Problemas Combinatorios

Cobertura de conjuntos

- Problemas de grupos que se deben cubrir
- Problemas de grupos que se particionan
- Problemas de packing

Sea:

- $S = \{1, 2, 3, \dots, n\}$ conjunto a cubrir
- $L = \{(1,2); (2), \dots\}$ conjunto formado por subconjuntos de S

- Elegir elementos de L tales que:
  - Cobertura: se cubran todos los elementos con solapamiento
  - Partición: se cubran todos los elementos sin solapamiento
  - Packing: se cubra la máxima cantidad de elementos sin solapamiento

Ejemplos

hay 5 ~~cro~~ ciudades (1, 2, 3, 4, 5)

6 posibles circuitos

A: (1, 2) B: (1, 3, 5) C: (2, 4, 5) D: (3) E: (1) F (4, 5)

se busca minimizar las tripulaciones

definimos Y_i variable binaria que define si se utiliza el circuito i

$$\text{Min} \sum_{i=1}^{6} Y_i$$

Cobertura: Todas las ciudades deben ser cubiertas al menos una vez

C1) $Y_A + Y_B + Y_E \geq 1$

C2) $Y_A + Y_C \geq 1$

C3) $Y_B + Y_D \geq 1$

C4) $Y_C + Y_F \geq 1$

C5) $Y_B + Y_C + Y_F \geq 1$

Partición: Las ciudades deben ser cubiertas exactamente 1 vez

C1) $Y_A + Y_B + Y_C = 1$

C2) $Y_A + Y_C = 1$

C3) $Y_B + Y_D = 1$

C4) $Y_C + Y_F = 1$

C5) $Y_B + Y_C + Y_F = 1$

Este problema puede no tener solución

Packing: Cubrir la mayor cantidad de elementos sin solapamiento

$$\text{MAX} \sum_{i=A}^{F} Y_i$$

C1) $Y_A + Y_B + Y_C \leq 1$

C2) $Y_A + Y_C \leq 1$

C3) $Y_B + Y_D \leq 1$

C4) $Y_C + Y_F \leq 1$

C5) $Y_B + Y_C + Y_F \leq 1$

otro planteo ; creamos V_i $\begin{cases} 1 & \text{si se visita ciudad } i \\ 0 & \text{si no} \end{cases}$

$$\text{MAX} \sum_{i=1}^{5} V_i$$

(d) $Y_A + Y_B + Y_E = V_j$ $\forall j \in (2, \ldots, 5)$

Problema del viajante

- Tiene que partir de un origen y visitar todos los vértices antes de volver al origen
- Se conocen las distancias o pesos entre los vértices.

Definimos

$$Y_{ij} = \begin{cases} 1 & \text{si el Tour sale de la ciudad } i \text{ y llega a } j \\ 0 & \text{si no} \end{cases}$$

exactamente una ciudad debe ser visitada despues de i

" " " " " " antes de j

$$\min Z = \sum_{\substack{i=0 \\ i \neq j}}^{n} \sum_{j=0}^{n} c_{ij} Y_{ij}$$

$\sum_{\substack{j=0 \\ i \neq j}}^{n} Y_{ij} = 1$ $\forall i = 0, \ldots, n$ sale y llega solo a 1 ciudad

$\sum_{\substack{i=0 \\ i \neq j}}^{n} Y_{ij} = 1$ $\forall j = 0, \ldots, n$ llega a j desde exactamente 1 ciudad.

$$U_i - U_j + n Y_{ij} \leq n - 1 \quad \forall i,j = 1, \ldots, n$$

define el orden de i, j.

