V Encuentro de trabajo del Grupo de Investigación
Nuevos Desafíos de la Matemática Combinatoria

Nuevas perspectivas para la aplicación de test-sets a problemas de optimización entera.

Noviembre 2014, Sanlúcar de Barrameda

Optimización entera con herramientas algebraicas: Introducción

Dado un problema (P) de optimización lineal entera

$$\boxed{\begin{array}{rl} \min & \mathbf{c}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}^n_+ \end{array}}$$

para $\mathbf{c}$ una función lineal y A y $\mathbf{b}$ con entradas enteras, las bases de Gröbner permiten calcular un test-set del problema **para todo b**:

Test-set

Un test-set asociado a $(A, \mathbf{c})$ es un conjunto finito $\mathcal{T} = \{\mathbf{t}_1, \ldots, \mathbf{t}_N\} \subset \mathbf{Z}^n$ con la siguiente propiedad: una solución $\mathbf{x}^\star$ es óptima si y sólo si

$$\mathbf{c}(\mathbf{x}^\star - \mathbf{t}_i) \geq \mathbf{c}(\mathbf{x}^\star)$$

para todos los $(\mathbf{x}^\star - \mathbf{t}^i), i = 1, \ldots, n$ que sean factibles para (P).

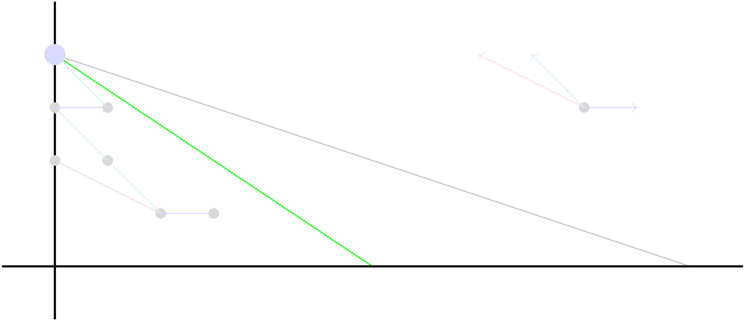
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1,0,-2), (-1,1,-1), (-2,1,1)\}.$$



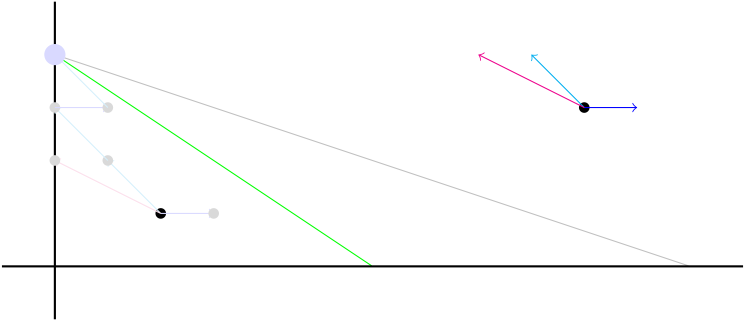
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1, 0, -2), (-1, 1, -1), (-2, 1, 1)\}.$$



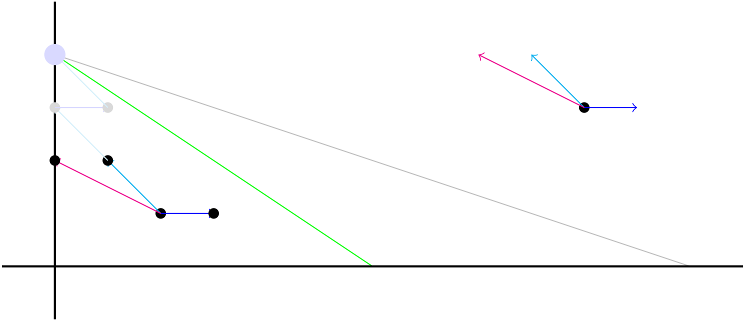
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1, 0, -2), (-1, 1, -1), (-2, 1, 1)\}.$$



