GLOBAL LONG-TERM OPTIMISATION OF VERY LARGE MINING COMPLEXES

Jeff Whittle BSc, FAusIMM, MACS – corresponding author
Technical Director
Whittle Consulting
Suite 13, 333 Canterbury
Victoria 3126
Australia
Tel: +613 9857 6679
Email: jeff@whittleconsulting.com.au

Gerald Whittle BEcon, GradDipFin, FCPA
Managing Director
Whittle Consulting
Suite 13, 333 Canterbury
Victoria 3126
Australia
Tel: +613 9836 9644
Email: gerald@whittleconsulting.com.au

ABSTRACT

Finding an optimal, or near optimal, long-term schedule for a very large mining complex can be computationally difficult, not only because of the size of the problem but because of its complexity. This is because very large complexes offer many more mining and processing options. Indeed, the analysis of such a complex can be a challenge in itself.

Here we consider a mining complex to be very large if it includes, say, a hundred pits, half-a-dozen different processing options with blending at the processing input and/or the product stage and, of course, stockpiles at the processing input and/or the product stage.

A program has been developed to handle just such large problems. It works by repeatedly creating a random initial feasible schedule and using linear programming iteratively to find the nearest local maximum to that initial schedule. The distribution of the Net Present Values of these local maxima then gives guidance as to when enough random starting points have been tried.

This paper defines the problem in detail and then concentrates on the search for the local maxima from random start points. The paper then gives examples of the program's application to real cases.

INTRODUCTION

Finding an optimal, or near optimal, long-term schedule for a very large mining complex can be computationally difficult not only because of the size of the problem but because of its complexity. This is because very large complexes offer many more mining and processing options. Indeed, the analysis of such a complex can be a challenge in itself.

Here we consider a mining complex to be very large if it includes, say, a hundred pits, half-a-dozen different processing options with blending at the processing input and/or the product stage and, of course, stockpiles at the processing input and/or the product stage.

Note that, without stockpiles, such a problem can be defined as a mixed-integer program and, with smaller problems, it can be optimised with standard software. However, solution times and memory requirements increase exponentially with problem size and very large problems defy solution by that method.

A program, ProberB, has been developed to handle just such large problems. It works by repeatedly creating a random initial feasible schedule and using linear programming iteratively to find the nearest local maximum to that initial schedule. The distribution of the Net Present Values of these local maxima then gives guidance as to when enough random starting points have been tried.

This paper defines the problem in detail and then concentrates on the search for the local maxima from random start points. The paper then gives examples of the program's application to real cases.

METHODOLOGY

Modelling the Resource

In our terminology, the hierarchy of objects in a resource is as follows:

- **Sequence**
  - **Panel**
    - **Parcel**

where each can contain multiples of the one below it. See Figure 1.

A *sequence* consists of a series of panels that can only be mined in order from first to last and where it is assumed that the mining of one panel must be completed before the next panel is started. A sequence can be a phase of an open pit where the panels are benches, or a series stopes in an underground mine that must be accessed in order. Each sequence has a name.

A *panel* consists of a number of parcels, but parcels have no specified location within a panel. If only part of a panel is mined during a particular time period, it is assumed that the same fraction of each parcel is mined. This is a mathematical fiction, but it is quite adequate for long-term planning. In the short-term the panel would probably not be mined in this way but only very rarely would this impact the long-term plan.

A *parcel* has a mine material type, a tonnage, a cost of mining per tonne, and one or more attribute grades. It is regarded as homogeneous and has no defined shape.

