Reporte

Iñaki Liendo

26 de abril de 2025

1. Resumen

Diseñé un método que encuentra todas las soluciones a problemas de la forma

$$\begin{aligned} \text{máx. } & c^t x, \\ \text{s.a. } & c^t x \leq s, \\ & x \in \mathbb{Z}^n, \end{aligned}$$

y que corre en tiempo lineal con respecto a la magnitud del lado derecho de la desigualdad pero exponencial con respecto al número de dimensiones. Además de encontrar todas las posibles soluciones, es más rápido que *branch & bound*. A pesar de tener complejidad lineal, los experimentos que realicé sugieren que en la práctica tiene complejidad constante. Aquellos dos últimos puntos contrastan significativamente con *B&B*, pues este algoritmo encuentra solamente una solución y tiene una complejidad más que lineal con respecto a la magnitud del lado derecho de la desigualdad (habíamos dicho que era exponencial en nuestras pláticas si mal no recuerdo, pero los experimentos que hice indican otra cosa).

2. Prerrequisitos

Definición 1 *Dados los enteros a y b, diremos que a divide a b (y escribimos $a \mid b$) si existe un entero r tal que $a \cdot r = b$.*

Definición 2 *Dados dos enteros a y b distintos de cero, su máximo común divisor es el entero d que satisface*

1. *$d \mid a$ y $d \mid b$ (d es un divisor común de a y b).*
2. *Si $d' \mid a$ y $d' \mid b$, entonces $d' \leq d$ (d es el máximo de los divisores comúnes).*

Escribimos $d = \gcd\{a, b\}$ para denotar que d es el máximo común divisor de a y b. Así también, si $a_1, a_2, \ldots, a_n$ son enteros distintos de cero, definimos su máximo común de manera inductiva: para $n = 2$ usamos la definición 2 y si $n > 2$ entonces

$$\gcd\{a_1, a_2, \ldots, a_n\} = \gcd\{a_1, \gcd\{a_2, \ldots, a_n\}\}.$$

Definición 3 *Sean $a, b, c \in \mathbb{Z}$. Decimos que c es combinación lineal entera de a y b si existen enteros x y y tales que $c = ax + by$.*

Teorema 1 *Sean $a, b, d \in \mathbb{Z} \setminus \{0\}$. Entonces $d = \gcd\{a, b\}$ si y solo si d es la mínima combinación lineal entera positiva de a y b.*

Definición 4 *Sea d el máximo común divisor de los enteros a y b. Llamamos coeficientes de Bézout asociados a a y b a los enteros x, y que satisfacen $d = ax + by$.*

Cabe mencionar que los coeficientes de Bézout no son únicos. Por ejemplo, 1 es el máximo común divisor de 3 y 5, de tal forma que se cumple $1 = 5(2) + 3(-3) = 5(5) + 3(-8)$. Así, $(2, -3)$ y $(5, -8)$ son dos pares de coeficientes de Bézout asociados a 3 y 5.

Definición 5 *Sean $a, b \in \mathbb{Z} \setminus \{0\}$. Decimos que a y b son coprimos si $\gcd\{a, b\} = 1$.*

Corolario 1 *Si $d = \gcd\{a, b\}$, entonces $\gcd\{\frac{a}{d}, \frac{b}{d}\} = 1$.*

Definición 6 *Sean $a, b, c \in \mathbb{Z}$. Una pareja de enteros (x_0, y_0) es una solución de la ecuación $ax + by = 0$ si se satisface $ax_0 + by_0 = c$.*

Teorema 2 *Sean $a, b \in \mathbb{Z} \setminus \{0\}$. La ecuación $ax + by = c$ tiene solución en los enteros si y solo si $\gcd\{a, b\} \mid c$.*

Teorema 3 *Sean $a, b, c \in \mathbb{Z}$ tales que $a, b \neq 0$ y sea (x_0, y_0) una solución particular de la ecuación $ax + by = c$. Entonces todas las soluciones de la ecuación están dadas por*

$$\begin{cases} x = x_0 + \frac{b}{d} \cdot t, \\ y = y_0 + \frac{a}{d} \cdot t, \end{cases}$$

donde $d = \gcd\{a, b\}$ y $t \in \mathbb{Z}$.

Dada la generalización del máximo común divisor a más de dos enteros, bien es cierto que podemos aplicar inducción para generalizar también las definiciones y resultados anteriores. Decimos, por ejemplo, que un vector $v \in \mathbb{Z}^n \setminus \{0\}$ es coprimo si $\gcd\{v_1, \ldots, v_n\} = 1$. De ser este el caso, existen coeficientes de Bézout $x_1, \ldots, x_n \in \mathbb{Z}$ asociados a v que satisfacen $v_1 x_1 + \ldots + v_n x_n = 1$. Podemos decir, en forma más compacta, que $x = (x_1, \ldots, x_n) \in \mathbb{Z}^n$ es un vector de Bézout asociado a v y por lo tanto se cumple $v^T x = 1$.