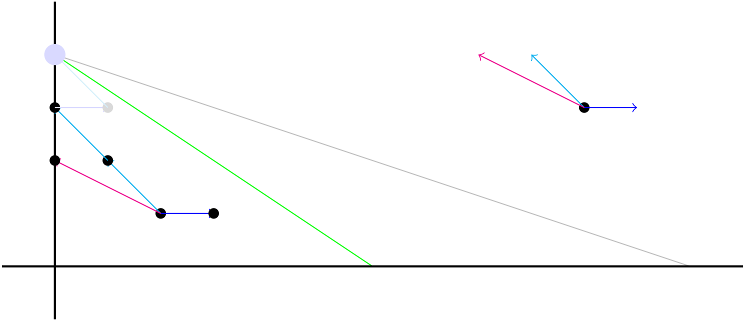
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1,0,-2),(-1,1,-1),(-2,1,1)\}.$$



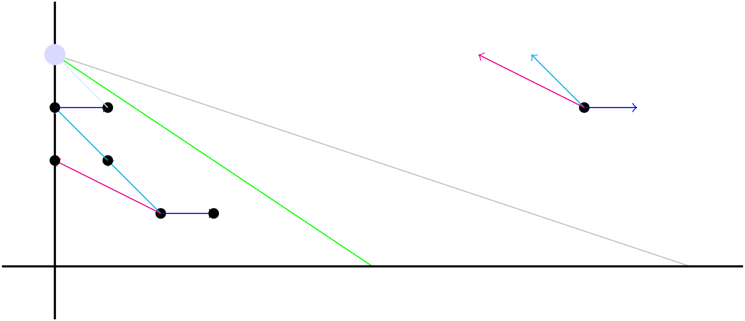
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1,0,-2),(-1,1,-1),(-2,1,1)\}.$$



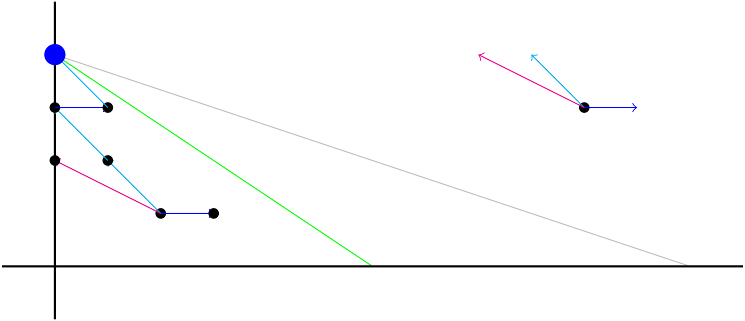
Óptimo lineal visual

Para calcular un test-set asociado al problema y, a partir de él, el óptimo $x^\star$

$$\begin{array}{rl} \max & x_1 + 3x_2, \\ \text{subject to} & 2x_1 + 3x_2 \leq 12, \end{array}$$

podemos usar 4ti2 (para este ejemplo y otros mucho mayores!) obteniendo

$$\mathcal{T} = \{(1,0,-2), (-1,1,-1), (-2,1,1)\}.$$



Óptimos lineales... y no lineales: walk back

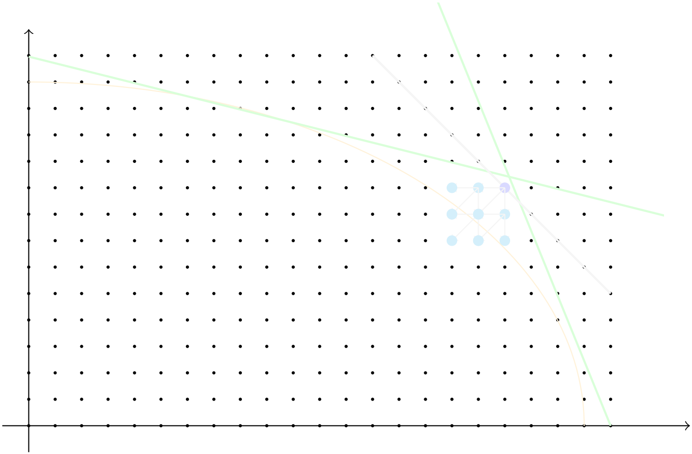
Recíprocamente, desde $x^\star$ **se puede reconstruir todo el conjunto factible aplicando los elementos del test-set inverso**. Este hecho se puede aprovechar para resolver problemas no lineales (P'):

$$\boxed{\begin{array}{rl} \min & \mathbf{c}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \in \mathcal{N} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}}$$

siempre que la pertenencia a $\mathcal{N}$ sea computable.

