Localizzazione

24/11/2005 12.37

Contenuto e scopo presentazione

Contenuto
- vengono introdotti modelli e metodi per problemi di localizzazione

Scopo
- fornire strumenti di supporto alle decisioni tattico strategiche in ambito logistico

Motivazioni

- Localizzazione di impianti (*facility locations*), centri di servizio o magazzini da dove partire per rispondere alla domanda dei clienti .
- Esempi:
  - magazzini di distribuzione di prodotti di largo consumo da cui rifornire i dettaglianti
  - centri commerciali
  - ospedali, caserme pompieri
  - ....

Tassonomia

- Localizzazione nel continuo vs. localizzazione su reti
  - Localizzazione nel continuo. Gli impianti e i clienti possono assumere una posizione qualunque in un insieme continuo, i costi sono usualmente espressi in termini di distanze tra punti secondo una data norma.
  - Localizzazione su reti. Gli impianti e i clienti sono di solito localizzati su alcuni vertici della rete, non necessariamente differenti.

Tassonomia

- Problemi di copertura (*covering*). Situa gli impianti in modo che tutti i clienti siano coperti, i.e., che giacciano a meno di una data distanza da un impianto. Problemi tipicamente associati con la disposizione di servizi pubblici, e.g., uffici postali, scuole.
- Problemi di centro (*center*). Situa p impianti in modo da minimizzare la massima distanza tra un punto di domanda e un impianto. Problemi tipicamente associati con la disposizione di servizi di emergenza, e.g., pronto soccorso, pompieri.
- Problemi di mediana (*median*). Situa p impianti in modo da minimizzare la distanza pesata tra i punti di domanda e gli impianti di riferimento. Problemi tipicamente associati con il trasporto merci.

Tassonomia

- Per numero di prodotti: *single* o *multi-commodity*
- Per numero di livelli: *single* o *multi-level*
- Per orizzonte di pianificazione: *single* o *multi-period*
- Per attori in gioco: *location* o *competitive location*
- Per modalità di trasporto: *full* o *not full load / truck load* o *less than a truck load / location* o *location routing*
- Per frazionabilitità della domanda.

Un impianto nel continuo

Istanza: un insieme $X \subseteq R^2$ convesso dove può essere localizzata il nuovo impianto. Un insieme S di clienti da cui bisogna minimizzare la distanza. Una funzione costo $d(x,s)$ funzione della distanza del punto x da s.

Soluzione: una potenziale localizzazione x.

Obiettivo: il costo totale sia minimo, i.e., $\min \Sigma_{s \in S} d(x,s)$, alternativamente il costo massimo sia minimo, i.e., $\min \max_{s \in S} d(x,s)$,

Commenti

- Nel caso dei trasporti la realtà è approssimabile con il continuo quando si considerano territori molto vasti, e.g., la Pianura Padana, gli Stati Uniti. In tali situazioni il costo del trasporto è circa proporzionale alla distanza in linea d'aria tra l'impianto e il cliente.
- L'obiettivo è il costo totale quando si devono ottimizzare le prestazioni medie, e.g., costi di trasporto. L'obiettivo è il costo massimo quando si devono ottimizzare le prestazioni peggiori, e.g., tempo di risposta di un servizio di emergenza.
- Quando deve essere localizzata un solo impianto i problemi sono facili si deve minimizzare una funzione convessa su un insieme convesso. Diventano difficili nel caso debbano essere localizzate più impianti. Le funzioni da minimizzare non sono più convesse. Una possibile euristica è partizionare a priori S e quindi affrontare il problema per un singolo impianto per ogni sottoinsieme.

Norme e Distanze

Una norma $N:R^n \rightarrow R$ è definita dalle seguenti tre proprietà:

- $N(x) \geq 0, \ \forall x \in R^n, \ N(x) = 0$ se e solo se $x=0$
- $N(ax) = |a|N(x), \ \forall x \in R^n, \ \forall a \in R$
- $N(x+y) \leq N(x) + N(y), \ \forall x,y \in R^n$, disuguaglianza triangolare

Ad ogni norma è associata una distanza

$$d(x,s) = N(x\text{-}y)$$

Tipica distanza norma l

$$d_l(x,s) = (\Sigma(x_i - s_i)^l)^{1/l}$$

$l = 1.4 - 1.7$ approssima ragionevolmente le distanze stradali

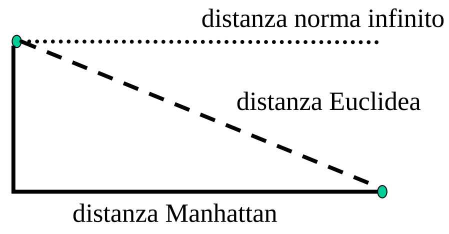
Distanze

- $d(x,s) = (\Sigma(x_i - s_i)^2)^{1/2}$ distanza euclidea - norma 2
- $d_A(x,s) = ((x - s)A^{-1}(x - s))^{1/2}$ distanza elissoidale, A matrice definita positiva. Se A diagonale distanza euclidea pesata
- $d_1(x,s) = \Sigma|x_i - s_i|$ distanza Manhattan - norma 1
- $d_\infty(x,s) = max_i\ (x_i - s_i)$ norma infinito

Proprietà

- $d(x,s) \leq d_1(x,s) \leq \sqrt{n}\, d(x,s)$ $\quad d_\infty(x,s) \leq d(x,s) \leq \sqrt{n}\, d_\infty(x,s)$
- $d_\infty(x,s) \leq d(x,s) \leq \sqrt{n}\, d_\infty(x,s)$ $\quad d_\infty(x,s) \leq d_1(x,s) \leq n\, d_\infty(x,s)$
- $\sqrt{\lambda}\, d(x,s) \leq d_A(x,s) \leq \sqrt{\Lambda}\, d(x,s)$ dove λ e Λ sono rispettivamente il minimo e i massimo autovalore di A

Distanze



1-mediana

Nel caso in cui l'obiettivo sia minimizzare i costi totali si deve risolvere il seguente problema di programmazione matematica

$$\min \Sigma_{s \in S}\, d_l(x,s)$$
$$x \in X$$

la funzione obiettivo è convessa e derivabile quasi ovunque, quindi il valore ottimo x^* può essere determinato tramite un algoritmo di discesa in tempi ragionevoli anche per cardinalità di S molto grandi.

Nel caso di norma 1 o infinito, se $X \supset S$, allora ogni x_i^* coincide con un coordinata di un punto di S.

1-centro

Nel caso in cui l'obiettivo sia minimizzare il costo massimo si deve risolvere il seguente problema di programmazione matematica

$$\min \max_{s \in S} d_l(x,s)$$
$$x \in X$$

la funzione obiettivo è convessa e derivabile quasi ovunque, quindi il valore ottimo x^* può essere determinato tramite un algoritmo di discesa in tempi ragionevoli anche per cardinalità di S molto grandi.

Esempio 1-mediana

$X = R^2$, $S = \{ (1,2), (3,2), (2,4), (0,5)\}$,

distanza Manhattan pesata con pesi $3, 5, 2, 6$.

$$\min 3(|x_1 -1| + |x_2 -2|) + 5 (|x_1 -3| + |x_2 -2|) +$$
$$+ 2(|x_1 -2| + |x_2 -4|) + 6(|x_1 -0| + |x_2 -5|)$$

Ogni coordinata del punto ottimo può essere calcolato come segue:

1. ordina i punti in base al valore della coordinata
2. calcola il peso cumulativo
3. scegli la coordinata corrispondente al valore del peso cumulato che è immediatamente superiore o uguale al peso cumulato totale diviso 2.

Esempio 1-mediana

$X = R^2$, $S = \{ (1,2), (3,2), (2,4), (0,5)\}$,

distanza Manhattan pesata con pesi $3, 5, 2, 6$.

coordinata x_1	peso	peso cumulato
0	6	6
1	3	9
2	2	11
3	5	16

coordinata x_2	peso	peso cumulato
2	3	3
2	5	8
4	2	10
5	6	16

punto ottimo $x^* = (1,2)$

1-centro

Il problema

$$\min \max_{s \in S} d_l(x,s)$$
$$x \in X$$

può anche essere scritto come

$$\min K$$
$$d_l(x,s) \leq K, \quad \forall s \in S$$
$$x \in X$$

nel caso di $l=1$ o ∞ e se X è un poliedro il problema diventa un problema di programmazione lineare. In ogni caso, il problema può essere risolto per generazione successiva di vincoli.

