University of Alberta

Intelligent 3D Interactive Open Pit Mine Planning and Optimization

By
Hooman Askari-Nasab ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

Mining Engineering

Department of Civil & Environmental Engineering

Edmonton, Alberta

Fall 2006

ABSTRACT

Numerous attempts have been made to address the computationally intractable open pit optimization and long-term planning problem. Heuristic methods, economic parametric analysis, operations research, and genetic algorithms have been used to formulate periodic open pit long-term schedules. In practice all these techniques have limitations in dealing with large industrial problems; stochastic processes governing mining technical and economic variables; temporal nature of exploitation; and mine to mill integration. In complex mining operations, small deviations from the optimal strategic plan could result in the loss of millions of dollars.

The primary research goal is to develop, analyze, and implement a 3-D intelligent open pit optimal production simulator (IOPS) based on reinforcement learning (RL) to maximize the net present value (NPV) of the venture. Also a continuous open pit simulator (COPS) based on the modified open pit geometrical model and a system of differential equations have been developed to capture the continuous-time open pit dynamics for tactical purposes. The Java Reinforcement Learning Library was chosen as the core of the IOPS application implementation. Java programming language and MATLAB were selected as the platform for programming and graphical user interface (GUI) implementation.

To verify and validate the research models, a case study on an iron ore deposit with 114,000 blocks was carried out. The final pit limits were determined using Lerch's Grossman's algorithm with the Whittle software. The optimized final pit limits show the total amount of 399 million tonnes of material consisting of 220 million tonnes of ore and 179 million tonnes of waste. The practical annual schedule generated by the industry

standard tool -- Milawa algorithm used in Whittle software -- yielded an NPV of $430 million over a 21-year mine life at a discount rate of 10% per annum. The practical learned scenario of 3000 simulation iterations using IOPS yielded an NPV of $438 million over the same time span. Experiments were also performed to compare the annual stripping ratio, average grade, annual waste, and the ore and concentrate production. The outcome of the research demonstrated a strong promise towards improving the expected net present value of mining investments. The algorithms developed can be the basis of the next generation of mine design software packages.

AKNOWLEDGMENT

I wish to express my sincere appreciation to my advisor, Dr. Samuel Frimpong, without whose guidance, encouragement, and support this work would not have been completed. I am also grateful to my co-supervisor Dr. Jozef Szymanski for his constructive and immediate feedback when I needed it.

I also appreciate the time and concern of my advisory committee members – Dr. Hilary Inyang, Dr. Tayfun Babadagli, Dr. Ergun Kuru, and Dr. Marek Reformat. Thank you for agreeing to contribute to the process; your comments and corrections are appreciated.

I am grateful for the insight and encouragement of my colleagues, Mr. George Dogbe, Ms. Nuray Erdogan Demirel and Mr. Kwame Awuah-Offei. My gratitude also goes to all my friends in Edmonton for their friendship which made things easier.

I am forever indebted to my parents, Nayer and Manouchehr for their support and encouragement throughout my life. Finally and most importantly I reserve the rest of my acknowledgements for my sweet Mina, whose love, friendship, and support have been a real blessing.

