Programación entera

Áurea Martínez

Programación lineal mixta (MILP)

Un problema de programación lineal mixta se puede escribir de manera general como:

$$\min c^T x + d^T y$$

$$\textit{sujeto a}$$

$$Ax + By \leq 0$$

$$x \in R^n$$

$$y \in Z^m$$

Los métodos para resolver problemas de programación lineal mixta (MILP) están basados en las técnicas de **ramificación y acotación** (Branch and bound) y sus variantes, donde cada subproblema lineal se resuelve usando el método del simplex.

Método de ramificación y acotación (Branch and bound):

Consiste en una enumeración en árbol en el cual el espacio de las variables enteras se divide de forma sucesiva dando lugar a problemas lineales que se resuelven en cada nodo del árbol. Estos problemas lineales se obtienen relajando las restricciones de integralidad y añadiendo restricciones adicionales.

- El procedimiento de ramificación y acotación establece inicialmente una **cota inferior** y una **cota superior** del valor óptimo de la función objetivo.

- El mecanismo de ramificación aumenta progresivamente el valor de la cota inferior y disminuye progresivamente el valor de la cota superior.

- La diferencia entre estas dos cotas es una medida de la proximidad del punto actual a la solución optima, si ésta existe.

Algoritmo de ramificación y acotación

A. **Inicio:**

- Se establece una cota superior y una cota inferior de la solución óptima.
- Se resuelve el MILP relajando las variables enteras como variables continuas.
- Si el problema relajado es no factible, el original también lo es y no hay solución.
- Si la solución obtenida satisface las restricciones de integralidad, es la solución óptima del problema original.
- En otro caso, se actualiza el valor de la cota inferior con el valor óptimo de la función objetivo del problema relajado.

B. Ramificación:

Empleando la variable x_k que ha de ser entera y no lo es, se generan mediante ramificación dos problemas. Si $x_k = a.b$, donde *a* es la parte entera y *b* su parte decimal, los problemas fruto de la ramificación son los siguientes:

- MILP inicial relajado con la restricción $x_k \leq a$
- MILP inicial relajado con la restricción $x_k \geq a+1$

Estos problemas se colocan en una lista de problemas a procesar que pueden resolverse secuencialmente o en paralelo.

C. **Solución:**

Se resuelve el problema siguiente en la lista de problemas a procesar.

D. **Actualización de cotas:**

- Si la solución del problema actual **satisface** las condiciones de integralidad y el valor óptimo de su función objetivo es menor que la cota superior actual, se actualiza la cota superior con este valor óptimo y la solución óptima del problema actual se almacena como el mejor candidato a minimizador del problema original.

- Si la solución del problema actual **no satisface** las condiciones de integralidad y el valor de la correspondiente función objetivo está entre las cotas inferior y superior, se actualiza el valor de la cota inferior con este valor óptimo, y se procede a ramificar.

E. Acotación:

- Si la solución del problema actual **cumple las restricciones de integralidad**, no tienen sentido ramificaciones adicionales relacionadas con esa solución. Se dice que la rama se poda por razones de integralidad.

- Si la solución del problema actual **no cumple las restricciones de integralidad**, y además el valor de la función objetivo correspondiente es mayor que la cota superior, no se pueden obtener soluciones mediante ramificaciones adicionales de esta rama. Se dice que la rama se poda por cotas.

- Si el problema es **no factible**, no tienen sentido ramificaciones en ese nodo. Se dice que la rama se poda por infactibilidad.

F. Optimalidad:

- Si la lista de problemas a procesar no está vacía, se continúa con el paso C.

- Si la lista de problemas a procesar está vacía, el procedimiento concluye:

  - Si existe un candidato a minimizador, este candidato es la solución óptima.
  - Si no existe candidato a minimizador, el problema es no factible.

OBSERVACIONES:

Existen muchas variantes del algoritmo básico. Las grandes cuestiones que diferencian los distintos algoritmos de la familia *Branch and bound* son:

- ¿Por qué nodo debe continuarse la búsqueda? Un criterio sencillo consiste en escoger aquel que tenga un valor de la función objetivo más bajo, en el caso de minimización. Sin embargo, existen otros criterios más sofisticados.