Esempio 1-centro

$X = R^2$, $S = \{ (1,2), (3,2), (2,4), (0,5)\}$, distanza Manhattan

min K

$|x_1 -1| + |x_2 -2| \leq K$

$|x_1 -3| + |x_2 -2| \leq K$

$|x_1 -2| + |x_2 -4| \leq K$

$|x_1 -0| + |x_2 -5| \leq K$

min K

$z_{11} + z_{12} \leq K$

$z_{21} + z_{22} \leq K$

$z_{31} + z_{32} \leq K$

$z_{41} + z_{42} \leq K$

$- z_{11} \leq x_1 -1 \leq z_{11}$

$- z_{12} \leq x_2 -2 \leq z_{12}$

$- z_{21} \leq x_1 -3 \leq z_{22}$

$- z_{22} \leq x_2 -2 \leq z_{22}$

... ...

$- z_{42} \leq x_2 -5 \leq z_{42}$

Un impianto su una rete

Istanza: una rete $G=(V,E)$ dove può essere localizzata il nuovo impianto. Un insieme S di clienti da cui bisogna minimizzare la distanza. I seguenti costi

- $d(i,j)$ distanza tra due vertici per ogni coppia di vertici
- $w(i)$ peso di un vertice
- $d(x,y)$ distanza tra due punti generici della rete

Soluzione: una potenziale localizzazione x.

Obiettivo: *problema di mediana:* il costo totale sia minimo, i.e.,

$$\min \Sigma_{i \in V} w(i)d(x,i),$$

problema di centro: il costo massimo sia minimo, i.e.,

$$\min \max_{i \in V} w(i)\, d(x,i),$$

1-mediana di una rete

Teorema (Hakimi)

In una rete connessa non orientata G esiste almeno un vertice v che è anche *1*-mediana.

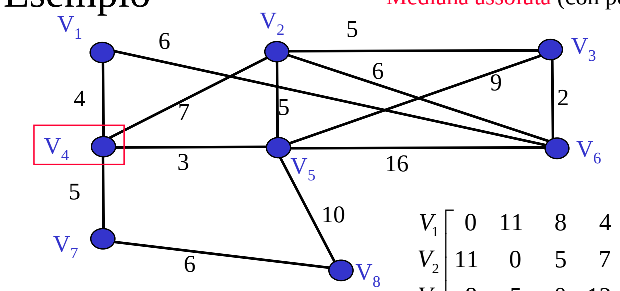
Algoritmo per il calcolo della mediana

(1) Calcolare la matrice $D(G)=[d(i,j)]$ delle distanze fra ogni coppia di vertici e moltiplicare ogni colonna j-ma per il peso $w(j)$. Sia $D'(G)$ la matrice così ottenuta.

(2) Sommare gli elementi di ciascuna riga di $D'(G)$ e scegliere la riga h-ma corrispondente alla somma minima. Il vertice h è la mediana di G.

Esempio

Mediana assoluta (con pesi dei vertici unitari)



$D(G) =$

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	
V_1	0	11	8	4	7	6	9	15	60
V_2	11	0	5	7	5	6	12	15	61
V_3	8	5	0	12	9	2	17	19	72
$\boxed{V_4}$	4	7	12	0	3	10	5	11	$\boxed{52}$
V_5									53
V_6	$\vdots$				$\vdots$			$\vdots$	71
V_7									72
V_8	15	15	19	11	10	21	6	0	97

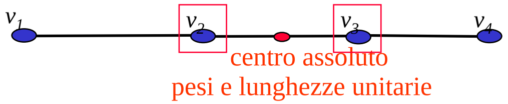
Eccentricità di un vertice e centro di una rete

Data una rete $G=(V, E)$ non orientata e connessa, con archi di lunghezza positiva arbitraria e con vertici con peso positivo arbitrario, si dice eccentricità di un punto x la distanza pesata di x dal vertice "più lontano"

$$e(x) = max_{i \in V}\{w(i)\, d(x,i)\}$$

Ogni punto di G con eccentricità minima prende il nome di centro assoluto.

Ogni vertice di G con eccentricità minima si dice centro-vertice di G.



Centro-vertice di una rete

Algoritmo per il calcolo del centro-vertice

(1) Calcolare la matrice $D(G)=[d_{ij}]$ delle distanze fra ogni coppia di vertici (i,j).

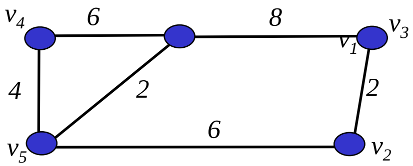
(2) Moltiplicare ogni colonna j-ma per il peso $w(j)$.

(3) Determinare il massimo su ogni riga ottenendo

$$e(i) = max_{i \in V}\{w(j)\, d(ij)\}$$

(4) Il centro vertice x è il vertice a cui corrisponde il valore minimo tra gli $e(i)$.

Esempio



$$\begin{bmatrix} 0 & 8 & 8 & 6 & 2 \\ 8 & 0 & 2 & 10 & 6 \\ 8 & 2 & 0 & 12 & 8 \\ 6 & 10 & 12 & 0 & 4 \\ 2 & 6 & 8 & 4 & 0 \end{bmatrix} \Rightarrow \begin{array}{l} e(v_1) = 8 \\ e(v_2) = 10 \\ e(v_3) = 12 \\ e(v_4) = 12 \\ e(v_5) = 8 \end{array}$$

Centro Vertice

Centro assoluto di una rete

Osservazione: Il centro assoluto di una rete può non giacere nella sottorete indotto dagli archi incidenti nei vari centro-vertici.

Metodo di Hakimi:

1. Per ciascun arco (v_i,v_h) determina il punto x_{ij} di eccentricità minima.
2. Tra i vari punti x_{ij} determina il punto di eccentricità minima in assoluto.

Centro assoluto di una rete

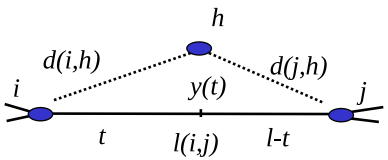
Calcolo punto x_{ij} di eccentricità minima.

$y(t)$: punto generico su arco (i,j) distante t dal vertice i

$l(i,j)$: lunghezza arco (i,j)

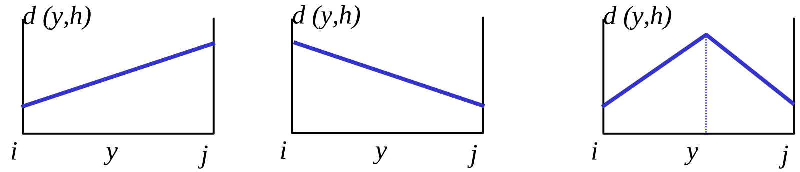
$d(y,h) = \min\{d(i,h)+t,\ d(j,h)+ l(i,j)-t\}$:

distanza di y dal nodo generico h



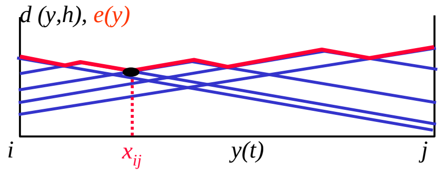
Centro assoluto di una rete

Al variare di y in (i, j) la distanza $d(y,h)$ può presentare uno tra i seguenti tre andamenti:



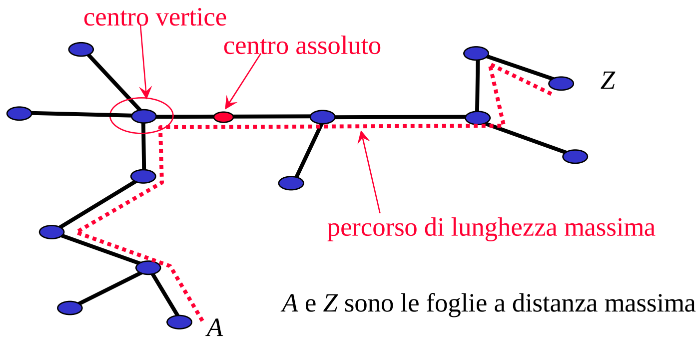
L'inviluppo dei grafici ottenuti per i vari h indica l'eccentricità dei punti $y(t)$ al variare della distanza t

x_{ij} è il punto ad eccentricità minima per (i,j).



