verdad para las variables x_i , que para nuestro problema de optimización se traduce en todas las posibles asignaciones de valores que satisfacen las restricciones.

Dicha gramática como se demuestra en [1] es un grafo acíclico dirigido, donde cada arista representa una producción y a su vez un valor de verdad para cada instancia de una variable. Luego para obtener el valor mínimo de la función objetivo se puede ponderar cada arista que represente la primera instancias de una variable con el valor de la función objetivo para dicha instancia y el resto de las aristas con 0. Luego identificar los estados nodos que representan los estados finales de la gramática y usar un algoritmo para minimizar el costo de llegar desde el distinguido a los estados finales.

El algoritmo propuesto será el algoritmo de Dijkstra, donde se considera el costo de llegar a un estado nodo como el valor de la función objetivo acumulado. En el caso de tener aristas negativas podemos utilizar la modificación del algoritmo para aristas de costo negativo (podemos garantizar el correcto funcionamiento de esta transformación ya que no existen ciclos y por tanto no existen ciclos de costo negativo) o también pudiéramos considerar el uso de Bellman Ford aunque esto implica una mayor complejidad.

Generalización del problema libre del contexto

Referencias

- [1] A. Fernández Arias, "El Problema de la Satisfacibilidad Booleana Libre del Contexto. Un Algoritmo polinomial.," La Habana, 2007.