Métodos de Descomposición para PL

Dpto. Ingeniería Industrial, Universidad de Chile

IN47B, Ingeniería de Operaciones

Contenidos

Algoritmo generico

1- Formulate a reduced problem (RP) by
2- Solve (RP)
3- Solve subproblem to identify profitable variables
4- If some variable is profitable, add to (RP)
 - else, STOP solution of (RP) is optimal
5- Goto 2

Cadena de Supermercados

- Gran numero de variables en cada tienda:
  – espacio de repisas, inventarios, servicios (limpieza, mantención), programación de turnos
- variables que relacionan tiendas:
  – compra de productos, presupuesto, contrato laboral

Para dos tiendas

$$
\begin{array}{llllll}
(P) & z = \max & c_1^T x_1 + & c_2^T x_2 & & \\
 & \text{s.t.} & B_1 x_1 + & B_2 x_2 & = b & \\
 & & A_1 x_1 & & = b_1 & \\
 & & & A_2 x_2 & = b_2 & \\
 & & x_1, & x_2 \geq 0 & &
\end{array}
$$

Descomposicion del problema

El método de generación de columnas funciona mejor cuando $m << m_1, m_2$.
Defina $S_i := \{x \mid A_i x = b_i,\ x \geq 0\}$, $i = 1, 2$. El problema es equivalente a

$$
\begin{array}{lrllr}
(P) & z = \text{máx} & c_1^T x_1 + & c_2^T x_2 & \\
 & \text{s.t.} & B_1 x_1 + & B_2 x_2 & = b \\
 & & x_1 \in S_1 & & \\
 & & & x_2 \in S_2 &
\end{array}
$$

Descomposicion del problema

Since S_i is a polyhedron, each point $x \in S_i$ can be written by

$$x = \sum_{k \in J_i} \lambda_i^k x_i^k + \sum_{k \in R_i} \theta_i^k w_i^k \quad \text{where} \quad \sum_{k \in J_i} \lambda_i^k = 1, \; \lambda_i^k \geq 0, \; \theta_i^k \geq 0$$

where x_i^k are extreme points of S_i, and w_i^k are extreme rays of S_i.
Use this representation to construct an equivalent problem (M) which has $m + 2$ constraints

How many variables do you need?

Generación de Columnas

To try to deal with a problem with a manageable number of variables, we will only consider a small subset of extreme points and rays ($\overline{J}_i \subset J_i$ and $\overline{R}_i \subset R_i$). This leads to the restricted master problem (RM):

$$
\begin{array}{llll}
z_{RM} = & & & \\
\text{máx} & \sum_{k\in\overline{J}_1} \lambda_1^k c_1^T x_1^k + \sum_{k\in\overline{R}_1} \theta_1^k c_1^T w_1^k + & \sum_{k\in\overline{J}_2} \lambda_2^k c_2^T x_2^k + \sum_{k\in\overline{R}_2} \theta_1^k c_2^T w_2^k & \\
\text{s.t.} & \sum_{k\in\overline{J}_1} \lambda_1^k B_1 x_1^k + \sum_{k\in\overline{R}_1} \theta_1^k B_1 w_1^k + & \sum_{k\in\overline{J}_2} \lambda_2^k B_2 x_2^k + \sum_{k\in\overline{R}_2} \theta_1^k B_2 w_2^k & = b \\
& \sum_{k\in\overline{J}_1} \lambda_1^k & & = 1 \\
& & \sum_{k\in\overline{J}_2} \lambda_2^k & = 1 \\
& \lambda_1^k, \theta_1^k \geq 0 & \lambda_2^k, \theta_2^k \geq 0 &
\end{array}
$$

Generación de Columnas

Demuestre que $z_{RM} \leq z$

How do we check that the optimal solution to (RM) is also optimal for (M)? What are the reduced costs and optimality conditions? $c_j - c_B B^{-1} A_j \leq 0$, let (y, σ_1, σ_2) be optimal dual variables of (RM):

To check over all extreme points in S_i consider the problems

$$(P_1) \qquad \begin{array}{rl} u_1 = \max & (c_1 - y^T B_1)x \\ \text{s.t.} & x \in S_1 \end{array} \quad \text{and} \quad (P_2) \qquad \begin{array}{rl} u_2 = \max & (c_2 - y^T B_2)x \\ \text{s.t.} & x \in S_2 \end{array}$$