Centro assoluto di un albero

Teorema: Il centro assoluto x di un albero T è il punto medio di un cammino di lunghezza massima tra due foglie di T. Se x è vertice allora è anche centro vertice, altrimenti il centro vertice è in uno o in entrambi i vertici adiacenti ad x.



A e *Z* sono le foglie a distanza massima

Problemi di *p*-mediana

Dato un insieme di p punti di $G=(V,E)$, $X_p=\{x_1, x_2,\ldots,x_p\}$ ed un vertice v, si dice distanza di v da X_p la quantità

$$d(v, X_p) = \min_{i\in X_p} d(x,i)$$

Se v rappresenta un cliente e X_p l'insieme degli impianti si ipotizza che il cliente si serva dal servizio più vicino.

Definizione

Si dice p-mediana assoluta di G ogni insieme di p punti X_p di G per cui è minima la funzione

$$f(X_p) = \Sigma_{v\in V}\, w(v)d(v, X_p)$$

Problemi di p-mediana

Teorema (Hakimi)

In una rete non orientata e connessa $G=(V,E)$ esiste almeno un insieme di p vertici che costituisce una p-mediana assoluta.

- Determinare la p-mediana su una rete qualunque è un problema NP-hard. Problema non approssimabile per nessun valore ε costante fissato, a meno di non tollerare qualche piccola variazione del valore di p
- Nel caso di alberi è un problema polinomiale.

Tecniche euristiche per la determinazione della p-mediana su grafi qualsiasi

<u>Approccio 1</u>

(1) Scegliere arbitrariamente p vertici $u_1,\ldots,u_p$

(2) Assegnare ogni altro vertice ad uno dei p vertici scelti in base alla minima distanza. Sia V_i l'insieme dei vertici assegnati ad u_i. Calcolare la somma S delle distanze

(3) Per ogni V_i cercare la mediana z_i e sostituirla a u_i. Calcolare la nuova somma S' delle distanze. Se $S'<S$ tornare al passo *2*, altrimenti l'algoritmo termina.

Tecniche euristiche per la determinazione della p-mediana su grafi qualsiasi

<u>Approccio 2</u>

Scegliere arbitrariamente p vertici quindi operare degli scambi (un nuovo vertice sostituisce uno tra quelli presenti nell'insieme selezionato) in modo che la funzione obiettivo abbia il massimo decremento.

Queste tecniche euristiche forniscono minimi locali.

Problemi di p-centro

Definizione

Si dice p-centro assoluto (o p-centro vertice) di $G=(V,E)$ ogni insieme di p punti (o vertici) X_p di G per cui è minima la funzione

$$r(X_p) = max_{v \in V}\, w(v)d(v, X_p)$$

La determinazione del p-centro vertice è un problema NP-hard.

Problemi di p-centro

Osservazioni:

- Il problema di determinare il p-centro assoluto (vertice) è interpretabile come la ricerca della localizzazione di p centri di servizio in una rete in modo da avere minima la distanza (tempo) per raggiungere il vertice più lontano dal centro più ad esso più vicino.
- Un problema collegato: data una distanza critica trovare il minor numero di centri e la loro localizzazione in modo che tutti i vertici siano ad una distanza minore o uguale a quella data da almeno uno dei centri.
- non approssimabile in meno di $3 - \varepsilon$

Formulazione MILP del problema della p-mediana

Descrizione del problema.

- **obiettivo:**
  - minimizzare costi di trasporto da impianti a cliente finale
- **vincoli:**
  - la domanda di ogni cliente deve essere soddisfatta da un impianto
  - devono essere aperti esattamente p impianti
- **leve decisionali:**
  - quali impianti aprire
  - come assegnare i clienti
- **dati tecnologici:**
  - c_{ij} costo assegnazione cliente j ad impianto i
  - p numero impianti
  - V insieme dei clienti
  - V' insieme delle possibili localizzazione degli impianti

Formulazione matematica

Formulazione del problema.

- Le variabili:

  x_{ij}: frazione domanda di j trasportata da impianto i a cliente j
  y_i: 1 se impianto aperto in i, 0 altrimenti

- La funzione obiettivo:
  minimizzazione costi

$$\Sigma_{i\in V'}\Sigma_{j\in V} c_{ij}\, x_{ij}$$

Formulazione matematica

- I vincoli:
  - la domanda di ogni cliente deve essere soddisfatta ...

$$\Sigma_{i\in V'} x_{ij} = 1, \qquad \forall j\in V$$

  - ... da un impianto aperto

$$x_{ij} \le y_i, \qquad \forall i\in V',\ \forall j \in V$$

  - devono essere aperti esattamente p impianti

$$\Sigma_{i\in V'} y_i = p$$

  - la domanda soddisfatta è espressa in percentuale, la localizzazione in modo discreto

$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \ \forall i\in V',\ \forall j \in V$$

Formulazione matematica

$$\min \sum_{i\in V'} \sum_{j\in V} c_{ij} x_{ij}$$
$$\sum_{i\in V'} x_{ij} = 1, \quad \forall j\in V$$
$$x_{ij} \le y_i, \qquad \forall i\in V', \ \forall j \in V$$
$$\sum_{i\in V'} y_i = p$$
$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \quad \forall i\in V', \ \forall j \in V$$

Simple Plant Location Problem

- ***p*-mediana**: numero impianti fissi, tutti gli impianti stesso costo apertura
- **simple plant location problem** o **uncapacitated plant location problem**: numero impianti variabili, costo di apertura f_i dipendente dalla localizzazione dell'impianto.

Formulazione matematica

$$\min \sum_{i\in V'} \sum_{j\in V} c_{ij} x_{ij} + \sum_{i\in V'} f_i y_i$$
$$\sum_{i\in V'} x_{ij} = 1, \quad \forall j\in V$$
$$x_{ij} \le y_i, \qquad \forall i\in V', \ \forall j \in V$$
$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \quad \forall i\in V', \ \forall j \in V$$

x_{ij}: frazione di domanda j soddisfatta dalla facility i

y_i: indicatore di apertura della facility

Data la struttura del problema, all'ottimo, x_{ij} è binaria e quindi può essere vista come una variabile che rappresenta un'assegnazione

Commenti

- problema NP-hard, ma approssimabile in *e/(e-1)*
- per grandi dimensioni deve essere risolto con euristiche, la più nota non elementare è DUALOC
- assomiglia al rilassamento lagrangiano del problema della *p-mediana*
- un lower bound può essere ottenuto attraverso il rilassamento continuo, eventualmente introducendo in modo dinamico i numerosi vincoli $x_{ij} \le y_i$.

