Codici di calcolo per la PL e la PLI

Premessa

Ai sostanziali progressi ottenuti nella programmazione matematica dagli anni '50 agli anni ‘80 non è corrisposto un adeguato utilizzo applicativo degli strumenti matematici sviluppati. Questo è dovuto, in gran parte, alle difficoltà computazionali ed informatiche insite nella stesura del modello, nella raccolta ed organizzazione dei dati, nella programmazione degli algoritmi solutori e nell'analisi dei risultati ottenuti.

Premessa (continua)

Quindi, gli sforzi ottenuti in campo matematico, risultavano inutilizzabili a livello applicativo poiché l'effettivo incremento nella velocità e potenza dei solutori si scontrava con la necessità di creare software che interrogasse questi solutori e che gestisse modelli e dati appartenenti al mondo reale.

Problemi di programmazione lineare (LP)

Min $c \bullet x$

Soggetto a:

$$Ax \geq b$$

$$x \geq 0$$

Problemi di programmazione lineare intera

Min $c \bullet x$

Soggetto a:

$Ax \geq b$

$x \geq 0$, $x \in Z^n$

Problemi di programmazione lineare mista intera (MIP)

Min $c \bullet x$

Soggetto a:

$Ax \geq b$

$x \geq 0$, $x_i \in S \subseteq X \subseteq Z^n$

Problemi di programmazione lineare binaria

Min $c \bullet x$

Soggetto a:

$Ax \geq b$

$x \in \{0, 1\}^n$

Alcune librerie di solutori per la LP e la MIP

- XMP (U. Arizona)
- OSL (IBM)
- Lindo (Lindo Inc)
- CPLEX (ILOG)
- XPRESS (Dash)
- GLPK (GNU)
- SOPLEX (ZIB)

Unix

DLL

WIN

Fortran
Anni 80

Stato dell'arte
C/C++
Anni 90

Free software
C/C++
Ultimi anni

Altri solutori

- CONOPT LP, NLP
- FortMP LP, MIP, BAR, QMIP (OptiRisk Systems)
- XA LP, MIP (Sunset Software Technologies)
- OML LP, MIP (Ketron Management)
- FrontLine LP, MIP (FrontLine Systems) in Ms Excel
- LPSolve LP, MIP (free)
- PCx BAR (Argonne National Laboratories)
- LSGRG2 LP, NLP (Optimal Methods) in MsExcel

Ambienti di sviluppo

Codici

programmazione a basso livello

Modelli

programmazione ad alto livello

Ambienti di sviluppo codici

Branch & Bound, Branch & Cut, Branch & Price
programmazione a basso livello, C e C++

- ABACUS (ZIB)
- MINTO (Georgia Tech)
- SYMPHONY (Cornell University) parallelo
- Concert Technology (ILOG)

Ambienti di sviluppo di modelli

- General Algebraic Modeling System (GAMS)
- Model Development Environment (MPL)
- LINGO (Lindo)
- AMPL (Cplex)
- XPRESS-L (Xpress)
- ZIMPL (Soplex)
- GLPK-L (Glpk)

Programmazione lineare e lineare intera

Obiettivi di questi ambienti

- fornire un linguaggio di programmazione ad alto livello che permetta di descrivere in modo semplice modelli reali anche molto complessi
- creare modelli indipendenti dal solutore utilizzato, in modo da poter sfruttare sempre i solutori più potenti presenti sul mercato
- descrivere modelli nei quali la struttura logica del problema e i dati utilizzati possano essere considerati come entità diverse

Principali caratteristiche

- semplicità: rappresentazione concisa ed elegante del modello in un unico documento
- leggibilità: l'input e l'output comprensibili ad altre persone ed accessibili con un editor di testo
- debugging: nell'output vengono rappresentate tutte le operazioni svolte dal solutore
- portabilità: indipendenti dalla piattaforma sulla quale vengono eseguiti programma e solutore
- database: interfaccia con database ODBC

Ambienti di sviluppo di modelli

- General Algebraic Modeling System (GAMS)

Il primo, il più diffuso, il più potente

- Model Development Environment (MPL)

Interfaccia utente, facilità d'uso, database

- Esempi di problemi LP e MIP

Il linguaggio GAMS (General Algebraic Modeling System)

versione student demo della release 21.5 www.gams.com

Tipi di modelli in GAMS

- Programmazione lineare (LP)
- Programmazione lineare mista intera(MIP)
- Programmazione non lineare (NLP)
- P non lineare con derivate discontinue (DNLP)
- Program non lineare mista intera (MINLP)
- Mixed Complementarity Problem (MCP)
- Sistema non lineare vincolato (CNS)
- Pr matematici con vincoli di equilibrio (MPEC)

Il linguaggio MPL (Model Development Environment)