Variables del problema del viajante

- Se agrega un nuevo medio para viajar (tren o coche)

$$MIN\ Z = \sum\sum T_{ij}\, YT_{ij} + C_{ij}\, YC_{ij}$$

se agregan al funcional los costos de cada medio y se relacionan con $Y_{ij} \Rightarrow$

$$Y_{ij} = YT_{ij} + YC_{ij}$$

- No se puede visitar al cliente de la ciudad D si antes no se visitó el de la ciudad G

$$U_D \geq U_G$$

- No se puede visitar al cliente de la ciudad F si antes no se visitó al de la ciudad E o B

$$U_F \geq U_E - MY$$

$$U_F \geq U_B - M(1-Y)$$

Problema de distribución o Transporte

- Un conjunto de lugares cada uno de los cuales tiene disponible una cantidad de unidades de un producto
- Un conjunto de lugares cada uno de los cuales demanda una cantidad de productos
- Se conoce el costo C_{ij} de enviar una unidad desde el origen i al destino j

Ejemplo
distribuidora

NOTA

ALMACÉN

DISTRIBUIDORA		CBA 1	TUCUMAN 2	ROSARIO 3	BS AS 4	OFERTA
MENDOZA	1	464	513	654	867	75
SAN LUIS	2	352	416	690	791	125
STA FE	3	995	682	388	685	100
DEMANDAS		80	65	70	85	300

En este caso demandas y ofertas totales son iguales, de no ser así, habría que agregar un origen o destino ficticio para equilibrar. Con costos muy grandes para que los elija el modelo.

X_{ij}: Cantidad de unidades que el origen i envió al destino j.

$X_{11} + X_{12} + X_{13} + X_{14} = 75$

$X_{21} + X_{22} + X_{23} + X_{24} = 125$

$X_{31} + X_{32} + X_{33} + X_{34} = 100$

Asegura que cada origen distribuya todos las unidades

$X_{11} + X_{21} + X_{31} = 80$

$X_{12} + X_{22} + X_{32} = 65$

$X_{13} + X_{23} + X_{33} = 70$

$X_{14} + X_{24} + X_{34} = 85$

Asegura la demanda de los destinos.

en gral.

MIN $Z = \sum \sum C_{ij} X_{ij}$ sujeto a:

$\sum_{j=1}^{n} X_{ij} = S_i \quad \forall i = 1 \ldots m$; $\sum_{i=1}^{m} X_{ij} = D_j \quad \forall j = 1 \ldots n$

Problema de transbordo

- En este problema las unidades no son enviadas directamente desde los orígenes hasta los destinos, si no que pasa del origen a un transbordo y de ahí a algún destino

Al problema anterior agregamos costos de transbordo

DISTRIBUIDORA	TRASBORDO 1	TRANSBORDO 2	OFERTA
MENDOZA 1	138	313	75
SAN LUIS 2	321	216	125
STA FE 3	356	182	100

TRANSBORDOS	CBA. 1	TUCUMAN 2	ROSARIO 3	BS. AS. 4
TRANSBORDO 1	464	513	654	867
TRANSBORDO 2	495	682	388	685
DEMANDAS	80	65	70	85

variables

XO_iT_j : Cantidad de unidades que el origen i envía al transbordo j.

XT_iD_j : Cantidad de unidades que el transbordo i envía al destino j.

para los orígenes los transbordos son destinos

$XO_1T_1 + XO_1T_2 = 75$

$XO_2T_1 + XO_2T_2 = 125$

$XO_3T_1 + XO_3T_2 = 100$

para los Transbordos destinos, los Trans. son origenes

$XT_1D_1 + XT_2D_1 = 80$

$XT_1D_2 + XT_2D_2 = 65$

$XT_1D_3 + XT_2D_3 = 70$

$XT_1D_4 + XT_2D_4 = 85$

se agrega una ecuación para cada Transbordo que relaciona lo que entra y lo que sale.