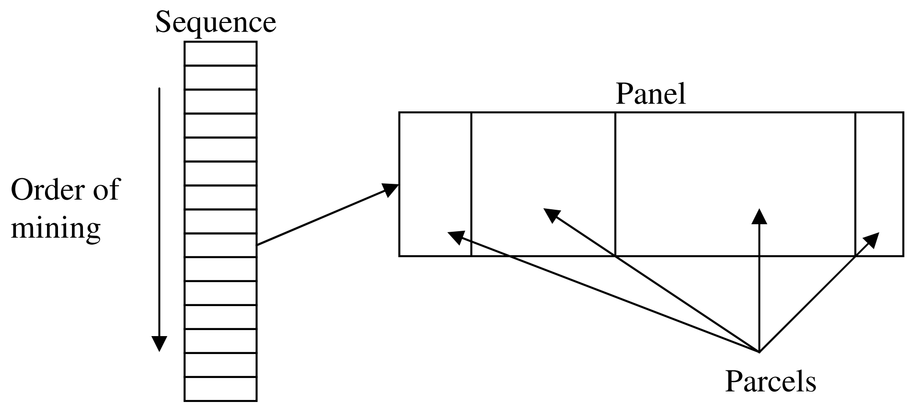

Figure 1: Sequence/Panel/Parcel relationship

Modelling Production

The hierarchy continues from each parcel as follows:

Parcel
Processing Path
Blend Feed

where each can contain multiples of the one below it. See Figure 2.

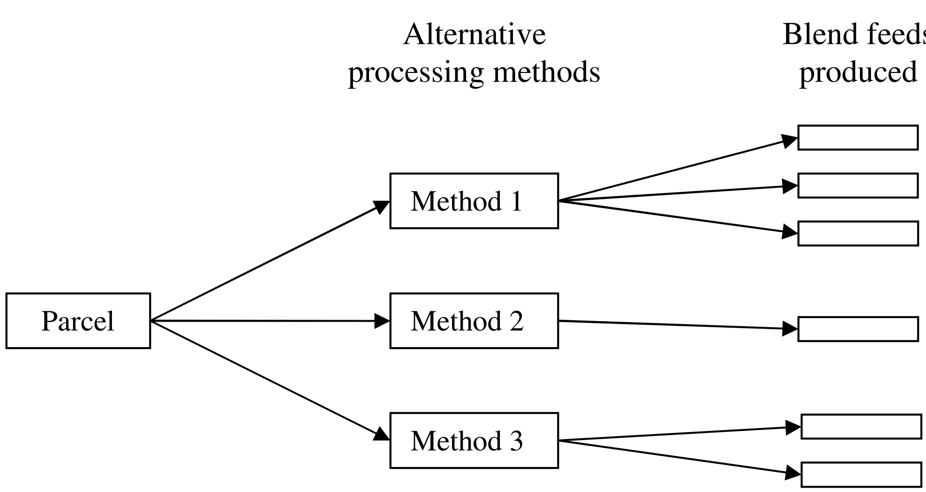

Figure 2: Parcel/Processing/Blend Feed relationships

A parcel can be processed along zero or more alternative *processing paths*. If there are zero processing paths then the parcel is waste. Each processing path has a processing method name, which identifies a particular plant, a processing cost per tonne and produces one or more *blend feeds*.

A blend feed is like a parcel in that it has a material type, a tonnage, a blending cost and one or more attribute grades.

Blend feeds, as their name implies, form the material that is blended into product. Blend feed material types might be, say, “Lump” or “Fines”, or they might have other characteristics that influence which product they are eligible for.

The Input File

The whole of the above hierarchy, from sequences to blend feeds, is included in the input file, which can be quite voluminous.

For example, if a particular parcel can be processed by two alternative methods, then two paths will be detailed for that parcel giving the method names and the resultant blend feeds in each case. Any complications, such as non-linearity, can be handled during the input preparation. The choice of which processing method to use for each parcel is made during the optimisation.

In addition to the resource information, the input file may contain one or more of any of the following:

Mining Constraints

Mining throughput limits: These are specified by a tonnage limit per year and the intersection of a list of mine material types and a list of sequences, each with a constant weight and, optionally, a weight depending on an attribute value. This last allows a mining limit to control the quantity of a contained attribute mined.

Minimum and maximum leads between sequences: For example, if two sequences A and B were physically adjacent, we might limit the mining of sequence A to be between one and four panels (benches) ahead of the mining of sequence B.

Maximum panels per year: For each sequence, it is possible to specify the maximum number of panels that can be mined in a year.

Sequence start conditions: Mining of a sequence may be constrained to start only after particular panels have been mined in one or more other sequences.

Processing and Production Constraints

Treatment of mine material types: Mined material can be processed, stockpiled or discarded, or treatment can be limited to any subset of these.

