Soluzioni

Esercizio 1

Si provi a realizzare un modello AMPL del *Set Covering Problem* (*SCP*)

$$\begin{aligned} \min z(x) = \sum_{j \in J} c_j x_j & \\ \sum_{j \in J} a_{ij} x_j \geq 1 & \qquad i \in I \\ x_j \in \{0,1\} & \qquad j \in J \end{aligned}$$

e ad applicare un risolutore alle istanze `scpb3.dat` e `scpb4.dat` fornite in allegato, limitando l'esplorazione in modo da rendere euristico il procedimento, operando sul tempo, sul numero di nodi, sul numero di soluzioni e sulle tolleranze per la condizione di ottimalità.

Soluzione Si veda il file `scp.mod`. Si osservi l'efficacia del *presolve*, in particolare nell'eliminare colonne dal problema.

Esercizio 2

Si provi a realizzare un modello AMPL del *Knapsack Problem* (*KP*)

$$\begin{aligned} \max z(x) = \sum_{j \in J} p_j x_j & \\ \sum_{j \in J} w_j x_j \leq c & \qquad i \in I \\ x_j \in \{0,1\} & \qquad j \in J \end{aligned}$$

e ad applicare un risolutore all'istanza `knapPI_16_10000_1000.txt` fornita in allegato, limitando l'esplorazione in modo da rendere euristico il procedimento, operando sul tempo, sul numero di nodi, sul numero di soluzioni e sulle tolleranze per la condizione di ottimalità[1].

Soluzione Si veda il file `kp.mod`. Il risolutore probabilmente troverà molto in fretta una soluzione quasi ottima, ma terminerà lasciando un divario fra soluzione euristica e *upper bound* a causa della tolleranza relativa δ (questo è quel che succede con Gurobi versione 9). Imponendo una tolleranza nulla, il risolutore impiegherà un tempo molto superiore, rimanendo bloccato a lungo su divari piccoli.

[1] Si può osservare una cosa interessante sul risultato ottenuto con i parametri di *default*.

Esercizio 3

Si provi a realizzare un modello AMPL del *Facility Location Problem* (*FLP*) con le due formulazioni discusse nella Lezione 1:

$$\min z(x) = \sum_{i \in I, j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i x_i$$
$$\sum_{i \in I} x_{ij} = d_i \qquad i \in I$$
$$\sum_{j \in J} x_{ij} \leq \left(\sum_{j \in J} d_j \right) y_j \qquad i \in I$$
$$x_{ij} \geq 0 \qquad i \in I, j \in J$$
$$y_i \in \{0,1\} \qquad i \in I$$

e

$$\min z(x) = \sum_{i \in I, j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i x_i$$
$$\sum_{i \in I} x_{ij} = d_i \qquad i \in I$$
$$x_{ij} \leq d_j y_j \qquad i \in I, j \in J$$
$$x_{ij} \geq 0 \qquad i \in I, j \in J$$
$$y_i \in \{0,1\} \qquad i \in I$$

e ad applicare un risolutore all'istanza `gs750a-5.dat` fornita in allegato, limitando l'esplorazione in modo da rendere euristico il procedimento, operando sul tempo, sul numero di nodi, sul numero di soluzioni e sulle tolleranze per la condizione di ottimalità.

Soluzione Si vedano il file `cflp.mod` e `cflp2.mod`. Il secondo modello risolve più velocemente il problema (la differenza non è forte, date le dimensioni ancora limitate dell'esempio), ma dovrebbe richiedere più tempo per risolvere il rilassamento continuo al nodo radice, dato il numero molto superiore di vincoli.

Esercizio 4

Si applichino le tecniche di presolve discusse nella lezione al seguente problema di *PLI*:

$$15 \leq 2x_1 - x_2 + 3x_3 \leq 32$$
$$x_1 \in [0,4], x_2 \in [1,6], x_3 \in [3,5]$$

Soluzione Il vincolo $2x_1 - x_2 + 3x_3 \leq 32$ è ridondante, perché $2u_1 - l_2 + 3u_3 = 2 \cdot 8 - 1 + 3 \cdot 3 = 16 < 32$.

Il vincolo $2x_1 - x_2 + 3x_3 \geq 15$ non è né ridondante né inammissibile. Rafforza i bound su x_1, imponendo $x_1 \geq 4$, dunque $x_1 = 4$; non modifica gli altri. Mi pare non succeda altro.

Esercizio 5

Si applichino le tecniche di presolve discusse nella lezione al seguente problema di *PLI*:

$$\begin{aligned} 3x_1 + 4x_2 &\leq 5 \\ 4x_1 + 2x_2 &\geq 2 \\ x_1, x_2 &\in \{0, 1\} \end{aligned}$$

Soluzione I vincoli dovrebbero essere tutti soddisfacibili e non ridondanti e non rafforzare i bound. Inducono però vincoli logici ausiliari:

- $3x_1 + 4x_2 \leq 5$ induce $x_1 + x_2 \leq 1$
- $4x_1 + 2x_2 \geq 2$ induce $x_1 + x_2 \geq 1$

da cui $x_2 = 1 - x_1$, e quindi $-x_1 \leq 1$ e $2x_1 \geq 0$, da cui non risulta nulla.

Esercizio 6

Si applichino le tecniche di presolve discusse nella lezione al seguente problema di *PLI*:

$$\begin{aligned} -x_1 + x_2 + x_3 + x_4 &\geq 1 \\ x_1 + x_2 + x_3 + 2x_4 &\leq 3 \\ x_1 + x_2 + 2x_3 + 4x_4 &\leq 3 \\ -x_1 - x_2 + x_3 - x_4 &\leq 0 \\ x_1, x_2, x_3, x_4 &\in \{0, 1\} \end{aligned}$$

Soluzione Il vincolo $x_1 + x_2 + 2x_3 + 4x_4 \leq 3$ implica che risulta $x_4 = 0$.

Questo, insieme a $-x_1 + x_2 + x_3 + x_4 \geq 1$ implica che $x_3 = 1$.

Ora il vincolo $x_1 + x_2 + x_3 + 2x_4 \leq 3$ è ridondante.

Ora il vincolo $-x_1 - x_2 + x_3 - x_4 \leq 0$ impone $x_1 + x_2 \geq 1$.