Solutori in MPL

- CPLEX LP, MIP, BAR
- XPRESS LP, MIP, BAR
- OSL LP, MIP, BAR
- Lindo LP, MIP
- CONOPT LP, NLP
- FortMP LP, MIP, BAR, QMIP (OptiRisk Systems)
- XA LP, MIP (Sunset Software Technologies)
- OML LP, MIP (Ketron Management)
- FrontLine LP, MIP (FrontLine Systems) in Ms Excel
- LPSolve LP, MIP (free)
- PCx BAR (Argonne National Laboratories)
- LSGRG2 LP, NLP (Optimal Methods) in MsExcel

Model Development Environment (MPL)

- un linguaggio di programmazione ad alto livello per descrivere modelli di Programmazione Matematica, poche parole chiave
- versione studenti di MPL con licenza d'uso semestrale al sito http://www.maximal-usa.com
- modelli specifici con dati assegnati, oppure modelli più generali, che vengono associati di volta in volta con i dati letti da un file esterno
- piccola libreria con esempi di modelli
- un programma MPL è diviso in due parti

Parole chiave
Prima parte: definizioni

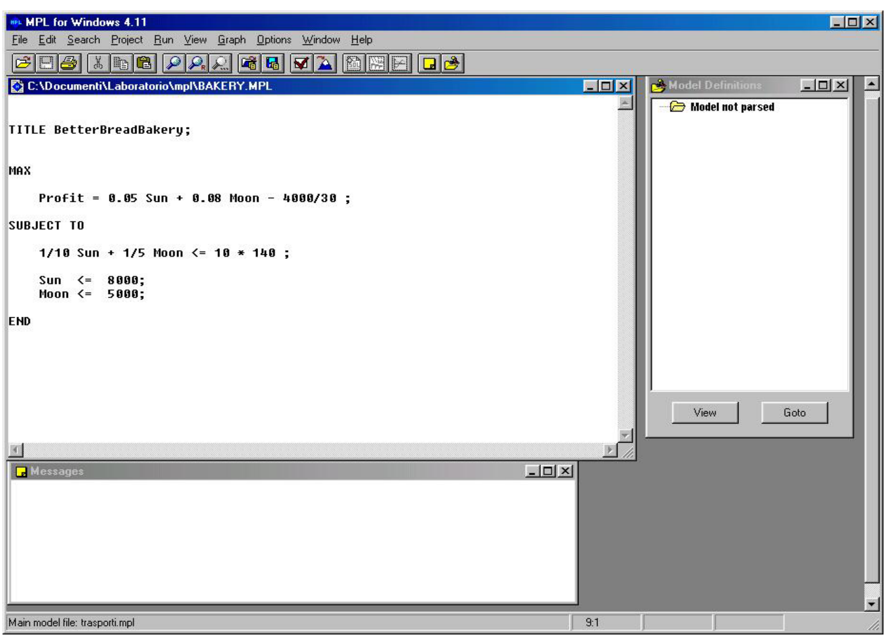
- TITLE - Il nome del modello
- INDEX - Gli indici del problema
- DATA - I dati (scalari, vettori e matrici multidimensionali)
- DECISION - Le variabili decisionali del problema
- MACRO – Simboli (=espressioni) utilizzabili nelle varie espressioni del modello
- Ogni riga finisce con ;