Algoritmo greedy

Inizializzazione (inserito il primo deposito nell'insieme D dei depositi aperti)

for all $i \in V'$ **do** $w_i = f_i + \Sigma_{j\in V} c_{ij}$, (calcola costo di apertura i)

$i^* = arg\ min_i \{w_i\}$

$D = \{i^*\}$,

continua = true

Iterazione (si inseriscono nuovi depositi se inducono una diminuzione dei costi)

while *contiuna* **do begin**

for all $i \in V' \setminus D$ **do** $\Delta_i = f_i + \Sigma_{j\in V} min\{c_{ij} - min_{k\in D}\{c_{kj}\},0\}$,

$i^* = arg\ min_i \{\Delta_i\}$

if $\Delta_{i^*} < 0$ **then** $D = D \cup \{i^*\}$ **else** *continua = false*

end

return(D)

Ricerca Locale

- Le procedure di ricerca locale esaminano un intorno (*neighbour*) della soluzione corrente cercando di trovarne una migliore.
- Le procedure di ricerca locale si distinguono per l'intorno considerato.
- La procedura *bump and shift* considera due intorni
  - *bumb*: soluzioni in cui è stato soppresso un impianto rispetto alla soluzione corrente
  - *shift*: soluzioni in cui è stato aperto un nuovo impianto rispetto alla soluzione corrente

Esempio applicazione: opening lockboxes

Esempio:

La Fin-ACME riceve pagamenti da tutti gli Stati Uniti. A causa dell'organizzazione del sistema bancario vi sono dei ritardi tra il momento in cui il versamento è effettuato e il momento in cui il capitale diventa disponibile. Questi ritardi sono maggiori quando il versamento avviene fuori piazza. Si supponga di ricevere pagamenti dalle quattro regioni degli USA (West, Midwest, East, e South) e che il valore medio dei pagamenti giornaliero sia: $70,000 dal West, $50,000 dal Midwest, $60,000 dal East, e $40,000 dal South. La mancata disponibilità del denaro (e quindi l'impossibilità di reinvestirlo) ha un costo equivalente ad un tasso del 20% annuo, la Fin-ACME sta valutando l'opportunità di aprire degli uffici di riscossione (*lockbox*) in Los Angeles, Chicago, New York, Atlanta. La gestione di un lockbox costa $50,000 per anno. Sapendo che i ritardi medi saranno quelli riportati in tabella formulare un modello di programmazione lineare intera che permetta di decidere quale lockbox dovranno essere aperti e a quali di essi le differenti regioni dovranno fare riferimento.

Esempio applicazione: opening lockboxes

Ritardi medi in giorni

	L.A.	Chicago	N.Y.	Atalanta
West	2	6	8	8
Midwest	6	2	5	5
East	8	5	2	5
South	8	5	5	2

Costo annuale ritardi in migliaia di dollari

	L.A.	Chicago	N.Y.	Atalanta
West	28	84	112	112
Midwest	60	20	50	50
East	96	60	24	60
South	64	40	40	16

Esempio applicazione: opening lockboxes

Descrizione del problema.

- **obiettivo:**
  - minimizzare costi attesi
- **vincoli:**
  - ogni regione deve fare riferimento ad un lockbox
  - una regione *i* può fare riferimento ad lockbox *j* solo se questo è stato aperto
- **leve decisionali:**
  - quali lockbox aprire
  - a che lockbox assegnare ogni regione
- **dati tecnologici:**
  - f_j costo gestione lockbox *j*-mo
  - c_{ij} costo assegnamento regione *i*-ma al lockbox *j*-mo

Esempio applicazione: opening lockboxes

Formulazione del problema.

- Le variabili:

  variabili binarie

$$y_j = \begin{cases} 1 & se\ aperto\ lockbox\ j \\ 0 & altrimenti \end{cases}$$

$$x_{ij} = \begin{cases} 1 & se\ regione\ i\ assegnata\ a\ lockbox\ j \\ 0 & altrimenti \end{cases}$$

- La funzione obiettivo:

  i costi attesi (costi gestione lockbox + costi dovuti ai ritardi)

$$\Sigma_{j=1..4}\ f_j\ y_j\ +\ \Sigma_{i=1..4}\ \Sigma_{j=1..4}\ c_{ij}\ x_{ij}$$

Esempio applicazione: opening lockboxes

I vincoli:

- (formulazione non lineare. NON utilizzare se si può fare altrimenti)

  ogni regione *i* deve essere assegnata ad un lockbox *j* e ciò può avvenire solo se *j* è aperto

$$\Sigma_{j=1..4}\, x_{ij}\, y_j = 1 \qquad \forall i$$

- (formulazione lineare)

  ogni regione *i* deve essere assegnata ad un lockbox *j*

$$\Sigma_{j=1..4}\, x_{ij} = 1 \qquad \forall i$$

le assegnazioni possono avvenire solo se il lockbox è aperto

$$x_{ij} \le y_j \qquad \forall i\ \forall j$$

Esempio applicazione: opening lockboxes

Formulazione (migliore) istanza specifica

min $50\, y_1 + 50\, y_2 + 50\, y_3 + 50\, y_4 +$

$+ 28\, x_{11} + 84\, x_{12} + 112\, x_{13} + 112\, x_{14} + 60\, x_{21} + 20\, x_{22} + 50\, x_{23} + 50\, x_{24}$
+

$+ 96\, x_{31} + 60\, x_{32} + 24\, x_{33} + 60\, x_{34} + 64\, x_{41} + 40\, x_{42} + 40\, x_{43} + 16\, x_{44}$

$$x_{11} + x_{12} + x_{13} + x_{14} = 1$$
$$x_{21} + x_{22} + x_{23} + x_{24} = 1$$
$$x_{31} + x_{32} + x_{33} + x_{34} = 1$$
$$x_{41} + x_{42} + x_{43} + x_{44} = 1$$

$x_{11} \le y_1$
$x_{21} \le y$
$x_{31} \le y_1$
………
$x_{44} \le y_4$

← solo se $y_i = 1$ x_{ij} può assumere valore 1, viceversa se $x_{ij} = 1$ y_j deve assumere valore 1

$$x_{ij}, y_j \in \{0,1\}$$

Capacitated Plant Location Problem

- ***p*-mediana**: numero impianti fissi, tutti gli impianti stesso costo apertura
- **simple plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i.
- **capacitated plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i, capacità massime di impianto q_i per soddisfare le domande d_j.

Formulazione matematica

$$\min \Sigma_{i\in V'} \Sigma_{j\in V} c_{ij} x_{ij} + \Sigma_{i\in V'} f_i y_i$$

$$\Sigma_{i\in V'} x_{ij} = 1, \qquad \forall j\in V$$

$$\Sigma_{j\in V} d_j x_{ij} \le q_i y_i, \qquad \forall i\in V'$$

$$x_{ij} \le y_i, \qquad \forall i\in V', \forall j \in V$$

$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \qquad \forall i\in V', \forall j \in V$$

x_{ij}: frazione di domanda j soddisfatta dalla facility i

y_i: indicatore di apertura della facility

Data la presenza di vincoli di capacità, all'ottimo, x_{ij} può assumere valori frazionari

Commenti

- una soluzione esiste se e solo se $\Sigma_{j\in V} d_j \le \Sigma_{j\in V'} q_i$
- problema NP-hard
- per grandi dimensioni deve essere risolto con euristiche (e.g., lagrangiane)
- la presenza di vincoli di capacità impedisce all'euristiche greedy e di bumb and shift di giungere a buoni risultati. Tra le euristiche di scambio sono da preferire quelle evolute: tabù search, simulated annealing ed algoritmi genetici.

Euristica Lagrangiana

Rilassamento lagrangiano

$$\min \Sigma_{i\in V'} \Sigma_{j\in V} c_{ij} x_{ij} + \Sigma_{i\in V'} f_i y_i + \Sigma_{j\in V'} \lambda_j (1- \Sigma_{i\in V'} x_{ij})$$

$$\Sigma_{j\in V} d_j x_{ij} \le q_i y_i, \qquad \forall i\in V'$$

$$x_{ij} \le y_i, \qquad \forall i\in V', \forall j \in V$$

$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \qquad \forall i\in V', \forall j \in V$$

- il problema può venire decomposto in tanti sotto problemi quanti sono le possibili localizzazioni degli impianti
- i valori λ_j vanno iterativamente aggiornati

Euristica Lagrangiana

Per ogni potenziale impianto i si deve risolvere il seguente facile problema

$$\min \Sigma_{j\in V}\, c_{ij}\, x_{ij} + f_i\, y_i - \Sigma_{j\in V}\, \lambda_j\, x_{ij}$$

$$\Sigma_{j\in V}\, d_j x_{ij} \le q_i y_i,$$

$$x_{ij} \le y_i, \qquad \forall j \in V$$

$$0 \le x_{ij} \le 1, \quad y_i \in \{0, 1\} \qquad \forall j \in V$$

- sia $lb_i(\lambda)$ la soluzione del problema per il potenziale impianto i
- rimane fuori dai sottoproblemi il termine costante $\Sigma_{j\in V}\, \lambda_j$, che non necessita di essere ottimizzato

Euristica Lagrangiana

Data la soluzione del rilassamento lagrangiano si determina una soluzione ammissibile come segue:

