Applications of Linear Programming

lecturer: András London

University of Szeged
Institute of Informatics
Department of Computational Optimization

Lecture 5

Multiperiod work scheduling

Problem: CSL is a chain of computer service stores. The number of hours of skilled repair time that CSL requires during the next five months is as follows:

1. Month 1 (January): 6,000 hours
2. Month 2 (February): 7,000 hours
3. Month 3 (March): 8,000 hours
4. Month 4 (April): 9,500 hours
5. Month 5 (May): 11,000 hours

Multiperiod work scheduling

Assumptions and Constraints: At the beginning of January, 50 skilled technicians work for CSL. Each skilled technician can work up to 160 hours per month.To meet future demands, new technicians must be trained. It takes one month to train a new technician. During the month of training, a trainee must be supervised for 50 hours by an experienced technician.Each experienced technician is paid $2,000 a month (even if he or she does not work the full 160 hours). During the month of training, a trainee is paid $1,000 a month.At the end of each month, 5% of CSL's experienced technicians quit to join Plum Computers. Formulate an LP whose solution will enable CSL to minimize the labor cost incurred in meeting the service requirements for the next five months.

Multiperiod work scheduling - solution

CSL must determine the number of technicians who should be trained during month t $(t = 1, 2, 3, 4, 5)$. Thus, we define

$$x_t = \text{number of technicians trained during month } t (= 1, 2, 3, 4, 5)$$

CSL wants to minimize total labor cost during the next five month

$$\text{Total labor cost} = \text{cost}(\text{trainees}) + \text{cost}(\text{experienced technicians})$$

To express the cost of paying experienced technicians, we need to define

$$y_t = \text{number of experienced technicians at the beginning of month } t$$

Then the total labor cost is

$$z = (1000x_1 + 1000x_2 + 1000x_3 + 1{,}000x_4 + 1000x_5) +$$

$$+ (2000y_1 + 2000y_2 + 2000y_3 + 2000y_4 + 2000y_5)$$

that CLS wants to minimize.

Multiperiod work scheduling - solution

What constraints does CSL face? Note that we are given $y_1 = 50$, and that for $t = 1, 2, 3, 4, 5$, CSL must ensure that

$$\begin{aligned}&\text{Number of available technician hours during month } t\\ \geq\ &\text{Number of technician hours required during month } t\end{aligned}$$

Because each trainee requires 50 hours of experienced technician time, and each skilled technician is available for 160 hours per month,

Number of available technician hours during month $t = 160y_t + 50x_t$

This yields the following five constraints:

1. $160y_1 - 50x_1 \geq 6000$ (month 1 constraint)
2. $160y_2 - 50x2 \geq 7000$ (month 2 constraint)
3. $160y_3 - 50x3 \geq 8000$ (month 3 constraint)
4. $160y_4 - 50x4 \geq 9500$ (month 4 constraint)
5. $160y_5 - 50x5 \geq 11000$ (month 5 constraint)

Multiperiod work scheduling - solution

Note that:
Experienced technicians available at beginning of month $t =$ Experienced technicians available at beginning of month $(t-1)$ + technicians trained during month $(t-1)$ - experienced technicians who quit during month $(t-1)$. For February, this yields

$$y_2 = y_1 + x_1 - 0.05y1 \text{ or } y_2 = 0.95y1 + x1$$

Similarly, for March, yields

$$y_3 = 0.95y_2 + x_2$$

and for April

$$y_4 = 0.95y_3 + x_3$$

and for May,

$$y_5 = 0.95y_4 + x_4$$

Multiperiod work scheduling - solution

Adding the sign restrictions $x_t \geq 0$ and $y_t \geq 0$ $(t = 1, 2, 3, 4, 5)$, we obtain the following LP:

$$
\begin{aligned}
\min z = {} & 1{,}000x_1 + 1{,}000x_2 + 1{,}000x_3 + 1{,}000x_4 + 1{,}000x_5 \\
& + 2{,}000y_1 + 2{,}000y_2 + 2{,}000y_3 + 2{,}000y_4 + 2{,}000y_5 \\
\text{s.t.} \quad & 160y_1 - 50x_1 \geq 6{,}000 \qquad y_1 = 50 \\
& 160y_2 - 50x_2 \geq 7{,}000 \qquad 0.95y_1 + x_1 = y_2 \\
& 160y_3 - 50x_3 \geq 8{,}000 \qquad 0.95y_2 + x_2 = y_3 \\
& 160y_4 - 50x_4 \geq 9{,}500 \qquad 0.95y_3 + x_3 = y_4 \\
& 160y_5 - 50x_5 \geq 11{,}000 \qquad 0.95y_4 + x_4 = y_5 \\
& x_t, y_t \geq 0 \qquad (t = 1, 2, 3, 4, 5)
\end{aligned}
$$

Vectors

- **Scalar** = a number; can be real ($\pi = 3.14\dots$), rational ($3/4$), integer (5, -8), etc.

- **Vector** = sequence of numbers, for example $(3, 1, 0, 2)$, we often write $x = \begin{bmatrix} 3 & 2 & 0 & 1 \end{bmatrix}$ **raw vector**, or $x = \begin{bmatrix} 3 \\ 1 \\ 0 \\ 2 \end{bmatrix}$ **column vector**

- **multiplying** $x = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}$ by a:

$$ax = \begin{bmatrix} ax_1 & ax_2 & \cdots & ax_n \end{bmatrix}$$

- **addition** of $x = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}$ and $y = \begin{bmatrix} y_1 & y_2 & \dots & y_n \end{bmatrix}$ vectors (of the same size):

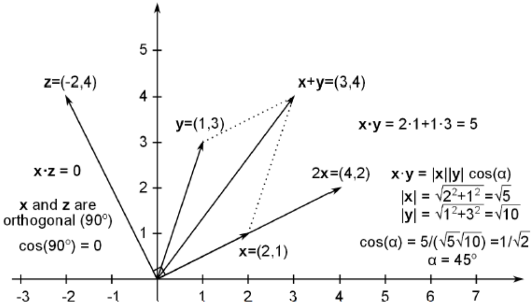
$$x + y = \begin{bmatrix} x_1 + y_1 & x_2 + y_2 & \dots & x_n + y_n \end{bmatrix}$$

Vectors

- **scalar product** of $x = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}$ and $y = \begin{bmatrix} y_1 & y_2 & \dots & y_n \end{bmatrix}$ vectors (of the same size):

$$xy = x_1 y_1 + x_2 y_2 \cdots + x_n y_n$$

- x and y **orthogonal**, if $xy = 0$



Matrices

- **Matrix** = 2-dimensional array of numbers, for example

$$A = \begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \\ 0 & 4 & 1 & 2 \end{bmatrix}$$

- $m \times n$ matrix:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ \vdots & & \ddots & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \qquad \underbrace{\begin{bmatrix} a_{i1} & a_{i2} & \cdots & a_{in} \end{bmatrix}}_{i\text{-th row of } \mathbf{A}} \qquad \left.\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix}\right\} j\text{-th column of } \mathbf{A}$$

- **multiplying a matrix by a scalar**:

$$2 \cdot \begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \\ 0 & 4 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 6 & 2 \\ 6 & 4 & 8 & 0 \\ 4 & 6 & 0 & 2 \\ 0 & 8 & 2 & 4 \end{bmatrix}$$

Matrices

- **adding matrices** of the same size

$$\begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \\ 0 & 4 & 1 & 2 \end{bmatrix} + \begin{bmatrix} 3 & 1 & 4 & 3 \\ 2 & 0 & 1 & 3 \\ 0 & 2 & 1 & 4 \\ 3 & 0 & 3 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 1 & 7 & 4 \\ 5 & 2 & 5 & 3 \\ 2 & 5 & 1 & 5 \\ 3 & 4 & 4 & 6 \end{bmatrix}$$

