Algoritmi di programmazione matematica per il Max Edge Weighted Clique problem with Multiple Choice constraints

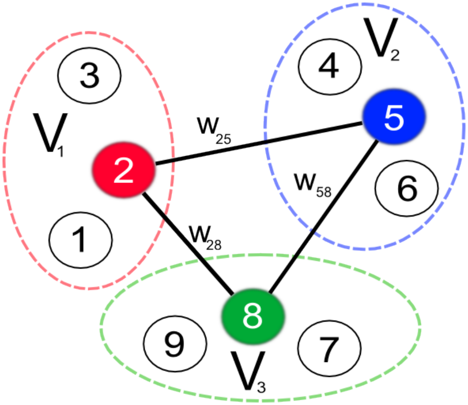
Yari Melzani (703242)

Università degli Studi di Milano
Facoltà di Scienze Matematiche Fisiche e Naturali

16 Ottobre 2009

Relatore:
Prof. Alberto Ceselli
Correlatore:
Prof. Roberto Cordone

UNIVERSITÀ DEGLI STUDI DI MILANO



Applicazioni

- Data Mining
- biologia computazionale
- reti di telecomunicazioni
- management science, ...

Esempio

$$M = \{v_2, v_5, v_8\}$$
$$z(M) = w_2 + w_5 + w_8 + w_{25} + w_{28} + w_{58}$$

Il MEWCMC in letteratura

- Algoritmo B&B basato sul rilassamento Lagrangeano (Bosio ’05)

Problema molto studiato nella versione senza M.C.

- F. Glover 1977, ha proposto un modello di PLI
- Algoritmi euristici (Tabu Search, GRASP)
- Algoritmi esatti (B&B, B&C)

Contenuto della tesi

- Nuovi modelli per il problema (QP, PLI, SDP)
- Algoritmi euristici
- Algoritmo esatto con tecniche di bounding innovative
- Estesa campagna sperimentale
- Confronto dei risultati con la letteratura e ILOG Cplex

$$z = max \sum_{i \in V} \sum_{j \in V} w_{ij} x_i x_j$$

Linearizziamo z imponendo $x_i x_j = y_{ij}$

$s.t.$

$$\sum_{i \in V_k} x_i = 1 \qquad \forall k \in K \quad (1)$$

$$x_i \in \{0, 1\} \qquad \forall i \in V \quad (2)$$

$$y_{ij} \leq x_i \qquad \forall i, j \in V \quad (3)$$

$$y_{ij} \leq x_j \qquad \forall i, j \in V \quad (4)$$

$$y_{ij} \geq x_i + x_j - 1 \qquad \forall i, j \in V \quad (5)$$

$$y_{ij} \in \{0, 1\} \qquad \forall i, j \in V \quad (6)$$

- n variabili decisionali
- $|K|$ vincoli di M.C.

- n^2 nuove variabili y_{ij} e $3n^2$ vincoli

$$
\begin{aligned}
z = \quad & max\, W \bullet Y && (7)\\
s.t. \quad & rank(Y) = 1 && (8)\\
& \sum_k S_k \bullet Y = 1 \ \forall k \in K && (9)\\
& Y \succeq 0 && (10)
\end{aligned}
$$

- Organizziamo le variabili y_{ij} in una matrice Y

$$
Y = xx^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} = \begin{bmatrix} x_1 x_1 & x_1 x_2 & \cdots & x_1 x_n \\ x_2 x_1 & x_2 x_2 & \cdots & x_2 x_n \\ \cdots & \cdots & \cdots & \cdots \\ x_n x_1 & \cdots & \cdots & x_n x_n \end{bmatrix}
$$

- Imponiamo su Y il vincolo di rango e richiediamo che Y sia semidefinita positiva

Euristica primale combinatoria

- Per ogni vertice $i \in V_s$ definisco un contributo potenziale

$$\eta(i) = w_{ii} + \sum_{V_t \in K, t \neq s} \max_{j \in V_t} \{ \frac{w_{jj}}{m-1} + w_{ij} \}$$

- Scelgo in ogni classe il vertice che massimizza tale contributo

Euristica di Tabu Search

1. Costruzione di una soluzione iniziale
2. Determinazione dell'intorno di una soluzione e scelta con politica best-improve
3. Due liste tabu ℓ_e, ℓ_u di lunghezza fissa per impedire l'esplorazione ciclica