Generación de Columnas

Prop *If* $u_1 \leq \sigma_1$ *and* $u_2 \leq \sigma_2$ *then the solution* z_{RM} *is also optimal for* (M).
Moreover, if $\infty > u_1 > \sigma_1$ *then variable* λ_1^j *has a positive reduced cost* ($c_1^T x_1^j - y^T B_1 x_1^j > \sigma_1$). *And if* $u_1 = \infty$ *then variable* θ_1^j *has a positive reduced cost. (Also true for sub-problem 2)*

Dantzig-Wolfe Algorithm

1- Transform and reduce problem to (RM)
2- Solve (RM), obtain (y, σ_1, σ_2)
3- Solve (P_i), obtain u_i and optimal solution x_i^j (or w_i^j)
4- If $u_i \leq \sigma_i$ for all i
- STOP: solution of (RM) is optimal for (M)
- else if $\infty > u_i > \sigma_i$
- add x_i^j to $\overline{J}_i$
- else if $\infty = u_i$
- add w_i^j to $\overline{R}_i$
5- Goto 2

Temas pendientes

- How do we find the initial (RM) and starting basis? We solve the following Phase I problem: Generate a point $x_1^1 \in S_1$ and $x_2^1 \in S_2$

$$
\begin{array}{rllll}
u = \text{mín} & e^T\alpha & & & \\
\text{s.t.} & \lambda_1^1 B_1 x_1^1 + & \lambda_2^1 B_2 x_2^1 & + (-)\alpha & = b \\
& \lambda_1^1 & & & = 1 \\
& & \lambda_2^1 & & = 1 \\
& \lambda_1^1, \lambda_2^1, \alpha \geq 0 & & &
\end{array}
$$

  This is the starting basis to apply Dantzig-Wolfe to solve the Phase I problem. If in the optimal solution $u = 0$, then (P) is feasible (might have to do some work). If $u > 0$, then
- What do we get if we can't wait for this to converge?

Temas pendientes

Prop $z_{RM} \leq z \leq z_{RM} + \sum_{i=1}^{2} u_i - \sum_{i=1}^{2} \sigma_i$
proof: Consider (P) and its dual:

$$
(P) \quad \begin{array}{rlllr}
z = \text{máx} & c_1^T x_1 + & c_2^T x_2 & & \\
\text{s.t.} & B_1 x_1 + & B_2 x_2 & = b & \\
& A_1 x_1 & & = b_1 & \\
& & A_2 x_2 & = b_2 & \\
& x_1, & x_2 \geq 0 & &
\end{array}
$$

$$
(D) \quad \begin{array}{rllll}
z = \text{mín} & b^T s + & b_1 \pi_1 + & b_2^T \pi_2 \\
\text{s.t.} & B_1^T s + & A_1^T \pi_1 & \\
& B_2^T s + & & A_2^T \pi_2
\end{array}
$$

and (P_i) and their dual:

$$
(P_i) \quad \begin{array}{rl}
u_i = \text{máx} & (c_i - B_i^T y)^T x \\
\text{s.t.} & A_i x = b_i \\
& x \geq 0
\end{array}
\qquad
(D_i) \quad \begin{array}{rl}
\text{mín} & b_i^T h \\
\text{s.t.} & A_i^T h \geq c_i - B_i^T y
\end{array}
$$

Muchas resticciones

Consider the problem in n variables, with $n << m$:

$$\begin{array}{ll} \max & c^T x \\ \text{s.t.} & a_i^T x \leq b_i \quad i = 1 \ldots m \end{array}$$

Propose a column generation type algorithm that exploits the problem structure to solve this problem efficiently

Descomponiendo directamente

Definimos un problema restringido usando solamente $k < m$ restricciones

$$\begin{array}{ll} \text{máx} & c^T x \\ \text{s.t.} & a_i^T x \leq b_i \quad i = 1 \ldots k \,. \end{array}$$

- If x^* optimal for (RM) is feasible for $i = 1 \ldots m$ then
- Otherwise need to identify a violated inequality

Aqui generación de columnas en el dual = planos cortantes en el primal
Cotas superior e inferior?