$XO_1T_1 + XO_2T_1 + XO_3T_1 = XT_1D_1 + XT_1D_2 + XT_1D_3 + XT_1D_4$

$XO_1T_2 + XO_2T_2 + XO_3T_2 = XT_2D_1 + XT_2D_2 + XT_2D_3 + XT_2D_4$

Problema de Asignación

Sean los conjuntos A y B ambos con m elementos:

El problema de asignación consiste en:

encontrar el conjunto P donde:

- Cada elemento de P es un par (a,b)
- a es un elemento de A
- b es un elemento de B

Tal que minimice la función $\Sigma C(a,b)$

- Cada elemento de A, B debe aparecer en P exactamente una vez

$$X_{ij} \begin{cases} 1 & \text{si } i \text{ (origen) es asignado a } j \text{ (destino)} \\ 0 & \text{si no} \end{cases}$$

MIN/MAX $\quad Z = \Sigma \Sigma C_{ij} X_{ij}$

$\Sigma X_{ij} = 1 \quad \forall\, i = 1, \ldots m \; ; \; \Sigma X_{ij} = 1 \quad \forall\, j = 1, \ldots, m$

Problema de asignación cuadrática

Es similar al problema de asignación pero los costos dependen de que se produzcan dos asignaciones simultáneas

$$C_{ijkl} = t_{ik} \cdot d_{jl}$$

si queremos resolver el problema mediante PLE deberemos generar un cambio de variables

$Y_{ijkl} = 1$ si $X_{ij} = X_{kl} = 1$

$Y_{ijkl} = 0$ en caso contrario

planteamos un AND de X_{ij} y X_{kl}

$$2Y_{ijkl} \leq X_{ij} + X_{kl} \leq 1 + Y_{ijkl}$$

$$\text{MIN/MAX} \quad Z = \sum\sum\sum\sum Y_{ijkl} C_{ijkl} \qquad \begin{matrix} \forall\, i \neq k \\ \forall\, j \neq l \end{matrix}$$

Problema de la mochila

Hay una persona que tiene una mochila con una cierta capacidad y tiene que elegir que elementos pondrá en ella. Cada uno de los elementos tiene un peso y aporta un beneficio. El objetivo es maximizar el beneficio sin excederse de la capacidad.

Definimos:

- c la capacidad
- p_i como el beneficio obtenido por ingresar el producto i en la mochila.
- w_i como el peso del producto i
- n como la cantidad de productos

NOTA

$$\sum w_i x_i \leq C \quad \text{(Capacidad)}$$

$$\max \quad Z = \sum p_i x_i$$

posibles variantes

Acotado: De cada objeto i hay disponibles una cantidad b_i, así que agregamos otra constante al modelo.

$$x_i \leq b_i \quad (\forall\, i = 1..n)$$

$$\sum w_i x_i \leq C$$

$$\max \rightarrow Z = \sum p_i x_i$$

Suma de subconjuntos: En esta variante el beneficio es igual al peso para cada uno de los elementos.

$$\sum w_i x_i \leq C$$

$$\max \rightarrow Z = \sum w_i x_i$$

Problema de cambio: Hay un caso particular que aparece cuando se exige cumplir exactamente la capacidad. En este caso además todos los $p_i = 1$.

$$\sum w_i x_i = C$$

$$\max \rightarrow Z = \sum x_i$$

multiples mochilas: Surge de la generalización del problema estandar cuando se tienen varios contenedores.

$$\sum_{j=1}^{m} x_{ij} \leq 1 \quad (\forall\, i)$$

$$\sum_{i=1}^{n} w_i x_{ij} \leq C_j \quad (\forall\, j)$$

$$\max \rightarrow Z = \sum_{i=1}^{n} \sum_{j=1}^{m} p_i x_{ij}$$

Problema de Scheduling

El problema se identifica por dos conceptos clave: prioridades y capacidades. En definitiva qué deberá hacerse primero y quién deberá hacerlo.

