A hybrid heuristic algorithm based on GRASP, VND, ILS and Path Relinking for the open-pit-mining operational planning problem

Igor Machado Coelho
Sabir Ribas
imcoelho@ic.uff.br
sribas@ic.uff.br
Fluminense Federal University, Computer Science Institute, Niterói, 24210-240, RJ, Brazil
Marcone Jamilson Freitas Souza
Vitor Nazário Coelho
marcone@iceb.ufop.br
vncoelho@gmail.com
Federal University of Ouro Preto, Department of Computer Science, Ouro Preto, 35400-000, MG, Brazil
Luiz Satoru Ochi
satoru@ic.uff.br
Fluminense Federal University, Computer Science Institute, Niterói, 24210-240, RJ, Brazil

Abstract. *This work deals with dynamic truck allocation in open-pit-mining operational planning. The objective is to optimize mineral extraction in the mines by minimizing the number of mining trucks used to meet production goals and quality requirements. According to literature, this problem is NP-hard, therefore heuristic strategies are justified. We present a hybrid algorithm that combines the power of three metaheuristics, Greedy Randomized Adaptive Search Procedure, Variable Neighborhood Descent and Iterated Local Search, as well as a mathematical programming module that is triggered periodically to solve smaller parts of the problem in optimality, and a post optimization module based on Path Relinking approach. The proposed algorithm was tested in a set of instances based on real-life problems and the results were compared with other metaheuristics-based strategies available in the literature and with an optimization solver. The computational experiments show that the developed algorithm is competitive, since it can obtain good quality solutions in an acceptable execution time.*

Keywords: *Open-pit-mining, Metaheuristics, Iterated Local Search, GRASP, Path Relinking*

1. INTRODUCTION

This work deals with dynamic truck allocation in Open-Pit-Mining Operational Planning (OPMOP). The situation involves mineral extraction (ore and waste rocks), as well as the equipment (shovels and mining trucks) used in the operation. Also taken into consideration are several ore control parameters (e.g., % iron, % silica, and water). The objective is to determine the extraction rate at each pit in a way that production and quality goals are satisfied, and to minimize the number of trucks needed for the process.

The problem considers that only mining trucks are used to transport the material from the loading points (pits) to the unloading points (either the crusher or the waste rock deposit). It is considered, too, that there are shovels of different productivities and their set is smaller than the number of pits they can be allocated to. Given the high cost of a shovel, a minimum quantity of extrac mineral is required to justify its use.

In the dynamic allocation system, the trucks are not fixed to a specific shovel/or pit. They can be directed to different pits, which increases the fleet productivity, reducing the amount of equipment needed to maintain a certain level of production. In this system it is also possible to decrease the time of the queue, since the truck can be allocated to different loading points. The disadvantages of dynamic vehicle allocation are: the demand for a greater number of operations; and a computerized dispatching system for the mining trucks.

The problem in focus has the Multiple Knapsack Problem (MKP) as a subproblem. In fact, the analogy can be made by considering each shovel like a knapsack and the loads (ore or waste rock) of the trucks as the items. In this analogy, the goal is to determine which loads are the most attractive to allocate to each knapsack, respecting its capacity (productivity). Thus, as MKP belongs to the NP-hard class (Papadimitriou and Steiglitz, 1998), OPMOP does too. Since in real cases the decision must be fast and it is unlikely that optimum solutions would be obtained by exact techniques in a short space of time, it is proposed to find sub-optimal solutions for the problem by means of heuristic techniques. The proposed heuristic algorithm is based on the procedures Greedy Randomized Adaptive Search Procedures - GRASP (Feo and Resende, 1995), Variable Neighborhood Descent - VND (Mladenovic and Hansen, 1997) and Iterated Local Search - ILS (Lourenço et al., 2003). To test its efficiency, the results from this algorithm were compared to those achieved by a market optimization software applied to a mathematical programming model also proposed in this work.

The contribution of this work is the presentation of a more complete mathematical programming model of OPMOP than those found in literature. This model seeks to more faithfully depict a real operational mining industry environment. Moreover, it presents a new heuristic model not yet found in literature in order to solve the problem in focus.

This paper is organized as follows. Section 2 shows the related work. Section 3 presents a mathematical programming formulation to OPMOP, while Section 4 presents a heuristic approach to the problem in focus. The testing scenarios are described in Section 5, while in the following section, the computational experiments are presented and analyzed. Section 7 concludes the work.

2. RELATED WORKS

White and Olson (1986) proposed an algorithm that is the basis for the *DISPATCH* System, which operates in many mines around the world. A solution is obtained in two steps. The first, based on linear programming, handles the problem of ore mixture optimizing by minimizing costs considering the mining rate, the quality of the mixture, the ore feed rate to the plant for beneficiation, and the material handling. The restrictions of the model are related to the production capacity of the shovels, the quality of the mixture and the minimum feeding rate to the processing plant. The second stage of the algorithm, which is solved by dynamic programming, uses a model similar to White et al. (1982), differing from this by using a decision variable for the volume of material transported per hour on a given route, instead of the truck working rate per hour. Also considered is the presence of storage piles. In this second stage of the algorithm, the objective is to minimize material transportation in the mine.