Hooman Askari-Nasab

TABLE OF CONTENTS

ABSTRACT

ACKNOWLEDGMENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

CHAPTER 1 ....INTRODUCTION

1.1 Background...1

1.2 Statement of the Problem 2

1.3 Objectives of the Study 7

1.4 Scope and Limitations of the Study...7

1.5 Research Methodology..8

1.6 Contributions and Industrial Significance of the Study 10

1.7 Organization of Thesis 11

CHAPTER 2 ....LITERATURE SURVEY

2.1 Open Pit Optimization Algorithms..13

2.1.1 The Manual Method 18

2.1.2 The Moving Cone Algorithm 18

2.1.3 Lerchs and Grossmann Algorithm..21

2.1.3.1 Lerchs and Grossmann 2-D Algorithm 21

2.1.3.2 Lerchs and Grossmann 3-D Algorithm 22

2.1.4 Neural Network (MCS/MFNN) Algorithm for Open Pit Optimization............25

2.2 Production Scheduling Algorithms 27

2.3 Haul Road Design and Incorporation..28

2.4 Rationale for PhD Research 30

2.5 Summary and Remarks..31

CHAPTER 3.. THEORETICAL FRAMEWORK

3.1 Foreword..33

3.2 Environment: Geostatistical Block Model 35

3.3 Simulation: Open Pit Production Simulator Components (OPPS)......................37

3.3.1 Geometrical Open Pit Models (GOPM) 37

3.3.2 Dynamics of Open Pit Expansion..41

3.3.2.1 Discrete Open Pit Expansion Model 42

3.3.2.2 Continuous Open Pit Expansion Model 45

3.3.3 Economic Pit Expansion Model (EPEM)...50

3.4 Agent: Intelligent Open Pit Agent (IOPA) Structure 53

3.5 Reinforcement Learning: IOPS Theoretical Framework......................................56

3.5.1 Elements of IOPS as an RL Problem 56

3.5.1.1 Rewards and Return..58

3.5.1.2 IOPS Action Selection Rule 59

3.5.1.3 Markov Property Assumption 59

3.5.1.4 IOPS Value Functions 60

3.5.1.5 Optimal Value Functions 62

3.5.2 Review on Solution Methods to the RL Problem 65

3.5.3 Temporal-Difference Learning 67

3.5.3.1 Q-Learning: Off-Policy TD Control Algorithm Core of IOPS 68

3.6 Summary and Remarks 69

CHAPTER 4 .... ALGORITHM DEVELOPMENT

4.1 Foreword 70

4.2 Algorithm Development 71

4.2.1 Intelligent Open Pit Optimal Production Simulator (IOPS) 72

4.2.2 Discrete Open Pit Simulator (DOPS) 76

4.2.3 Continuous Open Pit Simulator (COPS) 78

4.3 IOPS Design and Implementation 81

4.3.1 IOPS Classes 82

4.3.1.1 The Agent Class 84

4.3.1.2 The Environment Class 85

4.3.1.3 The Simulation Class 86

4.4 IOPS Graphical User Interface 86

4.5 Summary and Remarks 89

CHAPTER 5 .... EXPERIMENTAL DESIGN

5.1 Foreword 90

5.2 Verification and Validation 91

5.3 Experimental Design Framework for IOPS architecture 92

5.4 Case Study: Gol-E-Gohar Iron Ore Mine 94

5.4.1 Iron Ore Optimized Final Pit Limits Design 98

5.4.2 Whittle Long-Term Schedule 104

5.4.3 IOPS Long-Term Schedule 108

5.4.4 COPS Continuous Time Schedule 116

5.4.5 Comparison of IOPS vs. Whittle Schedule 125

5.4.6 Comparison of IOPS vs. COPS Schedule 128

5.5 Summary and Remarks 132

CHAPTER 6 .... CONCLUSIONS

6.1 Summary of Research 135

6.2 Conclusions 139

6.3 Contributions of PhD Research 141

6.4 Recommendations for Further Research 142

APPENDIX A 144

LIST OF REFERENCES 154

LIST OF TABLES

Table 2.1 Labeling guide for arcs 24

Table 3.1 Variables defining frustums in each region 40

Table 5.1 Summary of the ore and waste in the geological block model 98

Table 5.2 aE the sum of squares due to error 117

Table 5.3 aW the sum of squares due to error 118

Table 5.4 bN the sum of squares due to error 119

Table 5.5 bS the sum of squares due to error 120

Table 5.6 h the sum of squares due to error 121

Table 5.7 Block model information 132

Table 5.8 Final pit limits specifications, LG and IOPS 133

Table 5.9 NPV comparison 134

Table 6.1 Summary of the numerical application 139

LIST OF FIGURES

Figure 1.1 Sample block model (Thomas, 1996) 3
Figure 1.2 Flow diagram of research methodology 9