Processing input grade limits: Minimum and maximum average attribute grades input to each processing method in a period can be specified.

Processing throughput limits: These are specified by a tonnage limit per year and the intersection of a list of mine material types and a list of processing methods, each with a constant weight and, optionally, a weight depending on an attribute value. This last allows processing limits to control the quantity of a contained attribute processed.

Product grade limits: Minimum and maximum average attribute grades for each product in a period can be specified.

Treatment of blend feed material types: Blend feed material produced by processing can be blended into product, stockpiled or discarded, or treatment can be limited to any subset of these.

Product production limits: These are specified by a tonnage limit per year and the intersection of a list of blend feed material types and a list of products, each with a constant weight and, optionally, a weight depending on an attribute value. This last allows production limits to control the quantity of a contained attribute produced.

Other Details

Many values, including the period length, can vary with period.

The mining, processing and production capacities can themselves be optimized rather than being fixed.

The total tonnage of stockpiles of different types can be constrained.

The program, ProberB, is a processing engine. It has no input or output screens. It is run from a command window. It reads a text file and writes a text file, with minimal progress reporting on the screen.

Elaborate screens are used in preparing the input file from user data and in reporting the results, but these are not under discussion here.

A great deal of both business and computing skills are involved in preparing the input for a particular customer and in interpreting the output. This can involve days of consultation.

The Problem

Consider a case with 100 sequences, each containing 25 panels, and a project life of 20 periods. Assume that the sequences are phases of open pits and the panels are benches in these phases.

The aim is to find the mining, processing and production schedule that maximizes the Net Present Value (NPV) of the whole project over the life of the mine.

The set of mining depths in each sequence at the end of each period specifies the mining schedule.

If we fix these depths and ignore stockpiling for now, the optimal processing and production schedule can be obtained by standard linear programming. This is because we have fixed which parcels and parts of parcels are to be mined in each period, and optimisation is then merely a question of processing and blending from a number of sources, although this may involve a large number of variables. Consequently, in the larger sense, we can regard the mining depths in each sequence at the end of each period as the controlling variables in the problem.

With 100 sequences and 20 periods, we have 2,000 such variables.

A point in a space with 100 dimensions, which each range from zero to 25, can define the depth in 100 sequences at a point in time. If we have 20 periods, the whole mining schedule can be represented by a set of 20 such points. We can thus think of the whole mining schedule as being represented by one point in a space of 2000 dimensions. If the schedule meets the mining constraints, that is it is feasible, there is an NPV for that point. Thus we have a multi-dimensional volume which contains a smaller feasible volume within which each point has a corresponding NPV.

Figure 3 illustrates how a point in two dimensions can define the depth of mining in two sequences at the end of a particular period. This concept can be extended to cover 100 sequences and 20 periods by adding 1998 extra dimensions, in which case a point would define the whole mining schedule. The best processing and production schedule for that mining schedule can be obtained by standard linear programming because the material available from mining in each period is defined.