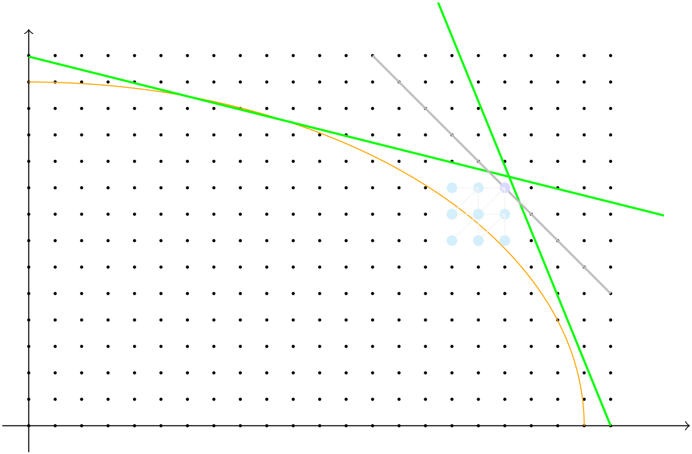
Walk back

Partiendo de un óptimo $x^\star$ de la parte lineal de (P'), vamos aplicando el test-set inverso (empeorando el coste) hasta obtener puntos que estén también en $\mathcal{N}$ y que pasan a ser candidatos a óptimos de (P').



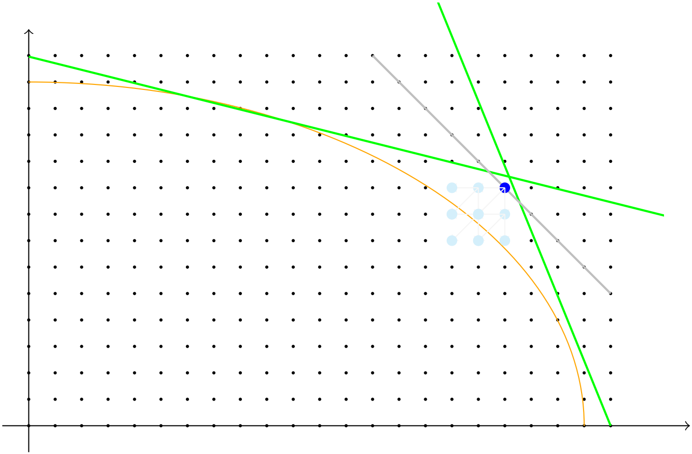
Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

Walking back visual



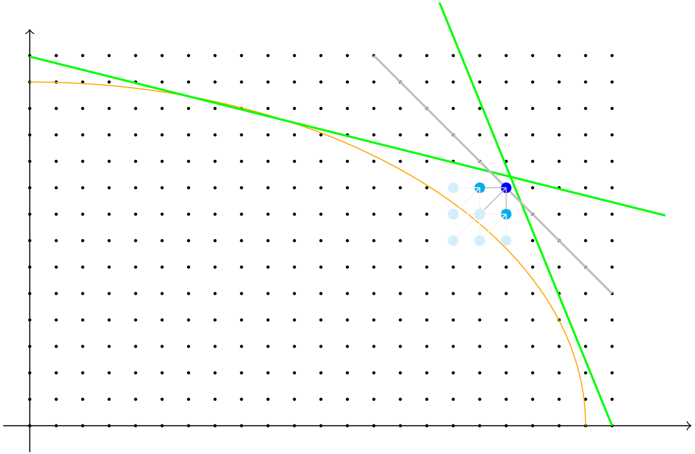
Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

