LABORATORIO DI RICERCA OPERATIVA 2020-2021

Laboratorio OPL- LEZIONE 4

Sommario

1. Problemi di PLI
   a) Problemi di Knapsack e sue varianti

Modelli di Programmazione Lineare Intera

$$PLI \begin{cases} \min & c^T x & & \\ & Ax & = & b \\ & x & \geq & 0 \\ & x & \in & \mathbb{Z}^n \end{cases}$$

numero di soluzioni ammissibili "finito" o "infinito"

```
int numVincoli = ...;
int numVariabili = ...;

range Vincoli    = 1..numVincoli;
range Variabili = 1..numVariabili;

float C[Variabili]=...;
float A[Vincoli][Variabili] = ...;
float B[Vincoli] = ...;

dvar int+ X[Variabili];
```

$$PL-01 \begin{cases} \min & c^T x & & \\ & Ax & = & b \\ & x & \in & [0,1]^n \end{cases}$$

numero di soluzioni ammissibili sempre "finito"

```
int numVincoli = ...;
int numVariabili = ...;

range Vincoli    = 1..numVincoli;
range Variabili = 1..numVariabili;

float C[Variabili]=...;
float A[Vincoli][Variabili] = ...;
float B[Vincoli] = ...;

dvar int+ X[Variabili] in 0..1;
```

Problema del Knapsack

Un insieme di n oggetti deve essere sistemato in un contenitore di capacità B. Ogni oggetto j ha un ingombro $w_j > 0$ ed un valore $v_j > 0$. Si vogliono sistemare gli oggetti nel contenitore in modo da massimizzare il valore totale.

$$x_j = \begin{cases} 1 & se\ l'oggetto\ j\ viene\ scelto \\ 0 & altrimenti \end{cases}$$

variabili decisionali

$$max \quad \sum_{j=1}^{n} v_j x_j$$

funzione obiettivo = valore totale degli oggetti scelti

$$\sum_{j=1}^{n} w_j x_j \leq B$$

vincolo di capacità

$$x_j \in \{0,1\} \qquad j = 1, \dots, n$$

Il problema del knapsack (bisaccia) binario è il più "semplice" problema di PLI (un solo vincolo!).

Applicazioni: pianificazione degli investimenti (B= budget, w_j costo dell'investimento j, v_j ricavo dell'investimento j); preparare un bagaglio; ……

Knapsack Intero

Di ciascun oggetto j sono presenti più copie nell'insieme di oggetti

x_j *numero di oggetti di tipo j scelti*

variabili decisionali

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} v_j x_j \\
& \sum_{j=1}^{n} w_j x_j \le B \\
& 0 \le x_j \le u_j \qquad j = 1, \dots, n \\
& x_j \in \mathbb{Z}_+^n \qquad j = 1, \dots, n
\end{aligned}
$$

Knapsack Bounded

u_j numero di copie dell'oggetto j disponibili

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{n} v_j x_j \\
& \sum_{j=1}^{n} w_j x_j \le B \\
& x_j \in \mathbb{Z}_+^n \qquad j = 1, \dots, n
\end{aligned}
$$

Knapsack Unbounded

Applicazioni: pianificazione degli investimenti (acquistare più «quote» dello stesso investimento); caricamento di una nave portacontainer;

Change Making

E' un caso particolare del Knapsack Bounded in cui il valore degli oggetti v_j sono tutti unitari, il vincolo di capacità è un uguaglianza ed la funzione obiettivo è di minimo.

$x_j \quad numero\ di\ oggetti\ di\ tipo\ j\ scelti$

$$
\boxed{
\begin{array}{lll}
min & \sum_{j=1}^{n} x_j & \\
& \sum_{j=1}^{n} w_j x_j = B & \\
& 0 \leq x_j \leq u_j & j = 1, \dots, n \\
& x_j \in \mathbb{Z}_+^n & j = 1, \dots, n
\end{array}
}
$$

Ogni oggetto j corrisponde ad un tipo di moneta-banconota; l'ingombro w_j corrisponde al valore della moneta-banconota j; B rappresenta una certa somma di denaro. Il problema è quello di determinare come «raggiungere» il valore di denaro B (ad esempio il resto) utilizzando il minor numero di monete.

Subset Sum

E' un caso particolare di Knapsack in cui il peso ed il valore di ciascun oggetto coincidono ($w_j = v_j$).

$$max \quad \sum_{j=1}^{n} v_j x_j$$

$$\sum_{j=1}^{n} v_j x_j \leq B$$

$$x_j \in \{0,1\} \qquad j = 1, \dots, n$$

Dati n numeri interi positivi $v_1, v_2, \dots, v_n$ ed un valore prefissato B, determinare un sottoinsieme dei numeri tali che la loro somma sia massima e non superiore a B.

Se $B = \frac{1}{2}\sum_{j=1}^{n} v_j$, il problema del Subset Sum è quello di suddividere l'insieme dei numeri in due sottoinsiemi il più bilanciato possibile.

