LABORATORIO DI RICERCA OPERATIVA 2020-2021

Laboratorio OPL - LEZIONE 2

Prototipo di un Modello di PL in OPL

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

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

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

dvar float+ X[Variabili];

minimize sum(j in Variabili)C[j]*X[j];

 subject to{
   forall(i in Vincoli)
   vincoli: sum(j in Variabili) A[i][j]*X[j] >= ( ==, <= ) B[i];
  }
```

$$\begin{array}{llll} \min & c^T x & & \\ & Ax & \leq (\geq, =) & b \\ & x & \geq & 0 \end{array}$$

Esercizio 1: Pianificazione della produzione

Una pasticceria vuole programmare la produzione di cioccolatini nei mesi di dicembre, gennaio, febbraio, marzo e aprile. La pasticceria ha una capacità produttiva di 100 kg di cioccolatini al mese, ed il costo di produzione è di 18 €/kg. Per far fronte a picchi di domanda, i pasticceri possono svolgere delle ore di straordinario incrementando così la produzione mensile di altri 25 kg, ad un costo di produzione di 22 €/kg. In base alle vendite degli anni precedenti, la pasticceria ha stimato la domanda di cioccolatini in 120Kg per il mese di dicembre, 110 kg per il mese di gennaio, 90kg per il mese di febbraio, 40kg per marzo, e 130kg per aprile. I cioccolatini hanno una data di scadenza di 5 mesi. Pertanto tutto ciò che è stato prodotto in un mese ma è rimasto invenduto viene conservato in un'apposita cella a temperatura controllata e costituisce, di fatto, una scorta per i mesi successivi. La cella ha una capacità di 20kg ed il suo costo di funzionamento è di 0,3€/kg. Assumendo che a dicembre il livello di scorte in cella sia 10kg e che ad aprile non devono rimanere cioccolatini invenduti, si vuole determinare il piano di produzione mensile che minimizzi i costi di produzione

Sintesi di un modello di Programmazione Matematica

Analisi del Problema

Cosa è noto? Cosa bisogna determinare? Qual è l'obiettivo? Quali sono i vincoli?

Costruzione del Modello

Scelta delle variabili decisionali. Formulazione matematica dell'obiettivo e dei vincoli.

Analisi del Modello

Studio delle proprietà matematiche del modello: esistenza soluzione ottima, criteri di ottimalità.

Soluzione

Calcolo della soluzione ottima mediante algoritmi di ottimizzazione eseguiti by «hand» oppure implementati in strumenti software.

Validazione

I risultati sono coerenti? Il modello è predittivo? Tale fase si effettua mediante un'analisi sperimentale oppure attraverso strumenti di simulazione.

Analisi del Problema: cosa è noto e cosa è da determinare

Dati:

1. Mesi di produzione: M = {Dic, Gen, Feb, Mar, Apr}={1,2,3,4,5}
2. Capacità produttiva 'normale' Q^N = 100 kg/mese
3. Costo produzione 'normale': C^N = 18 €/kg
4. Capacità produttiva 'extra': Q^E = 25 kg/mese
5. Costo produzione 'extra': C^E = 22 €/kg
6. Domanda Dicembre: D^1 = 120 kg
7. Domanda Gennaio: D^2 = 110 kg
8. Domanda Febbraio: D^3 = 90 kg
9. Domanda Marzo: D^4 = 40 kg
10. Domanda Aprile: D^5 = 130 kg
11. Costo conservazione: C^S = 0,3 €/kg
12. Stato iniziale delle scorte: S^0 = 10kg;
13. Capacità della cella: Q^C= 20kg;
14. Stato finale delle scorte;

Incognite:

- a. Quantità in kg da produrre in ciascun mese i = 1,2,3,4,5