- En cada nodo, ¿debe bifurcarse según una única variable? Lo más sencillo es, naturalmente, seleccionar una sola variable, pero cabe escoger varias a la vez.

- ¿Qué variable debe escogerse para ramificar en cada nodo? Normalmente se escogerá aquella que esté más cercana a tomar un valor entero. Existen, sin embargo, otros criterios.

Los algoritmos profesionales de BB tienen formas muy elaboradas para resolver cada una de estas cuestiones. Sin embargo, el conocimiento del problema, por parte del analista, puede permitirle reducir sustancialmente el tiempo necesario para calcular cuál es la solución óptima del problema.

- Cada **subproblema de programación lineal** se resuelve utilizando el **método del simplex** ya que esto puede hacerse de manera muy eficiente a partir del resultado del nodo antecesor. Esta propiedad no la comparten los métodos de puntos interiores, por lo que apenas se utilizan para resolver problemas MILP.
- En el peor de los casos el método de ramificación y acotación termina con la enumeración de todos los nodos del árbol.
- Dos desarrollos importantes han contribuido a reducir costes en la resolución de problemas MILP:
  - *Las técnicas de preprocesamiento*: eliminación automática de variables y restricciones, reducción de límites, reformulación de restricciones…
  - *Los planos de corte*: son restricciones extras añadidas al problema, que tienen el efecto de reducir la región factible del problema sin comprometer ninguna de las soluciones enteras del mismo.

Códigos para MILP

- CPLEX

http://www.ilog.com/products/cplex/

- XPRESS:

http://www.fico.com/en/Products/DMTools/xpress -overview/Pages/Xpress-and-Optimization.aspx

Programación no lineal con variables enteras y continuas (MINLP)

Un MINLP se puede formular de manera general como:

$$\min f(x)$$

$$x \in X \subset R^n$$

$$sujeto\ a\ c(x) \leq 0$$

$$x_i \in Z, para\ todo\ i \in I \subset \{1,\ldots,n\}$$

Las técnicas MINLP distinguen entre problemas convexos y no convexos.

Una diferencia importante entre MINLP y MILP es que la solución no es un punto extremo, en general estará en el interior del conjunto admisible. Para evitar esto podemos transformar el problema en la forma η-MINLP:

$$
\begin{aligned}
&\min \eta \\
&\eta \in R, x \in X \subset R^n \\
&\textit{sujeto a} \\
&\qquad\qquad f(x) \leq \eta \\
&\qquad\qquad c(x) \leq 0 \\
&x_i \in Z, \textit{para todo } i \in I \subset \{1,\ldots,n\}
\end{aligned}
$$

Herramientas básicas:

Relajación:

- Se usa para calcular una cota inferior para el óptimo.
- Se obtiene agrandando el conjunto admisible, por ejemplo, ignorando algunas restricciones.

Cumplimiento de restricciones:

- Excluye soluciones de las relajaciones que no son admisibles en el MINLP.
- Refinamiento o ajuste de la relajación, por ejemplo, añadiendo desigualdades válidas.

Cotas superiores:

- Obtenidas desde cualquier punto admisible, por ejemplo, resolviendo un problema NLP fijando las variables enteras.

Relajando la integralidad:

Se cambia:

$$x_i \in Z, \ para\ todo\ i \in I \subset \{1,\ldots,n\}$$

por:

$$x_i \in R, \ para\ todo\ i \in I \subset \{1,\ldots,n\}$$

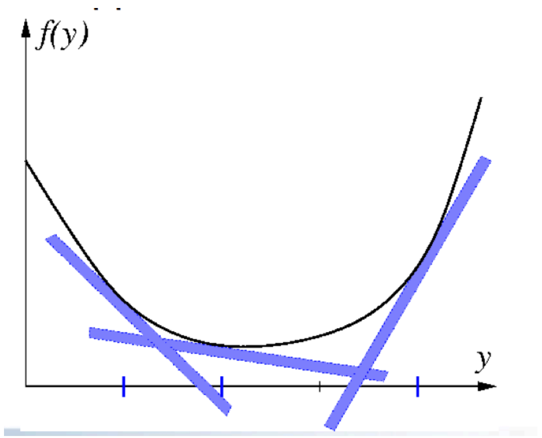
Esto conduce a un **NLP**:

$$\min f(x)$$
$$x \in X \subset R^n$$
$$sujeto\ a\ c(x) \leq 0$$

Relajación de restricciones convexas

Cuando f y c son convexas, las restricciones:

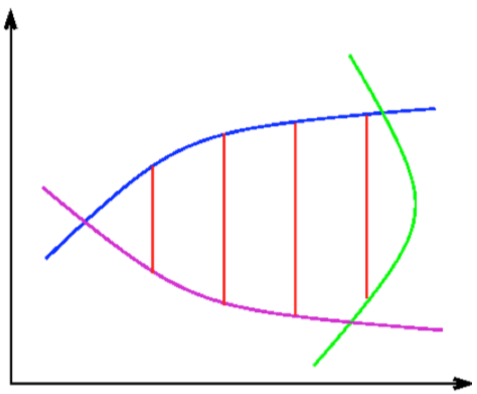
$$f(x) \le \eta$$
$$c(x) \le 0$$



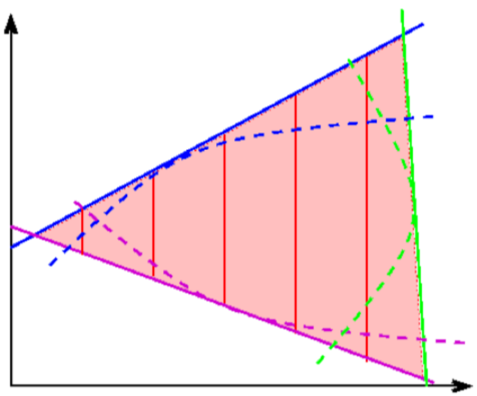
se relajan utilizando hiperplanos soporte:

$$f^{(k)} + \nabla f^{(k)T}(x - x^{(k)}) \le \eta$$
$$c^{(k)} + \nabla c^{(k)T}(x - x^{(k)}) \le 0$$

para un conjunto de puntos $x^{(k)}$, $k=1,\dots,K$.



restricciones enteras y convexas



restricciones relajadas

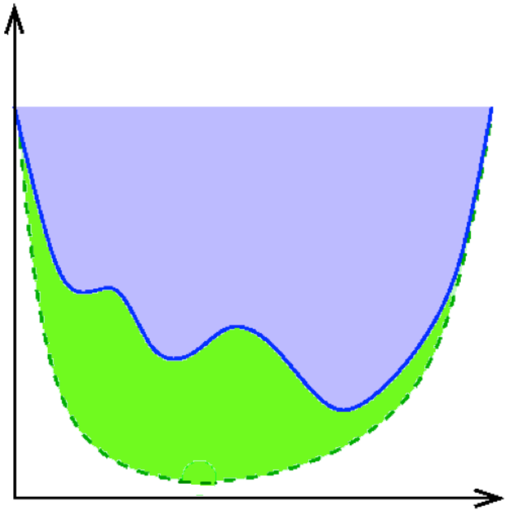
Relajación de restricciones no convexas

- Se construyen subestimadores convexos $f_s(x)$ y $c_s(x)$ para las funciones no convexas $f(x)$ y $c(x)$:

$$f_s(x) \leq f(x) \quad \text{y} \quad c_s(x) \leq c(x), \quad \forall\, x \in \text{conv}(X)$$

- Se relajan las restricciones $f(x) \leq \eta$ y $c(x) \leq 0$ como:

$$f_s(x) \leq \eta \quad \text{y} \quad c_s(x) \leq 0$$



Las relajaciones son útiles gracias al siguiente resultado:

Si la solución de la relajación del η-MINLP es admisible en el η-MINLP, entonces resuelve el MINLP

… pero si no es admisible, entonces es necesario forzar el cumplimiento de las restricciones.

Forzar el cumplimiento de las restricciones:

El objetivo es, dada una solución de relajación no admisible y en el MINLP, excluirla para asegurar la convergencia.