Walking back visual



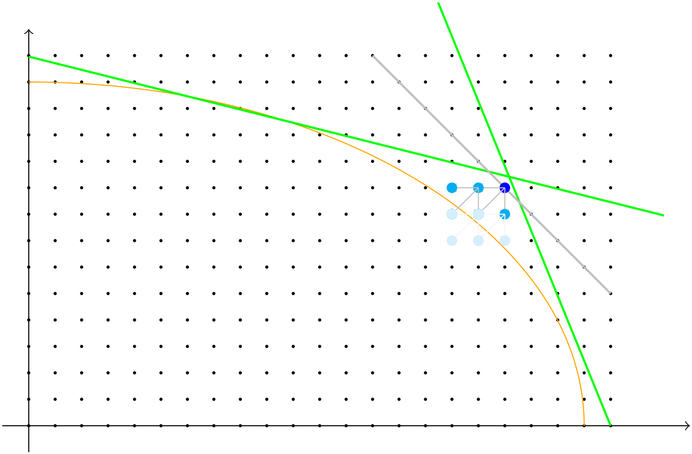
Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

Walking back visual



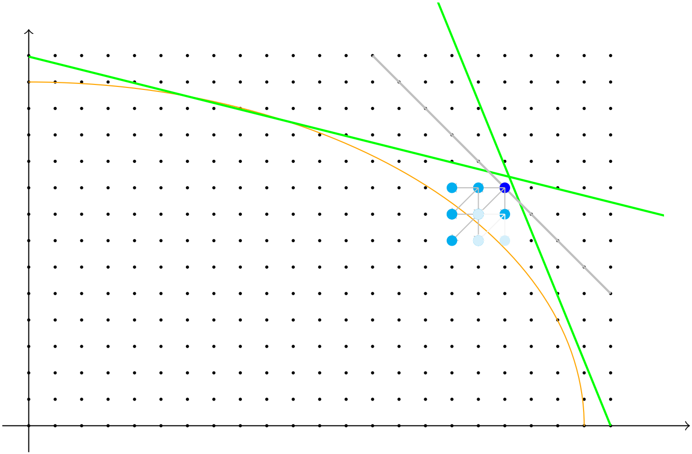
Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

Walking back visual



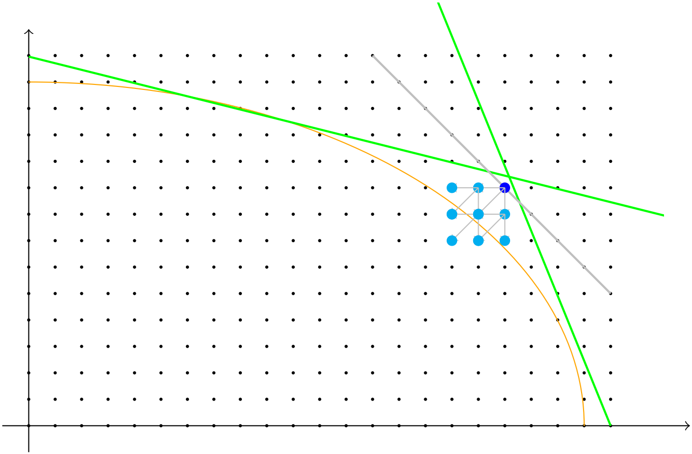
Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

Walking back visual



Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

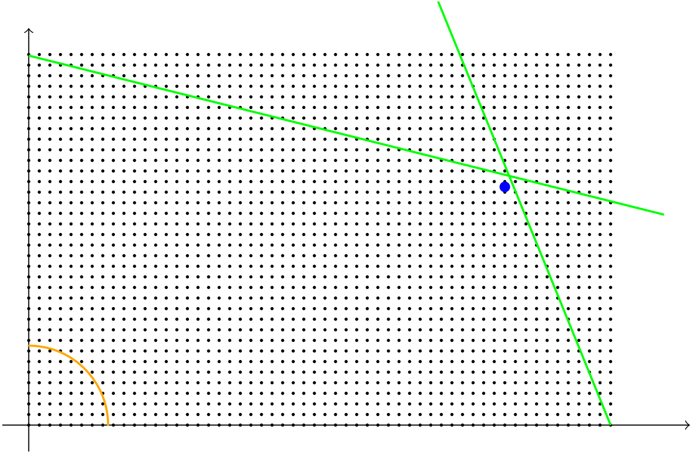
Walking back visual



Vamos recortando el árbol de búsqueda en cuanto entramos en $\mathcal{N}$ (en todo caso cuando llegamos a puntos peores de los que ya tenemos como candidatos).