Sgurev et al. (1989) describe an automated system for real-time control of truck haulage in open-pit mines. This system is called TRASY and it is designed towards the improvement of the technical-economical indices of the loading-unloading process in open-pit mines where trucks are used as vehicles. The authors describe the two ways of organizing the trucks work: on a closed-circuit system and on an open-circuit system, so called dynamic allocation system in the present paper. The benefits of the open-circuit system are shown and the authors describe the four modules of the TRASY system: configuration, control, monitoring and report. The authors conclude that the increase of the operation productivity in open-pit mines may be achieved by improving the effectiveness of the loading-haulage process control, so the introduction of automated systems for haulage vehicles control is one way to accomplish this goal. However, this system does not take into account the quality goals of the ore control parameters.

Chanda and Dagdelen (1995) developed a linear programming model that solves the problem of mixed minerals in the short-term planning of a coal mine. The objective function of this model is the weighted sum of three distinct objectives: to maximize an economic criterion, to minimize production deviations, and to minimize quality deviations from the desired values of the control parameters. No allocation for the loading and transport equipment was considered in this model.

Ezawa and Silva (1995) developed a system for dynamic truck allocation with the objective of reducing variability in the levels of the ore and increasing transport productivity. The system uses a heuristic to sequence the trucks in order to minimize changes in the levels. To validate it, the authors used a simulation and the theory of graphs for the mathematical modeling of the mine. Deploying this system transport productivity increased by 8% and management obtained more accurate data in real time.

Alvarenga (1997) developed a program for the optimal dispatch of trucks in the iron mining of an open pit mine, with the objectives of minimizing the queue time of the trucks in the fleet, increasing productivity and improving the quality of the extracted ore. In the work, which is the basis of the SMART MINE system widely used in various Brazilian mines, a technique of stochastic optimization was applied, using the genetic algorithm with parallel processing. Basically, the problem is to indicate the best point of tipping or loading and the trajectory for the movement, when there is a situation of choice to be made. The author pointed to productivity gains of 5% to 15%, proving the validity of the proposal.

Merschmann (2002) developed an optimization system and simulation for analyzing the production scenario in open pit mines. The system, called OTISIMIN (Simulator and Optimizer for Mining), was developed in two modules. The first is the optimization module where a linear programming model is constructed and solved, while the second is a simulation module that allows the user to use the results obtained by solving the linear programming model as input for the simulation. The optimization module was developed with the aim of optimizing the process of mixing the ores from the mining of several pits in order to meet the quality specifications imposed by the treatment plant and allocating the equipment (trucks, shovels and / or excavators) to the pits, considering both static and dynamic truck allocation. The developed model does not consider production optimization and quality targets, or reduction of the number of trucks required by the production system.

Godoy and Dimitrakopoulos (2004) deals with the open pit mine design and production scheduling problem, with a view to find the most profitable mining sequence over the life of a mine. According to the authors the dynamics of mining ore and waste and the spatial grade uncertainty make predictions of the optimal mining sequence a challenging task. The authors show a risk-based approach to life-of-mine production scheduling, including the determination of optimum mining rates for the life of mine, whilst considering ore production, stripping ratios, investment in equipment purchase and operational costs; and the generation of a detailed mining sequence from the previously determined mining rates, focusing on spatial evolution of mining sequences and equipment utilization. The production scheduling stage uses a specially-developed combinatorial optimization algorithm based on the Simulated Annealing metaheuristic. A new risk-based, multistage optimization process for long-term production scheduling is presented, and the results show the potential to considerably improve the valuation and forecasts for life-of-mine schedule.

Guimaraes et al. (2007) presents a computational simulation model to validate the results obtained by applying a mathematical programming model to determine the mining rate in open pit mines. LINGO solver, version 7.0, is used for optimizing the problem and ARENA, version 7.0, simulate the solver's solution. Contrary to belief, the modeling demonstrate that by increasing the number of vehicles, the production goal is not met and is further deterred due to increased queue time. Thus, increasing the number of vehicles does not necessarily optimize mining operations.

Boland et al. (2009) deals with the open pit mining production scheduling problem. The treated problem consists of finding the sequence in which the blocks should be removed from the pit, over the lifetime of the mine, such that the net present value (NPV) of the operation is maximized. Due to the large number of blocks and precedence constraints linking them, blocks are aggregated to form larger scheduling units. The authors investigated the characteristics of the problem and showed how the aggregates can be systematically divided into bins (groups of blocks) so that the solution of the linear programing (LP) relaxation with all processing decision variables fully disaggregated to block level (D-LP) can be recovered from the solution of our compactly disaggregated LP relaxation (B-LP) with processing decisions made at the level of bins. As the number of bins is much smaller than the number of blocks, using their binning approach, D-LP can be solved to optimality for much larger data instances than by a direct disaggregation approach. They showed that their approach can lead to significant improvements in NPV.

Coelho et al. (2008) presents a metaheuristics based algorithm to solve the OPMOP. It is also presented a more complete formulation of the mathematical programming model to solve the problem. The proposed model is an extension of the model of Costa (2005). Computational experiments compare the solutions of the metaheuristics approach and the mathematical programming model using the CPLEX solver. Results show that the metaheuristics approach can find good solutions with low variability quickly, what did not occur with CPLEX in the computational experiments. Since the decision maker needs fast answers the proposed algorithm is validated to real applications.