Se utilizan tres estrategias:

- Refinamiento de la relajación
- Ramificación
- Ramificación espacial (dividir la región en subregiones)

Refinamiento de la relajación:

Se añade una desigualdad que satisfagan todas las soluciones admisibles del MINLP.

Esa desigualdad se llama **corte** si excluye la solución no admisible ***y***.

Por ejemplo, si las restricciones son $c(x) \leq 0$ convexas,

y existe ***j*** tal que $c_j(\mathbf{y}) \geq 0$, entonces la restricción:

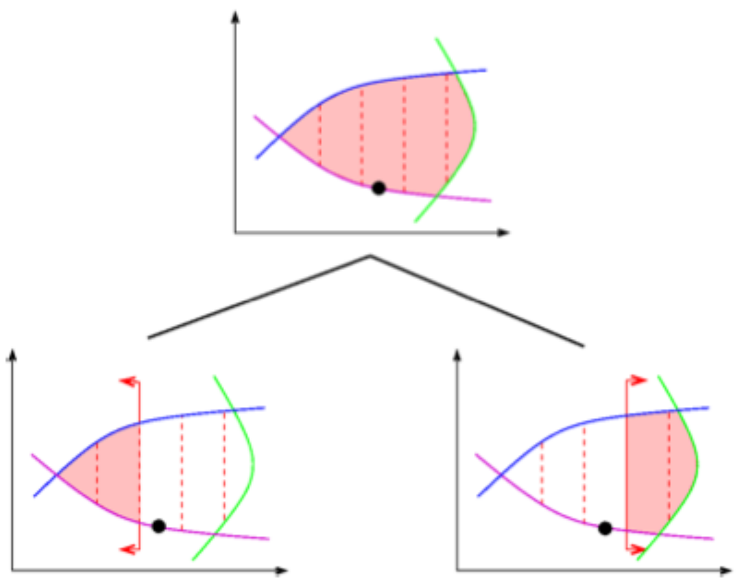
$$c_j(y) + \nabla c_j(y)^T (x - y) \leq 0$$

podará ***y*** .

Ramificación:

Eliminar la solución no admisible y ramificando en las variables enteras:

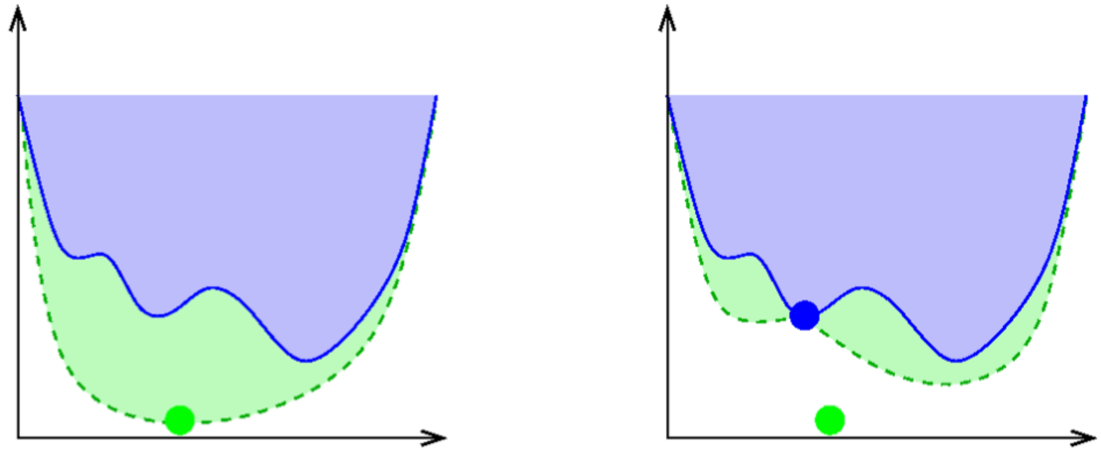
- Se elige la variable y_j que ha de ser entera y no lo es.
- Si $y_j = a.b$, con a parte entera y b parte decimal, se crean dos nuevas relajaciones añadiendo la restricción $y_j \leq a$ y la restricción $y_j \geq a+1$