Figure 2.1 Moving cone algorithm flowchart 20
Figure 2.2 Representation of the 1-9 block constraints (Hustrulid and Kuchta, 1998) 23
Figure 2.3 Labeling guide for arcs 24
Figure 2.4 Illustration of the steps in LG algorithm (Hochbaum and Chen, 1996) 25
Figure 2.5 Lerchs and Grossmann algorithm flow chart 26
Figure 3.1 The agent-environment interaction in reinforcement learning 33
Figure 3.2 Intelligent open pit optimal production simulator (IOPS) frame work 34
Figure 3.3 Sequence of actions in geostatistical modeling 35
Figure 3.4 Frustum geometry of open pit 38
Figure 3.5 Concentric ellipses 40
Figure 3.6 Modified geometrical frustum model 41
Figure 3.7 Open pit geometry, each bench is modeled by a modified frustum 42
Figure 3.8 Discrete open pit expansion model 44
Figure 3.9 Economic pit expansion model 51
Figure 3.10 Economic pit expansion model 51
Figure 3.11 Economic pit expansion flowchart 53
Figure 3.12 IOPS model as an RL architecture 55
Figure 3.13 Schematic of OPPS simulation at a discrete time step t 61
Figure 3.14 Backup diagrams for Figure 3.13 63
Figure 3.15 Tabular TD(0) for estimating V^{π} 67
Figure 3.16 Q-Learning an off-policy TD control algorithm 68
Figure 4.1 IOPS general flow chart 71
Figure 4.2 Q-learning algorithm flow diagram 73
Figure 4.3 Epsilon greedy algorithm flow chart 75
Figure 4.4 DOPS flowchart 77
Figure 4.5 COPS flowchart 80
Figure 4.6 IOPS class UML diagram 82
Figure 4.7 IOPS block specification tab 86

Figure 4.8 open pit specification tab 87

Figure 4.9 Q-Learning specification tab 88

Figure 5.1 Design, optimization, and planning experiments 93

Figure 5.2 Iron ore deposit borehole locations 94

Figure 5.3 Idealize schematic cross section of the deposit looking toward west 95

Figure 5.4 First topography of the area of study 96

Figure 5.5 3D view of the first topography 96

Figure 5.6 3D view of the deposit 97

Figure 5.7 bench by bench report of the geological block model 98

Figure 5.8 Whittle interface 100

Figure 5.9 Push-backs selection in pit by pit analysis 101

Figure 5.10 Incremental pit shells NPV 103

Figure 5.11 Pit shell 68, the optimized final pit limit 104

Figure 5.12 Best case and worst case schedule 105

Figure 5.13 Milawa schedule 106

Figure 5.14 Cumulative ore vs. waste by production period 107

Figure 5.15 Sensitivity analysis of discounted pit value to selling price and costs 108

Figure 5.16 Final pit limits captured by the modified geometrical model 109

Figure 5.17 Parameter initialization 110

Figure 5.18 Annual changes in the dimensions of the open pit 111

Figure 5.19 Push-backs years 1 to 6 112

Figure 5.20 Push-backs years 7 to 12 113

Figure 5.21 Push-backs years 13 to 18 114

Figure 5.22 Push-backs years 19 to 21 115

Figure 5.23 aE function approximation residuals and the best fit 117

Figure 5.24 aW function approximation residuals and the best fit 118

Figure 5.25 bN function approximation residuals and the best fit 119

Figure 5.26 bS function approximation residuals and the best fit 120

Figure 5.27 h function approximation residuals and the best fit 121

Figure 5.28 changes in a_E and a_W, COPS vs. IOPS 122

Figure 5.29 changes in b_N and b_S COPS vs. IOPS 123

Figure 5.30 changes in h, COPS vs. IOPS 124
Figure 5.31 COPS volume of material transferred over mine life 124
Figure 5.32 Annual production IOPS vs. Whittle 126
Figure 5.33 Annual ore and concentrate production, IOPS vs. Whittle 127
Figure 5.34 average annual grade IOPS vs. Whittle 128
Figure 5.35 Annual production schedule, IOPS vs. COPS 129
Figure 5.36 Ore and concentrate production, IOPS vs. COPS 130
Figure 5.37 Cash flow and average grade, IOPS vs. COPS 131
Figure 6.1 Research summary, models and algorithms developed 136
Figure 6.2 Research summary, numerical application and results 138