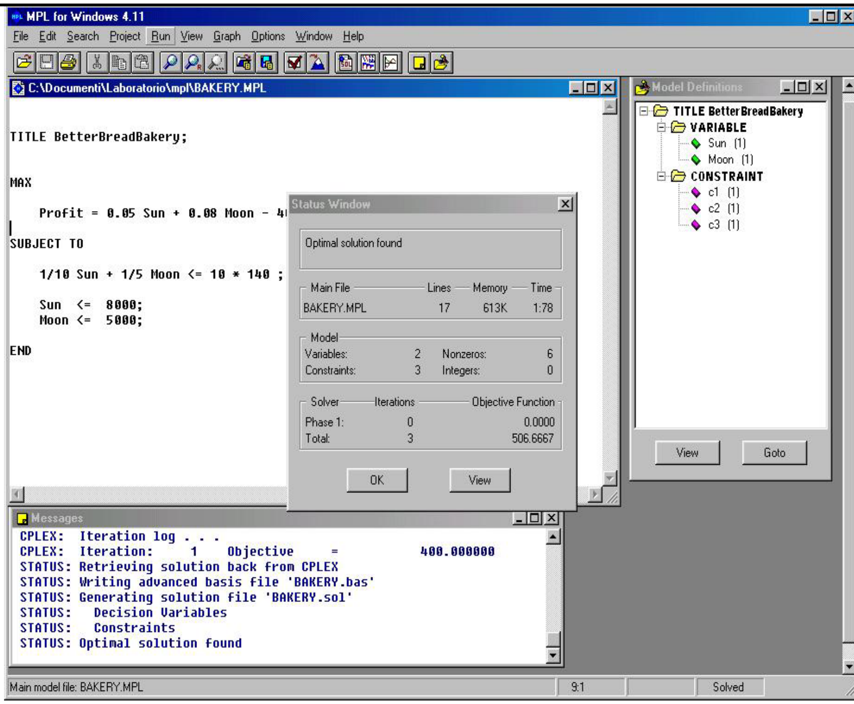
Parole chiave
Seconda parte: modello

- MODEL - Descrizione del problema
- MAX o MIN - Funzione obiettivo
- SUBJECT TO –Vincoli: [definizione: espressione]
- SUM(indici: sommando) –Σ_{indici}: sommando $_{\text{indici}}$
- BOUNDS - Estremi superiori ed inferiori delle variabili
- POSITIVE –Variabili non negative
- FREE - Variabili libere
- INTEGER - Variabili intere
- BINARY - Variabili binarie (0/1)
- END - Fine del modello

Scrivere e risolvere un Modello con MPL

- Installato MPL, il programma di installazione ha creato una voce nel menù di Avvio per eseguire l'applicazione MPL
- Scegliamo Open nel menù File
- Selezioniamo, per esempio, il file bakery.mpl
- Scegliere Solve CPLEX nel menù Run
- MPL registra automaticamente la soluzione in un file “ .sol”
- per visualizzare il file di soluzione bakery.sol premiamo il tasto View sul fondo della Status Window





Un modo semplice di usare MPL

Bakery.mpl
si può ottenere
con un
Cut &Paste
da un editor
di testo all'editor
in MPL

```
TITLE BetterBreadBakery;
MAX
    Profit = 0.05 Sun + 0.08 Moon - 4000/30 ;
SUBJECT TO
    1/10 Sun + 1/5 Moon <= 10 * 140 ;
    Sun <= 8000;
    Moon <= 5000;
END
```

```
    MPL Modeling System   -   Copyright (c) 1988-2000, Maximal Software, Inc.
--------------------------------------------------------------------------------
MODEL STATISTICS
Problem name:       BetterBreadBakery
Filename:           BAKERY.MPL
Date:               October 19, 2001
Time:               19:44
Parsing time:       0.11 sec
Solver:             CPLEX 300
Objective value:    506.666666667
Iterations:         3
Solution time:      0.77 sec
Constraints:        3
Variables:          2
Nonzeros:           4
Density:            67 %
SOLUTION RESULT
  Optimal solution found
    MAX Profit    =       506.6667
DECISION VARIABLES
PLAIN VARIABLES
    Variable Name               Activity     Reduced Cost
  -------------------------------------------------------
    Sun                        8000.0000          0.0000
    Moon                       3000.0000          0.0000
  -------------------------------------------------------
CONSTRAINTS
PLAIN CONSTRAINTS
    Constraint Name                Slack     Shadow Price
  -------------------------------------------------------
    c1                            0.0000          0.4000
    c2                            0.0000          0.0100
    c3                         2000.0000          0.0000
  -------------------------------------------------------
END
```

Bakery.sol

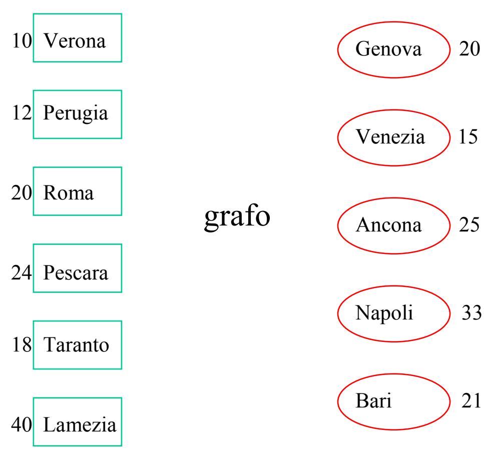
Un problema di trasporto

Una ditta di trasporto deve trasferire container vuoti dai propri magazzini ai principali porti nazionali

Le disponibilità di container vuoti ai magazzini e le richieste ai porti sono le seguenti:

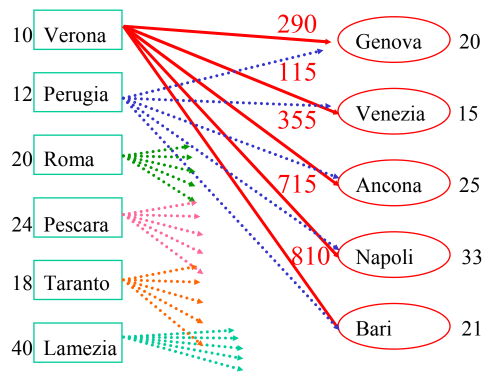
Tabella

Magazzini	Disponi bilità	Porti	richieste
Verona	10	Genova	20
Perugia	12	Venezia	15
Roma	20	Ancona	25
Pescara	24	Napoli	33
Taranto	18	Bari	21
Lamezia	40		