Ribas et al. (2009) proposes a parallel version of the Iterated Local Search metaheuristic to solve the OPMOP. The authors compare the sequential and parallel versions of the proposed algorithm, as well as with a metaheuristic-based algorithm from literature (Coelho et al., 2008). Also, a comparision was performed between the solutions found by the parallel algorithm and those obtained by a mathematical programming model. The results show the efficiency of the parallel implementation since it solves the OPMOP with gains in the solutions quality of up to 242,13%.

3. MATHEMATICAL FORMULATION

This section presents a mixed integer programming (MIP) model based on goal programming (Romero, 2004) to solve OPMOP. This model was proposed by Coelho et al. (2008) and it is an extension of the model of Costa (2005). The model refers to production planning for an hour, replicated while there isn't any exhausted pit and operational conditions of the mine remain the same. The objective is to minimize the deviations of the production and quality goals and to reduce the number of vehicles required for the operation.

Let the parameters be:

O : Set of ore pits;
W : Set of waste rock pits;
F : Set of ore and waste rock pits, ie, $F = O \cup W$;
P : Set of control parameters analysed in the ore (% Fe, SiO_2, etc);
S : Set of shovels;
T : Set of mining trucks;
O_r : Recommended rate of mining for ore (ton/h);
O_l : Minimum rate of mining for ore (ton/h);
O_u : Maximum rate of mining for ore (ton/h);
W_r : Recommended rate of mining for waste rocks (ton/h);
W_l : Minimum rate of mining for waste rocks (ton/h);
W_u : Maximum rate of mining for waste rocks (ton/h);
α^- : Penalty for negative deviation from the production of ore;
α^+ : Penalty for positive deviation from the production of ore;
β^- : Penalty for negative deviation from the production of waste rocks;
β^+ : Penalty for positive deviation from the production of waste rocks;
p_{ij} : Percentage of the control parameter j in pit i (%);
pr_j : Recommended percentage for the control parameter j in the mixture (%);
pl_j : Minimum allowable percentage for the control parameter j in the mixture (%);
pu_j : Maximum allowable percentage for the control parameter j in the mixture (%);
λ_j^- : Penalty for a negative deviation of the control parameter j in the mixture;
λ_j^+ : Penalty for a positive deviation of the control parameter j in the mixture;
ω_l : Penalty for use of the l-th truck;
Qu_i : Maximum rate of mining for the pit i (ton/h);
Tx_l : Maximum rate of use for the truck l (%);
Sl_k : Minimum productivity for the shovel k (ton/h);
Su_k : Maximum productivity for the shovel k (ton/h);
cap_l : Capacity of truck l (ton);
ct_{il} : Total cycle time of the truck l in the pit i (min);
g_{lk} : 1, if the truck l is compatible with the shovel k; and 0, otherwise.

Consider also the following variables of decision:

x_i : Mining rate of the pit i (ton/h);
y_{ik} : 1, if the shovel k operates in the pit i; e 0, otherwise.
n_{il} : Number of trips that a truck l performs to the pit i;
D_o^- : Negative deviation from the recommended ore production (ton/h);
D_o^+ : Positive deviation from the recommended ore production (ton/h);
D_w^- : Negative deviation from the recommended waste rock production (ton/h);
D_w^+ : Positive deviation from the recommended waste rock production (ton/h);

d_j^- : Negative deviation of the control parameter j in the mixture (ton/h);
d_j^+ : Positive deviation of the control parameter j in the mixture (ton/h);
U_l : 1, if the truck l is being used; and 0, otherwise.

Next, the equations (1)-(26) present the MIP model for the problem in focus.

$$\min \sum_{j \in P} \lambda_j^- d_j^- + \sum_{j \in P} \lambda_j^+ d_j^+ + \alpha^- D_o^- + \alpha^+ D_o^+ + \beta^- D_w^- + \beta^+ D_w^+ + \sum_{l \in T} \omega_l U_l \tag{1}$$

$$\sum_{i \in O} (p_{ij} - pu_j) x_i \leq 0 \quad \forall j \in P \tag{2}$$

$$\sum_{i \in O} (p_{ij} - pl_j) x_i \geq 0 \quad \forall j \in P \tag{3}$$

$$\sum_{i \in O} (p_{ij} - pr_j) x_i + d_j^- - d_j^+ = 0 \quad \forall j \in P \tag{4}$$