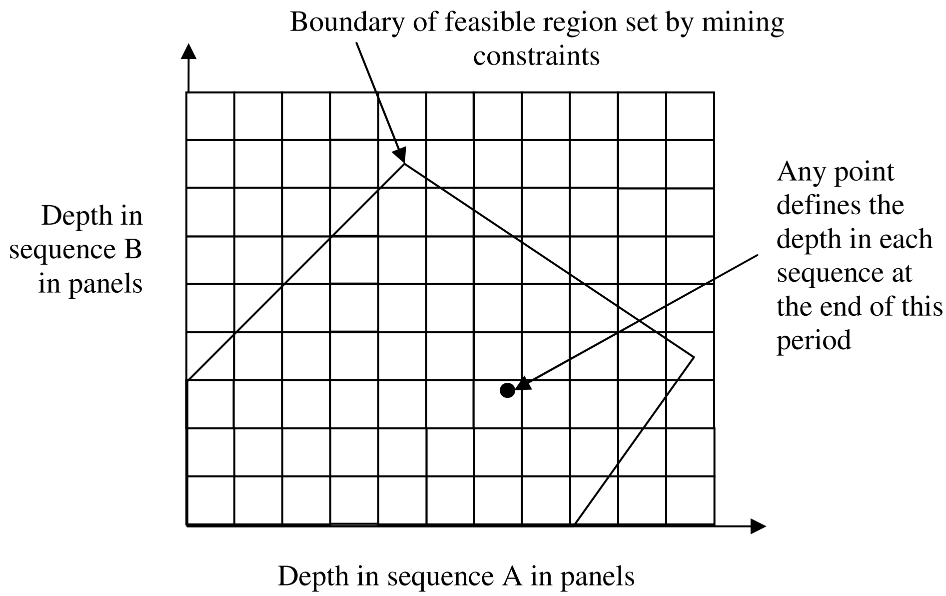

Figure 3: Defining the depth in two sequences with a point

However, the NPV is not a linear function of these variables. This is because in different squares we are mining different panels and the properties of each panel can be different. If we regard the NPV as a function in 2000 variables, it is continuous but has many local maxima..

Nevertheless, in theory, the whole optimisation could be done using mixed-integer linear programming. Unfortunately a mixed integer program to solve this problem would require at least 50,000 binary variables in addition to the 2000 mining variables and many more processing and production variables. It would be a serious challenge for a mixed integer programming package.

This would be bad enough, but it gets worse. With stockpiling, because we mix materials of different grades and then have to withdraw the same proportion of each, we require equality constraints on products of the variables. We then have to deal with not only a mixed-integer problem but a non-convex mixed-integer quadratic problem, which currently defies solution.

This is the real problem.

The Approach – without Stockpiles

Note that, within each part of the multi-dimensional volume for which the panels being mined do not change (eg a small square in figure 3), the NPV is linear because, if only a fraction of a panel is mined in a particular period, we assume that the same fraction of every parcel in the panel is mined.

We can think of the whole multi-dimensional volume as a matrix of small hypercubes within each of which the panels being mined at the end of each period are fixed, and thus the problem is linear.

ProberB finds a random initial mining schedule. This only has to obey the mining constraints. This identifies a hypercube within the matrix described above that is at least partially within the feasible volume. Linear programming, which controls the mining as well as the processing and production, is then used to find the point of highest NPV in that hypercube.

The program next examines this point. In particular it looks for cases where the point is up against a side or a corner of the hypercube. If it is, the program moves to the adjacent hypercube, and again uses linear programming to find the point of highest NPV.

ProberB works by repeatedly applying this approach until there is no further improvement in NPV. We have then reached the "nearest" local maximum to the initial feasible solution. Typically there will be 10 to 50 such steps required. Figure 4 illustrates this process in two dimensions. The moves between hypercubes are shown by the arrows.

A move to an adjacent hypercube cannot lead to a linear program (LP) with a result that is lower, because moving from the end of one panel in a sequence to the beginning of the next does not change what has been mined. If the move produces no improvement we have reached a local NPV maximum.