- b. Quantità extra in kg da produrre in ciascun mese i = 1,2,3,4,5
- c. Quantità in kg conservata in ciascun mese i = 1,2,3,4,5

Analisi del Problema: vincoli e obiettivo

Sulle capacità produttive

- la pasticceria ha una capacità produttiva di 100 kg di cioccolatini al mese
- i pasticceri possono svolgere delle ore di straordinario incrementando la produzione mensile di altri 25 kg

Sullo finale delle scorte

- ad aprile non devono rimanere cioccolatini invenduti

Sulla capacità della cella

- la cella ha una capacità di 20kg

Soddisfacimento della domanda mensile

- la quantità di cioccolatini disponibili per essere venduti in ciascun mese è pari alla quantità prodotta in quel mese + quella rimasta invenduta nei mesi precedenti

Vincoli

Obiettivo

- minimizzare i costi di produzione

Proviamo a scrivere il modello matematico in forma «compatta» con tutti vincoli <=

$$\boxed{\begin{array}{llll} min & c^T x & & \\ & Ax & \leq & b \\ & x & \geq & 0 \end{array}}$$

#Variabili Decisionali = 15 $\Longrightarrow x \in \mathbb{R}^{15}_{+}$

Quantità in kg da produrre in ciascun mese i = 1,2,3,4,5 $\Longrightarrow x_1, x_2, x_3, x_4, x_5$

Quantità extra in kg da produrre in ciascun mese i = 1,2,3,4,5 $\Longrightarrow x_6, x_7, x_8, x_9, x_{10}$

Quantità in kg conservata in ciascun mese i = 1,2,3,4,5 $\Longrightarrow x_{11}, x_{12}, x_{13}, x_{14}, x_{15}$

Scriviamo i vincoli: per ciascun mese la produzione std è al più 100kg di cioccolatini

mese 1: $x_1 \leq 100$ $\quad a_1^T = [1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0]$ riga 1 di A

mese 2: $x_2 \leq 100$ $\quad a_2^T = [0 \; 1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0]$ riga 2 di A

etc…. fino al mese 5

Scriviamo i vincoli: per ciascun mese la produzione extra è al più 25kg di

mese 1: $x_6 \leq 25$ $\quad a_6^T = [0 \; 0 \; 0 \; 0 \; 0 \; 1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0]$ riga 6 di A

mese 2: $x_7 \leq 25$ $\quad a_7^T = [0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0]$ riga 7 di A

etc…. fino al mese 5

Il modello sarà corretto ma illeggibile

Costruzione del Modello (scelta delle variabili decisionali)

Quantità di cioccolatini da produrre al mese in condizioni «normali»

- $x_i >= 0 \quad i = 1, 2, 3, 4, 5$

Quantità di cioccolatini da produrre al mese con lo «straordinario»

- $y_i >= 0 \quad i = 1, 2, 3, 4, 5$

Quantità di cioccolatini conservati nella cella per ciascun mese

- $z_i >= 0 \quad i = 1, 2, 3, 4, 5$

Costruzione del Modello (Formulazione del Problema)

Vincoli

Sulle capacità produttive

- $x_i \leq Q^N \quad (x_i \leq 100) \qquad i = 1,2,3,4,5$
- $y_i \leq Q^E \quad (y_i \leq 25) \qquad i = 1,2,3,4,5$

Sullo finale delle scorte

- $z_5 = 0$

Sulla capacità della cella

- $z_i \leq Q^C \quad (z_i \leq 20) \qquad i = 1,2,3,4,5$

Soddisfacimento della domanda (vincoli di bilancio)

- $x_1 + y_1 + S_0 = D^1 + z_1$
- $x_2 + y_2 + z_1 = D^2 + z_2$
- $x_3 + y_3 + z_2 = D^3 + z_3$
- $x_4 + y_4 + z_3 = D^4 + z_4$
- $x_5 + y_5 + z_4 = D^5 + z_5$

Funzione obiettivo da minimizzare

$$Q^N(x_1 + x_2 + x_3 + x_4 + x_5) + C^E(y_1 + y_2 + y_3 + y_4 + y_5) + C^S(z_1 + z_2 + z_3 + z_4)$$

Tipi di Dati Strutturati 1: Record