La triste realidad

La situación suele ser más bien algo así: muy lejos y muchos puntos!



Esta estrategia ha producido resultados competitivos con diferentes en problemas REALES

Algunas perspectivas que estamos desarrollando:

- Cálculo de todos los óptimos de problemas de optimización entera no lineal. (con José Manuel Jiménez Cobano).
- Adelgazamiento del test-set (con Jorge García).
- Optimización lineal entera multiobjetivo (con Haydee Jiménez).
- Métodos heurísticos: aprovechar el esqueleto que da el test-set.

Algunas perspectivas que estamos desarrollando:

- Cálculo de todos los óptimos de problemas de optimización entera no lineal. (con José Manuel Jiménez Cobano).
- Adelgazamiento del test-set (con Jorge García).
- Optimización lineal entera multiobjetivo (con Haydee Jiménez).
- Métodos heurísticos: aprovechar el esqueleto que da el test-set.

Algunas perspectivas que estamos desarrollando:

- Cálculo de todos los óptimos de problemas de optimización entera no lineal. (con José Manuel Jiménez Cobano).
- Adelgazamiento del test-set (con Jorge García).
- Optimización lineal entera multiobjetivo (con Haydee Jiménez).
- Métodos heurísticos: aprovechar el esqueleto que da el test-set.

Algunas perspectivas que estamos desarrollando:

- Cálculo de todos los óptimos de problemas de optimización entera no lineal. (con José Manuel Jiménez Cobano).
- Adelgazamiento del test-set (con Jorge García).
- Optimización lineal entera multiobjetivo (con Haydee Jiménez).
- Métodos heurísticos: aprovechar el esqueleto que da el test-set.

(1) Cálculo de todos los óptimos

Inspirado en **[Tsai, Jung-Fa; Lin, Ming-Hua; Hu, Yi-Chung. EJOR, 2008]** y otros (p.ej. problemas multiobjetivo de tipo portfolio).

Tardamos poco en encontrar el valor óptimo **c**, a veces mucho en certificarlo. Pero certificarlo equivale a encontrar todos los puntos en los que el óptimo se alcanza.

Los solvers tardan más que nosotros en encontrar todos los puntos que producen valor óptimo de la función de coste.

(1) Cálculo de todos los óptimos

Inspirado en **[Tsai, Jung-Fa; Lin, Ming-Hua; Hu, Yi-Chung. EJOR, 2008]** y otros (p.ej. problemas multiobjetivo de tipo portfolio).

Tardamos poco en encontrar el valor óptimo **c**, a veces mucho en certificarlo. Pero certificarlo equivale a encontrar todos los puntos en los que el óptimo se alcanza.

Los solvers tardan más que nosotros en encontrar todos los puntos que producen valor óptimo de la función de coste.

(1) Cálculo de todos los óptimos

Inspirado en **[Tsai, Jung-Fa; Lin, Ming-Hua; Hu, Yi-Chung. EJOR, 2008]** y otros (p.ej. problemas multiobjetivo de tipo portfolio).

Tardamos poco en encontrar el valor óptimo **c**, a veces mucho en certificarlo. Pero certificarlo equivale a encontrar todos los puntos en los que el óptimo se alcanza.

Los solvers tardan más que nosotros en encontrar todos los puntos que producen valor óptimo de la función de coste.