- si ordinano i potenziali impianti i in ordine crescente di $lb_i(\lambda)$
- si selezionano i primi T potenziali impianti che permettano di soddisfare la domanda, i.e., tale che $\Sigma_{j\in V}\, d_j \le \Sigma_{j\in T}\, q_i$
- si riassegnano i clienti agli impianti

$$\min \Sigma_{i\in V'} \Sigma_{j\in V}\, c_{ij}\, x_{ij} + \Sigma_{i\in T}\, f_i$$

$$\Sigma_{i\in T}\, x_{ij} = 1, \qquad \forall j \in V$$

$$\Sigma_{j\in V}\, d_j x_{ij} \le q_i y_i, \qquad \forall i \in T$$

$$0 \le x_{ij} \le 1, \qquad \forall j \in V$$

Multicommodity Capacitated Plant Location Problem

- ***p*-mediana**: numero impianti fissi, tutti gli impianti stesso costo apertura
- **simple plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i.
- **capacitated plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i, capacità massime di impianto q_i per soddisfare le domande d_j.
- **multicommodity capacitated plant location problem**: come il capacitated plant location problem, ma con la gestione di più prodotti p, ognuno richiedente una capacità v_p degli impianti per essere trattato. Capacità massime di impianto q_i di cui al più q_{pi} per il prodotto p. In generale $\Sigma_{p\in P}\, q_{pi} \ge q_i$.

Formulazione matematica

$$\min \Sigma_{p\in P} \Sigma_{i\in V'} \Sigma_{j\in V}\, c_{pij}\, x_{pij} + \Sigma_{i\in V'}\, f_i\, y_i$$

$$\Sigma_{i\in V'}\, x_{pij} = d_{pj}, \qquad \forall p\in P, \forall j\in V$$

$$\Sigma_{p\in P} \Sigma_{j\in V}\, v_p\, x_{pij} \le q_i y_i, \qquad \forall i\in V'$$

$$v_p\, \Sigma_{j\in V} x_{pij} \le q_{ip} y_i, \qquad \forall p\in P,\ \forall i\in V'$$

$$v_p x_{pij} \le \min\{v_p\, d_{pj}, q_{pi}\}\, y_i, \qquad \forall p\in P,\ \forall i\in V',\ \forall j \in V$$

$$x_{pij} \ge 0, \quad y_i \in \{0, 1\} \qquad \forall p\in P,\ \forall i\in V',\ \forall j \in V$$

x_{pij}: quantità di domanda j soddisfatta dalla facility i

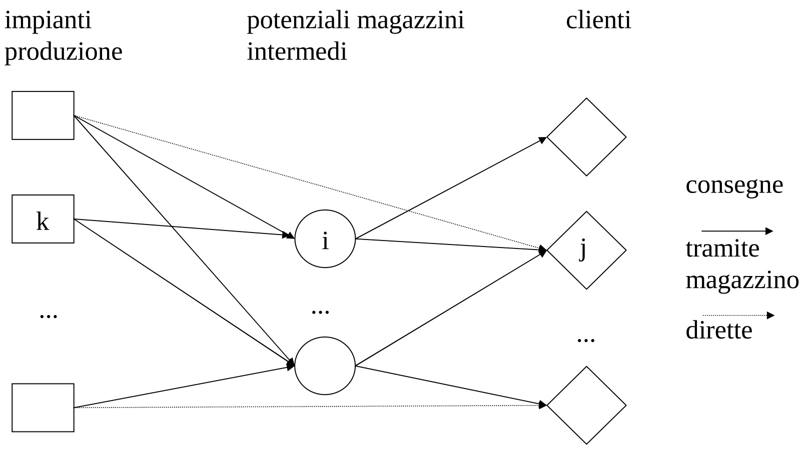
y_i: indicatore di apertura della facility

Valgono gli altri commenti del capacitated location problem

Production Distribution Problem

- ***p*-mediana**: numero impianti fissi, tutti gli impianti stesso costo apertura
- **simple plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i.
- **(multicommodity) capacitated plant location problem**: numero impianti variabili, costo di apertura dipendente dalla localizzazione dell'impianto f_i, capacità massime di impianto q_i per soddisfare le domande d_j.
- **production distribution problem**: più beni p sono prodotti in un insieme V'' di impianti di capacità q_k (al più q_{pk} per il prodotto p) e devono essere consegnati ai clienti, direttamente o tramite dei magazzini intermedi. Si devono localizzare gli eventuali magazzini intermedi di numero variabile, costo di apertura dipendente dalla localizzazione del magazzino f_i, capacità massime di magazzino q_i per soddisfare le domande d_{pj}.

Production Distribution Problem



Formulazione matematica

$$\min \Sigma_{p\in P} \Sigma_{k\in V''} \Sigma_{i\in V'} \Sigma_{j\in V} c_{pkij}\, x_{pkij} + \Sigma_{i\in V'} f_i y_i$$

$$\Sigma_{k\in V''} \Sigma_{i\in V'} x_{pkij} = d_{pj}, \qquad \forall p\in P, \forall j\in V$$

$$\Sigma_{p\in P} \Sigma_{k\in V''} \Sigma_{j\in V} v_p x_{pkij} \le q_i y_i, \qquad \forall i\in V'$$

$$\Sigma_{p\in P} \Sigma_{i\in V'} \Sigma_{j\in V} v_p x_{pkij} \le q_k, \qquad \forall k\in V''$$

$$v_p \Sigma_{j\in V} x_{pij} \le q_{ip} y_i, \qquad \forall p\in P, \forall i\in V'$$

$$v_p \Sigma_{j\in V} x_{pij} \le q_{ik} y_i, \qquad \forall p\in P, \forall k\in V'',$$

$$v_p x_{pkij} \le \min\{v_p d_{pj}, q_{pi}, q_{pk}\}\, y_i, \qquad \forall p\in P, \forall k\in V'', \forall i\in V', \forall j \in V$$

$$x_{pkij} \ge 0, \quad y_i \in \{0, 1\} \qquad \forall p\in P, \forall k\in V'', \forall i\in V', \forall j \in V$$

x_{pkij}: quantità di domanda j del prodotto p soddisfatta dall'impianto k e passante per il magazzino i

y_i: indicatore di apertura della facility

Commenti

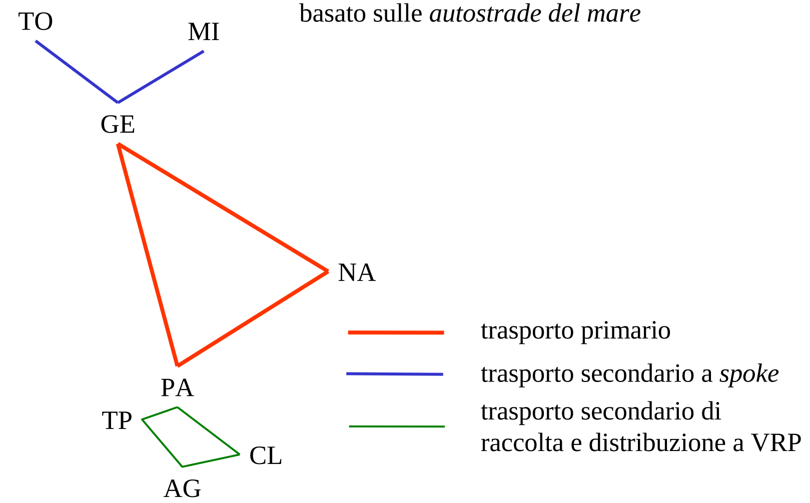
- problema NP-hard, per grandi dimensioni deve essere risolto con euristiche (e.g., le euristiche lagrangiane funzionano molto bene)
- nella formulazione matematica un prodotto che passa attraverso il magazzino intermedio *0* "dummy" è in realtà consegnato direttamente.

  Tale magazzino o non ha costo fisso oppure, in generale, ha costo fisso differenziato per prodotto (in questo caso la formulazione cambia leggermente)
- il problema non è banale se NON vale la disuguaglianza triangolare, i.e., se il costo unitario per una consegna diretta è maggiore (a meno dei costi fissi di apertura dei magazzini) della somma dei costi per una consegna indiretta. I costi unitari delle consegne indirette sono minori grazie al consolidamento.