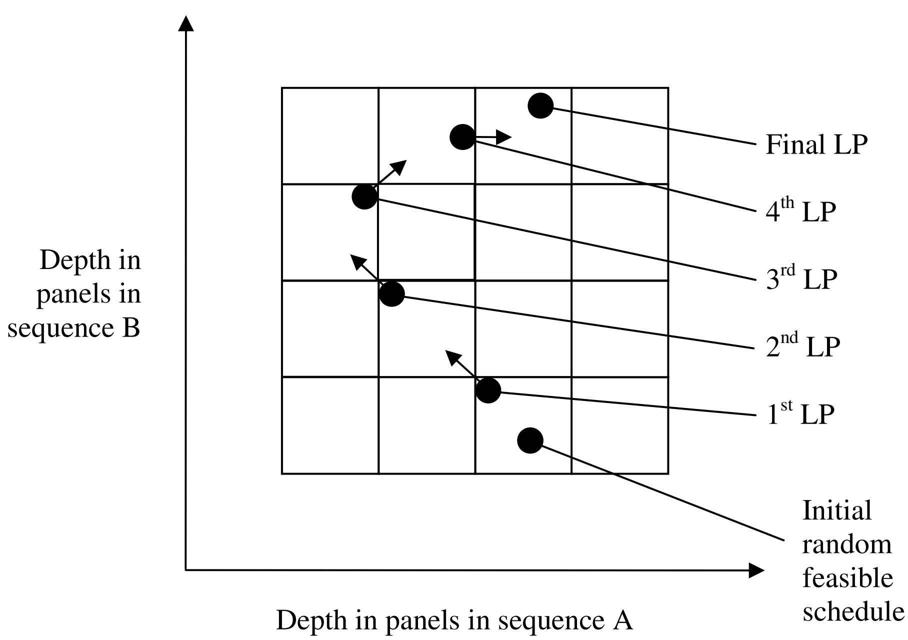

Figure 4: Movement between hypercubes of just two dimensions

The Approach – with Stockpiles

As explained earlier, the use of stockpiles makes the problem non-convex quadratic because of the requirement for equality constraints on products of the LP variables.

Our approach is to iteratively estimate the grades of the stockpiles, solve an LP with the grades fixed, and then use the results to get a better estimate of the grades. With real data this generally converges very quickly. With test data, where only a handful of parcels with widely different grades are involved, it can prove more difficult, so that adjustment heuristics have to be applied and some residual oscillation may remain.

Note that this approach can, in theory, distort the function being optimised because the grades would change as LP variable values changed and this could affect the LP result. However, in real cases, we believe this effect to be very small because many parcels contribute to a stockpile.

Technicalities

The Program consists of 30,000 lines of Intel Fortran® code and uses calls to the Ilog CPLEX® callable library for the linear programming.

LPs with a million variables and half a million constraints are not unusual.

Some very large models can take 24 hours to find one local maximum, so a bank of computers is used to process one case. Note that different random samples can be produced on different machines by using different random seeds.

Apart from the over-all addressing limit, there are no limits on the numbers of resource items or constraints since all required memory is allocated at run time.

Termination

ProberB finds the nearest local maxima to each of a large number of different random feasible starting schedules and we plot a distribution of the NPVs of the local maxima found.

Figures 5a and 5b show ranked results for a particular case after 50 samples and after 500 samples of the same model.

In this case, it seems reasonable to conclude that the best possible NPV is very likely to be close to $2,100,000,000. We usually stop when the top ten results lie within a small fraction of a percent of the best value.

The shapes of the curves are different for different models.

