An Evolutionary Model for Underground Mining Planning

José Saavedra
Facultad de Ingeniería, Universidad de Los Andes.
Marco Alfaro
Facultad de Ingeniería, Universidad de Los Andes and Metalica Consultores S.A.
Jorge Amaya
Centro de Modelamiento Matemático, Universidad de Chile.

In this work we present a methodological approach for finding near optimal solutions to the problem of defining plans for underground mines. This problem of obvious combinatorial nature is intractable by means of traditional techniques (Mixed Integer Programming for example). The approach proposed here is based in the mechanism of natural selection, we construct a Genetic Algorithm to conduct the search of a approximate solution to the problem. In order to acquire this objective, we first need a model which characterizes the gravitational flow of material from the drawpoints, the chosen model for this objective is a cellular automata specifically designed with very simple rules of local evolution. The model was implemented and tested, the time needed for a solution in real type cases is much less than the time human planner needs actually for the same task. Possible extensions to this model are presented.

I. INTRODUCTION

Underground mining operations are very complex in nature. The main factors that determine this complexity (between others) are the unsuitable knowledge of ore resources contained in the mine and the parameters that characterize them. From this point of view, we can obviously see that human planners are unable to handling this complexity and as a consequence very poor solutions are obtained.

The problem of underground mining planning can be defined as choosing the best production plan in order to maximize the benefits derived from the operations. This planning could be made in three distinct scenarios: short term (operational), mid term (tactical) and long term (strategical). In this paper we focus our attention to the operational case.

Many resources are wasted in preparing mining plans. Usually this task takes two or three workers dedicated only to this work. Moreover, they don't have any tools available and this imposes on the final solutions the human planner bias. Some efforts have been made to solve this situation. From the point of view of classical optimization we can consider a model that has the following features:

- The mine is subdivided into blocks of homogeneous dimensions that conform a domain without holes (*edge connected*).
- When we extract a block from a given drawpoint, the blocks that are in the same column descend in one position (precedence restrictions) .
- Ore extraction is supposed to be realized in a soft manner, in the sense that adjacent columns can't show high differences in height, this restriction prevents dilution entry.
- Usually the objective function of this kind of models is the Net Present Value (NPV) of the economic result of the business derived from the operations.

this point of view is the dominant one presented in the work [6].

One of the problems that isn't resolved by this approach is the incorporation of the stochastic behavior of Gravitational Flow. Because of this, we call to this kind of models *deterministic*.

Another important source of problems in this deterministic perspective is the combinatorial explosion of the problem. We have the following result:

Proposition I.1
If we call $S_{k,l,m}$ the number of feasible sequences in a sector of k blocks width by l blocks length by m blocks height, with $m \le k,l$, the we have

$$(k \cdot l)^m < S_{k,l,m} < (k \cdot l \cdot m)!$$

the proof of this proposition can be found in the work [7]. As an application of this result, if we consider a sector with dimensions $n = k \cdot l \cdot m = 20 \cdot 20 \cdot 15$ then we are in presence of at least $400^{15} \approx 1.07 \times 10^{39}$ possible sequences, if we take 1 second in evaluating each one of this sequences then we need at least 34048129883307965499746321664130 years in order to resolve this problem. So an obvious conclusion is that exhaustive search is a very bad strategy for this class of problems.

Of course not every sequence is feasible. One possible choice, in order to reduce this high number of sequences, is to make an algorithm that can generate feasible sequences. It's not hard to see that this approach faces other problems that are not easy to resolve, for example to decide if a given sequence is feasible.

In this paper we propose a model that breaks this classical approach to the problem of underground mining planning. We choose as an alternative an Evolutionary Model because of the flexibility and good empirical results in problems of higher complexity (like the one presented here).

II. EVOLUTIONARY ALGORITHMS

Genetic Algorithms (GA from now) were introduced by John Holland in 1975. They are inspired in Darwin's mechanisms of natural selection. Such mechanisms establish that an individual is generated as a mixture of the genes from his parents by means of crossover, added to this process of mixture there is a process called mutation (change in some segments of genetic material).

This last mechanism implies in some way evolution because add novel elements not present in the genetic information from parents. Finally, the adaptation to the medium makes that some individuals survive and inherit their genes to his sons. The general form of an *Evolutionary Algorithm* is the following:

```
t := 0;
initialize(P(0));
evaluate(P(0));
While Not has been done Do
   P'(t) := select parents(P(t));
   P''(t) := recombinate(P'(t));
   P'''(t) := mutate(P''(t));
   evaluate(P'''(t));
   P(t + 1) := natural selection(P(t); P'''(t));
   t := t + 1;
Next
```

In this Algorithm t is the counter of generations and evaluate (P) implies to evaluate fitness function to every member of population P. This algorithm finish when the fitness value of actual population $P(t)$ in time t don't innovate or after a fixed number of iterations.