Ejemplo

TAREA	TIEMPO EN MAQ 1	TIEMPO EN MAQ 2
A	3 min	2 min
B	6 min	8 min
C	5 min	6 min
D	7 min	4 min

definimos

I_{ij}: minuto en que empieza la tarea i en la maq. j

F_{ij}: minuto en el cual finaliza la tarea i en la maq. j

$F_{ij} = I_{ij} + \text{Tiempo\_de\_tarea\_i\_en\_maq\_j}$

$FA1 = IA1 + 3$

$FA2 = IA2 + 2$

Para que no empiece en maq 2 antes de haber terminado en maq 1

$$F_{A1} \leq I_{A2}$$

FINAL: Minuto en el cual finaliza la última tarea

$FA2 \leq FINAL$; $F_{B2} \leq FINAL$; $F_{C2} \leq FINAL$;

$F_{D2} \leq FINAL$

$MIN \to Z = FINAL$

para que no haga dos tareas al mismo tiempo en la máquina

hace en maq 1 la tarea A antes que la B, o la B antes que la A

$$F_{A1} \le I_{B1} + M\, Y_{ANULO\,AB}$$
$$F_{B1} \le I_{A1} + M\, Y_{ANULO\,BA}$$

$$Y_{ANULO}AB + Y_{ANULO}BA = 1$$

PARA MAQ J:

$$F_{ij} \le I_{kj} + M\, Y_{ANULO}IK$$
$$F_{kj} \le I_{ij} + M\, Y_{ANULO}KI$$

$$Y_{ANULO}IK + Y_{ANULO}KI = 1$$

para todo par de tareas i, k

en gral:

$$F_{ij} \le I_{kj} + M\, Y_{ANULO}JIK$$
$$F_{kj} \le I_{ij} + M\, Y_{ANULO}JKI$$

$$Y_{ANULO}JIK + Y_{ANULO}JKI = 1$$

para todo par de tareas i, k
para toda maquina J

Problema de coloreo de grafos

modelo clásico

$$MIN \to Z = \sum_{j=1}^{n} w_j$$

V: vertices

sujeto a

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i \in V \quad ; \quad x_{ij} + x_{kj} \le w_j \text{ si } [i,k] \in E \quad \forall j = 1 \dots n$$

X_{ij} $\begin{cases} 1 & \text{si el vertice } i \text{ se colorea con el color } j \\ 0 & \text{si no} \end{cases}$

funcional: el modelo clásico minimiza la cantidad de colores

Res:

- si dos vertices son adyacentes no pueden tener el mismo color
- si algun vertice usa el color j se fuerza w_j a valer 1.
- sumatoria igualada a 1: Cada vertice debe tener exactamente 1 color

__Modelo por conjuntos independientes__

sea $S = \{S_1, S_2, \dots, S_t\}$ el conjunto de todos los conjuntos independientes de G.

X_{S_i} $\begin{cases} 1 & \text{si todos los vertices de } S_i \text{ son coloreados de un mismo color} \\ 0 & \text{si no} \end{cases}$

$$\text{Min} \rightarrow Z = \sum_{i=1}^{t} X_{S_i}$$

sujeto a

$$\sum_{i: v \in S_i} X_{S_i} = 1 \qquad \forall v \in V$$

$$X_{S_i} \in \{0, 1\} \qquad \forall i = 1 \dots t$$

funcional: minimizar la cantidad de conjuntos ind.

sumatoria: Cada vertice debe pertenecer a un conjunto independiente

modelo por conjuntos independientes maximales

$$\text{Min} \rightarrow Z = \sum_{i=1}^{p} X_{M_i}$$

sujeto a

$$\sum_{i: v \in S_i} X_{M_i} \geq 1 \qquad \forall\, v \in V$$

funcional: utilizar la menor cantidad de conjuntos ind. maximales

sumatoria: Cada vertice debe pertenecer a un conjunto ind. o más.

OR

$Y_5 \leq Y_1$

$Y_6 \leq Y_1$

Si se produce 5 o 6 entonces se produce 1

AND

Si se ~~po~~ produce 3 y 4 entonces se produce 1

$2 \cdot Y_1 \leq Y_3 + Y_4 \leq 1 + Y_1$