Relajación Lagrangiana

Dpto. Ingeniería Industrial, Universidad de Chile

IN47B, Ingeniería de Operaciones

Contenidos

1. Problema de CMR
2. Relajación Lagrangiana
3. Flujos Multicommodity

Definición

Dado $G = (N, A)$, cada $(i, j) \in A$ tiene c_{ij} y t_{ij}.
Queremos llegar de s a t tal que minimizamos el costo y el tiempo no es muy alto.

- Camino Minimo: facil!
- Camino Minimo Restringido: NP-hard

Por ejemplo, considere el siguiente problema:

Ejemplo

¿Que pasa si cobramos un peaje de $1 por unidad de tiempo en cada arco?

- ¿y si cobramos 0.1?
- ¿y si cobramos $10?
- ¿ Cual es la solucion correcta?

El problema de optimización de camino mínimo restringido es:

$$\begin{aligned} \min \quad & c^T x \\ & Nx = b \\ & t^T x \leq T \\ & x \in \{0,1\}^{|A|} \end{aligned}$$

Simplificar el Problema

La idea de relajación Lagrangiana es remover restricciones que hacen que el problema sea dificil. Se cobra por violar estas restricciones.

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b \\ & v^T x \leq g \end{array} \quad \text{se convierte} \quad \begin{array}{ll} \min & c^T x + 2(v^T x - g) \\ \text{s.t.} & Ax = b \end{array}$$

Para Camino Mínimo Restrigido: relajar la restriccion de tiempo... deja un simple camino minimo.

Ejemplo: Generalized Assignment Problem

Dados trabajos I y maquinas J, con c_{ij} el costo lineal de asignar trabajo i a maquina j. Determine la asignación de trabajos a maquinas de minimo costo.
Este problema se escribe como:

$$
\begin{array}{llll}
\min & \sum_{i\in I}\sum_{i\in J} c_{ij}x_{ij} & & (a) \\
\text{s.t} & \sum_{j\in J} x_{ij} = 1 & \text{for all } i \in I & (b) \\
 & \sum_{i\in I} a_{ij}x_{ij} \leq d_j & \text{for all } j \in J & (c) \\
 & x_{ij} \geq 0, \ \text{integer} & \text{for all } (i,j) \in A & (d)
\end{array}
$$

- ¿Que pasa si relajamos (c)?
- ¿Que pasa si relajamos (b)?

¿Como se compara la relajación lineal con la relajación Lagrangiana?

Definición

Dado costos μ (multiplicador Lagrangiano) definimos el Problema Lagrangiano, donde $L(\mu)$ denota su valor optimo, e.g.

$$
\begin{array}{rll}
z_* = & \min & c^T x \\
& \text{s.t.} & Ax = b \\
& & M_1 x \leq g_1 \\
& & M_2 x \geq g_2 \\
& & M_3 x = g_3
\end{array}
\quad \Rightarrow \quad
\begin{array}{l}
L((\mu_1, \mu_2, \mu_3)) = \\
\quad \min \;\; c^T x + \sum_{i=1}^{3} \mu_i^T (M_i x - g_i) \\
\quad \text{s.t.} \quad Ax = b \\
\text{con } \mu_1 \geq 0,\; \mu_2 \leq 0,\; \mu_3 \text{ unconstrained}
\end{array}
$$

El problema Lagrangeano Dual es escojer el costo μ que maximiza $L(\mu)$:

$$L^* = \max_{\mu \geq 0} L(\mu) \; .$$

Graficamente

Para el problema de camino minimo restrigido

- Dibuje el costo de cada camino como funcion del peaje
- Encuentre el Lagrangeano Dual
- ¿Ha resuelto el problema original?

Esto significa que $P = NP$??

Un poco de teoria

Proposition Considere $\min c^T x \mid Ax = b,\ Mx \leq g$, el problema Lagrangeano obtenido al relajar $Mx \leq g$ para todo multiplicador $\mu \geq 0$ satisface $L(\mu) \leq z_*$

Theorema Si $\mu \geq 0$ y $x = \mathrm{argmin} L(\mu)$ es tal que $Mx \leq g$ y $\mu^T (Mx - g) = 0$, entonces $L^* = z_*$

Ejercicio: El PL dual

Derive el PL dual usando el Dual Lagrangeano

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Motivación

Del mono pareciera que se puede resolver el dual Lagrangeano

$$\max_{\mu \geq 0} L(\mu) = \begin{array}{ll} \min & c^T x + \mu^T (Mx - g) \\ \text{s.t.} & Ax = b \end{array}$$

escogiendo la menor entre funciones lineales de μ. Por ejemplo si I representa el conjunto de puntos extremos de $Ax = b$

$$L_I = \begin{array}{lll} \max_{\mu, v} & v & \\ \text{s.t.} & v \leq c^T x^k + \mu^T (Mx^k - g) & \forall\, x^k \in I \\ & \mu \geq 0 & \end{array}$$

nos daria $L_I = L^*$.

Primer Algoritmo

Construir L_I gradualmente:

Set $k = 0$

Let x^0 a solution to problem $L(0)$

Set $I^k = \{x^0\}$, and $\mu^k = \text{argmax}|_\mu L_{I^k}$

while $L_{I^k} > L(\mu^k)$ do Let $k = k + 1$

Let $x^k = \text{argmin} L(\mu^k)$

Set $I^k = I^{k-1} \cup \{x^k\}$

Set $\mu^k = \text{argmax}|_\mu L_{I^k}$

endwhile

Algoritmo 2

Algoritmo de optimización non-lineal para optimizar máx $L(\mu)$.