Ramificación espacial:

Fuerza el cumplimiento de las restricciones no convexas relajadas:

- Combinar la ramificación y el refinamiento de la relajación
- Ramificar sobre las variables continua y dividir el dominio en dos
- Crear una nueva relajación sobre subdominios reducidos
- Mezclar con técnicas de intervalos para eliminar subdominios



Principales métodos para resolver problemas MINLP

- ***Ramificación y acotación*: ** extensión directa de los métodos de ramificación y acotación empleados para resolver MILP, con la diferencia de que *en cada nodo* se debe resolver un problema de *programación no lineal* y la resolución de los diferentes problemas NLP a partir de su inmediato antecesor ya no es tan eficiente como en el caso lineal.
- ***Métodos de descomposición*** que iteran entre dos subproblemas. El primero es un problema de programación no lineal con valores fijos de las variables enteras cuya solución es una *cota superior* de la solución óptima del problema. El segundo es un problema maestro (en general un problema MILP) cuya solución da una *cota inferior* de la solución óptima del problema. Los más importantes son la **descomposición de Benders generalizada** y el **método de aproximaciones exteriores**.
- ***LP-NLP*:** basado en ramificación y acotación, es un método intermedio entre los dos anteriores. En este caso en lugar de resolver el problema maestro hasta la optimalidad, tan pronto como se encuentra un iterante donde todas las variables discretas toman valores enteros, se resuelve un NLP y se actualizan los nodos abiertos del árbol de búsqueda.
- ***Método de plano de corte extendido***, es una variación del anterior que no requiere la solución de NLPs.

Ramificación y acotación:

El método comienza resolviendo el problema relajado:

$$\min f(x)$$
$$x \in X \subset R^n$$
$$sujeto\ a\ c(x) \leq 0$$

- Si x_i es entero para todo índice i de I, entonces está resuelto el MINLP.
- Si la relajación es inadmisible, entonces el MINLP es inadmisible.
- En otro caso resolvemos el NLP(l,u):

$$\min f(x)$$
$$x \in X \subset R^n$$
$$sujeto\ a\ c(x) \leq 0$$
$$l_i \leq x_i \leq u_i, para\ todo\ i \in I$$

Si la solución x' del NLP(l,u) es admisible pero no cumple las restricciones de integrabilidad:

Se elige una variable no entera, $x_i' = \boldsymbol{a.b}$ ($\boldsymbol{a}$ parte entera, $\boldsymbol{b}$ parte decimal).

Se introducen dos nodos correspondientes a las cotas:

$$(l^-,u^-) := (l^+,u^+) := (l,u)$$

donde se cambian:

$$u_i^- := \boldsymbol{a} \qquad \text{y} \qquad l_i^+ := \boldsymbol{a+1}$$

Se resuelven entonces los dos nuevos NLPs: NLP(l^-,u^-) y NLP(l^+,u^+) que dan lugar a dos nuevas ramas del árbol. Estos problemas producen cotas inferiores para los subproblemas de los nodos descendientes. La eliminación de un nodo y los descendientes ocurre cuando la cota inferior sobrepasa a la actual cota superior, cuando el subproblema es no factible o cuando todas las variables cumplen las restricciones de integrabilidad. En este caso se tiene una cota superior para la solución del problema.

PUEDE SER MUY LENTO !!

Aproximaciones exteriores (OA):

- El método comienza resolviendo un problema de relajación continua o bien con un conjunto fijado de variables discretas y se obtiene un iterante $x^{(1)}$, K=1.
- La solución óptima de este problema se utiliza para generar el primer problema maestro:

$$\min \eta$$

$$\eta \in R, x \in X \subset R^n$$

$$sujeto\ a$$

$$f^{(k)} + \nabla f^{(k)T}(x - x^{(k)}) \le \eta$$

$$c^{(k)} + \nabla c^{(k)T}(x - x^{(k)}) \le 0$$

$$para \quad x^{(k)}, k = 1, \ldots K$$

$$x_i \in Z, para\ todo\ i \in I \subset \{1, \ldots, n\}$$

- La solución de ***y*** este problema maestro genera un nuevo conjunto de variables discretas: $\boldsymbol{y_i}$ para $\boldsymbol{i}$ en $\boldsymbol{I}$.