- **multiplying matrices**: A of size $n \times m$-es multiplied by B of $m \times k$

$$\begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \\ 0 & 4 & 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 3 & 1 & 4 & 3 \\ 2 & 0 & 1 & 3 \\ 0 & 2 & 1 & 4 \\ 3 & 0 & 3 & 4 \end{bmatrix} = \begin{bmatrix} 6 & 7 & 10 & 19 \\ 13 & 11 & 18 & 31 \\ 15 & 2 & 14 & 19 \\ 14 & 2 & 11 & 24 \end{bmatrix} \qquad (2,3,0,1)\cdot(1,0,2,0) = 2$$

  - Note that $AB \neq BA$
  - except for this, matrix addition and multiplication obey exactly the same laws as numbers
  - from now on vector with m entries is to be treated as a $m \times n$ matrix (column)

Matrices

- multiplying matrix by a vector = just like multiplying two matrices

$$\begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \\ 0 & 4 & 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 7 \\ 11 \\ 2 \\ 2 \end{bmatrix}$$

- transpose of a matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \end{bmatrix} \qquad \mathbf{A}^{\mathrm{T}} = \begin{bmatrix} 1 & 3 & 2 \\ 0 & 2 & 3 \\ 3 & 4 & 0 \\ 1 & 0 & 1 \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} 1 \\ 0 \\ 3 \\ 1 \end{bmatrix} \qquad \mathbf{x}^{\mathrm{T}} = \begin{bmatrix} 1 & 0 & 3 & 1 \end{bmatrix}$$

  - Note that $(A^T)^T = A$ and $(AB)^T = B^T A^T$

System of linear equations

- A system of linear equations has the following form

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & = & b_2 \\ \vdots & & & & \ddots & & & & \vdots \\ a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & = & b_m \end{array}$$

- Using the matrix notation we can simply write it as $Ax = b$, where

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ \vdots & & \ddots & \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

System of linear equations

- For example

$$\begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix}$$

- Let us multiply (from the left) both sides of the equation by this matrix

$$\begin{bmatrix} -4 & 3 & -2 \\ 8/3 & -2 & 5/3 \\ 5/3 & -1 & 2/3 \end{bmatrix}$$

- This operation does not change the solutions to this system (determinant $\neq 0$)

$$\underbrace{\begin{bmatrix} -4 & 3 & -2 \\ 8/3 & -2 & 5/3 \\ 5/3 & -1 & 2/3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 3 & 1 \\ 3 & 2 & 4 & 0 \\ 2 & 3 & 0 & 1 \end{bmatrix}}_{\begin{bmatrix} 1 & 0 & 0 & -6 \\ 0 & 1 & 0 & 13/3 \\ 0 & 0 & 1 & 7/3 \end{bmatrix}} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \underbrace{\begin{bmatrix} -4 & 3 & -2 \\ 8/3 & -2 & 5/3 \\ 5/3 & -1 & 2/3 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix}}_{\begin{bmatrix} 5 \\ -10/3 \\ -4/3 \end{bmatrix}}$$

$$\begin{bmatrix} 1 & 0 & 0 & -6 \\ 0 & 1 & 0 & 13/3 \\ 0 & 0 & 1 & 7/3 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 5 \\ -10/3 \\ -4/3 \end{bmatrix}$$

System of linear equations

- We can expand it back to the system of linear equations

$$\begin{array}{rrrrrrrrl} x_1 & & & - & 6x_4 & = & 5 \\ & x_2 & & + & \frac{13}{3}x_4 & = & -\frac{10}{3} \\ & & x_3 & + & \frac{7}{3}x_4 & = & -\frac{4}{3} \end{array} \quad \to\to \quad \begin{array}{rcrcl} x_1 & = & 5 & + & 6x_4 \\ x_2 & = & -\frac{10}{3} & - & \frac{13}{3}x_4 \\ x_3 & = & -\frac{4}{3} & - & \frac{7}{3}x_4 \end{array}$$

- The system on the right is in a **dictionary** form: In particular, we can set $x_4 = 0$ in which case we have a basic solution.

