

# Stochastic Models and Optimization: The coal mining problem

Roger Garriga Calleja, José Fernando Moreno Gutiérrez, David Rosenfeld, Katrina Walker

April 4, 2017

We have a company that has a  $N$  years concession of 6 mines of coal from which extracts two types of coal (thermal coal and coking coal). Both types are mixed in the same rock, so one does not decide which type to mine, when the coal is mined a certain percentage is coking and the rest thermal. Coking coal is more valuable than thermal coal. Each mine has its own maximum production capacity per year ( $m_k^i$ ,  $i = 1, \dots, 6$ ), a finite amount of coal ( $C^i$ ,  $i = 1 \dots, 6$ ) and a certain composition (percentage of coking coal  $\rho^i$  can be fixed as first approach and then if we have time we put it as a random variable following a distribution  $\eta_k^i$  that depends on the observations and the prior we have).

The mines are connected to the port (where they are sailed to sell) by a railway that has a total transport capacity  $T$ . The price of each type of coal is a random variable that we know at the end of the year  $w_k^c, w_k^t$ . The expected value of the price increases each year in a linear form  $\mathbb{E}[w_k^c] = a_c k + b_c$ ,  $\mathbb{E}[w_k^t] = a_t k + b_t$ . There is a cost of extraction per tone of coal  $c$  and if the coal is mined but cannot be transported it can be used in the mines contributing with a value of  $s < c$ .

## Primitives:

$x_k^i$ : The amount of coal remaining on each mine at period  $k$ .

$u_k^i$ : How much coal to mine at period  $k$ .

$w_k^c, w_k^t$ : Price per tone of each type of coal.

Constrains:  $u_k^i \leq m_k^i$ ,  $x_k^i \geq 0$ , transportation capacity  $T$

## Information:

$c$ : Cost of extracting a tone of coal.

$s$ : Salvage value for the coal not sold.

Dynamics:  $x_{k+1} = x_k - u_k$ .

## Profit:

$g_N(x_N) = 0$

$g_k(x_k, u_k, w_k) = v_k^c w_k^c + v_k^t w_k^t - cu_k + s \max\{0, u_k - T\}$ , where

$$\begin{pmatrix} v_k^c \\ v_k^t \end{pmatrix} = \begin{cases} \begin{pmatrix} \rho u_k \\ (1 - \rho)u_k \end{pmatrix} & \text{if } u_k \leq T \\ \begin{pmatrix} \min\{\rho u_k, T\} \\ T - \min\{\rho u_k, T\} \end{pmatrix} & \text{if } u_k > T \end{cases}$$

## DP algorithm:

First approach,  $T = \infty$ . (no salvage value).  $J_N(x_N) = 0$ .

$$J_k(x_k) = \max_{\substack{u_k \leq m_k, \\ u_k \leq x_k}} \mathbb{E}[u_k(\rho w_k^c + (1 - \rho)w_k^t - c) + J_{k+1}(x_k - u_k)].$$

Solving the DP problem:

$$J_{N-1}(x_{N-1}) = \max_{\substack{u_{N-1} \leq m_{N-1}, \\ u_{N-1} \leq x_{N-1}}} \mathbb{E}[u_{N-1}(\rho w_{N-1}^c + (1 - \rho)w_{N-1}^t - c)] = \quad (1)$$

$$= \max_{\substack{u_{N-1} \leq m_{N-1}, \\ u_{N-1} \leq x_{N-1}}} \{u_{N-1}(\rho \mathbb{E}[w_{N-1}] + (1 - \rho)\mathbb{E}[w_{N-1}] - c)\} = \quad (2)$$

$$\max_{\substack{u_{N-1} \leq m_{N-1}, \\ u_{N-1} \leq x_{N-1}}} \{u_{N-1}(\rho(a_c(N-1) + b_c) + (1 - \rho)(a_t(N-1) + b_t) - c)\}. \quad (3)$$

Since  $\rho(a_c(N-1) + b_c) + (1 - \rho)(a_t(N-1) + b_t) > c$ , the function is always increasing on  $u_{N-1}$ , so the maximum will be accomplished on the more restrictive constrain.  $u_{N-1} = \min\{m_{N-1}, x_{N-1}\}$ . Now, for  $N-2$ :

$$J_{N-2}(x_{N-2}) = \max_{\substack{u_{N-2} \leq m_{N-2}, \\ u_{N-2} \leq x_{N-2}}} \mathbb{E}[u_{N-2}(\rho w_{N-2}^c + (1 - \rho)w_{N-2}^t - c) + J_{N-1}(x_{N-2} - u_{N-2})] = \quad (4)$$

$$= \max_{\substack{u_{N-2} \leq m_{N-2}, \\ u_{N-2} \leq x_{N-1}}} \{u_{N-2}(\rho(a_c(N-2) + b_c) + (1 - \rho)(a_t(N-2) + b_t) - c) + \quad (5)$$

$$+ \min\{m_{N-1}, x_{N-2} - u_{N-2}(\rho(a_c(N-1) + b_c) + (1 - \rho)(a_t(N-1) + b_t) - c)\}\}. \quad (6)$$

Since  $\rho(a_c(N-1) + b_c) + (1 - \rho)(a_t(N-1) + b_t) - c > \rho(a_c(N-2) + b_c) + (1 - \rho)(a_t(N-2) + b_t) - c$ , in order to maximize we need  $\min\{m_{N-1}, x_{N-2} - u_{N-2}\}$  to be as big as possible. That means

$$\max \begin{cases} m_{N-1} & \text{if } x_{N-2} - u_{N-2} > m_{N-1} \\ x_{N-2} - u_{N-2} & \text{otherwise} \end{cases},$$

which will be attained when  $u_{N-2} = (x_{N-2} - m_{N-1})^+$ . However, since could happen that  $(m_{N-1} - x_{N-2})^+ > m_{N-2}$ , the optimal policy would be  $u_k = \min\{(x_{N-2} - m_{N-1})^+, m_{N-2}\}$ .

Backward induction proves that  $u_k = \min\{(x_k - \sum_{i=k+1}^{N-1} m_i)^+, m_k\}$  (I will write it tomorrow)