(2) Adelgazamiento del test-set

En **[Urbaniak, Regina; Weismantel, Robert; Ziegler, Günter M. SIAM J. Discrete Math., 1997]** aparece un algoritmo para calcular test-sets de (P) **para un b fijo**. Esto produce a veces un test-set sutancialmente más pequeño que el test-set general y, por tanto, un árbol de búsqueda para hacer walk back con muchos menos puntos que recorrer.

El test-set particular se obtiene depurando un test-set universal que sirve para toda función de coste.

Problema interesante: ¿Cuándo coinciden?

En **[Urbaniak, Regina; Weismantel, Robert; Ziegler, Günter M. SIAM J. Discrete Math., 1997]** aparece un algoritmo para calcular test-sets de (P) **para un b fijo**. Esto produce a veces un test-set sutancialmente más pequeño que el test-set general y, por tanto, un árbol de búsqueda para hacer walk back con muchos menos puntos que recorrer.

El test-set particular se obtiene depurando un test-set universal que sirve para toda función de coste.

Problema interesante: ¿Cuándo coinciden?

(2) Adelgazamiento del test-set

En **[Urbaniak, Regina; Weismantel, Robert; Ziegler, Günter M. SIAM J. Discrete Math., 1997]** aparece un algoritmo para calcular test-sets de (P) **para un b fijo**. Esto produce a veces un test-set sutancialmente más pequeño que el test-set general y, por tanto, un árbol de búsqueda para hacer walk back con muchos menos puntos que recorrer.

El test-set particular se obtiene depurando un test-set universal que sirve para toda función de coste.

Problema interesante: ¿Cuándo coinciden?

En **[Blanco, Víctor; Puerto, Justo. SIAM J. Discrete Math., 2009]** se presentó el concepto de *base de Gröbner parcial* para problemas multiobjetivo:

$$\begin{array}{rl} \min & (\mathbf{c}_1(\mathbf{x}), \ldots, \mathbf{c}_p(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

La base parcial sirve para construir todo el conjunto de soluciones Pareto-óptimas de manera exacta. Desgraciadamente, puede coincide en muchos problemas la base de Graver, que después hay que ordenar en cadenas maximales (según el orden parcial que de manera natural definen las funciones objetivo).
Tiempos

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \end{array} \right.$$

(3) ϵ-restricciones: una alternativa clásica [Marglin, 1967]

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \end{array} \right.$$

(3) ϵ-restricciones: una alternativa clásica [Marglin, 1967]

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \end{array} \right.$$

(3) ϵ-restricciones: una alternativa clásica [Marglin, 1967]

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \end{array} \right.$$

(3) ϵ-restricciones: una alternativa clásica [Marglin, 1967]

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \end{array} \right.$$

(3) ϵ-restricciones: una alternativa clásica [Marglin, 1967]

Partimos del problema multiobjetivo:

$$\begin{array}{ll} \min & f(\mathbf{x}) := (f_1(\mathbf{x}), \dots, f_p(\mathbf{x})) \\ \text{s.t.} & \mathbf{x} \in S \end{array}$$

El método consiste en:

- Elegir arbitrariamente un objetivo f_k para minimizar, y convertir los $p-1$ objetivos restantes en restricciones, que se añaden a las que ya teníamos.
- Para ello, hay que elegir una cota ε_j asociada a cada objetivo $f_j(\mathbf{x})$ que se transforma en restricción: $\boxed{f_j(\mathbf{x}) \leq \varepsilon_j}$.
- *Si se conoce* el rango de variación del objetivo f_j, entonces la cota se elige dentro de él: $\varepsilon_j \in [f_j^{min}, f_j^{max}]$.

El problema resultante es:

$$(P_k(\varepsilon)) \left\{ \begin{array}{ll} \min & f_k(\mathbf{x}) \\ \text{s.t.} & \mathbf{x} \in S \\ & f_j(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \dots, p,\ j \neq k \end{array} \right.$$

ϵ-restricciones con test-set

Esta estrategia se ha utilizado en programación lineal entera (p.ej en **[Mavrotas, George. Appl. Math. Comput., 2009]**.