Hub (and Spoke) Network Problem

- ***p*-mediana**
- **simple plant location problem**
- **(multicommodity) capacitated plant location problem**
- **production distribution problem**
- **hub (and spoke) network problem**: i prodotti vengono raccolti dai produttori (trasporto secondario (*spoke*) o VRP) consolidati negli *hub*, vengono poi trasportati da hub ad hub con veicoli di grande dimensione (trasporto primario) e quindi distribuiti a partire dall'hub di arrivo ai clienti finali tramite trasporto secondario.

Rete per trasporto collettame multimodale basato sulle *autostrade del mare*



Commenti

- problema che si formula in modo analogo al production distribution, ma servono variabili x_{pkilj}: quantità di domanda *j* del prodotto *p* soddisfatta dall'impianto *k* e passante per l'hub *i* in partenza e l'hub *l* in arrivo
- problema NP-hard, per grandi dimensioni deve essere risolto con euristiche (lagrangiane o tabù search)
- ovviamente se la domanda giustifica il trasporto diretto, vi sono anche alcune merci che vengono trasferite direttamente da produttore a cliente, come nel caso production distribution

Commenti

- sistema di organizzazione dei trasporti molto usato nei trasporti navali e aerei intercontinentali. Utilizzato anche nei trasporti su lunga distanza su gomma e ferro
- il trasporto aereo passeggeri nazionale italiano è in realtà un production distribution (con tappe intermedie negli hub di Malpensa o Fiumicino). E' organizzato ad hub and spoke nel caso di trasporto intercontinentale, e.g., con gli USA, hub italiani Malpensa o Fiumicino, hub americani New York, Chicago, Los Angeles, Las Vegas
- l'organizzazione ad hub and spoke introduce notevoli risparmi di scala, ed è l'unico economicamente giustficabile per trasporti di merci con volumi ridotti, introduce ovviamente dei ritardi a causa delle attese agli hub. Questi ritardi sono particolarmente gravosi per i trasporti espressi e per il trasporto di passeggeri.

Set covering, partitioning, packing

Siano dati un insieme finito $M=\{1,...,m\}$, S un insieme (non necessariamente completo) dei sottoinsiemi di M ed un insieme $F=\{F_1, F_2, \ldots, F_n\}$ di sottoinsiemi di M t.c. $F \subseteq S$

- l'insieme F si dice **Covering di M** se $\cup_i F_i = M$
- l'insieme F si dice **Packing di M** se $F_i \cap F_j = \emptyset, \forall F_i, F_j \in F$
- l'insieme F si dice **Partitioning di M** se F è sia un covering che un packing di M.

Covering, partitioning

Problema di set covering $SCP(M,\mathbf{c}, min)$

Istanza: dati un insieme finito $M=\{1,...,m\}$ e una funzione $\mathbf{c}:S \to N$ che assegni un costo ad ogni sottoinsieme F_i di M (si noti che non necessariamente tutti i sottoinsiemi devono essere considerati).

Soluzione: un covering $F=\{F_1, F_2, \ldots, F_n\}$ di M.

Obiettivo: il costo del covering sia minimo, i.e. min $\Sigma_{Fi\in F}\, c(F_i)$

Problema di set partitioning: analogo al precedente, ma si cerca un partitioning

Packing

Problema di packing $SPP(M,\mathbf{c}, max)$

Istanza: dato un insieme finito $M=\{1,...,m\}$ e una funzione $\mathbf{c}$ che assegni un valore ad ogni sottoinsieme F_i di M (si noti che non necessariamente tutti i sottoinsiemi devono essere considerati).

Soluzione: un packing $F=\{F_1, F_2, \ldots, F_n\}$ di M.

Obiettivo: il valore del packing sia massimo, i.e. max $\Sigma_{Fi\in F}\, c(F_i)$

Covering, partitioning, packing

Una matrice A si definisce **matrice di incidenza** di S se ha una riga per ogni elemento k di M, una colonna per ogni sottoinsieme F_j in S e le sue *entries* a_{kj} soddisfano la seguente condizione

$$\mathbf{A} = \begin{array}{cc} \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} & \begin{array}{c} 1 \\ \vdots \\ m \end{array} \\ \begin{array}{ccc} F_1 & \cdots & F_n \end{array} & \end{array} \qquad a_{kj} = \begin{cases} 1 & se\ \ k \in F_j \\ 0 & se\ \ k \notin F_j \end{cases}$$

Un vettore $\boldsymbol{x}$ si definisce **vettore di incidenza** di F se

$$x_j = \begin{cases} 1 & se\ F_j \in F \quad \text{(viene scelto l'insieme } F_j\text{)} \\ 0 & se\ F_j \notin F \quad \text{(non viene scelto l'insieme } F_j\text{)} \end{cases}$$

Covering, partitioning, packing

I vincoli delle tre classi di problemi si possono scrivere come:

- Covering: $\boldsymbol{Ax} \geq \boldsymbol{1}$
- Packing: $\boldsymbol{Ax} \leq \boldsymbol{1}$
- Partitioning: $\boldsymbol{Ax} = \boldsymbol{1}$

dove $\boldsymbol{1}$ è un vettore a m elementi tale che $\boldsymbol{1}^T=[1,...,1]$

L'obiettivo è min $\boldsymbol{cx}$ (covering, partitioning) o max $\boldsymbol{cx}$ (packing)

Covering, partitioning, packing

Commenti:
- il concetto di matrice di incidenza generalizza il concetto di matrice di incidenza dei grafi non orientati;
- analogamente l'insieme S generalizza il concetto di rete, ogni sottoinsieme F_i definisce una relazione $|F_i|$-aria tra i suoi elementi (invece che relazioni binarie come nelle reti)

Esempio applicazione: covering

Localizzazione di impianti (Facility Location).

Sono date m comunità che devono essere protette da stazioni antiincendio ed n possibili localizzazioni delle centrali. Il costo pagato per posizionare una stazioni nel sito j è c_j. Una centrale situata in j è in grado di proteggere un certo sottoinsieme di comunità.

Si vuole determinare dove localizzare le centrali in modo che tutte le comunità siano protette minimizzando il costo complessivo.

$M=\{1,...,m\}$ è l'insieme delle comunità da proteggere.

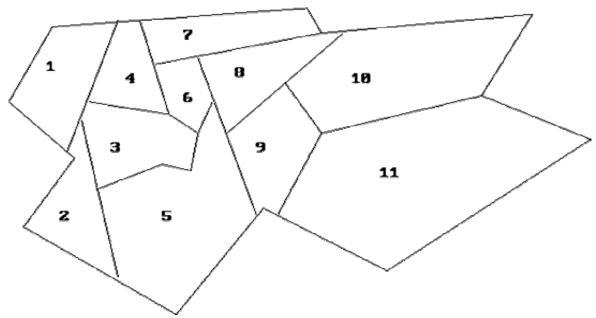
$N=\{1,...,n\}$ è l'insieme delle possibili localizzazioni delle centrali.

F_j è il sottoinsieme delle comunità che sono protette da una centrale situata in j (ad esempio, perché è raggiungibile entro un tempo massimo).

La matrice $\boldsymbol{A}$ ha una riga per ciascuna comunità da proteggere ed una colonna per ciascuna possibile localizzazione di una centrale.

Il problema è un *Set Covering*. Le variabili costituiscono il vettore di incidenza che indica i siti scelti per le stazioni.

Esempio applicazione: covering



In particolare si supponga che una stazione dei pompieri in una delle regioni rappresentate in figura possa servire sia la regione di appartenenza che quelle ad essa limitrofe (e.g., una stazione in *9* copre anche *5*, *8*, *10* e *11*). Sia inoltre il costo di installazione uguale in tutte le regioni. Si vuole formulare un modello che porti a determinare il numero minimo di stazioni necessarie.

Esempio applicazione: covering

Descrizione del problema.