- Este nuevo conjunto de variables discretas se utiliza para resolver un nuevo problema (PNL($\boldsymbol{y}$)) con las variables discretas fijadas:

$$
\begin{aligned}
&\min f(x) \\
&x \in X \subset R^n \\
&\textit{sujeto a } c(x) \leq 0 \\
&x_i = y_i, \quad \textit{para todo } i \in I
\end{aligned}
$$

- Si este problema no es factible, se resuelve el problema de factibilidad (PNLF($\boldsymbol{y}$)) :

$$
\begin{aligned}
&\min \alpha \\
&\alpha \in R, x \in X \subset R^n \\
&\textit{sujeto a } c(x) \leq \alpha \\
&x_i = \mathrm{y}_i, \textit{para todo } i \in I
\end{aligned}
$$

- Las linealizaciones de la solución de este problema se añaden al problema maestro continuando así de forma iterativa.

- El algoritmo se para cuando la diferencia entre la mejor solución y la peor solución almacenadas para el problema maestro, está dentro de una tolerancia especificada.

Descomposición de Benders Generalizada (GDB):

- Es similar al método de las aproximaciones exteriores. Las diferencias están en la definición del problema maestro.

$$\min \eta$$

$$\eta \in R, x \in X \subset R^n$$

$$sujeto\ a$$

$$f^{(k)} + \mu^{(k)T}(x_I - x_I^{(k)}) \leq \eta$$

$$para \qquad x^{(k)}, k = 1, \ldots K$$

$$x_I = (x_i)_{i \in I \subset \{1,\ldots,n\}} \text{ variables enteras}$$

donde $\mu^{(k)}$ es el multiplicador asociado a las restriccion $\boldsymbol{x_I}=\boldsymbol{x_I^{(k)}}$ en las condiciones KKT de PNL($\boldsymbol{x^{(k)}}$)

LP/NLP basado en ramificación y acotación (LP/NLP-BB):

- El método comienza resolviendo un subproblema NLP inicial, el cual se linealiza como en el problema maestro de OA.

- La idea básica consiste en utilizar una búsqueda por ramificación y acotación, resolviendo problemas de programación lineal, en cada rama del árbol. En cuanto un nodo presenta una solución $\boldsymbol{y}$ con valores enteros, se resuelve un problema (PNL($\boldsymbol{y}$)) y se actualiza la representación del problema maestro en todos los nodos del árbol, añadiendo las correspondientes linealizaciones.

Método del Plano de Corte Extendido (ECP):

- No resuelve problemas NLP.
- Resuelve iterativamente problemas maestro, añadiendo linealizaciones sucesivas a aquella restricción más violada en el punto predicho $\boldsymbol{x}^k$.
- La convergencia se obtiene cuando la violación máxima está dentro de una tolerancia especificada.
- Es posible añadir linealizaciones a todas las restricciones violadas o a todas las no lineales (más caro).
- La función objetivo debe ser lineal. Esto se consigue fácilmente añadiendo una nueva variable para transferir las no linealidades de la función objetivo a una desigualdad.

Caso no convexo:

- Si ***f*** o ***c*** no son convexas , los sub-problemas que aparecen pueden no tener un único mínimo local …..

- Solución posible: se desarrollan envolventes convexas o sub-estimadores que permiten formular problemas que sean límites inferiores rigurosos.

- Se utilizan técnicas de optimización global

Códigos para MINLP

- AlphaECP (ECP)

http://www.abo.fi/~twesterl/

- AIMMS/AOA (OA)

http://main.aimms.com/aimms/solvers/aoa/

- GAMS/DICOPT (OA+relajación de igualdades+penalización aumentada)

http://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Fsolvers%2Findex.html

- BARON (modificación de BB)

http://archimedes.cheme.cmu.edu/?q=baron

- MINOPT (OA,GBD)

http://titan.princeton.edu/MINOPT/

- Bonmin (código libre): (LP/NLP-BB,OA, …)

http://www.coin-or.org/Bonmin/