$$\sum_{i \in O} x_i \leq O_u \tag{5}$$

$$\sum_{i \in O} x_i \geq O_l \tag{6}$$

$$\sum_{i \in O} x_i + D_o^- - D_o^+ = O_r \tag{7}$$

$$\sum_{i \in W} x_i \leq W_u \tag{8}$$

$$\sum_{i \in W} x_i \geq W_l \tag{9}$$

$$\sum_{i \in W} x_i + D_w^- - D_w^+ = W_r \tag{10}$$

$$x_i \leq Qu_i \quad \forall i \in F \tag{11}$$

$$x_i \geq 0 \quad \forall i \in F \tag{12}$$

$$d_j^-, d_j^+ \geq 0 \quad \forall j \in P \tag{13}$$

$$D_o^-, D_o^+ \geq 0 \tag{14}$$

$$D_w^-, D_w^+ \geq 0 \tag{15}$$

$$\sum_{k \in S} y_{ik} \leq 1 \quad \forall i \in F \tag{16}$$

$$\sum_{i \in F} y_{ik} \leq 1 \quad \forall k \in S \tag{17}$$

$$y_{ik} \in \{0, 1\} \quad \forall i \in F,\ \forall k \in S \tag{18}$$

$$x_i - \sum_{k \in S} Cu_k\, y_{ik} \leq 0 \quad \forall i \in F \tag{19}$$

$$x_i - \sum_{k \in S} Cl_k\, y_{ik} \geq 0 \quad \forall i \in F \tag{20}$$

$$n_{il} ct_{il} - 60 \sum_{k \in S,\ g_{lk}=1} y_{ik} \leq 0 \quad \forall i \in F,\ \forall l \in T \tag{21}$$

$$x_i - \sum_{l \in T} n_{il}\, cap_l = 0 \quad \forall i \in F \tag{22}$$

$$\frac{1}{60} \sum_{l \in T} n_{il} ct_{il} \leq Tx_l \quad \forall i \in F \tag{23}$$

$$U_l - \frac{1}{60}\sum_{l \in T} n_{il} ct_{il} \geq 0 \qquad \forall i \in F \tag{24}$$

$$n_{il} \in Z^+ \qquad \forall i \in F,\ \forall l \in T \tag{25}$$

$$U_l \in \{0,1\} \qquad \forall l \in T \tag{26}$$

The objective function (1) seeks to minimize the differences with regard to production goals of ore and waste rock, quality targets of the mixture, as well as to reduce the number of trucks used. The constraints (2)-(15) model the classic problem of blending with goals. In this group, the constraints (7) and (10) relate respectively to the care of the production targets of ore and waste rock, while the constraints (11) limit the maximum mining rate defined by the user for each pit.

The other constraints which complement the model can be divided into two groups. The first concerns the allocation of shovels and productivity range in order to justify the equipment use. The second is related to the allocation of trucks for material transport in the mine.

For the first group, the constraints (16) define that at most one shovel can be allocated to each pit, while the constraints (17) define that each shovel can be allocated to one pit at most. The constraints (18) define that the variables y_{ik} are binary. The constraints (19) and (20) limit respectively the maximum and minimum mining rate defined by the shovel allocated to the pit.

In the second group of constraints, each constraint (21) forces the truck to only perform trips where there is compatible shovel allocated. The constraints (22) are such that the mining rate of a pit is equal to the total production of the trucks allocated to the pit. The constraints (23) ensure that each truck l is in operation for at most $Tx_l\%$ in an hour. The constraints (24), together with the objective function, force the number of trucks used to be penalized. The constraints (25) force the number of trips that a truck performs to a pit to be a positive integer value. The constraints (26) indicate that the variables U_l are binary.

4. HEURISTIC MODEL

4.1 Representation of a Solution

A solution is represented by a matrix $R = [Y \mid N]$, where Y is a matrix $|F| \times 1$ and N is a matrix $|F| \times |T|$.

Each cell y_i of the matrix $Y_{|F|\times 1}$ represents a shovel k allocated to the pit i. A value -1 means that there isn't any shovel allocated to that pit. If there aren't any trips made to a pit i, the shovel k associated to that pit is considered *inactive* and it isn't penalized for a production below the minimum for a shovel (restrictions (18) of the mathematical programming model).

In the matrix $N_{|F|\times|T|}$, each cell n_{il} represents the number of trips that each truck $l \in T$ performs to a pit $i \in F$. A value 0 (zero) means that there aren't any trips allocated to that truck, while a value -1 indicates that the truck and the shovel allocated to that pit aren't compatible.

From Y, N and the cycle times from the matrix CT ($|F| \times |T|$ dimensional) the extraction rate at each pit is determined, as well as the sum of the cycle times for each truck.

4.2 Neighborhood

To explore the solution space of the problem, eight movements were developed and are presented below. Each movement is associated to a neighborhood defined by the function $N(.)$.

Movement Number of Trips - $N^{NT}(s)$. This movement increases or decreases the number of trips of a truck l to a pit i where there's an allocated compatible shovel. Thus, in this movement, a cell n_{il} of the matrix N has its value increased or decreased by one trip.

Movement Load - $N^L(s)$**.** Consists of changing two separate cells y_i and y_k of the matrix Y, i.e. exchanging the shovels that operate in the pits i and k, if both pits have allocated shovels. If only one of the two pits has an allocated shovel and the other is available, this movement will relocate the shovel to the available pit. In order to maintain compatibility between shovels and trucks, the trips made to the pit are relocated along with the shovel.

Movement Relocate Trip from a Truck - $N^{TT}(s)$**.** Consists of choosing two cells n_{il} and n_{kl} from the matrix N and passing a trip from n_{il} to n_{kl}. Thus, in this movement, a truck l ceases to do a trip to a pit i and does it at another pit k. Compatibility restrictions between equipment are respected in this movement, so the trip relocation is only done when there's compatibility between them.