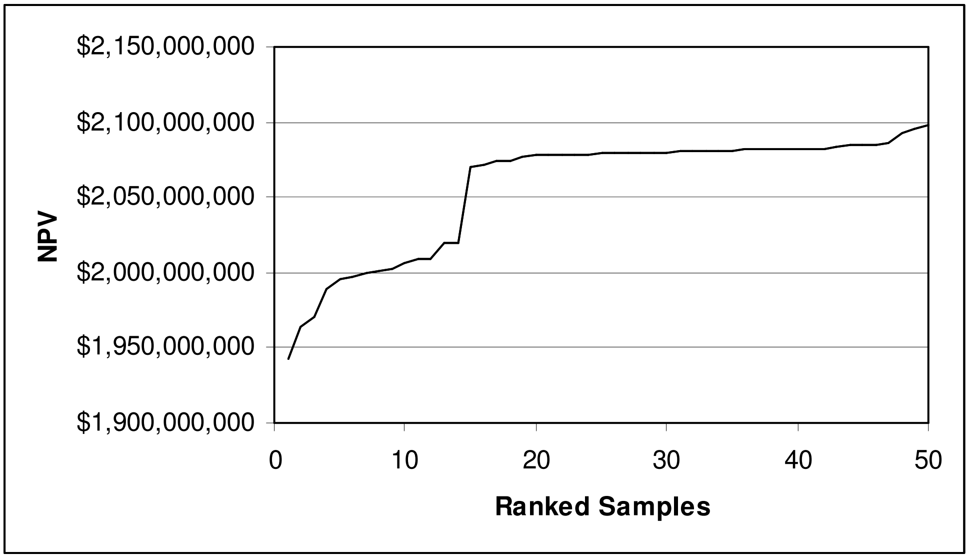

Figure 5a: 50 ranked random local maxima

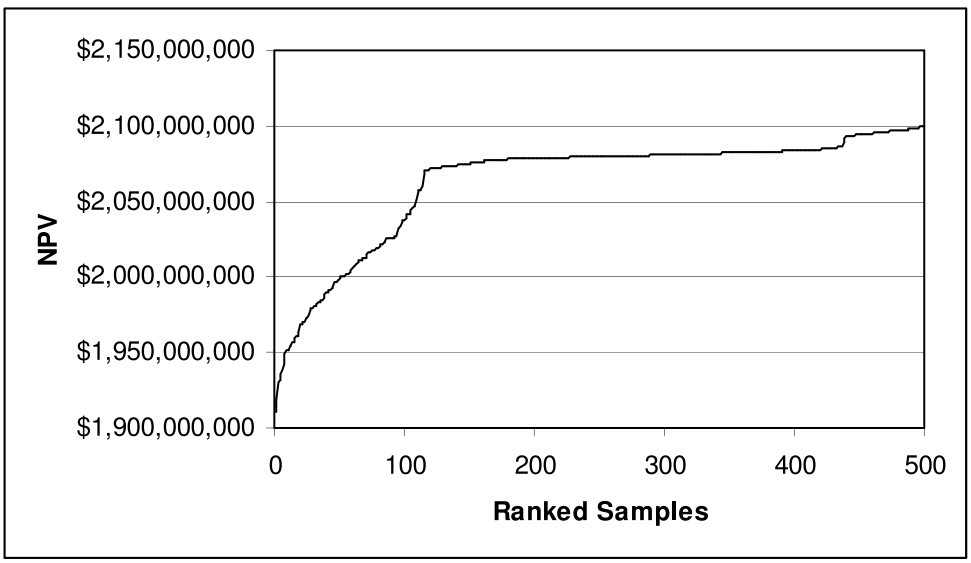

Figure 5b: 500 ranked random local maxima

ACTUAL CASES

Case Study 1.

A base metal resource company operates a number of pits and underground mines.

The schedule for one underground system was considered inflexible and so was represented as a single sequence based on a previously determined annual mining plan. Another underground system was modelled as eleven separate sequences with interdependency rules applied. A set of independent mines providing feed were included as a single sequence according to the contract production profile. One major pit with 5 pushback phases containing fifteen different material types was included as five sequences of up to 20 benches linked by minimum lead rules. A second pit system was included as two separate pits, each with three phases, making six related sequences of up to 28 benches, containing 5 material types. For the pits there was no predetermined cut-off grade, and stockpiling was possible.

A total of five concentrators were available, with some mines being able to contribute to more than one. Two of the concentrators produced both a high grade concentrate and a secondary recovery of low grade concentrate. Three of the concentrators had alternative configurations which affected the recovery versus concentrate grade, as well as the operating cost and throughput. The availability of the concentrators varied over time due to construction plans and future modifications to improve recoveries on certain material types.

Downstream options included a smelter, with restrictive feed characteristic requirements, a leach process with more flexible input requirements but lower recovery, and a refinery that could inefficiently process concentrates or more efficiently, but with less value added, process the intermediate products from the smelter or leach. The capacity and performance of the downstream plant changed during the forecast time frame due to planned upgrade and expansion projects.

Although the business was focussed on one metal product, five other trace metal by-products contributed to revenues, the yield on these varying greatly according to the choice of downstream option. It was possible to sell concentrates, smelter and leach intermediate products, and refined metal. The variation between metal paid for these alternatives was significant. Metal prices were forecast to reduce over time from current high levels to more modest long term expectations.

The optimiser controlled 37 mining sequences, 37 material types with 16 grade attributes, through 18 processing paths, to produce potentially 16 different concentrates with 52 attributes (representing the detailed outcomes of the various downstream options) to produce potentially 14 products.

The optimiser controlled mining rate and location, cut-off grade and stockpiling, choice and calibration of concentrator, blending of concentrates into downstream plant, production volume mix and specification, over the 30 year planning horizon within the business constraints specified.

The optimiser typically took about twenty steps between hypercubes to find an NPV local maximum from an initial random feasible schedule. Each step required the solution of around four LPs with an average of 500,000 variables, 100,000 constraints and 200,000 non-zeroes. Each sample took about two hours in a 3Gh Pentium 4. Approximately sixty per cent of this time was spent in solving the LPs.