Hacerlo con test-sets tiene algunas ventajas potenciales. La idea central es observar que el problema inicial

$$\begin{array}{rl} \min & (\mathbf{c_1}(\mathbf{x}), \ldots, \mathbf{c}_p(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

Se reduce a múltiples problemas

$$\begin{array}{rl} \min & \mathbf{c_k}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{c_j}(\mathbf{x}) \leq \varepsilon_j,\ j = 1, \ldots, p,\ j \neq k \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

todos con el mismo test-set. Una vez calculado, resolver cada uno es muy barato.

Esta estrategia se ha utilizado en programación lineal entera (p.ej en **[Mavrotas, George. Appl. Math. Comput., 2009]**.
Hacerlo con test-sets tiene algunas ventajas potenciales. La idea central es observar que el problema inicial

$$\begin{array}{rl} \min & (\mathbf{c_1}(\mathbf{x}), \ldots, \mathbf{c}_p(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

Se reduce a múltiples problemas

$$\begin{array}{rl} \min & \mathbf{c_k}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{c_j}(\mathbf{x}) \leq \varepsilon_j, \; j = 1, \ldots, p, \; j \neq k \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

todos con el mismo test-set. Una vez calculado, resolver cada uno es muy barato.

ϵ-restricciones con test-set: obteniendo todos los Pareto-óptimos

De hecho, se puede utilizar esta idea para calcular todo el conjunto de soluciones Pareto-óptimas de manera exacta.

Para el caso biobjetivo, por ejemplo:

$$\begin{array}{rl} \min & (\mathbf{c_1}(\mathbf{x}), \mathbf{c}_2(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

Resolvemos primero

$$\begin{array}{rl} \min & \mathbf{c_1}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

y a partir del óptimo $P_1^\star$ de coste $c_1^\star$ podemos conseguir todos los Pareto-óptimos resolviendo los problemas

$$\begin{array}{rl} \min & \mathbf{c_k}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{c_1}(\mathbf{x}) \leq c_1^\star + k_1, \;\; k_1 \in \mathbf{N} \\ & \mathbf{x} \in \mathbf{Z}^n \end{array}$$

ϵ-restricciones con test-set: obteniendo todos los Pareto-óptimos

De hecho, se puede utilizar esta idea para calcular todo el conjunto de soluciones Pareto-óptimas de manera exacta.
Para el caso biobjetivo, por ejemplo:

$$\begin{array}{rl} \min & (\mathbf{c_1}(\mathbf{x}), \mathbf{c}_2(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

Resolvemos primero

$$\begin{array}{rl} \min & \mathbf{c_1}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}_+^n \end{array}$$

y a partir del óptimo $P_1^\star$ de coste $c_1^\star$ podemos conseguir todos los Pareto-óptimos resolviendo los problemas

$$\begin{array}{rl} \min & \mathbf{c_k}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{c_1}(\mathbf{x}) \leq c_1^\star + k_1, \ \ k_1 \in \mathbf{N} \\ & \mathbf{x} \in \mathbf{Z}^n \end{array}$$

ϵ-restricciones con test-set: obteniendo todos los Pareto-óptimos

De hecho, se puede utilizar esta idea para calcular todo el conjunto de soluciones Pareto-óptimas de manera exacta.
Para el caso biobjetivo, por ejemplo:

$$\begin{array}{rl} \min & (\mathbf{c_1}(\mathbf{x}), \mathbf{c}_2(\mathbf{x})) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}^n_+ \end{array}$$

Resolvemos primero

$$\begin{array}{rl} \min & \mathbf{c_1}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{x} \in \mathbf{Z}^n_+ \end{array}$$

y a partir del óptimo $P_1^\star$ de coste $c_1^\star$ podemos conseguir todos los Pareto-óptimos resolviendo los problemas

$$\begin{array}{rl} \min & \mathbf{c_k}(\mathbf{x}) \\ \text{s.t.} & A\mathbf{x} \in= \mathbf{b} \\ & \mathbf{c_1}(\mathbf{x}) \leq c_1^\star + k_1, \;\; k_1 \in \mathbf{N} \\ & \mathbf{x} \in \mathbf{Z}^n \end{array}$$