Los siguientes teoremas, lemas y definiciones fueron tomadas de [BH09].

Definición 7 *Sea $v \in \mathbb{R}^n$ un vector. Decimos que el hiperplano afino*

$$H_{v,t} := \ker(x \mapsto v^T x) + t \cdot v$$

es una capa entera si contiene al menos un punto entero, es decir, si contiene un punto cuyas entradas son todas enteras.

Lema 1 *Sea $v \in \mathbb{R}^n$ un vector distinto de cero y sea $x \in \mathbb{R}^n$ un punto. Sea $t_x := \frac{v^T x}{||v||_2^2}$. Entonces $x \in H_{v,t_x}$.*

Definición 8 *Sea $v \in \mathbb{R}^n$ un vector. Decimos que v es esencialmente entero si existe un vector $w \in \mathbb{Z}^n$ y un escalar $k \in \mathbb{R}$ tal que $v = kw$. De otra forma, decimos que v es esencialmente irracional.*

Es decir, decimos que v es esencialmente entero si es un múltiplo real de un vector entero. Por las siguientes razones descartamos a los vectores esencialmente irracionales de nuestro análisis: en primer lugar, porque la motivación de este trabajo es la asignación de recursos que tienen un costo asociado, el cual es necesariamente racional; en segundo lugar, porque todo número representable en cualquier sistema de aritmética finita es necesariamente racional. Independientemente del caso, solamente son de nuestro interés los vectores racionales, pero todo vector v en $\mathbb{Q}^n$ es esencialmente entero.

Definición 9 *Sea $v \in \mathbb{R}^n$ un vector esencialmente entero y sea $(w_1, \ldots, w_n) \in \mathbb{Z}^n$ un múltiplo entero de v. Decimos entonces que el vector*

$$\frac{1}{\gcd\{w_1, \ldots, w_n\}} \cdot (w_1, \ldots, w_n)$$

es un múltiplo coprimo de v.

Para asegurar unicidad, forzamos a que la primera entrada del múltiplo coprimo de v sea positivo. Por el Corolario 1 tenemos que el múltiplo coprimo de un vector esencialmente entero es, en efecto, un vector coprimo.

Método simple

La idea del método es la siguiente. Para empezar, nos concentramos exclusivamente en vectores de utilidad $c \in \mathbb{R}^n$ proyectivamente racionales. Sea c' el múltiplo coprimo de c, entonces el número de *c-layers* entre 0 y la frontera de la restricción $p^t x \leq s$ está dado por $\eta := \lfloor s \cdot p'_n / p_n \rfloor$.

Ahora bien, buscamos caracterizar la colección de puntos enteros que se encuentran en el k-ésimo *c-layer* con $k \in \{0, 1, \ldots, \eta\}$. Es decir, buscamos resolver la ecuación entera

$$p'_1 x_1 + p'_2 x_2 + \cdots + p'_n x_n = k.$$

Como $\gcd(p'_1, \ldots, p'_n) = 1$, se sigue que para cualquier $k \in \mathbb{Z}$ existe una infinidad de soluciones enteras parametrizadas por una solución inicial $(x_1^{(0)}, \ldots, x_n^{(0)})$ y $n-1$ parámetros $t_1, \ldots, t_{n-1}$. Para obtener la solución inicial notemos que podemos resolver el sistema auxiliar

$$p'_1 x_1 + p'_2 x_2 + \cdots + p'_n x_n = 1,$$

donde una solución son los coeficientes de Bézout de $p'_1, \ldots, p'_n$. Luego, a esta solución la multiplicamos por k y de esta forma obtenemos $x^{(0)}$. Finalmente, para obtener una solución factible (no negativa) "simplemente" acotamos los parámetros. La forma en la que acoto actualmente los parámetros es recursivamente.

Consideremos el caso $n = 2$. La solución a la ecuación entera está dada por

$$\begin{cases} x_1 = kx_1^{(0)} + p'_2 t, \\ x_2 = kx_2^{(0)} - p'_1 t, \end{cases}$$

donde $x_1^{(0)}$ y $x_2^{(0)}$ son los coeficientes de Bézout de p'_1 y p'_2. Como buscamos factibilidad, se debe cumplir

$$\frac{-k \cdot x_1^{(0)}}{p'_2} \leq t \leq \frac{k \cdot x_2^{(0)}}{p'_1}, \quad t \in \mathbb{Z}.$$