How did we choose the matrix to multiply?

$\Longrightarrow$ we can choose the **inverse** matrix of the first three column of A.

- inverse of A is an A^{-1} matrix, for which $AA^{-1} = A^{-1}A = I$ (I is the identity matrix)

Simplex alg. in matrix formulation

Given a **LP in standard form**:

$$\begin{aligned} \sum_{j=1}^{n} a_{ij} x_j &\leq b_i \qquad && i = 1,2,\dots,m \\ x_j &\geq 0 && j = 1,2,\dots,n \end{aligned}$$

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

Adding the non-negative artificial variables:

$$\begin{aligned} \sum_{j=1}^{n} a_{ij} x_j + x_{n+i} &= b_i \qquad && i = 1,2,\dots,m \\ x_j &\geq 0 && j = 1,2,\dots,n+m \end{aligned}$$

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

Simplex alg. in matrix formulation

In **matrix form** we can write as:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & 1 & & & \\ a_{21} & a_{22} & \cdots & a_{2n} & & 1 & & \\ & & \vdots & & & & \ddots & \\ a_{m1} & a_{m2} & \cdots & a_{mn} & & & & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ x_{n+1} \\ \vdots \\ x_{n+m} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

$$\begin{pmatrix} c_1 & c_2 & \cdots & c_n & 0 & \cdots & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ x_{n+1} \\ \vdots \\ x_{n+m} \end{pmatrix} = z$$

Simplex alg. in matrix formulation

Matrices and vectors used:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & 1 & & & \\ a_{21} & a_{22} & \cdots & a_{2n} & & 1 & & \\ & & \vdots & & & & \ddots & \\ a_{m1} & a_{m2} & \cdots & a_{mn} & & & & 1 \end{pmatrix}$$

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ x_{n+2} \\ \vdots \\ x_{n+m} \end{pmatrix} \qquad c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ 0 \\ \vdots \\ 0 \end{pmatrix} \qquad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

Simplex alg. in matrix formulation

The problem is **matrix form** then

$$\begin{aligned} Ax &= b \\ x &\geq 0 \end{aligned}$$
$$\overline{\max c^T x \qquad}$$

We know that a dictionary is uniquely determined by its the basic variables

- let $\mathcal{B}$ be the index set of the basic variables, $\mathcal{N}$ be the index set of the non-basic variables

Divide the matrices and vectors for two parts, based on the role (basic or non-basic) of their elements in the dictionary

Simplex alg. in matrix formulation

Let B denote the **basis** matrix formed by taking the columns of A corresponding to the basic variables $x_{\mathcal{B}}$

Let N denote the columns of A corresponding to the non-basic variables in $x_{\mathcal{N}}$

Divide the vectors as $c = \begin{pmatrix} c_{\mathcal{B}} \\ c_{\mathcal{N}} \end{pmatrix}$, $x = \begin{pmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{pmatrix}$

Simplex alg. in matrix formulation

We obtain that

$$Ax = \begin{pmatrix} B & N \end{pmatrix} \begin{pmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{pmatrix} = Bx_{\mathcal{B}} + Nx_{\mathcal{N}}$$

$$c^T x = \begin{pmatrix} c_{\mathcal{B}} & c_{\mathcal{N}} \end{pmatrix} \begin{pmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{pmatrix} = c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}}$$

and hence the optimization problem is given in the form:

$$\begin{aligned} Bx_{\mathcal{B}} + Nx_{\mathcal{N}} &= b \\ x &\geq 0 \end{aligned}$$