The considerations in the moment of implementing GA strategies are: chromosomic representation, population size, fitness function, crossover and mutation operators, crossover and mutation probabilities. For further reading on this technique go to references [3], [5], [7].

III. MODELS FOR GRAVITATIONAL FLOW

If we want to incorporate the stochastic behavior of granular flow in our model we need first consider models for this phenomenon.

In the last decade, some efforts have been made. The main results were obtained by chilean well known scientists, Eric Goles [1] and Servet Martínez [4]. These two approaches are similar in the kind of technique used, both are cellular automata models.

Another interesting model is the one proposed by Gregorio González [2]. In this work the ideas presented in [1] are refined. Applications to underground mining are presented.

The most recent development in this area is the model presented by Marco Alfaro [8]. This model is a Cellular Automata that has the benefit of simple rules of evolution, and as a consequence, it's possible to obtain a efficient implementation [9].

Independent of the chosen model, it's fundamental for the proposed methodology to have some gravitational flow model. As we will see in the next section, our model consider a gravitational flow model in the kernel of the evaluation function of the proposed genetic algorithm.

IV. PROPOSED EVOLUTIONARY MODEL

A. Extraction Charts

Operations in underground mining are realized in turns. Such turns are usually of 8 hours each one and each day is divided in three turns. Given a set of drawpoints $\{\eta_i\}_{i=1}^{n}$ a *Extraction Chart* is an assignment of tons to extract for each drawpoint en each turn. We can if we wish consider another kinds of periods like days, weeks, months, etc. Formally:

Definition IV.1

Given a set of drawpoints $\{\eta_i\}_{i=1}^{n}$ we define a *Extraction Chart* as a matrix $M \in M_{m\times n}(\mathbb{Z})$ with *n* the number of drawpoints and *m* the number of turns. The coefficient m_{ij} of this matrix is defined as:

$m_{ij} :=$ tons (expressed in shovels) to extract in drawpoint *j* in turn *i*

Observation IV.1
Extraction Chart *M* don't have to be a square matrix.

If we call D_i to demand (in shovels) in turn *i* then we have

$$\sum_{k=1}^{n} m_{ik} \le D_i$$

"$\le$" in the last restriction gives the choice of not satisfy the hole demand, if not we could force our algorithm to extract blocks that gives worse solutions.

B. Chromosome Representation

The natural chromosome representation for a extraction chart is a matrix like the one previously defined. This will be the formal structure in which we will define the crossover and mutation operators.

C. Crossover Operator

Given two individuals (matrix) from a given population, for example *M;N*, we define the associated crossover operator as:

- We select randomly a number in the set $\{1,\ldots,m\}$ (i.e. we select randomly a turn). Let i^* such number.
- We consider the submatrices of *M* and *N* given by $M_1 = (m_{ij})_{i=1}^{i^*}$, $j \in \{1,\ldots,n\}$ and $M_2 = (m_{ij})_{i=i^*+1}^{m}$, $j \in \{1,\ldots,n\}$ and analog for *N*.
- We define the new matrices $\tilde{M} = \begin{pmatrix} M_1 \\ N_2 \end{pmatrix}$ and $\tilde{N} = \begin{pmatrix} N_1 \\ M_2 \end{pmatrix}$

This crossover operator guarantees feasibility from turn to turn of extraction charts, this because we maintain demand inequality in each turn.

D. Mutation Operator

To define a mutation operator we have to randomly select a turn (row in the extraction matrix). We proceed then to re-balance the selected row with a number randomly chosen between 0 and the demand of the turn, then we distribute randomly in to the selected turn. This operator guarantees feasibility of the selected row (the number chosen is less than demand).

E. Fitness Function

In order to evaluate the fitness of a extraction chart we have to run a simulation of the chart and then obtain a list of blocks with laws of ore grades. With this information we can obtain the benefit given by that extraction chart incorporating NPV in this calculation.