Distanze

	Genova	Venezia	Ancona	Napoli	Bari
Verona	290	115	355	715	810
Perugia	380	340	165	380	610
Roma	505	530	285	220	450
Pescara	655	450	155	240	315
Taranto	1010	840	550	305	95
Lamezia	1072	1097	747	372	333



Modello di PL

- i := magazzino, i ∈I
- j := porto, j ∈J
- a(i) = disponibilità di i
- b(j) = richiesta di j
- d(i,j) = distanze tra i e j
- C = costo al km di trasporto di un container
- x(i,j) = variabili, numero di containter spediti dal magazzino i al porto j

Formulazione del problema del trasporto a costo minimo

$$\text{Min} \sum_{\forall (i,j) \in I \times J} C \bullet d_{ij} \bullet x_{ij} \qquad (1)$$

$$\text{Soggetto a:} \quad \sum_{\forall (i,j) \text{ tale che } j \in J} x_{ij} \leq a(i) \quad \text{per } i \in I \qquad (2)$$

$$\sum_{\forall (j,i) \text{ tale che } i \in I} x_{ji} \geq b(j), \text{ per } j \in J \qquad (3)$$

$$x_{ij} \geq 0, \text{ per } (i,j) \in I \times J \qquad (4)$$

Modello MPL

```
      TITLE
         Problema_di_Trasporto;

      INDEX
i        magazzini = ( Verona, Perugia, Roma, Pescara, Taranto, Lamezia );
j        porti = ( Genova, Venezia, Ancona, Napoli, Bari );

      DATA
a(i)     disponibilita[magazzini] = ( 10, 12, 20, 24, 18, 40 );
b(j)     richiesta[porti] = ( 20, 15, 25, 33, 21 );

d(i,j)   distanze[magazzini, porti] = DATAFILE ("Distanze.dat");

C        costo_unitario = 300;        ! costo per chilometro

      DECISION
         ! numero di containter spediti dal magazzino i al porto j
x(i,j)   x[magazzini, porti]

      MODEL
         ! funzione obiettivo
(1)      MIN  costoTotale = Sum(magazzini, porti: distanze * x * costo_unitario);

      SUBJECT TO
(2)      vincolo_disponibilita[magazzini] :  Sum(porti: x) <= disponibilita;
(3)      vincolo_richiesta[porti] :  Sum(magazzini: x) >= richiesta;

      END
```

Distanze.dat

{datafile per TRASPORTI.MPL }

	Genova	Venezia	Ancona	Napoli	Bari
Verona	290	115	355	715	810
Perugia	380	340	165	380	610
Roma	505	530	285	220	450
Pescara	655	450	155	240	315
Taranto	1010	840	550	305	95
Lamezia	1072	1097	747	372	333