Movement Relocate Trip from a Pit - $N^{TP}(s)$**.** Two cells n_{il} and n_{ik} from the matrix N are chosen and a unit of n_{il} is relocated to n_{ik}. So this movement consists of relocating a trip from a truck l to a truck k that are both working at the pit i. Compatibility restrictions between equipment are respected in this movement, so the trip relocation is only done when there's compatibility between them.

Movement Pit Operation - $N^{PO}(s)$**.** Consists of removing from operation the shovel that is allocated to the pit i. The movement removes all the trips made to this pit, leaving this shovel *inactive*. The shovel is again put in operation as soon as a new trip is associated to it.

Movement Truck Operation - $N^{TO}(s)$**.** Consists of selecting a cell n_{il} from the matrix N and zero-fill its content, meaning that the truck l doesn't operate in the pit i anymore.

Movement Swap Trips - $N^{ST}(s)$**.** Two cells of the matrix N are selected and a trip is relocated between them. This movement can occur in any cell of the matrix N if compatibility restrictions between equipment are respected.

Movement Swap Shovels - $N^{SS}(s)$**.** Consists of swapping two separate cells y_i and y_k from the matrix Y, i.e. changing the shovels that operate in the pits i and k. This movement is similar to the movement L, because the shovels are also exchanged, but the trips made to the pits are not exchanged. To maintain compatibility between the shovels and trucks, the trips made to incompatible equipment are removed.

4.3 Evaluation of a Solution

As the developed movement can generate infeasible solutions, a solution is evaluated by a single-objective function $f : S \leftarrow \mathbb{R}$, where S represents the set of all possible s solutions generated from the movements developed in the previous section. This function f, defined by Eq. (27), to be minimized, consists of two parts: first, the objective function itself (Eq. (1) from the mathematical programming model) and second, a group of functions that penalize the occurrence of infeasibility in the current solution.

$$f(s) = f^{MP}(s) + f^{p}(s) + \sum_{j \in P} f_j^q(s) + \sum_{l \in T} f_l^u(s) + \sum_{k \in S} f_k^c(s) \tag{27}$$

In Eq. (27), $f^{MP}(s)$ is the objective function from the mathematical programming model given by Eq. (1), i.e. $f^{MP}(s)$ evaluates $s \in S$ considering the production and quality goals, as well as the number of trucks used; $f^p(s)$ evaluates s for disrespect of the production limits established for the ore and waste rock; $f_j^q(s)$ evaluates s considering the infeasibility of the j-th control parameter; $f_l^u(s)$ evaluates s regarding disrespect of the maximum use rate of the l-th truck; and $f_k^c(s)$ evaluates s for disrespect of the productivity limits of the shovel k.

4.4 Initial Solution Generation

An initial solution to the problem is obtained by a partially greedy constructive procedure, similar to the construction phase of GRASP (Feo and Resende, 1995). The construction is done in two steps. The allocation of the shovels and the distribution of trips to the pits are done in the first step for the waste rock pits, and secondly, for the ore pits.

This strategy is adopted because in the waste rock pits, it is important to meet production and not necessary to observe the quality of the control parameters. The classification of the candidate elements to be inserted in the solution is made considering that for the waste rock pits, the best is the pit that has the greatest mass, the best is the shovel that provides the greatest production and the best is the truck that is the largest. For the ore pits, it is considered that the best pit is the one that has the least deviation of the control parameter levels in relation to the targets, the best is the shovel that provides the greatest production and the best is the truck that has the smallest capacity.

The choice of the ore pit in the second stage of the construction is done by a guide function, as in Bresina (1996). First, all pit candidates are sorted with respect to the deviation values of the quality goals. To the r-th better classified pit, that pit is associated to the guide function $bias(r) = 1/(2r)$. Each candidate pit is chosen with probability $p(r) = bias(r)/\sum bias(r)$. In this strategy, it is more likely to choose the pit that best helps to minimize the deviations from the quality targets.

4.5 A Hybrid Algorithm

We present a hybrid algorithm, called H-GVILS, that combines the power of the heuristic procedures GRASP (Feo and Resende, 1995), *Variable Neighborhood Descent* - VND (Mladenovic and Hansen, 1997) and *Iterated Local Search* - ILS (Lourenço et al., 2003), with a mathematical programming based module. When ILS reaches a certain level of perturbation, this module is triggered to solve smaller parts of the problem in optimality.

Figure 1 shows the H-GVILS algorithm.

Algorithm 1: ***H-GVILS***

Input: sets M, E, T, C, V (See parameters in Section 3.) - **Output**: Solution s

$s_0^{'} \leftarrow$ *BuildWasteSolution*()
$s_0 \leftarrow$ *BuildOreSolution*($s_0^{'}$)
$s \leftarrow$ *VND*(s_0)
while *stop criterion not satisfied* **do**
    $s' \leftarrow$ *Perturbation*(s, level p)
    $s'' \leftarrow$ *VND*(s')
    if $f(s'') < f(s)$ **then**
        $s \leftarrow s''$
    end
end
return s

Figure 1: H-GVILS Algorithm

Building an initial solution s_0 (lines 1 and 2 of the Algorithm in Fig. 1) is made by the procedure described in Subsection 4.4. The local search (line 3 of the Algorithm in Fig. 1), in turn, uses the VND procedure with the movements described in Subsection 4.2. Strategically, the local search operates in the neighborhoods in a pre-defined order, beginning from the ones requiring less computational effort to those that require more effort. As in the preliminary tests some neighborhoods did not produce great local optimums or spend too much processing time to achieve a local optimum; only a small group of neighborhoods has been used in the local search. Thus, the VND uses the following order for the exploration of the neighborhoods: N^L, N^{NT}, N^{TT} and N^{TP}. The movements on neighborhoods N^{SS}, N^{TO}, N^{PO} and N^{ST} were used only as a disturbance, or perturbation, of the current solution.

