Lecture 3: engineering solutions of LPs

- Engineering LPs
- Matrix form of a LP
- How to solve a LP in MATLAB
- A graphical illustration of the SIMPLEX method
- Engineering algorithms

In general, LPs encountered in engineering are big

It is very unlikely that your boss will give you something that simple to solve:

$$
\begin{array}{ll}
\textbf{min:} & Z = 140x_1 + 160x_2 \\
\textbf{Subject to:} & 2x_1 + 4x_2 \leq 28 \\
& 5x_1 + 5x_2 \leq 50 \\
& x_1 \leq 8 \\
& x_2 \leq 6 \\
& x_1 \geq 0 \\
& x_2 \geq 0
\end{array}
$$

In general, you get something that looks like this

Example: air traffic control

$$\mathbf{min}_{w^j_{ft}} \sum_{f \in \mathcal{F}} \left[c^g_f g_f + c^a_f a_f \right]$$

such that:

$$\sum_{f:P(f,1)=k} \left(w^k_{ft} - w^k_{f,t-1}\right) \leq D_k(t) \qquad \forall k \in \mathcal{K},\ t \in \mathcal{T}$$

$$\sum_{f:P(f,N_f)=k} \left(w^k_{ft} - w^k_{f,t-1}\right) \leq A_k(t) \qquad \forall k \in \mathcal{K},\ t \in \mathcal{T}$$

$$\sum_{(f,j'):P(f,i)=j,P(f,i+1)=j',i<N_f} \left(w^j_{ft} - w^{j'}_{ft}\right) \leq S_j(t) \qquad \forall j \in \mathcal{J},\ t \in \mathcal{T}$$

$$w^{j'}_{f,t+t_{fj}} - w^j_{ft} \leq 0 \qquad \forall f \in \mathcal{F},\ t \in T^j_f,\ j = P(f,i), j' = P(f,i+1),\ i < N_f$$

$$w^k_{ft} - w^k_{f',t-s_{f'}} \leq 0 \qquad \forall (f',f) \in \mathcal{C},\ t \in T^k_f, k = P(f,1) = P(f',N_f)$$

$$w^j_{ft} - w^j_{f,t-1} \geq 0 \qquad \forall f \in \mathcal{F},\ j \in P_f,\ t \in T^j_f$$

[Bertsimas, Stock-Patterson, 1997]

Reminder: general formulation of a LP

$$
\begin{array}{lll}
\textbf{min:} & c_1x_1 + c_2x_2 + \cdots + c_Nc_N & \\
\textbf{s.t.:} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N & \leq b_1 \\
& a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N & \leq b_2 \\
& \vdots & \vdots \\
& a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N & \leq b_M
\end{array}
$$

Or equivalently,

$$
\begin{array}{lll}
\textbf{min:} & \sum_{i=1}^N c_jx_j & \\
\textbf{s.t.:} & \sum_{i=1}^N a_{1,j}x_j & \leq b_1 \\
& \sum_{i=1}^N a_{2,j}x_j & \leq b_2 \\
& \vdots & \vdots \\
& \sum_{i=1}^N a_{M,j}x_j & \leq b_M
\end{array}
$$

Cost function is a dot product

$$\textbf{min:} \quad c_1x_1 + c_2x_2 + \cdots + c_Nc_N$$

This cost function is in fact a dot product (scalar product between a vector x and a vector c:

$$\left[\ c_1,\ c_2, \cdots, \cdots, c_N\ \right] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_N \end{bmatrix}$$

$$\textbf{min:} \qquad \mathbf{c^T \cdot x}$$

Inequality constraints

$$
\begin{array}{lll}
\textbf{min:} & c_1x_1 + c_2x_2 + \cdots + c_Nc_N & \\
\textbf{s.t.:} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N & \leq b_1 \\
& a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N & \leq b_2 \\
& \vdots & \vdots \\
& a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N & \leq b_M
\end{array}
$$

Inequality constraints

$$
\begin{array}{ll}
\textbf{s.t.:} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N \quad \leq b_1 \\
& a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N \quad \leq b_2 \\
& \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
& a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N \quad \leq b_M
\end{array}
$$

Inequality constraints

$$
\begin{array}{ll}
\textbf{s.t.:} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N \quad \leq b_1 \\
& a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N \quad \leq b_2 \\
& \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
& a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N \quad \leq b_M
\end{array}
$$

The constraints are in fact in vector form:

$$
\begin{bmatrix} a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N \\ \vdots \\ a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix}
$$

Inequality constraints

$$
\begin{array}{ll}
\textbf{s.t.:} & a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,j}x_j \cdots + a_{1,N}x_N \quad \leq b_1 \\
& a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,j}x_j \cdots + a_{2,N}x_N \quad \leq b_2 \\
& \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
& a_{M,1}x_1 + a_{M,2}x_2 + \cdots + a_{M,j}x_j \cdots + a_{M,N}x_N \quad \leq b_M
\end{array}
$$

The constraints are in fact in vector form:

$$
\begin{bmatrix} a_{1\,1} & a_{1\,2} & \cdots & a_{1N} \\ a_{2\,1} & a_{2\,2} & \cdots & a_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NM} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_N \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix}
$$

LP in matrix form

The constraints are in fact in vector form:

$$\mathbf{A}\ \mathbf{x} \leq \mathbf{b}$$

The constraints are in fact in vector form:

$$
\begin{bmatrix} a_{1\,1} & a_{1\,2} & \cdots & a_{1N} \\ a_{2\,1} & a_{2\,2} & \cdots & a_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NM} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_N \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix}
$$