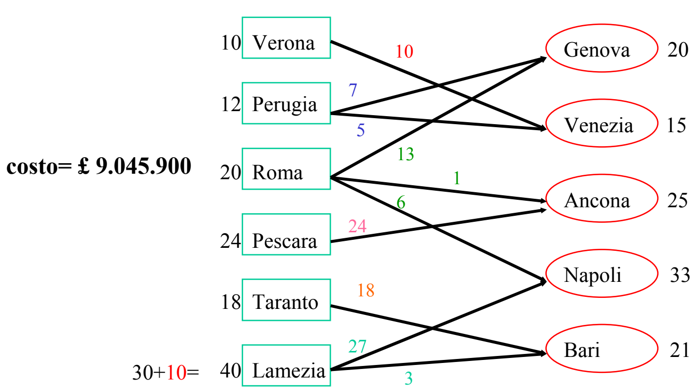
```
 MPL Modeling System   -   Copyright (c) 1988-2000, Maximal Software, Inc.
--------------------------------------------------------------------------------
MODEL STATISTICS

Problem name:       Problema_di_Trasporto

Filename:           trasporti.mpl
Date:               October 18, 2001
Time:               23:36
Parsing time:       0.00 sec

Solver:             CPLEX 300
Objective value:    9045900.00000
Iterations:         0
Solution time:      0.00 sec

Constraints:        11
Variables:          30
Nonzeros:           60
Density:            18 %

SOLUTION RESULT

  Optimal solution found

    MIN costoTot =    9045900.0000
```

```
DECISION VARIABLES

VARIABLE x[magazzini,porti] :

   magazzini  porti             Activity     Reduced Cost
  --------------------------------------------------------
   Verona     Genova              0.0000       40500.0000
   Verona     Venezia            10.0000           0.0000
   Verona     Ancona              0.0000      126000.0000
   Verona     Napoli              0.0000      253500.0000
   Verona     Bari                0.0000      293700.0000
   Perugia    Genova              7.0000           0.0000
   Perugia    Venezia             5.0000           0.0000
   Perugia    Ancona              0.0000        1500.0000
   Perugia    Napoli              0.0000       85500.0000
   Perugia    Bari                0.0000      166200.0000
   Roma       Genova             13.0000           0.0000
   Roma       Venezia             0.0000       19500.0000
   Roma       Ancona              1.0000           0.0000
   Roma       Napoli              6.0000           0.0000
   Roma       Bari                0.0000       80700.0000
   Pescara    Genova              0.0000       84000.0000
   Pescara    Venezia             0.0000       34500.0000
   Pescara    Ancona             24.0000           0.0000
   Pescara    Napoli              0.0000       45000.0000
   Pescara    Bari                0.0000       79200.0000
   Taranto    Genova              0.0000      177300.0000
   Taranto    Venezia             0.0000      138300.0000
   Taranto    Ancona              0.0000      105300.0000
   Taranto    Napoli              0.0000       51300.0000
   Taranto    Bari               18.0000           0.0000
   Lamezia    Genova              0.0000      124500.0000
   Lamezia    Venezia             0.0000      144000.0000
   Lamezia    Ancona              0.0000       93000.0000
   Lamezia    Napoli             27.0000           0.0000
   Lamezia    Bari                3.0000           0.0000
  --------------------------------------------------------
```

```
CONSTRAINTS

CONSTRAINT vincolo_disponibilita[magazzini] :

   magazzini                 Slack      Shadow Price
  ----------------------------------------------------
   Verona                   0.0000     -150600.0000
   Perugia                  0.0000      -83100.0000
   Roma                     0.0000      -45600.0000
   Pescara                  0.0000      -84600.0000
   Taranto                  0.0000      -71400.0000
   Lamezia                 10.0000           0.0000
  ----------------------------------------------------

CONSTRAINT vincolo_richiesta[porti] :

   porti                   Slack      Shadow Price
  --------------------------------------------------
   Genova                 0.0000     197100.0000
   Venezia                0.0000     185100.0000
   Ancona                 0.0000     131100.0000
   Napoli                 0.0000     111600.0000
   Bari                   0.0000      99900.0000
  --------------------------------------------------

END
```

Configurazione di trasporto ottimo

	Genova	Venezia	Ancona	Napoli	Bari
Verona	-	10	-	-	-
Perugia	7	5	-	-	-
Roma	13	-	1	6	-
Pescara	-	-	24	-	-
Taranto	-	-	-	-	18
Lamezia	-	-	-	27	3



Alcune applicazioni di modelli di PL

Un modello di produzione

•Il modello di bilanciamento della produzione è un problema di distribuzione della forza produttiva tra i prodotti, con lo scopo di determinare il livello di produzione di ogni singolo prodotto per soddisfare una data domanda
•Tre prodotti: A1, A2 e A3
•Ai prodotti associamo un indice e, definiamo un vettore a tre variabili che rappresenta il livello di produzione di ogni singolo prodotto: $Prod_{prodotto}$, prodotto $\in \{A1,A2,A3\}$

•Il prezzo di vendita di ogni prodotto è:
Euro 120.00 per A1, Euro 100.00 per A2 e Euro 115.00 per A3
•Per ogni prodotto, esiste anche un limite massimo di domanda:
4300 per A1, 4500 per A2 e 5400 per A3
•La quota di produzione di un prodotto è misurata in quanti pezzi vengono prodotti ogni giorno
•In questo problema abbiamo un totale di 22 giorni di produzione disponibili in un mese
•Nella tabella seguente sono indicati la quota e il costo di produzione di ogni prodotto

Produzione	A1	A2	A3
Costo di Produzione	Euro 73.30	Euro 52.90	Euro 65.40
Quota di Produzione	500	450	550

Formulazione del problema del bilanciamento della produzione

$$\text{Max profitto} = \sum_{\text{prodotto} \in \{A1,A2,A3\}} (\text{Ricavov}_{\text{prodotto}} - \text{costi}_{\text{prodotto}}) \bullet \text{Prod}_{\text{prodotto}}$$

Soggetto a:

$$\sum_{\text{prodotto} \in \{A1,A2,A3\}} \frac{1}{\text{QuotaP}_{\text{prodotto}}} \text{Prod}_{\text{prodotto}} \leq \text{Giorni\_produzione}$$

$$0 \leq \text{Prod}_{\text{prodotto}} \leq \text{domanda}_{\text{prodotto}}, \text{ prodotto} \in \{A1,A2,A3\}$$