LIST OF NOMENCLATURE

EBV_{ij} = economic block value

CBV_{ij} = cumulative block value

TBV_{ij} = temporary block value

P_{0j} = maximum value at the surface

A_1 = area of top ellipse in the open pit geometrical frustum

A_2 = area of bottom ellipse in the open pit geometrical frustum

θ = stable and economical overall pit slope

V = the volume of material in the frustum

a = one half of the length of the major axis of the top ellipse

b = one half of the length of the minor axis of the top ellipse

Δt = changes in production time

ΔV = changes in the volume of material transferred

NF_N = number of concurrent active faces in the north-south

NF_E = number of concurrent active faces east-west directions

NE_N = number of loading equipment in active faces in north-south directions

NE_E = number of loading equipment in active faces in east-west directions

A = equipment availability

U = equipment utilization

L_N = length of cuts in the north-south direction

L_E = length of cuts in the east-west direction

Δa = increments along the major axis of the frustum

$\Delta b =$ increments along the minor axis of the frustum

A_{NW} = area of north-west quarter of the top ellipse

A_{NE} = area of north-east quarter of the top ellipse

A_{SW} = area of south-west quarter of the top ellipse

A_{SE} = area of south-east quarter of the top ellipse

a_W = top ellipse major axis in west region

a_E = top ellipse major axis in east region

b_N = top ellipse minor axis in north region

b_S = top ellipse minor axis in south region

$h =$ frustum height

P_c = annual production capacity

F_s = fleet size

F_c = fleet capacity

C_T =cycle time

γ_l =loose density of material

$C(x_0, y_0)$ = center of the ellipse

(x_0, y_0) = starting point of extraction

α_{ij} = the angle between the center of the ellipse

d = the distance between the center of each block and the center of the ellipse

r = the distance between the center of the ellipse to the perimeter of it in the direction of d

s_t =a state of the open pit at time step t

$S =$ the set of possible states

$a_t =$ an action taken at time step t

$A(s_t) =$ the set of changes possible in the pit geometry in state s_t

$r_{t+1} =$ numerical reward, which is the cash flow of the $t+1$ period of mining operation

$\pi_t =$ agent's policy

$\pi_t(s,a) =$ is the probability that $a_t = a$ if $s_t = s$

$R_t =$ expected return

$\gamma =$ discount factor

$\mathrm{P}^a_{ss'} =$ given any state and action, s and a, the probability of possible next state, s'

$R^a_{ss'} =$ the expected value of the next reward is given by equation

$V^\pi(s) =$ The *value* of state s under policy π

$Q^\pi(s,a) =$ action-value function the value of taking action a in state s under a policy π