The objective of the perturbation is to diversify the search, generating a different solution, which becomes increasingly *distant* from the current location in the search space. To fulfill this mission, different levels of disturbance are set. For a given level p of perturbation, $p + 2$ movements are applied to the current solution, each movement randomly chosen among 6 of those described in Subsection 4.2. Each movement is chosen according to a certain probability, given by: 30% for N^{NT}, 20% for N^{ST}, 20% for N^L, 10% for N^{SS}, 10% for N^{PO} and 10% for N^{TO}. This probability varies due to the greater influence of changing the number of trips, exchanging the trips and exchanging the equipment (mining trucks and shovels) in the quality of the final solution. The local search applied on the disrupted solution is based on the procedure VND (line 6 of the Algorithm in Fig. 1). After *IterMax* iterations without improvement in a given level, this level increases by one unit. If the local search method finds a better solution, the perturbation returns to the lowest level, in this case, $p = 0$.

When the perturbation level p reaches a limit H, the Mathematical Programming Module (MPM) is triggered to solve smaller parts of the problem in optimality. Initially, 90% of the pits are locked and the mathematical programming module finds the optimal solution of this reduced solutions space. At each step it considers 10% more pits until p reaches $H + 9$ and the complete mathematical programming model is solved. The MPM stops in two situations: (i) after finding the optimal solution or (ii) after τ seconds.

4.6 Post Optimization: Path Relinking

The Path Relinking technique was proposed by Glover (1996) as an intensification strategy to explore links between elite solutions generated during Tabu Search or Scatter Search (Glover and Laguna, 1997) metaheuristics. This search consists in generating and exploring paths in the solution space, starting from one or more solutions (called *base solutions*) in direction to other elite solutions (called *guide solutions*).

In our case, the *base solution* is the final solution generated by the H-GVILS procedure (Section 4.5) and the *guide solution* is the initial solution generated by GRASP procedure (lines 1 and 2 of the Algorithm in Fig. 1), as presented in Section 4.4. Our Path Relinking approach is based on *attributes locking* and the considered attribute was the *shovel position* in the solution. At each step, the procedure selects a pit in the *guide solution* and moves the shovel k (the one assigned to the selected pit) to the same pit in the *base solution*. The trips made to that pit are also exchanged to keep compatability between shovels and trucks. Finally, a new local search is made in the *base solution* having both shovel k and its trips *locked* during the search procedure. The process is repeated until the *base solution* has all *guide solution* attributes, i.e., all the shovels and trips of the *base solution* are assigned to the same pits and trucks of the *guide solution*.

The local search is also based on the VND procedure of the H-GVILS algorithm (line 6 of the Algorithm in Fig. 1) and the same order of exploration of neighborhoods is used. However, in this case these neighborhoods have an *extra* ability of preserving the locked attributes in the solution during the search.

5. SCENARIOS DESCRIPTION

The scenarios utilized for the tests refer to an iron mining company located in the state of Minas Gerais, Brazil, and it was considered an 85% maximum utilization rate for the trucks. Instances are available at http://www.decom.ufop.br/prof/marcone/projects/mining.html.

Considered were four scenarios, giving origin to the instances OPM01, OPM02, OPM03 e OPM04. On the two first instances, there are 17 pits, 8 shovels and 30 mining trucks available. The trucks from 1 to 15 have 50-ton capacity and are compatible with shovels Car01 to Car04, while the other trucks have 80-ton capacity and are compatible with shovels Car05 to Car08. For the instances OPM03 and OPM04, there are 32 pits, 7 shovels and 30 mining trucks, with all trucks having a 50-ton capacity and compatibility with all the shovels. Ten control parameters (chemical and/or granulometric) are considered in all instances.

6. COMPUTATIONAL RESULTS AND ANALYSIS

The proposed algorithm, so-called H-GVILS-PR, is composed by the H-GVILS algorithm (Section 4.5) and the Post Optimization Module based on the Path Relinking procedure (Section 4.6). H-GVILS-PR was coded in C++ programming language using the g++ 4.0 compiler and the IDE Eclipse 3.1. The mathematical programming model was done in AMPL and solved by CPLEX solver, version 11. The hybrid Mathematical Programming Module used the GLPK solver, version 4.29. Both heuristic and exact models were tested in a PC Intel Core 2 Quad (Q9550), 2.83 GHz, with 8 GB of RAM, running Ubuntu 9.04 kernel 2.6.28-15.

The H-GVILS-PR algorithm was applied 30 times for each instance and it was interrupted after 2 and 15 minutes of processing. The CPLEX solver also was interrupted within this time limit. This strategy was adopted because the time for the decision maker is small and 2-15 minutes is a typical value for the maximum tolerance in a real case.