Modello del problema del bilanciamento della produzione

$$\text{Max } (120-73.30)\,\text{Prod}_{A1} + (100-52.9)\,\text{Prod}_{A2} + (115-65.4)\,\text{Prod}_{A3}$$

Soggetto a:

$$\frac{1}{500}\text{Prod}_{A1} + \frac{1}{450}\text{Prod}_{A2} + \frac{1}{550}\text{Prod}_{A3} \leq 22$$

$$0 \leq \text{Prod}_{A1} \leq 4300$$

$$0 \leq \text{Prod}_{A2} \leq 4500$$

$$0 \leq \text{Prod}_{A3} \leq 5400$$

Modello MPL

```
TITLE
 Producton_Planning3;

INDEX
 product := (A1, A2, A3);

DATA
 Price[product] := (120.00, 100.00, 115.00);
 Demand[product] := (4300, 4500, 5400);
 ProdCost[product] := (73.30, 52.90, 65.40);
 ProdRate[product] := (500, 450, 550);
 ProdDaysAvail := 22;

VARIABLES
 Produce[product] -> Prod;

MACROS
 TotalRevenue := SUM(product: Price * Produce);
 TotalCost := SUM(product: ProdCost * Produce);

MODEL

 MAX Profit = TotalRevenue - TotalCost;

SUBJECT TO
 ProdCapacity -> PCap:
 SUM(product: Produce / ProdRate) <= ProdDaysAvail;

BOUNDS
 Produce <= Demand;

END
```

Soluzione

```
SOLUTION RESULT
  Optimal solution found

 MAX Profit = 544566.6364

MACROS
 Macro Name Values
 -----------------------------------------------
 TotalRevenue 1298181.8182
 TotalCost 753615.1818
 -----------------------------------------------

DECISION VARIABLES
   VARIABLE Produce[product] :

 product Activity Reduced Cost
 ---------------------------------------------
 A1 4300.0000 4.3100
 A2 1611.8182 0.0000
 A3 5400.0000 11.0636
 ---------------------------------------------

CONSTRAINTS
PLAIN CONSTRAINTS

 Constraint Name Slack Shadow Price
 -----------------------------------------------------
 ProdCapacity 0.0000 21195.0000
 -----------------------------------------------------

END
```

le entrate totali sono di 1,298 milioni di Euro
le spese sono circa 754.000 Euro
il profitto circa 545.000 Euro

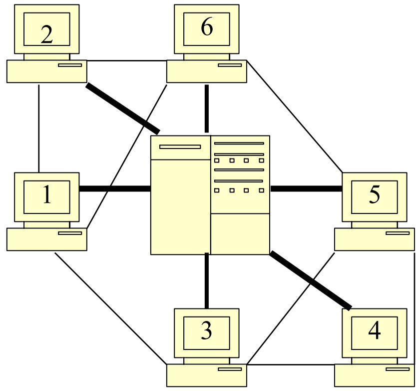
C'è un solo vettore di variabili (Produce)
per A1 e A3 sono prodotti 4.300 e 5.400 pezzi (= domanda)
per A2 viene suggerito di produrre 1.611 pezzi, (< domanda)
non si produce abbastanza da coprire il max della domanda di tutti i prodotti

un vincolo di tipo semplice ProdCapacity.
Lo slack del vincolo vale zero:
la nostra produzione funziona a pieno regime.
I prezzi ombra ci informano a quanto ammonterebbe il mancato guadagno se ci trovassimo nella necessità di ridurre di una unità i giorni disponibili per la produzione
i profitti diminuiscono di 21.195 Euro al giorno

Alcune applicazioni di modelli di PLI

Localizzazione di server front end in una rete di computer

- Una azienda ha il suo database principale in un server a cui si collegano numerose volte al giorno i terminali nelle sue filiali sparse nel territorio
- Le sue filiali si collegano al server del database con protocolli (TCP/IP) e collegamenti telefonici
- Per garantire una connessione remota, si devono installare dei server front end per gestire la connessione
- Ogni filiale è una possibile località di installazione
- Sono stati misurati i tempi medi di risposta per ogni possibile collegamento tra filiali
- Si desidera che ogni collegamento abbia un tempo di risposta non superiore ai 15 millisecondi
- Per garantire la semplicità della rete si vuole minimizzare il numero dei server installati



Tempi medi di collegamento

A = a (f,l) =

	l1	l2	l3	l4	l5	l6
f1	0	10	20	30	20	23
f2	10	0	30	20	17	14
f3	20	30	0	11	17	40
f4	30	20	11	0	9	20
f5	20	17	15	9	0	13
f6	23	14	40	20	13	0