After between 100 and 200 samples, the top ten results had NPVs within 0.1% of each other, which provided sufficient confidence that the global maximum NPV had been identified. This model, which was optimized a number of times, was generally run on four to eight PC’s simultaneously to minimise elapsed time.

The Global Optimisation model was used as a strategic planning tool to question conventional thinking on business configuration. The results provided significant insights into mine and plant expansion capacities, cut-off grades, appropriate calibration of concentrators for different operating and market conditions, use of downstream plant, and the conservation of certain ore bodies that were critical to the blend in certain plant.

Case Study 2.

A base metal resource company has a significant shallow but widespread medium grade resource made up of 18 deposits. The pit optimisation analysis resulted in 130 pits spread over 70kms, with up to 30 benches, and 5 material types. No predetermined cut-off grade was set. Stockpiling by material type and grade characteristics was allowed in 9 different locations, near the mines to defer haulage.

Milling, screen recovery and associated beneficiation was a function of material type. Screen rejects were stockpiled for potential later use.

A detailed model of the pressure leach plant captured cost, throughput and recovery details. The material type blend determined leach slurry viscosity/density and therefore throughput. Acid consumption was a function of five grade attributes of the leach feed, the chosen free acid setting and the density. Certain corrosive elements were controlled by blending, and consideration of changing characteristics of ground water over time. Plant throughput was controlled in m3 of slurry, and respected acid cost and availability. Recovery in the leach was formulated to consider a constant tailings loss, which was a function of the density, free acid setting and residence time.

A second detailed model of an atmospheric leach process that would run in combination with the above, sharing infrastructure at some points in the process flow, was added. This model had similar complexity but very different sensitivities to feed characteristics.

An approach of "acid balance" was applied, where the limited acid was effectively "mined" by the optimiser and "blended" at the point in the process flow of ore curing, pressure leach and atmospheric leach. Blending constraints of acid balance equalling zero were applied at these three points.

Metal prices were expected to change over time for the two metals produced, as was the cost of sulphur which is the raw material for acid production and therefore the most significant variable cost.

The optimizer typically took about ten steps between hypercubes to find an NPV local maximum from an initial random feasible schedule. Each step required the solution of around four LPs with an average of 620,000 variables, 480,000 constraints and 2,000,000 non-zeroes. Each sample took about four hours in a 3Gh Pentium 4. Almost all this time was spent in solving the LPs.

After between thirty and sixty samples, the top ten results had NPVs within 0.1% of each other, which provided sufficient confidence that the global maximum NPV had been identified. This model, which was optimized a number of times, was generally run on four to eight PC's simultaneously to minimise elapsed time.

The number of samples required is less than the previous case. Although the model dimensions are greater, the stratigraphic nature of the ore body means that material changes more regularly with pit depth resulting in fewer local maxima than in the deep and faulted case 1 above.

The optimiser controlled 179 sequences, 93 material types with 16 grade attributes, through 6 processing paths, to produce potentially 24 blend feed types with 32 attributes to produce potentially 4 products.

The optimiser therefore simultaneously controlled two main blends with different sensitivities, in the context of significant mining flexibility, grade control and stockpiling options over 30 years. The two leach streams had to compete for material and acid, the result reflecting their different performance characteristics.

The Global Optimisation model was used to evaluate the business case for several expansion and reconfiguration projects, and to determine the appropriate mining, stockpiling and blending strategies for the various operational and market scenarios. It also served the purpose of a communication medium between geology, mining, processing, marketing and finance for alignment of objectives and the evaluation of improvement proposals with company-wide repercussions.

CONCLUSION

Global Optimisation has been applied to ten cases over the last four years, most on an ongoing basis following an initial optimisation project. These involved very different combinations of modelling, optimisation and business challenges. A high degree of collaboration between the technical specialists involved was required.

By solving the extreme mathematical challenges involved in optimisation of complex mining and mineral processing operations, the optimiser acts as a type of artificial intelligence to support business decision making and strategy development with outcomes typically measured in tens or hundreds of millions of dollars.