- L'euristica fornisce una buona soluzione iniziale per l'algoritmo esatto

Euristica di Rounding

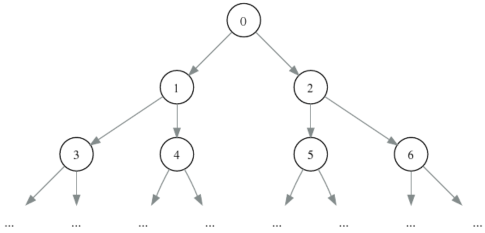
- Arrotondamento delle soluzioni frazionarie fornite dai modelli

Algoritmo esatto basato sulla tecnica Branch and Bound

- Esplorazione di un albero di ricerca in cui i nodi rappresentano i sottoproblemi
- Tecniche di bounding
  - forniscono stime per `eccesso` e per `difetto` della soluzione ottima
  - permettono di stabilire quali sottoproblemi tralasciare

Efficienza dell'algoritmo esatto

- bound combinatorio
- bound semidefinito
- euristiche primali
- strategia di branching
- enumerazione esplicita



$$
\begin{aligned}
z = \quad & \max W \bullet Y \\
s.t. \quad & \cancel{rank(Y) = 1} \\
& \sum_{k \in K} S_k \bullet Y = 1 \\
& Y \succeq 0
\end{aligned}
$$

1. Eliminando il vincolo di rango otteniamo un problema di ottimizzazione convessa nel continuo
2. La soluzione fornisce un upper bound valido per MEWCMC
3. Rafforzamento della formulazione con vincoli di taglio

$$\sum_{j \in V} y_{ij} = m x_i \ \forall i \in V$$

Algoritmo e solutore per SDP

- Algoritmo del cammino centrale permette di risolvere il modello
- Utilizzo del solutore sperimentale DSDP 5.8 (Benson, Ye, 2005)

Rounding

- Applichiamo la tecnica del rounding sulla soluzione del rilassamento

- Bound duale che sfrutta le proprietà combinatorie del problema
- Per ogni classe consideriamo il vertice che ha il massimo contributo potenziale $\eta(i)$

$$\eta(i) = w_{ii} + \sum_{V_t \in K, t \neq s} \max_{j \in V_t} \{ \frac{w_{jj}}{m-1} + w_{ij} \}$$

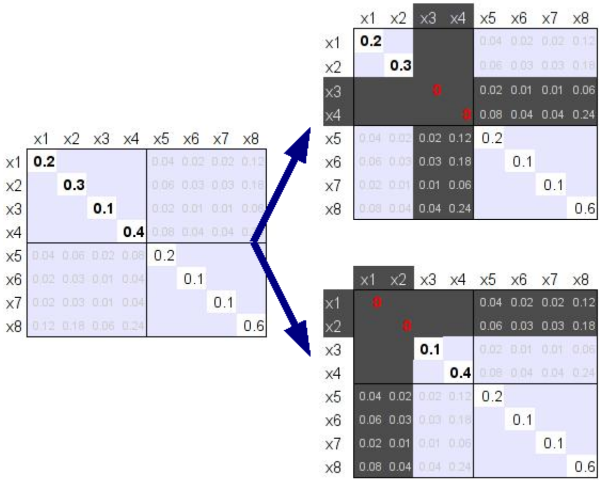
- Considerando tutte le classi della partizione otteniamo una maggiorazione del valore della soluzione ottima

$$\frac{1}{2} \sum_{V_s \in K} \max_{i \in V_s} \{ \eta(i) \}$$

Preprocessing

Riduzione del problema iniziale sfruttando il bound combinatorio

Branching binario bilanciato



$$V_1 = \{x_1, x_2, x_3, x_4\}$$
$$V_2 = \{x_5, x_6, x_7, x_8\}$$

- Le variabili fissate vengono eliminate → il problema diventa più facile
- Soluzione diventa sempre meno frazionaria → il bound si stringe

- La procedura di branching riduce progressivamente la dimensione dei sottoproblemi
- Il numero di soluzioni ammissibili associate ad un sottoproblema P_i è esponenziale nel numero di classi

$$S(P_i) = \prod_{k \in K_i} \ell_k$$

- La valutazione del bound semidefinito ha complessità polinomiale ma è computazionalmente onerosa
- Se la dimensione del sottoproblema è sufficientemente piccola si procede con l'enumerazione esplicita