- **obiettivo:**
  – minimizzare numero stazioni
- **vincoli:**
  – ogni regione deve essere coperta
- **leve decisionali:**
  – dove localizzare le stazioni
- **dati tecnologici:**
  – c_j costo localizzazione una stazione nella regione j-ma
  – a_{kj} parametro binario che vale 1 se regione j-ma è adiacente alla k-ma, i.e., $k\in F_j$, 0 altrimenti

Esempio applicazione: covering

Formulazione del problema.

- Le variabili:
  variabili binarie

$$x_j = \begin{cases} 1 & \text{se una stazione è nella regione } j \\ 0 & \text{altrimenti} \end{cases}$$

- La funzione obiettivo:
  i costi di assegnazione

$$\Sigma_{j=1..11}\ c_j\, x_j$$

- I vincoli:

  ogni regione i deve essere coperta da almeno una stazione

$$\Sigma_{j:k\in F_j}\, x_j \geq 1 \qquad \forall k$$

Esempio applicazione: covering

Formulazione istanza specifica

$$\begin{array}{ll}
\min & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} \\
1: & x_1 + x_2 + x_3 + x_4 \geq 1 \\
2: & x_1 + x_2 + x_3 + x_5 \geq 1 \\
3: & x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \geq 1 \\
4: & x_1 + x_3 + x_4 + x_6 + x_7 \geq 1 \\
5: & x_2 + x_3 + x_5 + x_6 + x_8 + x_9 \geq 1 \\
6: & x_3 + x_4 + x_5 + x_6 + x_7 + x_8 \geq 1 \\
7: & x_4 + x_6 + x_7 + x_8 \geq 1 \\
8: & x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} \geq 1 \\
9: & x_5 + x_8 + x_9 + x_{10} + x_{11} \geq 1 \\
10: & x_8 + x_9 + x_{10} + x_{11} \geq 1 \\
11: & x_9 + x_{10} + x_{11} \geq 1 \\
 & x_j \in \{ 0,1 \}
\end{array}$$

Esempio applicazione: covering

Formazione di gruppi di lavoro.

Sono date m attività ed n persone, ciascuna in grado di svolgere un sottoinsieme di tali attività. Ad ogni persona j, $j=1,\ldots,n$, è associato un costo c_j.

Si vuole costituire un gruppo di lavoro formato da un sottoinsieme di persone in maniera che il gruppo sia in grado di svolgere tutte le attività al costo complessivo minimo.

$M=\{1,\ldots,m\}$ è l'insieme delle attività.

$N=\{1,\ldots,n\}$ è l'insieme delle persone.

F_j è il sottoinsieme delle attività che la persona j è in grado di svolgere: ad esempio, se la persona j è in grado di svolgere l'attività k allora $k\in F_j$.

La matrice ***A*** ha una riga per ciascuna attività ed una colonna per ciascuna persona.

Il problema è un *Set Covering*. Le variabili costituiscono il vettore di incidenza che indica le persone scelte.

Esempio applicazione: covering

Problemi di instradamento (Routing Problem).

Sono dati *m* clienti che devono essere serviti da una azienda di trasporti ed *n* possibili rotte effettuate dai mezzi dell'azienda. Il costo pagato per seguire una rotta *j* è c_j.

Una rotta *j* è in grado di servire un certo sottoinsieme di clienti.

Si vuole determinare quali rotte effettuare in modo che tutte i clienti siano serviti con il minimo costo complessivo.

$M=\{1,...,m\}$ è l'insieme dei clienti da servire.

$N=\{1,...,n\}$ è l'insieme delle possibili rotte.

F_j è il sottoinsieme dei clienti che sono serviti da un trasporto che segua la rotta *j*.

La matrice ***A*** ha una riga per ciascun cliente da servire ed una colonna per ciascuna possibile rotta da far eseguire ai mezzi di trasporto.

Il problema è un *Set Covering*. Le variabili costituiscono il vettore di incidenza che indica quali rotte sono state scelte.

Esempio applicazione: partitioning

Formazione dei collegi elettorali (Political Districting).

Devono essere suddivisi *m* comuni in un insieme di distretti elettorali. Sono specificate a priori *n* possibili suddivisioni in maniera che rispondano a determinati requisiti. Se si sceglie di costituire il distretto *j* si paga un costo c_j. Ogni comune deve essere incluso esattamente in un solo distretto elettorale.

Si vuole determinare come formare i distretti elettorali in maniera da minimizzare il costo complessivo.

$M=\{1,...,m\}$ è l'insieme dei comuni.

$N=\{1,...,n\}$ è l'insieme dei possibili distretti.

F_j è il sottoinsieme di comuni che sono inclusi nel distretto elettorale *j*.

La matrice ***A*** ha una riga per ciascun comune ed una colonna per ciascun possibile distretto elettorale.

Il problema è un *Set Partitioning*. Le variabili costituiscono il vettore di incidenza che indica quali sono i distretti che si vogliono formare.

Esempio applicazione: partitioning

Formazione equipaggi (Crew Scheduling)*

One interesting (and profitable!) area where set partitioning is used is in airline crew scheduling. Next to fuel costs, personnel costs are the largest cost an airline faces; captain salaries of $150,000 per year are not unusual. The effective use of personnel can have a tremendous impact on the bottom line. There are, however, a tremendous number of restrictions on how airline personnel are used. Between FAA regulations and union contracts, determining a feasible crew schedule is a very difficult problem. About ten years ago, American Airline was interested in applying OR techniques to crew scheduling. In their formulation, a variable is created for each feasible schedule for a single crew. For instance, a schedule where the crew leaves Pittsburgh at 7:15AM on Tuesday, arrives in NY at 8:30, leaves NY at 9:00, arrives Atlanta at 11:00, leaves Atlanta at 12:00 and arrives Pittsburgh at 2:30 would correspond to a single variable, , say. The cost of setting that variable to 1 (meaning to have a crew fly that schedule) would be based on the union contract and other regulations.

* Quanto segue è tratto dal sito del prof. M. Trick (1998) (da esso sono anche ispirati gli esempi di capital budgeting, opening lockboxes, facility location ed alcuni degli esercizi).

Esempio applicazione: partitioning

Formazione equipaggi (Crew Scheduling).

As you can imagine, there are a tremendous number of variables.

The constraints, however are very simple. For each flight (called a leg), it is necessary to choose a schedule that includes that flight. This leads to a constraint of the form

$$x_1 + x_4 + x_{55} + x_{5534} + \ldots = 1$$

if schedules 1, 4, 55, 5534 and so on all contain the 7:15 Tuesday flight from Pittsburgh to NY (for example). While the resulting is very large, the savings that result from finding good solutions are so large that a tremendous amount of work has gone into finding techniques for solving these problems. In fact, the group within American has spun off into a consulting operation (now part of Sabre Technologies), and is one of the most profitable areas for American Airlines.

Esempio applicazione: packing

Formazione di gruppi di lavoro (notare le differenze con il caso di covering)

Sono date m persone ed n possibili squadre di lavoro, ciascuna in grado di svolgere una specifica attività. Ad ogni attività j, $j=1,...,n$, è associato un profitto p_j.

Sapendo che una persona non può appartenere a più squadre contemporaneamente, si devono costituire le squadre di lavoro in modo da svolgere l'insieme di attività che danno profitto complessivo massimo.

$M=\{1,...,m\}$ è l'insieme delle persone.

$N=\{1,...,n\}$ è l'insieme delle squadre di lavoro.

F_j è il sottoinsieme delle persone appartenenti alla squadra j: ad esempio, se la squadra j include la persona k allora $k\in F_j$.

La matrice $\boldsymbol{A}$ ha una riga per ciascuna persona ed una colonna per ciascuna squadra.

Il problema è un *Set Packing*. Le variabili costituiscono il vettore di incidenza che indica le squadre formate.

Euristiche per il set covering

Nel seguito verranno presentate alcune euristiche per il problema di *unicost* set covering, i.e., dove tutti gli insiemi sono caratterizzati dallo stesso costo e si voglia quindi minimizzare il numero di insiemi utilizzati per la copertura

Limite inferiore e prestazioni euristiche

- **Limite inferiore**

  Un ovvio limite inferiore (*lower bound*) per il problema del set covering (*SCP*) è ottenibile rilassando per continuità il problema MIP associato.