Luego, para acotar los parámetros en n dimensiones, definimos $w := p'_1 x_1 + \ldots + p'_{n-1} x_{n-1}$ y resolvemos el sistema $w + p'_n x_n = k$. Con esto regresamos al caso $n = 2$ y tenemos que el parámetro $n-1$ debe satisfacer

$$\frac{-k \cdot w^{(0)}}{p'_n} \leq t_{n-1} \leq \frac{k \cdot x_n^{(0)}}{1}, \quad t_{n-1} \in \mathbb{Z}.$$

No es difícil ver que $w^{(0)} = 1$ y $x_n^{(0)} = 0$. Una vez resuelto este sistema, buscamos resolver

$$p_1' x_1 + \cdots + p_{n-1}' x_{n-1} = w = kw^{(0)} + p_n' t_{n-1}$$

para cada t_{n-1} factible y repetimos el proceso anterior hasta poder acotar t_1. Es de esta forma que la complejidad con respecto a la dimensión d es entonces n^d. Notemos que $t_i = 0$ es factible para todo $i \in \{2, 3, \ldots, n-1\}$, así que en términos prácticos valdría la pena resolver primero $p_1' x_1 + p_2' x_2 = k$ y checar si existe una t_1 factible. Si este es el caso entonces obtendríamos inmediatamente una solución factible con $(x_1, x_2, 0, \ldots, 0)$.

Finalmente, como nos encontramos en un problema de maximización, para obtener el óptimo empezamos nuestra búsqueda en los *c-layers* más cercanos a la frontera. Es decir, resolvemos las ecuaciones con $k = \eta$, luego con $k = \eta - 1$, etcétera y terminamos el algoritmo una vez que encontremos una solución factible.

3. Experimentos numéricos

Por el momento solo he realizado experimentos cuando la dimensión n es 2. En ambas figuras 1 y 2 podemos notar que el método que diseñé ("diophantine") es relativamente constante con respecto al *slack*, que denota la magnitud del lado derecho en la desigualdad $p^t x \leq s$.

También se cumple en ambas figuras que el algoritmo de *B&B* sí tiene una dependencia inherente con respecto a s: en un inicio se mantiene constante pues s es demasiado pequeña y no hay efecto real en cuanto al tiempo de convergencia; luego, el tiempo aumenta en proporción a n (cabe mencionar que las gráficas están a escala logarítmica, no sé si haya sido la mejor elección para interpretarlas) y eventualmente aumentos en s no afectan al tiempo. Esto último coincide con la presencia de múltiples soluciones enteras, lo cual parece razonable: mientras más soluciones enteras haya, más probabilidad tiene *B&B* de encontrarse con una y por lo tanto el tiempo de convergencia esperado es compensado por esta mayor probablidad.

Una consecuencia sorprendente de lo expuesto es la siguiente: *Branch & Bound* es sumamente sensible al redondeo numérico. En efecto, mientras más cifras decimales tengan las entradas $p_1, \ldots, p_n$, los puntos enteros que se encuentran sobre la k-ésima c-layer están más distanciados, por lo que el número de puntos enteros factibles se reduce drásticamente (en un orden de 10), y entonces *B&B* tomará significativamente más tiempo en encontrar una solución. Así pues, para compensar de nuevo el tiempo de convergencia necesitamos aumentar el *slack* s, pero esto afecta completamente el problema original.

4. Siguientes pasos

- Implementar el algoritmo "diophantine" para más de dos dimensiones y comparar resultados con *B&B*.
- Pensar en otras alternativas para generalizar el método a más de dos dimensiones. Encontré algo que se llama la forma normal de Smith para resolver sistemas de ecuaciones enteras (ecuaciones lineales enteras en más de dos dimensiones se pueden transformar a sistemas de ecuaciones). Investigar.
- Gráficas, gráficas gráficas: tiempos vs cifras decimales, tiempos vs dimensión, gráficas que transmitan mejor el mensaje.

../figs/cmp-expanded-three-digits.png

Figura 1: Comparación de tiempos entre el método de *Branch & Bound* y el que diseñé cuando los precios tienen tres cifras decimales.

- Posibles aplicaciones (?) o exponerlo como casos degenerados del simplex/b&b (?).

5. Notas

- Siento que este método puede generalizarse un poco más para agregar otro tipo de restricciones al problema original. Solo habría que ver cómo determinar t_i's factibles. Esto puede verse como un problema lineal no-entero sobre t y por lo tanto puede resolverse más rápido (?).
- Si el punto anterior es cierto, entonces este método es otro acercamiento para resolver problemas simétricos (c.f. [BH09]) sin hacer uso de órbitas ni mucho menos de transitividad y por lo tanto es un poco más general.

Referencias

[BH09] R. Bödi and K. Herr. Symmetries in integer programs, 2009.

`../figs/cmp-expanded-four-digits.png`

Figura 2: Comparación de tiempos entre el método de *Branch & Bound* y el que diseñé cuando los precios tienen cuatro cifras decimales.