Metodo iterativo en que $\mu^{k+1} = \mu^k + \theta_k \nabla L(\mu^k)$

Set μ^0, $k = 0$
Let x^0 a solution to problem $L(\mu^0)$
Set $\mu^1 = \mu^0 + \theta_0(Ax^0 - b)$
while $|\mu^{k+1} - \mu^k| > \mathrm{TOL}$ Let $k = k + 1$
  Let x^k a solution to problem $L(\mu^k)$
  Set $\mu^{k+1} = \mu^k + \theta_k(Ax^k - b)$
endwhile

Converge si $\theta_k \to 0$ y $\sum_{k=1}^{n} \theta_k \to \infty$.

Flujos Multicommodity: Definition

Una generalisación del flujo de costo minimo.
Hay materiales/cosas diferentes viajando por la red.
Por ejemplo: La red representas una linea de producción y los distintos flujos son los distintos productos.
El modelo matematico es

$$\begin{aligned} \text{mín} \quad & \sum_{(i,j)\in A} \sum_{1\leq k\leq K} c_{ij}^k x_{ij}^k \\ \text{s.t.} \quad & Nx^k = b^k \\ & \sum_{1\leq k\leq K} x_{ij}^k \leq u_{ij} \\ & 0 \leq x_{ij}^k \leq u_{ij}^k \end{aligned}$$

Flujos Multicommodity: Definition

Commodities pueden identificar

- distintos pares origen destino
- todo el flujo que emana de una fuente
- vehiculos con distinta capacidad

- Homogeneous goods
- Solucion puede ser no entera
- ¿Que es lo especial en este problema?

Example: Airline Scheduling

Una aerolina tiene que decidir que aviones vuelan en que rutas en su itinerario. Tomando en cuenta que distintos aviones pueden tener distintas capacidades modele esto como un problema de flujo multicommodity.
Si la solución puede ser no entera... ¿De que sirve?

Relajación Lagrangiana

Relajamos las capacidades mixtas:

$$\begin{aligned} \min \quad & \sum_{(i,j)\in A} \sum_{1\leq k\leq K} c_{ij}^k x_{ij}^k + \sum_{(i,j)\in A} \mu_{ij} \left(\sum_{1\leq k\leq K} x_{ij}^k - u_{ij} \right) \\ \text{s.t.} \quad & N^k x^k = b^k \\ & 0 \leq x_{ij}^k \leq u_{ij}^k \end{aligned}$$

¿Que obtenemos?

Proposition Para PL $L^* = z_*$

Condiciones de Optimalidad

Obtenemos condiciones de optimalidad para flujos multicommodity de la dualidad de PL. Consider $u_{ij}^k = +\infty$, esto nos da:

La condicion de optimalidad basicas son

- Primal feasibility
- Dual feasibility/non-negative reduced costs
- Same objective function/complementary slackness

Formulación por caminos

Por simplicidad consideremos que cada commodity k tiene un unico origen s^k y destino t^k, con demanda d^k, y que $u_{ij}^k = +\infty$.
Defina

- P^k cjto de caminos dirigidos de s^k a t^k en $G = (N, A)$.
- Variables de decision: $f(P)$ flujo en camino P.
- $\delta_{ij}(P)$ indicador si arco (i, j) esta en P.
- $c^k(P) = \sum_{(i,j) \in A} c_{ij}^k \delta_{ij}(P)$

Problema y Condiciones de optimalidad

$$
\begin{aligned}
\text{mín} \quad & \sum_{1 \leq k \leq K} \sum_{P \in P^k} c^k(P) f(P) \\
\text{s.t.} \quad & \sum_{1 \leq k \leq K} \sum_{P \in P^k} \delta_{ij}(P) f(P) \leq u_{ij} \\
& \sum_{P \in P^k} f(P) = d^k \\
& f(P) \geq 0
\end{aligned}
$$

- Primal feasibility
- Dual feasibility/non-negative reduced costs
- Same objective function/complementary slackness

Algoritmo de generación de Columnas

We solve the problem considering only a subset $Q_i^k \subset P^k$ of possible paths at each iteration. We add paths (solutions) to Q_i^k as needed, by solving one subproblem per commodity.

The master problem at iteration i will be

$$
\begin{aligned}
\text{mín} \quad & \sum_{1 \leq k \leq K} \sum_{P \in Q_i^k} c^k(P) f(P) \\
\text{s.t.} \quad & \sum_{1 \leq k \leq K} \sum_{P \in Q_i^k} \delta_{ij}(P) f(P) \leq u_{ij} \\
& \sum_{P \in Q_i^k} f(P) = d^k \\
& f(P) \geq 0
\end{aligned}
$$

Algoritmo de generación de Columnas

for each commodity k, we solve a subproblem

$$\min_{P \in P^k} \sum_{(i,j) \in P} c_{ij}^k + w_{ij} - \sigma^k$$

which is a shortest path problem with costs defined with optimal dual variables of the primal

Algoritmo de generación de Columnas

Set $Q_0^1, \ldots, Q_0^K$
Set $i = 0$
Solve master problem$_i$, and get optimal prim and dual sol.
while path flow optimality conditions are not satisfied do
    for each $k = 1, \ldots, K$ do
        solve subproblem$_k$ and get P^*
        $Q_{i+1}^k = Q_i^k \cup \{P^*\}$
    endfor
    $i = i + 1$
    Solve master problem$_i$, get optimal p/d sol.
endwhile