Knapsack Multiplo

Un insieme di n oggetti deve essere sistemato in un insieme di m contenitori $(m \leq n)$ ciascuno di capacità b_i. Ogni oggetto j ha un ingombro w_j ed un valore v_j . Si vuole decidere come sistemare gli oggetti nei contenitori in modo tale da massimizzare il profitto totale.

$$x_{ij} = \begin{cases} 1 & se\ l'oggetto\ j\ viene\ inserito\ nel\ contenitore\ i \\ 0 & altrimenti \end{cases}$$

variabili decisionali

$$max \quad \sum_{i=1}^{m} \sum_{j=1}^{n} v_j x_{ij}$$

$$\sum_{i=1}^{m} x_{ij} \leq 1 \qquad j = 1, \dots, n$$

vincolo di "scelta multipla": ogni oggetto può essere inserito in al più un contenitore

$$\sum_{j=1}^{n} w_j x_{ij} \leq b_i \quad i = 1, \dots, m$$

vincolo di capacità per ciascun contenitore

$$x_{ij} \in \{0,1\} \qquad i = 1, \dots, m, j = 1, \dots, n$$

Bin Packing

Un insieme di n oggetti deve essere sistemato in m contenitori ciascuno di capacità B. Ogni oggetto j ha un ingombro w_j. Si vogliono sistemare tutti gli oggetti nei contenitori in modo tale da minimizzare il numero di contenitori utilizzati.

$$x_{ij} = \begin{cases} 1 & se\ l'oggetto\ j\ viene\ inserito\ nel\ contenitore\ i \\ 0 & altrimenti \end{cases}$$

$$y_i = \begin{cases} 1 & il\ contenitore\ i\ viene\ utilizzato \\ 0 & altrimenti \end{cases}$$

variabili decisionali

$$min \sum_{i=1}^{m} y_i$$

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \dots, n$$

vincoli di "scelta multipla": ogni oggetto deve essere inserito in esattamente un contenitore

$$\sum_{j=1}^{n} w_j x_{ij} \le B y_i \qquad i = 1, \dots, m$$

vincolo di capacità per ciascun contenitore effettivamente utilizzato: se il contenitore i è utilizzato ($y_i = 1$), allora l'ingombro totale degli oggetti inseriti in i non può superare B; se il contenitore i non è utilizzato ($y_i = 0$), nessuno oggetto può essere inserito in i.

$$x_{ij} \in \{0,1\} \qquad i = 1, \dots, m, j = 1, \dots, n$$

$$y_i \in \{0,1\} \qquad i = 1, \dots, m$$

Dati per un problema di Knapsack Multiplo

$m = 2, n = 10$ numero contenitori, numero di oggetti

$v = [78,35,89,36,94,75,74,79,80,16]^T$ valore degli oggetti

$w = [18,9,23,20,59,61,70,75,76,30]^T$ ingombro degli oggetti

$b = [103,156]^T$ capacità di contenitori

Dati per un problema di Bin Packing (Esercizio per casa)

m=10, n=50;
B=150

w=[92 91 90 89 85 84 81 80 80 78 78 77 77 76 75 74 73 69 69 68 68 67 67 65 64 63 63 61 60 56 54 54 51 49 45 43 42 39 39 39 38 36 35 34 34 33 32 31 30 50]

Assegnamento Generalizzato

Un insieme di n oggetti deve essere sistemato in m contenitori ciascuno di capacità b_i. Ogni oggetto j ha un ingombro w_{ij} ed un valore v_{ij} che dipendono quindi anche dal contenitore. Si vogliono sistemare tutti gli oggetti nei contenitori in modo tale da massimizzare il valore totale.

$$x_{ij} = \begin{cases} 1 & se\ l'oggetto\ j\ viene\ inserito\ nel\ contenitore\ i \\ 0 & altrimenti \end{cases}$$

variabili decisionali

$$max \sum_{i=1}^{m} \sum_{j=1}^{n} v_{ij} x_{ij}$$

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \dots, n$$

vincolo di "scelta multipla": ogni oggetto deve essere inserito in esattamente un contenitore

$$\sum_{j=1}^{n} w_{ij} x_{ij} \leq b_i \qquad i = 1, \dots, m$$

vincolo di capacità per ciascun contenitore

$$x_{ij} \in \{0,1\} \qquad i = 1, \dots, m, j = 1, \dots, n$$

Applicazioni: contenitori = macchine, oggetti = lavorazioni, w_{ij} = tempo impiegato dalla macchina i per eseguire la lavorazione j, v_{ij} valore aggiunto alla lavorazione j se eseguita dalla macchina i (oppure costo sostenuto per eseguire la lavorazione j sulla macchina i in caso di problema di minimo)

Esercizio per casa 1 – Assegnamento Generalizzato

$m = 5, n = 10$

$$v = \begin{bmatrix} 110 & 65 & 19 & 89 & 62 & 37 & 89 & 78 & 74 & 88 \\ 16 & 69 & 93 & 31 & 17 & 115 & 102 & 96 & 27 & 97 \\ 25 & 54 & 45 & 72 & 77 & 87 & 98 & 87 & 99 & 99 \\ 78 & 28 & 45 & 83 & 18 & 59 & 74 & 55 & 91 & 99 \\ 59 & 71 & 9 & 20 & 39 & 97 & 61 & 77 & 5 & 51 \end{bmatrix}$$

$$w = \begin{bmatrix} 95 & 54 & 3 & 72 & 44 & 20 & 72 & 75 & 68 & 69 \\ 1 & 53 & 91 & 30 & 1 & 99 & 96 & 82 & 8 & 83 \\ 21 & 44 & 43 & 56 & 71 & 87 & 97 & 83 & 87 & 98 \\ 66 & 26 & 42 & 72 & 13 & 52 & 73 & 44 & 74 & 88 \\ 59 & 60 & 5 & 9 & 27 & 85 & 49 & 59 & 4 & 45 \end{bmatrix}$$

Tempi di lavorazione macchina-task

$b = [91, 87, 109, 88, 64]^T$

Tempo massimo di operatività per ciascuna macchina

Con i dati riportati, risolvere il problema di assegnamento generalizzato interpretando i coefficienti v_{ij} $i = 1, \dots, m$, $j = 1, \dots, n$

1. come valore aggiunto al task j se eseguito dalla macchina i (funzione obiettivo max)
2. come costo sostenuto per eseguire il task j sulla macchina i (funzione obiettivo min)

Confrontare la due soluzioni.