Una Compañia de Electricidad

Como satisfacer demanda a costo minimo. In the case of a thermal plant and a hydro plant, satisfying the demand over the next two periods can be written as:

$$
\begin{aligned}
\text{mín} \quad & 3x_1 + 3x_2 \\
& x_1 + h_1 \geq 10 \\
& x_2 + h_2 \geq 12 \\
& h_1 \leq 5 \\
& h_2 \leq V_2 \\
& V_2 + h_1 = 5 + r \\
& x_i, h_i \geq 0
\end{aligned}
$$

Note that this is a production and inventory problem.

Una Compañia de Electricidad

Suppose
- 2nd period demand can be 15 or 10 each with prob. 1/2
- 2nd period thermal cost can be 1 or 5 each with prob. 1/2
- rain can be $r = 0$ or $r = 10$ each with prob. 1/2

Question: Best strategy to satisfy 1st period demand considering uncertainty?

scen.	prob.	2^o dem	costo term.	lluvia	best strategy
1	0.125	15	5	0	save water, x_1 high
2	0.125	10	3	10	use water, x_1 low
⋮	⋮				

Una Compañia de Electricidad

Solution: Minimize the expected value: $\sum_{i=1}^{8} p_i z_i$. The z_i is the optimal solution for each scenario. Note that some variables have to be decided before the uncertainty (x_1 and h_1) and some after the uncertainty (x_2 and h_2). *Recourse*.

Esto genera un problema:

Formulaciones y notación

Problems for which Bender's Decomposition (constraint generation) methods work best, are those that have a large number of constraints and the following structure

$$
\begin{array}{llllll}
\min\limits_{x,y_1,\ldots,y_k} & c^T x + & f^T y_1 & \ldots & f^T y_k & \\
\text{s.t.} & Ax & & & & = b \\
& B_1 x & Dy_1 & & & = d_1 \\
& B_2 x & & Dy_2 & & = d_2 \\
& \vdots & & \ddots & & = \vdots \\
& B_k x & & & Dy_k & = d_k \\
& x,\ y_1,\ \ldots,\ y_k \geq 0 & & & &
\end{array}
$$

Formulaciones y notación

This structure is exploited by doing each minimization separately:

$$\begin{array}{ll} \min_x & c^T x + \sum_{i=1}^{k} \phi_i(x) \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$

donde

$$\phi_i(x) = \begin{array}{ll} \min_{y_i} & f^T y_i \\ \text{s.t.} & Dy_i = d_i - B_i x \\ & y_i \geq 0 \end{array} \qquad \text{para todo } i = 1, \ldots, k.$$

Formulaciones y notación

What does the function $\phi_i(x)$ look like?
The functions $\phi_i(x)$ are constructed sequentially through cuts:

- Optimality cut: If $\gamma \geq \phi_i(x)$ then
- Feasibility cut: If x makes some subproblem infeasible, $\phi_i(x) = \infty$, then

The master problem that is solved is

$$\begin{array}{ll} \min_x & c^T x + \sum_{i=1}^{k} \gamma_i \\ \text{s.t.} & Ax = b \\ & x \geq 0 \\ & d_i^T z^k - x^T B_i z^k \leq \gamma_i \ \text{ for } z^k \text{ BFS of } D^T z \leq f \\ & d_i^T w^k - x^T B_i w^k \leq 0 \ \text{ for } w^k \text{ extreme ray of } D^T z \leq f \end{array}$$

Ojo que D y f pueden variar por escenario tambien.

Algoritmo

1- Formulate a master problem (i.e. find a BFS for $D^T z \leq f$)
2- Obtain (x^*, γ^*) the optimal solution for the master problem
3- Solve the subproblem $\phi_i(x^*)$ for every scenario i
4- If all subproblems have optimal solution $\phi_i(x^*) \leq \gamma_i^*$
- STOP: (x^*, γ^*) is optimal as it is feasible for all cuts
- else if some i has $\infty > \phi_i(x^*) > \gamma_i^*$
- add optimality cut $d_i^T z^k - x^T B_i z^k \leq \gamma_i$
- else if some i has $\phi_i(x^*) = \infty$
- add feasibility cut $d_i^T w^k - x^T B_i w^k \leq 0$
5- Goto 2