```
tuple nomeRecord {
      tipo campo_1;
      ...
      tipo campo_n;
};
```

`tipo` è un tipo base (integer, float, string) oppure un tipo composto (un record, un array di int o di float, ….). Non è possibile usare array di stringhe o array multidimensionali

```
nomeRecord t = <campo_1, campo_2,.., campo_n>
```

```
tuple Point {
      int x;
      int y
};
```

`Point p = <2, 5>;` un punto

`Point S[1..5] = [<1,2>,<3,5>,<7,2>,<10,6>,<7,0>];` array di punti

```
int ascissa = p.x; (ascissa =2)
int ordinata = S[4].y (ordinata = 6)
```

Tipi di Dati Strutturati 2: Set

```
{tipo} nomeSet = {listaElementi};

{int} SetA = {3, 5, 7, 1, 4, 10};

{string} Days = {Lun, Mar, Mer, Gio, Ven, Sab, Dom};
```

- non sono ammessi i duplicati;
- sono ammesse le operazioni classiche sugli insiemi;
- gli elementi di un set possono essere usati come indici di array

```
    int oreLavorate[Days] = [10, 2, 13, 42, 25, 6, 0];
```

- è possibile usare i set come argomenti di forall e sum

Inizializzazione di Set

Inizializzazione Interna

```
{int} s1 = {1,2,3};
{int} s2 = {1,4,5};
```

Elencando i valori

```
{int} intersezione  = s1 inter s2;
```

Come il risultato di un'operazione

Dichiarandone le proprietà: Insieme dei numeri pari nel range [1,10]

```
{int} even = {i | i in 1..10: i mod 2 == 0};
```

Filtro

Inizializzazione Esterna

Nel file "mod" la dichiarazione

```
tuple punto {
      int x;
      int y;
};
{punto} InsiemePunti = ...;
```

Nel file "dat" la lista dei valori

```
InsiemePunti ={<1,2>, <1,3>,<1,5>}
```

Grafi in OPL

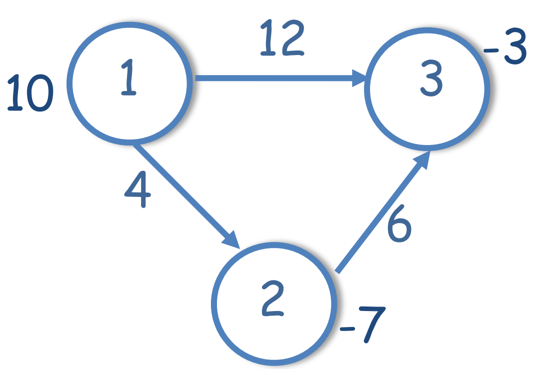
Un grafo G è la coppia $G =< V,E >$, dove V è l'insieme dei nodi e E è un sottoinsieme di $V \times V$ ed è l'insieme di archi o rami, a seconda se le coppie $(i,j) \in E$ sono ordinate o meno. Gli elementi di V e A possono avere degli "attributi" (pesi).

Digrafi pesati su nodi ed archi - Implementazione 1

```
tuple nodo {
        int IndiceNodo;
        int PesoN_1;
        ...
        int PesoN_k;

};
{nodo} InsiemeNodi = ...;
```

```
tuple arco {
        int NodoOut;
        int NodoIn;
        int PesoA_1;
        ...
        int PesoA_q;
};
{arco} InsiemeArchi = ...;
```




```
InsiemeNodi={<1,10>, <2,-7>, <3,-3>}

InsiemeArchi={<1,3,12>, <2,3,6>, <1,2,4>}
```

Nota: i pesi di archi e nodi possono essere eterogenei, ovvero di tipo diverso

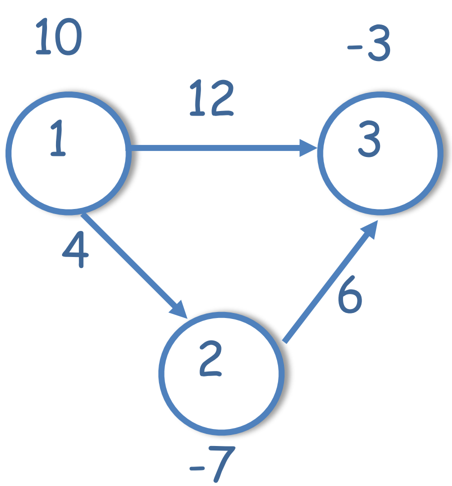
Grafi in OPL

Grafi orientati pesati su nodi ed archi – Implementazione 2

```
int NNodi = ...;
range Nodi = 1..NNodi;

int PesoN_1[Nodi] = ...;
int PesoN_2[Nodi] = ...;
...
int PesoN_k[Nodi] = ...;
```

```
tuple arco {
        int NodoOut;
        int NodoIn;
};
{arco} InsiemeArchi = ...;
int PesoA_1[InsiemeArchi]= ...;
...
int PesoA_q[InsiemeArchi]= ...;
```




```
InsiemeArchi={<1,3>, <2,3>, <1,2>}
PesoA[InsiemeArchi]=[12, 6,3];
PesoN[Nodi]=[10,-7,-3]
```

- I pesi di nodi ed archi sono «separati» dagli elementi
- I nodi sono rappresentati come range (poco flessibile nel caso si voglia rappresentare un grafo in cui i nodi hanno “nomi” piuttosto che “indici”

Flussi di costo minimo

$$\min \sum_{(i,j)\in A} c_{ij} f_{ij}$$

$$\sum_{j|(i,j)\in A} f_{ij} - \sum_{j|(j,i)\in A} f_{ji} = b_i \qquad \forall i \in V$$

$$f_{ij} \geq 0 \qquad (i,j) \in A$$