The following weights were adopted in the evaluation function: $\alpha^- = \alpha^+ = \beta^- = \beta^+ = 100$, $\lambda_j^- = \lambda_j^+ = 1 \;\; \forall j \in T$, $\omega_l = 1 \;\; \forall l \in V$, $Tx_l = 85\% \;\; \forall l \in V$. After a preliminary sequence of tests the three parameters of the proposed algorithm, *IterMax*, H and τ were set to: 60, 8 and 30, respectively.

Results were also compared to a pure metaheuristic version of the algorithm found in the literature, so-called GVILS (Coelho et al., 2008). This version of the algorithm is also based on the GRASP, VND and ILS metaheuristics, without the Post Optimization Module and the Mathematical Programming Module. The only parameter of the algorithm $IterMax$ was also set to 60 after a preliminary set of tests.

Tables 1 and 2 compare the results found by the CPLEX, GVILS and H-GVILS-PR algorithms. In these tables, “Instance” column means the analyzed instance; “Best Known” represents the best known value found in literature for the instance; “CPLEX” indicates the value (upper bound) found by CPLEX; “Best” and “Average” indicate the best and average values, respectively, in thirty executions of both GVILS and H-GVILS-PR algorithms; “Gap” for CPLEX, GVILS and H-GVILS-PR are calculated as equations (28), (29) and (30), and indicate the deviation between the CPLEX, GVILS and H-GVILS-PR solutions, respectively, in relation to the best known results. Column “Imp” measures the improvement of the GVILS (or H-GVILS-PR), in terms of average value, over the CPLEX value.

$$gap_i^{CPLEX} = \frac{f_i^{CPLEX} - f_i^*}{f_i^*} \tag{28}$$

$$gap_i^{GVILS} = \frac{\bar{f}_i^{GVILS} - f_i^*}{f_i^*} \tag{29}$$

$$gap_i^{H\text{-}GVILS\text{-}PR} = \frac{\bar{f}_i^{H\text{-}GVILS\text{-}PR} - f_i^*}{f_i^*} \tag{30}$$

In Eq. (29) and Eq. (30), for each instance i, $\bar{f}_i^{GVILS}$ is the average value found in thirty executions by the GVILS algorithm; and $\bar{f}_i^{H\text{-}GVILS\text{-}PR}$ is the average value for the thirty H-GVILS-PR executions; and f_i^* is the best known value. In Eq. (28), f_i^{CPLEX} represents the upper bound found in instance i within the time limit.

Table 1: Results comparison between CPLEX and GVILS

Instance	Time	Best	CPLEX		GVILS			Imp.
	(min)	Known	value	Gap (%)	Best	Average	Gap (%)	(%)
OPM01	2	227.12	227.90	0.34	227.12	227.44	0.14	0.00
OPM02	2	252.41	254.06	0.66	253.37	259.68	2.88	-0.38
OPM03	2	164029.15	164029.76	0.00	164044.41	164099.43	0.04	-0.01
OPM04	2	164054.04	164054.04	0.00	164106.14	164204.60	0.09	-0.03
OPM01	15	227.12	227.90	0.34	227.12	227.16	0.02	0.00
OPM02	15	252.41	254.06	0.66	252.41	253.32	0.36	0.00
OPM03	15	164029.15	164029.76	0.00	164040.91	164059.98	0.02	-0.01
OPM04	15	164054.04	164054.04	0.00	164100.21	164121.80	0.04	-0.03

Table 2: Results comparison between CPLEX and H-GVILS-PR

Instance	Time	Best	CPLEX		H-GVILS-PR			Imp.
	(min)	Known	value	Gap (%)	Best	Average	Gap (%)	(%)
OPM01	2	227.12	227.90	0.34	227.12	227.15	0.01	0.00
OPM02	2	252.41	254.06	0.66	225.28	225.39	-10.71	10.75
OPM03	2	164029.15	164029.76	0.00	164017.23	164017.28	-0.01	0.01
OPM04	2	164054.04	164054.04	0.00	164017.66	164018.22	-0.02	0.02
OPM01	15	227.12	227.90	0.34	227.12	227.14	0.01	0.00
OPM02	15	252.41	254.06	0.66	225.27	225.28	-10.75	10.75
OPM03	15	164029.15	164029.76	0.00	164017.18	164017.29	-0.01	0.01
OPM04	15	164054.04	164054.04	0.00	164017.68	164017.88	-0.02	0.02

As can be observed in Tab. 1, the H-GVILS-PR algorithm was able to generate better solutions than the best known solutions available in the literature for the instances OPM02, OPM03 and OPM04, with the best improvement of 10.75% (OPM02). Both heuristic algorithms GVILS and H-GVILS-PR had obtained gaps up to 2.88%. In addition, H-GVILS-PR was found capable to obtain better solutions when compared to CPLEX in all instances. Finally, H-GVILS-PR produced better results than the GVILS algorithm, with respect to the quality of the final solutions and with smaller gaps.

7. CONCLUSIONS

This work dealt with the operational planning of mines considering the dynamic allocation of trucks. Because of the complexity of this combinatorial problem, we proposed a hybrid heuristic algorithm, called H-GVILS-PR, which combines the heuristic procedures GRASP, Variable Neighborhood Descent and Iterated Local Search, with a Mathematical Programming Module and a Post Optimization Module based on Path Relinking to solve it.