Implementazione e competitor

Algoritmi implementati in **C** compilati con *gcc* 4.3.3 ed eseguiti su *PC x*86 con processore Intel$^{(R)}$ Core^{TM} 2 Duo a 3.0GHz con 2GB di RAM, in ambiente GNU Linux^{TM}

1. ILOG Cplex 11.2 (modello di PLI con vincoli di taglio)
2. B&B basato sul rilassamento Lagrangeano proposto da Bosio (tesi 2005)

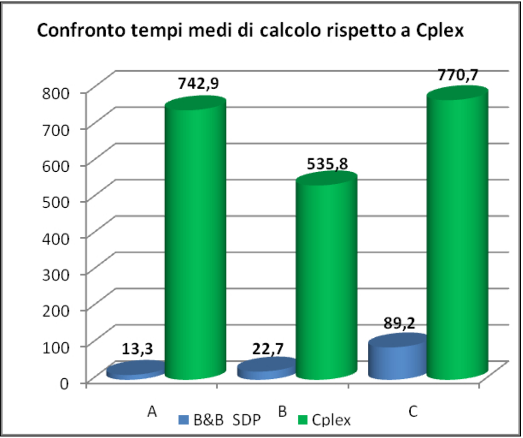
Dataset

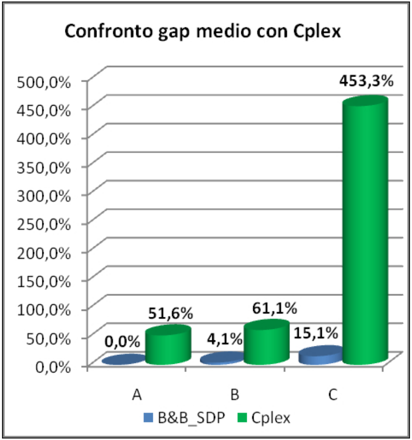
- Test condotti su dataset di 330 istanze (tempo limite di 1ora)
  - **dataset** 1: 165 istanze divise in 3 classi
  - **dataset** 2: 168 istanze divise in 4 classi
- Caratteristiche delle istanze
  - **numero di vertici**: $n = 30, ..., 300$
  - **pesi lati/vertic**i: distanza euclidea, distribuzione uniforme intera
  - **rapporto** $\frac{m}{c}$: $m = 3, ..., 150 \quad c = 2, ..., 100$

Risultati dataset 1 (165 istanze)

Tempo limite di 1 ora:

- *$B\&B_{SDP}$* chiude all'ottimo 127 (77%)
- *Cplex* chiude all'ottimo: 73 (44%)

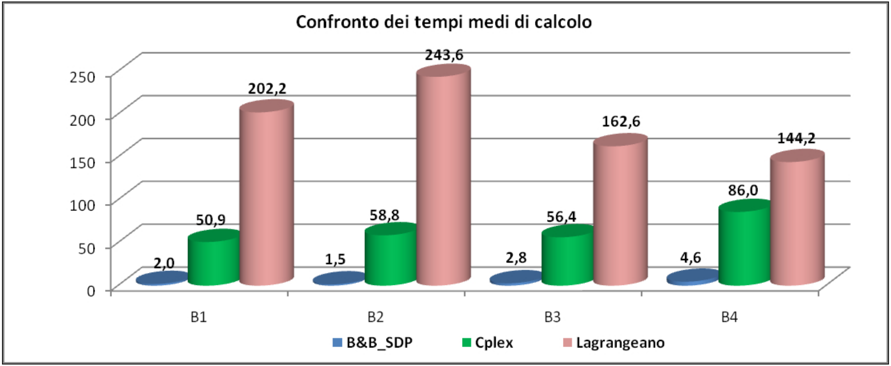




Risultati dataset 2

Tempo limite di 1 ora:

- 168 istanze divise in 4 classi
- Tutti gli algoritmi risolvono all'ottimo le istanze



Conclusioni

- Studio di proprietà combinatorie del problema
- Tecniche di bounding innovative
- L'algoritmo esatto mostra tempi di calcolo fino a 2-3 ordini di grandezza inferiori rispetto a *Cplex*
- Tabu Search individua la soluzione ottima nel 84% delle istanze (gap di ottimalità inferiore al 2%)

Sviluppi futuri

1. Molti problemi reali possono essere ricondotti al MEWCMC tramite l'introduzione di vincoli aggiuntivi
2. Il successo delle tecniche di bounding suggerisce l'utilizzo di tecniche di *second-order conic* programming, ancora fortemente sperimentali