Implicazioni terminal server

$$A' = a'(f,l) = \begin{array}{c|cccccc} & l1 & l2 & l3 & l4 & l5 & l6 \\ \hline f1 & 1 & 1 & 0 & 0 & 0 & 0 \\ f2 & 1 & 1 & 0 & 0 & 0 & 1 \\ f3 & 0 & 0 & 1 & 1 & 0 & 0 \\ f4 & 0 & 0 & 1 & 1 & 1 & 0 \\ f5 & 0 & 0 & 1 & 1 & 1 & 1 \\ f6 & 0 & 1 & 0 & 0 & 1 & 1 \end{array}$$

Se $a(f,l) \leq 15$ **allora** $a'(f,l) = 1$, **altrimenti** $a'(f,l) = 0$

Modello di PLI

- l := località-terminal l, $l \in L$
- x_l = variabili 0-1, server attivo in l

Modello

- Obiettivo: minimizzare il numero di server da attivare
- ogni terminale deve essere collegato ad almeno un server

MIN $\underline{1}\ \underline{x}$

Soggetto a:

$A'\underline{x} \geq \underline{1}$

$\underline{x} \geq \underline{0}$ binarie

Modello di PLI

MIN

$$x[l1] + x[l2] + x[l3] + x[l4] + x[l5] + x[l6]$$

Soggetto a

$$
\begin{array}{lllllllll}
q1: & x[l1] & + x[l2] & & & & & \geq 1 \\
q2: & x[l1] & + x[l2] & & & & + x[l6] & \geq 1 \\
q3: & & & x[l3] & + x[l4] & + x[l5] & & \geq 1 \\
q4: & & & x[l3] & + x[l4] & + x[l5] & & \geq 1 \\
q5: & & & & x[l4] & + x[l5] & + x[l6] & \geq 1 \\
q6: & & x[l2] & & & + x[l5] & + x[l6] & \geq 1
\end{array}
$$

$x[l1], x[l2], x[l3], x[l4], x[l5], x[l6] \geq 0$ e binarie

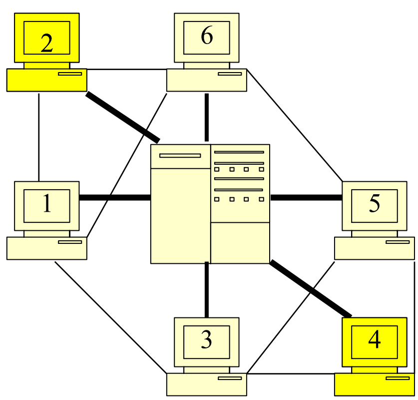
MPL

```
{Modello per la localizzazione di Server}

TITLE
   server_model

INDEX
    l  =  (l1, l2, l3, l4, l5, l6)
    f  =  (f1,f2,f3,f4,f5,f6);

DATA

  A[l,f] = DATAFILE(serv.dat) ;

BINARY  VARIABLES
  x[l]   ->  ""   !  il server e' attivato in l;

MODEL

   MIN  Cost = SUM(l: x) ;

SUBJECT TO
 nb[f]-> "" :   SUM(l: A*x) > 1;

END
```

```
{   serv.dat   }

  {   Datafile per server.mpl}

! implicazioni server terminal

      l1  l2  l3  l4  l5  l6
q1    1   1   0   0   0   0
q2    1   1   0   0   0   1
q3    0   0   1   1   0   0
q4    0   0   1   1   1   0
q5    0   0   1   1   1   1
q6    0   1   0   0   1   1 ;
```

Server da aprire

- I server da aprire sono 2
- Si installano dei server in località 2 e 4



Localizzazione di un server (II)

- Nuova ipotesi: si desidera che ogni collegamento abbia un tempo non superiore ai 10 millisecondi
- Devo cambiare solo il datafile serv.dat

Se $a(f,l) \leq 10$ **allora** $a'(f,l) = 1$,
altrimenti $a'(f,l) = 0$

Implicazioni terminal server (II)

	l1	l2	l3	l4	l5	l6
f1	1	1	0	0	0	0
f2	1	1	0	0	0	0
f3	0	0	1	0	0	0
f4	0	0	0	1	1	0
f5	0	0	0	1	1	0
f6	0	0	0	0	0	1

File serv.dat

Server da aprire (II)

- I server da aprire sono 4
- Si aprono server in località 2, 3, 4 e 6

Localizzazione di un Centro Unificato di Prenotazione

- in ogni quartiere è stata individuata una possibile località di installazione
- sono stati misurati i tempi medi di spostamento
- si desidera che ogni spostamento abbia un tempo non superiore ai 15 minuti