- **Prestazioni euristiche**

  Per il SCP non possono esistere schemi di approssimazione polinomiale, a meno che $P=NP$. Esistono algoritmi la cui performance è $O(1+a\log n)$ e $O(k(1-(b/k)^{1/k})$ dove a e b sono opportune costanti e k il numero massimo di elementi non nulli per riga.

Euristiche greedy per il set covering

Euristica *Greedy*

– ad ogni passo è scelto l'insieme che copre il maggiore numero elementi non coperti dai sottoinsiemi scelti ai passi precedenti, i.e., ad ogni passo è fissata uguale ad 1 la variabile che rende soddisfatto il maggior numero di vincoli non già soddisfatti dalle variabili già poste ad 1 ai passi precedenti. In caso di possibilità alternative si sceglie l'insieme/variabile di indice minimo.

Euristica *Randomized Greedy*

– come *Greedy*, ma in caso di possibilità alternative si sceglie a caso.

Euristiche greedy per il set covering

Euristica *Greedy Alternante*

si iterano due passi

passo 1: identico a quello dell'algoritmo greedy. Sia r il numero di nuovi elementi coperti nell'iterazione generica con la selezione di un nuovo sottoinsieme, ma non coperti nelle iterazioni precedenti;

passo 2: si deselezionano i sottoinsiemi ridondanti, i.e., quegli insiemi che se deselezionati non conducono alla scopertura di alcun elemento oppure che al più scoprono complessivamente $r-1$ elementi. Vengono deselezionati prima i sottoinsiemi che scoprono il minor numero di elementi, in caso di parità si sceglie il sottoinsieme di indice minimo.

Euristiche greedy il set covering

Commenti:

- l'euristiche *Greedy Alternante* e *Randomized Greedy* sono computazionalmente più onerose dell'euristica *Greedy*. Si noti che di solito l'algoritmo *Randomized Greedy* viene fatto girare più volte su uno stesso problema e poi viene scelto il risultato migliore;
- l'euristica *Randomized Greedy* sembra presentare in media migliori prestazioni nel caso di istanze generate casualmente;
- nel caso di istanze di specifici problemi combinatori l'euristica *Greedy Alternante* o, addirittura, la *Greedy* hanno prestazioni migliori. Nessuna delle euristiche presentate domina le altre;

Euristiche lagrangiane

- Nel caso in cui ai sottoinsiemi da selezionare siano associati costi differenti, le euristiche viste vengono modificate;
- ad ogni passo viene scelto il sottoinsieme j che presenta un punteggio (*score*) maggiore;
- il punteggio è tipicamente funzione del costo del sottoinsieme j, del numero k_j di elementi i non ancora coperti che j riesce a coprire e dei moltiplicatori lagrangiani λ_k associati alle righe degli elementi k non ancora coperti;
- nelle euristiche più semplici non si tiene conto dei moltiplicatori e il punteggio si basa tipicamente su c_j, c_j / k_j, $c_j / \log k_j$, $c_j / k_j \log k_j$, in quelle più avanzate i moltiplicatori λ_k vengono calcolati risolvendo un rilassamento lagrangiano o surrogato del SCP.

Euristiche lagrangiane

Il rilassamento lagrangiano del problema di SCP, supponendo di rilassare tutti i vincoli, è il seguente:

$$L(\lambda) = \min \textstyle\sum_{j\in S} c_j x_j + \sum_k \lambda_k (1 - \sum_{j:k\in F_j} x_j) = \sum_{j\in S} c_j(\lambda) x_j + \sum_k \lambda_k$$

$$x_j \in \{0, 1\}, \lambda_k \geq 0 \qquad \forall j \in S$$

dove $c_j(\lambda) = c_j - \sum_{k\in F_j} \lambda_k$ è detto costo lagrangiano.

Per ogni valore λ fissato, determinare i valori ottimi di x_j è banale

- $x_j = 1$ se $c_j(\lambda) < 0$
- $x_j = 0$ se $c_j(\lambda) > 0$
- $x_j = 0$ oppure 1 a scelta se $c_j(\lambda) = 0$

Euristiche lagrangiane

- per ogni valore λ, $L(\lambda)$ è un lower bound per il valore ottimo del SCP;
- si determina un valore (sub)ottimo di λ, i.e., quello che massimizza il valore di $L(\lambda)$, attraverso una procedura di *subgradient optimization*, *i.e.*, una procedura che modifica il valori dei λ, in modo da fare aumentare i valori dei $c_j(\lambda) < 0$, ed eventualmente diminuire i valori dei $c_j(\lambda) > 0$;
- un possibile modo di aggiornamento è

$$\lambda_k(r+1) = \max(\lambda_k(r) + q\, s_k(\lambda_k(r))(UB-L(\lambda_k(r)))/||s(\lambda_k(r))||^2)$$

dove:
  - UB è un upper bound per il SCP,
  - $s_k(\lambda_k(r)) = 1 - \sum_{j:k\in F_j} x_j$,
  - $q>0$ è il passo nella direzione del subgradiente, q progressivamente diminuisce (tipicamente si dimezza) quando, dopo un certo numero di iterazioni, il valore di $L(\lambda)$ non aumenta.

Euristiche lagrangiane

- le x_j determinate con il rilassamento lagrangiano NON sono una copertura, i.e., non sono ammissibili per il problema SCP;
- date le x_j si può sempre determinare una soluzione ammissibile per il SCP cambiando alcuni dei valori $x_j = 0$ in $x_j = 1$ in modo greedy, e.g., selezionando iterativamente i sottoinsiemi di costo minimo che coprono gli elementi non già coperti ed eliminando passo passo quei sottoinsiemi che risultino ridondanti;
- noto $L(\lambda)$, $c_j(\lambda)$ e UB è possibile fissare il valore di alcune variabili:
  - $x_j = 0$ se $L(\lambda) + c_j(\lambda) \geq UB$
  - $x_j = 1$ se $L(\lambda) - c_j(\lambda) \geq UB$

  infatti $c_j(\lambda)$ è un lower bound sull'incremento marginale dei costi in cui si incorre selezionando il sottoinsieme j.

Commenti

- si riescono a risolvere in modo esatto problemi SCP reali con centinaia di righe e migliaia di colonne;
- le euristiche lagrangiane sono quelle che hanno prestazione migliore. Le loro prestazioni possono essere migliorate riducendo le dimensioni del problema limitando l'euristica alle variabili che appartengono ad un *core problem* (che può essere statico o dinamicamente aggiornato). Tali variabili vengono selezionate sulla base dei costi ridotti determinati nella prima fase in cui si considera il problema rilassato. Le variabili con costo ridotto più basso hanno infatti maggiore probabilità di appartenere alla soluzione ottima esatta;
- sono state sperimentate anche con qualche successo euristiche basate sugli algoritmi genetici e sulla simulated annealing;
- la difficoltà nei SCP reali risiede anche nel generare tutti i possibili sottoinsiemi ammissibili in modo automatico.

Bibliografia

- Crainic, T.G. and Laporte, G. (1997). Planning Models for Freight Transportation. European Journal of Operational Research, pages 409-438.
- Campbell, A., Clarke, L., Kleywegt, A., and Savelsberg, M. (1998). The Inventory Routing Problem. In T.G. Crainic and G. Laporte, editors, Fleet Management and Logistics, pages 95-113. Kluwer, Norwell, MA.
- Cornuèjols, G., Sridharan, R., and Thizy, J.M. (1991). A Comparison of Heuristics and Relaxations for the Capacitated Plant Location Problem. European Journal of Operational Research, 50:280-297.
- Caprara, A., Fischetti, M., Toth, P. (2000). Algorithms for the Set Covering Problem, Annals of Operations Research 98, 353-371.
- Owen, S.H., Daskin, M.S.(1998), Strategic facility location: A review European Journal of Operational Research 111, 423-447
- Hale, T.S. (2003), Location Science Research: A Review, Annals of Operations Research 123, 21–35

Esercizi

1) Provare che il problema di localizzazione di una singola facility su un piano ha come soluzione ottima la media dei punti in S nel caso in cui la distanza considerata sia la distanza quadratica $d_2(x,s) = \Sigma(x_i - s_i)^2$