$$\overline{\max c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}}}$$

Simplex alg. in matrix formulation

Assuming that B **is invertible**, we can rewrite:

$$\begin{aligned} Ax &= b \\ Bx_{\mathcal{B}} + Nx_{\mathcal{N}} &= b \\ B^{-1}(Bx_{\mathcal{B}} + Nx_{\mathcal{N}}) &= B^{-1}b \\ B^{-1}Bx_{\mathcal{B}} + B^{-1}Nx_{\mathcal{N}} &= B^{-1}b \\ x_{\mathcal{B}} + B^{-1}Nx_{\mathcal{N}} &= B^{-1}b \\ x_{\mathcal{B}} &= B^{-1}b - B^{-1}Nx_{\mathcal{N}} \end{aligned}$$

Now we can substitute $x_{\mathcal{B}}$ to the objective function

$$\begin{aligned} z = c^T x &= c_{\mathcal{B}}^T x_{\mathcal{B}} + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T (B^{-1}b - B^{-1}Nx_{\mathcal{N}}) + c_{\mathcal{N}}^T x_{\mathcal{N}} \\ &= c_{\mathcal{B}}^T B^{-1}b + (c_{\mathcal{N}}^T - c_{\mathcal{B}}^T B^{-1}N)x_{\mathcal{N}} \end{aligned}$$

Simplex alg. in matrix formulation

We put it together to obtain the corresponding dictionary:

$$x_{\mathcal{B}} = B^{-1}b - B^{-1}Nx_{\mathcal{N}}$$

$$\overline{\qquad\qquad\qquad\qquad}$$

$$z = c_{\mathcal{B}}^T B^{-1}b + (c_{\mathcal{N}}^T - c_{\mathcal{B}}^T B^{-1}N)x_{\mathcal{N}}$$

The **Basic solution**, when $x_{\mathcal{N}} = 0$ is given as:

$x_{\mathcal{B}} = B^{-1}b$, with the value of the objective function: $z = c_{\mathcal{B}}^T B^{-1}b$.

The solution is optimal(maximal) if, if $c_{\mathcal{N}}^T - c_{\mathcal{B}}^T B^{-1}N \leq 0$, means that the constant coefficients of the non-basic variables are non-positive.

Totally unimodular matrices

A square matrix A is **totally unimodular** (TU) if its every square submatrix has determinant 1, -1, or 0. (It follows that a TU matrix has only $0, +1$ or -1 entries)

Theorem. If A is totally unimodular, all entries of b and c vectors are integers, then the basic solutions of the $\max c^T x$ s.t. $Ax \leq b$ linear program are integers. In other words, the coordinates of the corner points of the $P = \{x : Ax \leq b\}$ polyhedron are integers.

Proof. The dictionary for $x_{\mathcal{B}}$ basic solution is given in the form

$$x_{\mathcal{B}} = B^{-1}b - B^{-1}Nx_{\mathcal{N}}$$

$$z = c_{\mathcal{B}}^T B^{-1}b + (c_{\mathcal{N}}^T - c_{\mathcal{B}}^T B^{-1}N)x_{\mathcal{N}}$$

Totally unimodular matrices

We can calculate the entries of B^{-1} using the *Cramer-rule*, such that

$$(B^{-1})_{i,j} = \frac{(-1)^{i+j}\det(B^{ji})}{\det(B)}$$

where B^{ij} is the matrix obtained from B by omitting raw j and column i.

Since now A is TU and $\det(B) \neq 0$, then $\det(B) = \pm 1$. It is easy to see that each entry of B^{ji} is integer.
Since $x_{\mathcal{B}} = B^{-1}b$ it immediately follows that $x_{\mathcal{B}}$ is an integer vector. ■

Remark. It is not obvious how to decide whether a given matrix is TU or not. Good news is that TU matrices can be characterized, moreover, a fast algorithm can be given to decide this characteristic.