NB: Problema di set covering

$$\text{MIN} \quad \underline{c}.\underline{x}$$

Soggetto a:

con $\underline{c} = \underline{1}$

$$A\underline{x} \geq \underline{1}$$

$$\underline{x} \geq 0 \text{ binarie}$$

Turni in ospedale: ipotesi

- Turni degli infermieri in un reparto
- Ogni infermiere lavora 5 giorni consecutivi, indipendentemente da come sono collocati nella settimana, poi ha diritto a 2 giorni di riposo
- Esigenze di servizio variano di giorno in giorno

Presenze richieste

G	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato	Domenica
N	17	13	15	19	14	16	11

Combinazioni giorno infermiere

	Lun	Mar	Mer	Gio	Ven	Sab	Dom
Lun	1	0	0	1	1	1	1
Mar	1	1	0	0	1	1	1
Mer	1	1	1	0	0	1	1
Gio	1	1	1	1	0	0	1
Ven	1	1	1	1	1	0	0
Sab	0	1	1	1	1	1	0
Dom	0	0	1	1	1	1	1

Se un infermiere comincia il turno il lunedì lavora fino a venerdì e sta a casa sabato e domenica

La somma degli infermieri che iniziano lunedì, più quelli di giovedì, venerdì, sabato e domenica deve essere almeno 17

Modello

x[g] = il numero di infermieri il cui turno inizia il giorno g

MIN

$$x[Lun] + x[Mar] + x[Mer] + x[Gio] + x[Ven] + x[Sab] + x[Dom]$$

Soggetto a:

$$
\begin{array}{lllllllll}
\text{pLu:} & x[Lun] & & & + x[Gio] & + x[Ven] & + x[Sab] & + x[Dom] & \geq 17 \\
\text{pMa:} & x[Lun] & + x[Mar] & & & + x[Ven] & + x[Sab] & + x[Dom] & \geq 13 \\
\text{pMe:} & x[Lun] & + x[Mar] & + x[Mer] & & & + x[Sab] & + x[Dom] & \geq 15 \\
\text{pGi:} & x[Lun] & + x[Mar] & + x[Mer] & + x[Gio] & & & + x[Dom] & \geq 19 \\
\text{pVe:} & x[Lun] & + x[Mar] & + x[Mer] & + x[Gio] & + x[Ven] & & & \geq 14 \\
\text{pSa:} & & x[Mar] & + x[Mer] & + x[Gio] & + x[Ven] & + x[Sab] & & \geq 16 \\
\text{pDo:} & & & x[Mer] & + x[Gio] & + x[Ven] & + x[Sab] & + x[Dom] & \geq 11
\end{array}
$$

$x[Lun], x[Mar], x[Mer], x[Gio], x[Ven], x[Sab], x[Dom] \geq 0$ e interi

MPL

```
{   inf.mpl   }
{
Modello per la programmazione settimanale di infermieri in un reparto
}
TITLE
   program_model
INDEX
    p  =  (pLun, pMar, pMer, pGio, pVen, pSab, pDom)
    g =  ( Lun, Mar, Mer, Gio, Ven, Sab, Dom );
DATA
 b[p] = (17,  13,  15,  19, 14,  16, 11 );

 A[p,g] = DATAFILE(richieste.dat) ;
INTEGER  VARIABLES
   x[g]   ->  ""   !  il numero di infermieri il cui turno inizia il giorno g ;
MODEL
   MIN  Cost = SUM(g: x) ;
SUBJECT TO
 nb[p]-> "" :   SUM(g: A *x) > b;
END
```

```
{   richieste.dat   }
   {   Datafile for inf.mpl   }

! Richieste di combinazioni giorno infermiere
!combinazioni giorno infermiere
       Lun   Mar  Mer  Gio  Ven  Sab  Dom
pLun   1     0    0    1    1    1    1
pMar   1     1    0    0    1    1    1
pMer   1     1    1    0    0    1    1
pGio   1     1    1    1    0    0    1
pVen   1     1    1    1    1    0    0
pSab   0     1    1    1    1    1    0
pDom   0     0    1    1    1    1    1
```

Turni Ottimali infermieri

G	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato	Domenica
N	7	3	3	7		3	

Totale infermieri è 23

Problema di set covering particolare

$$\text{MIN} \quad \underline{c}.\underline{x}$$

Soggetto a:

con $\underline{c} = \underline{1}$

$$A\underline{x} \geq \underline{d}$$

$$\underline{x} \geq \underline{0} \text{ e interi}$$

Sito internet di riferimento

- **Michael Trick's Operations Research Page:**

  è la principale fonte di informazioni in rete sulla Ricerca Operativa: contiene collegamenti a corsi in rete, pubblicazioni, software, siti di organizzazioni, riviste, gruppi di ricerca, aziende e quant'altro riguardi ogni aspetto della Ricerca Operativa

  http://www.informs.org/Resources/

- Raggiungibile anche da

  http:// www.dei.unipd.it/~brunetta