Using instances from literature, the proposed heuristic algorithm was compared to the optimizer CPLEX 11.0 applied to a mathematical programming model, also developed in this work. It was found that the H-GVILS-PR algorithm is competitive with CPLEX 11.0 solver, since H-GVILS-PR is able to find good quality solutions quickly with low variability. The proposed algorithm is also compared to its pure heuristic version (GVILS) and shows better results than this one. Since staff decisions have to be made quickly, the results validate the use of the proposed algorithm, as a tool for decision support.

Acknowledgements

The authors acknowledge CNPq (process 474831/2007-8) and FAPEMIG (process CEX 2991-06.1/07) for supporting the development of this research.

REFERENCES

Alvarenga, G. B., 1997. Optimal dispatch of trucks in an iron mine using genetic algorithms with parallel processing (in portuguese). Master's thesis, Programa de Pós-Graduação em Engenharia Elétrica, Escola de Engenharia, UFMG, Belo Horizonte, Minas Gerais, Brazil.

Boland, N., Dumitrescu, I., Froyland, G., & Gleixner, A. M., 2009. Lp-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity. *Computers and Operations Research*, vol. 36, pp. 1064–1089.

Bresina, J. L., 1996. Heuristic-biased stochastic sampling. In *Proceedings of the 13th National Conference on Artificial Intelligence, AAAI Press*, pp. 271–278, Portland.

Chanda, E. K. C. & Dagdelen, K., 1995. Optimal blending of mine production using goal programming and interactive graphics systems. *International Journal of Surface Mining, Reclamation and Environment*, vol. 9, pp. 203–208.

Coelho, I. M., Ribas, S., & Souza, M. J. F., 2008. Um algoritmo baseado em grasp, iterated local search para a otimização do planejamento operacional de lavra. In *Proceedings of the XI Encontro de Modelagem Computacional*, pp. 8 p., Volta Redonda (RJ).

Costa, F. P., 2005. Aplicações de técnicas de otimização a problemas de planejamento operacional de lavras em mina a céu aberto. Master's thesis, Programa de Pó-Graduação em Engenharia Mineral, Escola de Minas, UFOP, Ouro Preto.

Ezawa, L. & Silva, K. S., 1995. Dynamic allocation of trucks aiming quality (in portuguese). In *Proceedings of the VI Congresso Brasileiro de Mineração*, pp. 15–19, Salvador, Bahia, Brazil.

Feo, T. A. & Resende, M. G. C., 1995. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, vol. 6, pp. 109–133.

Glover, F., 1996. Tabu Search and adaptive memory programming - advances, applications and challenges. In Barr, R., Helgason, R., & Kennington, J., eds, *Interfaces in Computer Sciences and Operations Research*, pp. 1–75. Kluwer Academic Publishers.

Glover, F. & Laguna, M., 1997. *Tabu Search*. Kluwer Academic Publishers, Boston.

Godoy, M. & Dimitrakopoulos, R., 2004. Managing risk and waste mining in long-term production scheduling of open-pit mines. *SME Transactions*, vol. 316, pp. 43–50.

Guimaraes, I. F., Pantuza, G., & Souza, M. J. F., 2007. A computational simulation model to validate results by dynamic allocation of trucks in open-pit mines (in portuguese). In *Proceedings of the XIV Simpósio de Engenharia de Produção (SIMPEP)*, pp. 11, Bauru, São Paulo, Brazil.

Lourenço, H. R., Martin, O. C., & Stützle, T., 2003. Iterated local search. In Glover, F. & Kochenberger, G., eds, *Handbook of Metaheuristics*. Kluwer Academic Publishers, Boston.

Merschmann, L. H. C., 2002. Development of an optimization and simulation system for the analysis of production scenarios in open-pit mines (in portuguese). Master's thesis, Programa de Engenharia de Produção/COPPE, UFRJ, Rio de Janeiro, Brazil.

Mladenovic, N. & Hansen, P., 1997. A variable neighborhood search. *Computers and Operations Research*, vol. 24, pp. 1097–1100.

Papadimitriou, C. H. & Steiglitz, K., 1998. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, Inc., New York.

Ribas, S., Coelho, I. M., Souza, M. J. F., & Menotti, D., 2009. Parallel iterated local search aplicado ao planejamento operacional de lavra. In *Proceedings of the XLI SBPO*, pp. 12 p., Porto Seguro (BA).

Romero, C., 2004. A general structure of achievement function for a goal programming model. *European Journal of Operational Research*, vol. 153, pp. 675–686.

Sgurev, V., Vassilev, V., Dokev, N., Genova, K., Drangajov, S., Korsemov, C., & Atanassov, A., 1989. Trasy - an automated system for real-time control of the industrial truck haulage in open-pit mines. *European Journal of Operational Research*, vol. 43, pp. 44–52.

White, J. W., Arnold, M. J., & Clevenger, J. G., 1982. Automated open-pit truck dispatching at Tyrone. *Engineering and Mining Journal*, vol. 183, n. 6, pp. 76–84.

White, J. W. & Olson, J. P., 1986. Computer-based dispatching in mines with concurrent operating objectives. *Mining Engineering*, vol. 38, n. 11, pp. 1045–1054.