F. The Algorithm

The search algorithm is described with the following pseudo-code:

Algorithm.
Inputs:
Block Model
Location of Drawpoints
Population Size *n*
Parameters: Crossover and Mutation Probabilities
Iterations ν
Probabilities for Cellular Automata
Outputs:
Optimal Extraction Chart
Algorithm:
Initialize Extraction Charts Population (*n* Extraction Charts): $\mapsto$ *P*(0);
Fitness Evaluation(*P*(0));
For *t* = 0 To ν
Begin
  Crossover(*P*(*t*)): $\mapsto$ *P*(*t* + 1);
  Mutation(*P*(*t* + 1));
  Fitness Evaluation(*P*(*t* + 1));
  Next Generation Selection(*P*(*t*),*P*(*t* + 1));
End For

Function Fitness Evaluation(*P*(*t*));
For *i* = 1 To *n*
Begin
  Simulate(Extraction Chart *i*(*t*));
    ↲ Cellular Automata(Extraction Chart *i*(*t*));
    ↲ Economic Evaluation(Extraction Chart *i*(*t*));
End For
End Function

V. NUMERICAL RESULTS

A. Trivial Case

This case is a sector of 4 by 4 by 4 blocks, all of them with grade 0, i.e. sterile. This simple example has only two extraction points, one in coordinates $(1; 1; 0)$ and the other in coordinates $(3; 2; 0)$. The obvious solution to this problem is to extract nothing from drawpoints. This example was tested with the following parameters:

Application Parameters	
Turn Number	5
Max. Demand in each Turn	10

GA Parameters	
Iterations Number	100
Population Size	10
Crossover Probability	0.8
Mutation Probability	0.2
Selection Policy	Between Parents and Sons
Selection Method	Drawing

The Cellular Automata transition Probabilities are given by the following table:

0.02	0.10	0.02
0.10	0.51	0.10
0.02	0.10	0.02

The solution to this problem was obtained in 63 iterations.

The results of the iterations are summarized in the following picture. This picture illustrate the behavior of the best solution in each iteration:

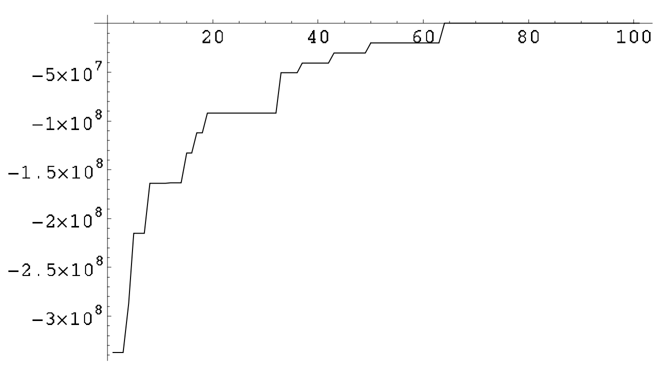

FIG. 1: Best Individual Evolution.

As we can see, the algorithm generates a sequence of solutions in which each one is at least equal or better than the previous.

B. Calibration of Model Parameters

In order to operate this GA, we need calibrate the functional parameters. The used example was the previous one. We will vary crossover and mutation probabilities.

To denote the instances of the problem we will use the notation *An*1/*n*2*M*, where *n*1 is crossover probability, *n*2 mutation probability, selection method *A* (Parents and Sons: *PH*; Only Sons: *SH*) and parent selection *M* (Roulette: *R*; Drawing: *S*).

Each instance was runned 30 times and we determine in which generation we reach the optimum. The following table resumes the results.

Instance	Mean	Standard Deviation
*PH*80/20*S*	64.93	22.89
*PH*90/30*S*	50.17	21.55
*PH*80/30*S*	45.70	20.36
*PH*80/40*S*	38.33	20.33
*PH*80/50*S*	35.40	15.70
*PH*90/20*S*	61.13	27.64
*PH*100/20*S*	55.23	27.17
*PH*100/0*S*	100.00	0.00 (*)
*PH*100/100*S*	31.70	9.50
*SH*100/100*S*	100.00	0.00 (*)
*SH*80/20*S*	100.00	0.00 (*)
*PH*100/100*R*	28.97	15.01

(*) means that instance doesn't converge never in 100 iterations.

C. Another Factors

We prove many others effects: Maximum Demand, Number of Blocks, Population Size, Drawpoints Number.