$\pi^* =$ optimal policy

$V^* =$ optimal state-value function

$Q^* =$ *optimal action-value function*

$\alpha =$ constant step-size parameter

CHAPTER 1
INTRODUCTION

1.1 Background

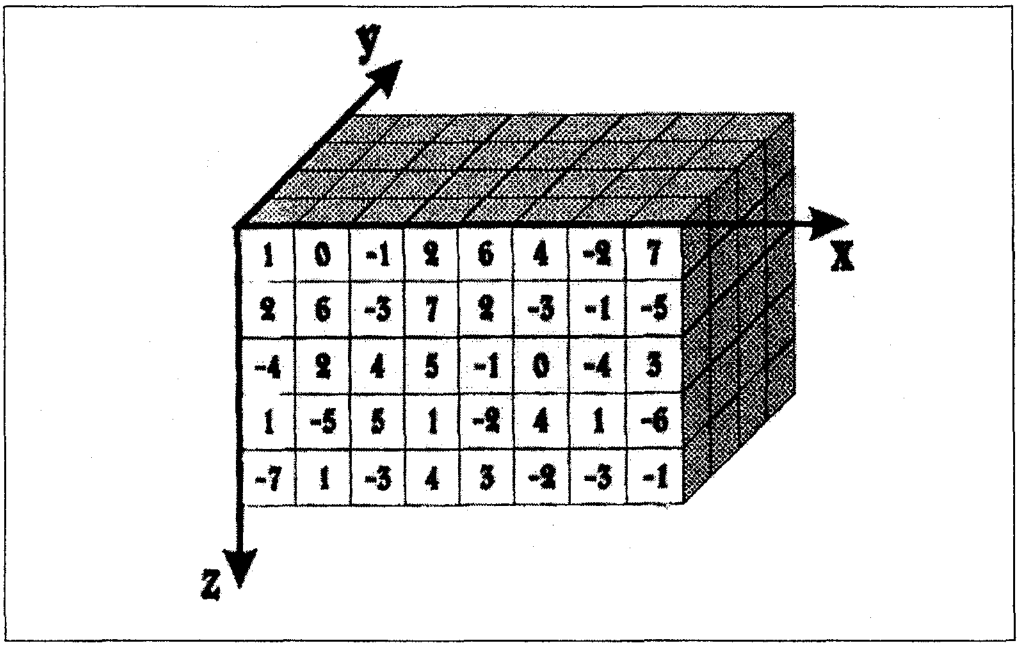
An *open pit mine* is an excavation or cut made at surface of the ground for the purpose of extracting ore, which is open to the surface for the duration of the mine's life. To expose and mine the ore it is generally necessary to excavate and relocate large quantities of waste rock. The main objective in any commercial mining operation is the exploitation of the mineral deposit at the lowest possible cost to maximize the net present value of the operation. The selection of physical design parameters and the scheduling of the ore and waste extraction program are complex engineering decisions of enormous economic significance. The planning of an open pit mine is, therefore, basically an economic exercise constrained by certain geologic, technical, and operating aspects. Open pit mining layouts consist of concentric shells with near ellipsoidal cross-sections. These shells decrease in size with increasing depth from the surface. Wall slopes, whose angles depend on the rock mechanics and geological characteristics of the ore body and host rocks, bound an open pit layout. The challenge to mine planning engineers is to plan, design and optimize the pit layouts to minimize waste removal, ensure safety and maximize the net value of the minerals in the pit (Asa, 2002).

Mine planning has its objective, the specification of an extraction sequence, which maximizes the profit or cash flow over the economic life of the ore body. The optimal plan must determine: (i) the ultimate economic limit (pit limits), which defines the final size and shape of the pit; and (ii) the mining schedule, which depletes the ore body defined by the pit limit and maximizes the chosen financial objective over the life of the mine (Koenigsberg, 1982). According to Whittle (1989), the pit outline with highest net present value can not be determined until the values of the blocks are known; the values of the blocks can not be determined until the extraction sequence has been established; and the extraction sequence can not be prepared without knowledge of the final pit outline.

Lerchs and Grossman (1965) presented a 2D algorithm to determine the optimum ultimate pit limits based on dynamic programming. Simultaneously, they demonstrated a 3D algorithm based on graph theory for optimum ultimate pit limits problem. Some of the researchers used variants of the original algorithm derived from graph theory (Picard, 1976; Whittle, 1988; Zhao and Kim, 1992). Other categories of algorithms namely heuristic methods, include different versions of floating or moving cone (Pana and Davey, 1965; Williams, 1970; Gauthier and Gray, 1971; Robinson and Prenn, 1973; Lemieux, 1976; Lemieux, 1979), the Korobov algorithm (Korobov, 1974), the corrected form of Korobov algorithm (Dowd and Onur, 1993), and dynamic programming (Johnson and Sharp, 1971; Koenigsberg, 1982; Wilke and Wright, 1984; Yamatomi et al., 1995). Rigorous methods including network or maximum flow techniques (Johnson and Barnes, 1988; Yegulalp and Arias, 1992), transportation algorithm (Huttagosol and Cameron, 1992), network flow algorithm based on dual of Lerchs and Grossmann (Underwood and Tolwinski, 1998) and parameterization techniques (Francois-Bongarcon and Marechal, 1976; Francois-Bongarcon and Guibal, 1982). Stochastic methods including genetic algorithm (Denby and Schofield, 1994; Denby et al., 1996), modified conditional simulation/multi-layer feed forward neural networks (MCS/MFNN) (Frimpong and Achireko, 1997), simulative optimization model (Erarslan and Celebi, 2001), simulated annealing and neurogentic algorithm (Asa, 2002), and recently open pit optimization including mineral processing criteria using non-linear goal programming (Esfandiari et al., 2004).

