OPTIMAL SCHEDULING FOR OPEN PIT MINE EXTRACTION: A HEURISTICS BASED UPON GITTINS INDEX

Michel De Lara[1]

4th October 2007

[1]CERMICS, École des ponts, ParisTech, France

Credits

- Jorge AMAYA (CMM) for introducing us to the open-pit mine optimal scheduling
- Alain RAPAPORT (INRIA) for discussions on the dynamic programming approach, for supervising Marie-Charlotte DRUESNE (ENPC student),

Outline of the presentation

A TWO-DIMENSIONAL OPEN PIT MINE

OPTIMAL EXTRACTION MODEL

A two-dimensional open pit mine

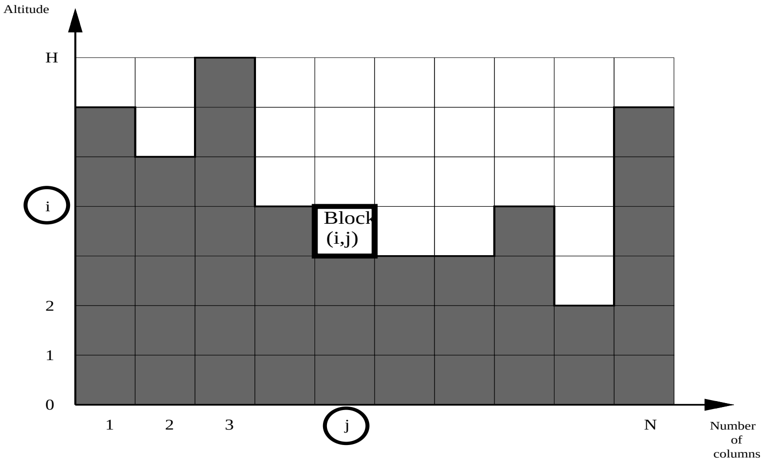

Figure: An extraction profile in an open pit mine

Each block is a two-dimensional rectangle

- identified by its horizontal position $j \in \{1, \ldots, N\}$,
- and by its vertical position $i \in \{1, \ldots, H\}$,
- containing ore with net value $\mathcal{R}(i, j)$.

The mine as a collection of columns

In the sequel, it will also be convenient to see the mine as a **collection of columns** indexed by $j \in \{1, \ldots, N\}$, each column containing H blocks.

Sequential extraction assumptions

We assume that blocks are extracted sequentially under the following hypothesis:

- it takes **one time unit to extract one block**;
- only **blocks at the surface may be extracted**;
- a block cannot be extracted if the slope made with one of its two neighbours is too high, due to physical requirements (the **slope constraints** are mathematically decribed in the sequel).

States

Denote **discrete time** by $t = 0, 1, \ldots, T$, where the number of time steps $T + 1 = N \times H$ equals the number of blocks.

At time t, the **state** of the mine is a **profile**

$$x(t) = (x_1(t), \ldots, x_N(t))$$

where $x_j(t) \in \{1, \ldots, H + 1\}$ is the vertical position of the top block with horizontal position $j \in \{1, \ldots, N\}$ (see Figure 1).

- The initial profile is $x = (1, 1, \ldots, 1)$.
- The mine is totally exhausted in state $x = (H + 1, H + 1, \ldots, H + 1)$.
- The set of states is $\mathbb{X} = \{1, \ldots, H + 1\}^N$.

Admissible states

An admissible profile is one that respects **slope constraints**, due to physical requirements.

A state $x = (x_1, \ldots, x_N) \in \mathbb{X}$ is said to be **admissible** if

$$\sup\{|x_2 - x_1|, |x_3 - x_2|, \ldots, |x_{N-1} - x_{N-2}|, |x_N - x_{N-1}|\} \leq 1 .$$

The **set of admissible states** is denoted by

$$\mathbb{X}_a \subset \mathbb{X} .$$

Decisions

A decision is the **selection of a column whose top block will be extracted**. Thus a decision u is an element of the set

$$\mathbb{U} = \{1, \dots, N\}.$$

Of course, not all decisions $u = j$ are possible

- either because there are no blocks left in column j $(x_j = H + 1)$
- or because of slope constraints.

Controlled dynamics

At time t, once a column $u(t) \in \{1, \dots, N\}$ is chosen at the surface of the open pit mine, the corresponding block is extracted and the profile $x(t) = (x_1(t), \dots, x_N(t))$ becomes

$$x_j(t+1) = \begin{cases} x_j(t) - 1 & \text{if } j = u(t) \\ x_j(t) & \text{else.} \end{cases}$$

In other words, the **dynamics** is given by $x(t+1) = F(x, u)$ where

$$F(x,u)_j = \begin{cases} x_j - 1 & \text{if} \quad j = u \\ x_j & \text{if} \quad j \neq u \,. \end{cases}$$

Indeed, the top block of column j is no longer at altitude $x_j(t)$ but at $x_j(t) - 1$, while all other top blocks remain.

Example

For instance, consider a mine with $N = 5$ columns of height $H = 3$. Starting from state $x = (2, 3, 1, 4, 3)$ in Figure 2

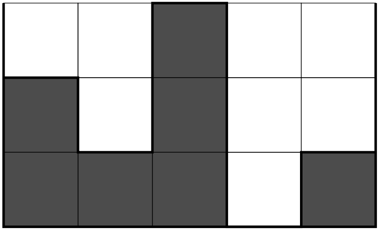

Figure: State (2,3,1,4,3)

Applying control $u = 3$, one obtains the following state $(2, 3, 2, 4, 3)$ as in Figure 3.