Almost all of them gives a linear dependence between the number of iterations needed to reach the optimum and the increase of the values. The only factor that shows exponential behavior is Maximum Demand, i.e., if we vary the Maximum Demand parameter then the number of iterations needed to converge grows exponentially. We summarize this effects in the following table:

Effect	Variation	Result
Max. Demand	Grow	Exponential Grow
Number of Blocks	Grow	Linear Grow
Population Size	Grow	Linear Decrease
Drawpoints Number	Grow	Linear Grow

D. Real Scale Example

This example was runned with the following parameters:

Application Parameters	
Turn Number	12
Max. Demand in each Turn	100
Number of Drawpoints	171
Number of Blocks	400000

GA Parameters	
Iterations Number	34
Population Size	8
Crossover Probability	1
Mutation Probability	1
Selection Policy	Between Parents and Sons
Selection Method	Drawing

The results are summarized in the following table:

Results	
Execution Time	3:11
Iterations Number	34

We can extrapolate this result and see that in real situations it would take about 8 hours to finish the optimization process.

VI. EXTENSIONS TO THE MODEL

The proposed model don't consider downstream operations. These operations are in general the most restrictive operations. For example in some downstream operations smooth ore grade is required, the fine mid term promise has to be accomplished, etc. In all of these cases the proposed model don't give an answer.

Recently, an extension of the model proposed in this paper have been implemented [10]. This model uses Genetic Algorithms too and the main characteristics are:

1. Genetic Algorithms mechanisms in the search of solutions.
2. Mixture Models for Ore Unload.
3. Operations are considered as transport problem.
4. Restrictions on the quantity of ore to be extracted from drawpoints are imposed.
5. Capacity constraints in downstream operations are considered.
6. NPV evaluation.
7. Drawpoint Grade behavior is assumed.

In real type situations, the response time of this implementation are in the order of 3:00 hrs [10].

The next challenge is to integrate the model proposed in [7] whit the one proposed in [10]. Both models were constructed with the same philosophy so we can expect some kind of integration and scale economies between + both models.

In the future we hope to construct a model that incorporates gravitational flow simulations and downstream operations.

Another source of extensions to the model is to consider operational events as stochastic processes. In this way simulations of extraction charts would be more realistic. In order to accomplish this objective is needed historical data to calibrate the parameters of the involved stochastic processes.

VII. CONCLUSIONS

As an obvious first conclusion we have that this problem has a very large number of involved variables.

In this moment it's really very difficult consider the hole complexity of this problem. For example at the moment there aren't appropriate models to handle the breaking of the solid rock.

Another important conclusion is that the stochastic nature of the phenomenon is hard to include in the modeling process. Some attempts have been made but at the moment this efforts are in initial development. The proposed methodology could be applied in real type situations. The response times are good compared with human planners. The most important advantages of the methodology is that we can test in an efficient way many choices and that the search procedure is well conducted.

With this methodology we are in presence of a flexible model that can be customized to satisfy the planner needs.

VIII. ACKNOWLEDGMENTS

J.S. wish to acknowledge the support of FAI (Fondo de Ayuda a la Investigación), Proyecto Ingeniería 2004, Universidad de Los Andes.

REFERENCES

[1] Eric Goles y Sebastián Peña: *Modelamiento y Simulación del Flujo Gravitacional*. Informe Final PB5-10004, Apéndice N°2, 1996.

[2] Gregorio González: *Estudio del Comportamiento de un Material Granular Mediante Modelos Computacionales*. Memoria para optar al título de Ingeniero Civil Matemático, 1999.

[3] John R. Koza: *Genetic Programming: on the programming of computers by means of natural selection*, The MIT Press, Cambridge, Massachussets, 1992.

[4] Servet Martínez: *Consideraciones Acerca del Modelamiento de Flujo Gravitacional*. Informe Final PB5-10004, Apéndice N°3, 1996.

[5] Zbigniew Michalewicz: *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, 1996.

[6] Nelson Morales: *Modelos Matemáticos para Planificación Minera*. Memoria para optar al título de Ingeniero Civil Matemático, Universidad de Chile, 2002.

[7] José Saavedra: *Secuencias, Flujo Gravitacional y Evolución en Planificación Minera*. Memoria para optar al título de Ingeniero Civil Matemático, Universidad de Chile, 2002.

[8] Marco Alfaro, José Saavedra: *Predictive Models for Gravitational Flow*. To be presented in MassMin 2004.

[9] Carlo Calderón, Marco Alfaro, José Saavedra: *Computational Model for Simulation and Visualization of Gravitational Flow*. To be presented in MassMin 2004.

[10] Andrés Donoso: *Modelos para la Planificación Operacional de Producción en Minería Subterránea*. Memoria para optar al título de Ingeniero Civil Industrial, Universidad de Los Andes, 2004.