LP in matrix form

The constraints are in fact in vector form:

$$
\begin{array}{ll}
\textbf{min:} & \mathbf{c^T} \cdot \mathbf{x} \\
\textbf{s.t.} & \mathbf{A}\ \mathbf{x} \leq \mathbf{b}
\end{array}
$$

Introduce objective function vector:

$$\mathbf{c}^T = \left[\ c_1,\ c_2, \cdots, \cdots, c_N\ \right]$$

The constraints are in fact in vector form:

$$
\begin{bmatrix} a_{1\,1} & a_{1\,2} & \cdots & a_{1N} \\ a_{2\,1} & a_{2\,2} & \cdots & a_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NM} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_N \end{bmatrix} \leq \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix}
$$

LP in matrix form

The constraints are in fact in vector form:

$$
\begin{array}{ll}
\textbf{min:} & \mathbf{c^T} \cdot \mathbf{x} \\
\textbf{s.t.} & \mathbf{A}\ \mathbf{x} \leq \mathbf{b}
\end{array}
$$

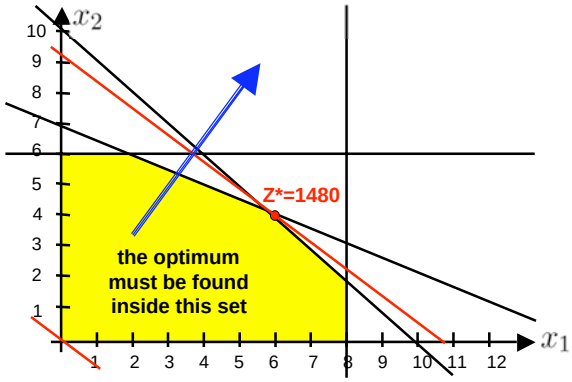
This is exactly the way you will use this in matlab:

```
x = linprog(c,A,b)
```

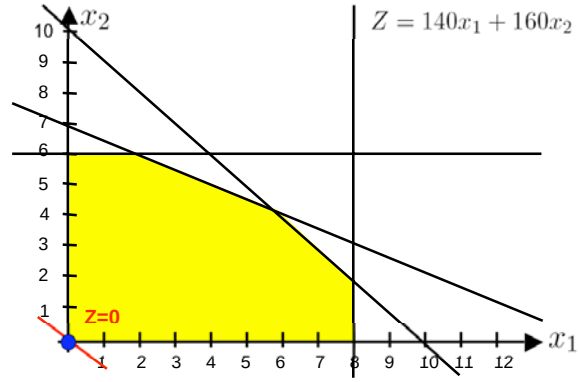
Several very powerful methods have been developed to solve linear programs, in particular:

- **Simplex method**
- **Interior point method**

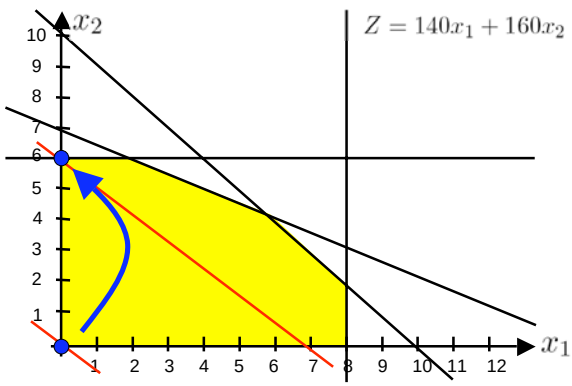
A simplified illustration of the simplex method



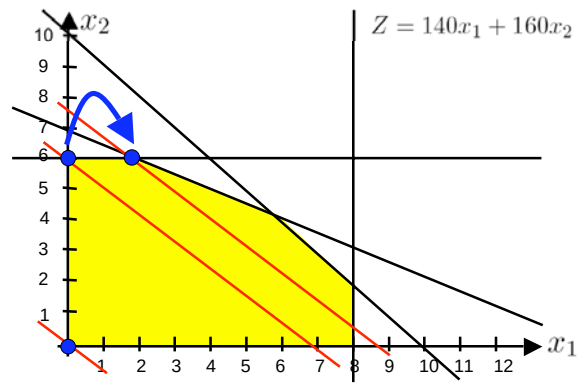
A simplified illustration of the simplex method



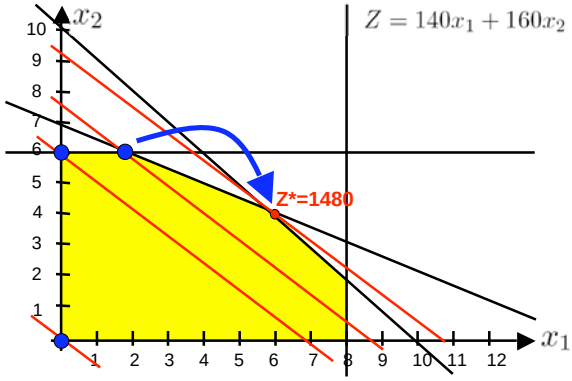
A simplified illustration of the simplex method



A simplified illustration of the simplex method



A simplified illustration of the simplex method



Engineering algorithms

As an engineer, you formulate a physical problem in a known format (for example a linear program). Then, you can use an algorithm to solve it (either your own, or a toolbox, for example using MATLAB).

```
BEGIN
    input: guess, starting point
    DO
        repeat simple task
    UNTIL
        stopping criterion is satisfied
END
```