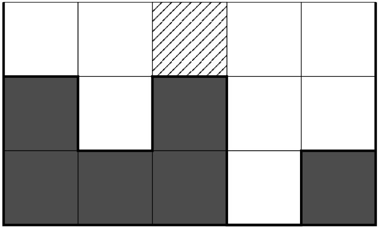

Figure: State (2,3,2,4,3)

Decision constraints

When in state $x \in \mathbb{X}_a$, the decision $u \in \mathbb{U} = \{1, \ldots, N\}$ is **admissible** if the future state $F(x,u)$ is an admissible state that is

$$u \, \text{admissible} \iff F(x,u) \in \mathbb{X}_a \; .$$

Thus, the decision u is admissible whenever $u \in \mathcal{U}(x)$, where

$$\mathcal{U}(x) = \{u \in \mathbb{U} \mid F(x,u) \in \mathbb{X}_a\} \, .$$

We shall also say that the corresponding top block is admissible.

Intertemporal profit maximization

1. Selecting the column $j = u(t) \in \mathbb{U}$,
2. and extracting the corresponding upper block with depth $i = x_j(t)$ at the surface of the open pit mine
3. yields the profit

$$\mathcal{R}\big(x_j(t), j\big) = \mathcal{R}\big(x_{u(t)}(t), u(t)\big) .$$

With discounting $\rho \in [0, 1]$, the **optimization problem** is

$$\sup_{u(0),\dots,u(T-1)} \sum_{t=0}^{T-1} \rho^t \mathcal{R}\big(x_{u(t)}(t), u(t)\big) .$$

The optimal extraction path problem

A "good" mine

11.	12.	13.	14.	15.
6.	7.	8.	9.	10.
1.	2.	3.	4.	5.

A "bad" mine

1.	2.	3.	4.	5.
6.	7.	8.	9.	10.
11.	12.	13.	14.	15.

DYNAMIC PROGRAMMING ALGORITHM

Numerical considerations

N	H	$\mathbb{X}^\sharp$	$\mathbb{X}_a^\sharp$	$\mathbb{X}_a^\sharp/\mathbb{X}^\sharp$
N	H	$(H+1)^N$	$\leq (H+1) \times 3^{N-1}$	
4	9	10^4	≤ 270	≤ 3 %
10 000	9	$10^{10\ 000}$	$\leq 10^{5\ 000}$	$\leq 10^{-5\ 000}$

Table: Cardinals of states and acceptable states sets

$$x = (x_1, \ldots, x_N) \mapsto \varphi(x) = (x_1, x_2 - x_1, \ldots, x_N - x_{N-1}) .$$

Let $x \in \mathbb{X}_a$ and $y = \varphi(x)$. Since x satisfies the admissibility condition $\sup_{j=1,\ldots,N} |x_j - x_{j-1}| \leq 1$, y satisfies $\sup_{j=1,\ldots,N} |y_j| \leq 1$, and thus

$$\mathbb{X}_a \subset \varphi^{-1}(\{0,1\} \times \{-1,0,1\}^{N-1}) .$$

A new incremental state

The dynamic programming algorithm will be released with the **new state**

$$y = (y_1, \dots, y_N) \in \mathbb{Y} := \{0, 1\} \times \{-1, 0, 1\}^{N-1}$$

corresponding to the **increments of the state** x given by the inverse mapping

$$y = (y_1, \dots, y_N) \in \mathbb{Y} \mapsto \varphi^{-1}(y) = (y_1, y_1+y_2, \dots, y_1+y_2+\dots+y_N) \in \mathbb{N}^N$$

A GITTINS INDEX SUBOPTIMAL ALGORITHM

A bandit problem

If we see the mine as a collection of columns indexed by $j \in \{1, \dots, N\}$, selecting a control amounts to selecting a column and changing its height.

This is typical of a so called **job** or **bandit problem**:

- a job is selected at each time step;
- its state changes, but no the states of the other jobs.

The **state** of the job/column $j \in \{1, \dots, N\}$ is the height $i = x_j(t) \in \{1, \dots, H\}$.

Gittins index

To each column is associated a **Gittins index** which is a function of the state of this column.

For each job/column $j \in \{1, \ldots, N\}$ and state $i \in \{1, \ldots, H\}$, the Gittins index is

$$\mu_j(i) := \sup_{\tau=1,2,\ldots} \left(\frac{\sum_{t=0}^{\tau-1} \rho^t \mathcal{R}(i+t, j)}{\sum_{t=0}^{\tau-1} \rho^t} \right)$$

Gittins index Theorem

When

1. selecting one job only changes its state
2. and provides a reward depending only on this state
3. the criterion is a infinite discounted sum

selecting the job whose Gittins index is the highest at each time step provides the optimal path.

Heuristics

Select the column whose Gittins index is the highest among all admissible columns.

Change the Gittins index in

$$\mu_j^{(I)}(i) := \sup_{\tau=1,2,\dots,I} \left(\frac{\sum_{t=0}^{\tau-1} \rho^t \mathcal{R}(i+t,j)}{\sum_{t=0}^{\tau-1} \rho^t} \right)$$

NUMERICAL SIMULATIONS

Data from a Chilean mine

```
cd /home/delara/cermics/MATHEMATIQUES/mine/CodeScilab

scilab

exec mine_Gittins.sce
```