1.2 Statement of the Problem

A problem of central concern to the mining industry is to determine the most profitable material, which can be feasibly removed from an open pit mine. An early task in mine management is the establishment of an accurate model of the deposit. The problem is approached by first taking samples of the ore from boreholes and applying geostatistical techniques to estimate the ore's distribution in terms of quality and quantity (Underwood and Tolwinski, 1998; Caccetta and Hill, 2003). There are a number of models available for reserve modeling. According to Kim (1978), they are classified into:



1	0	-1	2	6	4	-2	7
2	6	-3	7	2	-3	-1	-5
-4	2	4	5	-1	0	-4	3
1	-5	5	1	-2	4	1	-6
-7	1	-3	4	3	-2	-3	-1

Figure 1.1 Sample block model (Thomas, 1996)

(i) regular 3D fixed model; (ii) 3D variable block model; (iii) girded seam model; (iv) 2D irregular model; and (v) 3D irregular block model. The 3D fixed block model is the most commonly used and the most suitable model for computerized optimization techniques. This model is based on the ore body being divided into fixed size blocks. The block dimensions are dependent on the physical characteristics of the mine, such as pit slopes, dip of deposit and grade variability, as well as, the production equipment (Gignac, 1975). A grade is assigned to the center of each block using geostatistical methods such as distance weighted interpolations, kriging (Krige, 1951; Matheron, 1962), sequential Gaussian simulation (SGS) (Deutsch, 2002), or modified conditional simulation (MCS) (Frimpong and Achireko, 1997). Each block is identified by its location coordinates. The net value of the block, referred to as economical block value (EBV), is the difference between the value of the estimated or simulated ore in a block and the cost of extracting that block. The value of the ore, in general depends on its grade richness. In most of the cases a cut-off grade is established; all ore with grades below this cut-off is considered as waste. A second cut-off grade may be defined to differentiate ores sent to the processing plant from ores sent to stockpiles. The gross return of the block is determined on the basis of mineral grade, cut-off grades, and block grades. The cost of mining a block depends on the type of material and the production equipment. The value of the block is the difference between the gross return and the cost of mining (Koenigsberg, 1982).

The general definition of the ultimate pit limits could be described as size and shape of mine-able reserves and associated waste materials to be excavated based on technical, economical, and safety constraints. It can also be defined as the final dimensions of an open pit at the end of the mining period, which maximizes the objective function. The management's objective could be based on: recovered metal, profit, or net present value. The model usually provides information for evaluating: (i) the economic potential of a mineral deposit; (ii) financing and taxation; (iii) short and long-term mine plans; and (iv) the boundaries outside which mine plant and structures should be located.

The objective of pit design and optimization is to maximize the net value of an open pit operation subject to geological, geotechnical, economic and capacity constraints. Constrained by variable field slopes, blocks are removed to optimize the pit layouts following algorithmic procedures that mimic the extraction sequence of a mining operation (Frimpong et al., 2002). The problem of finding the optimal plan is compounded by the interaction between the schedule (which depends, in part on the pit limit) and the pit limit (which depends, in part on the schedule) and by the physical and technical constraints on mining operations. The physical constraints include the type of soil or rock in the substrate, which affects the limiting slopes of the pit, the speed, and cost of excavation. The technical constraints would include the number and size of the excavation and transportation equipment, which affects the speed and cost of excavation and, perhaps, the extraction sequence of the ore body. Other technical constraints depend on the rate and cost of processing the ore (i.e., on economic considerations external to pit development). These may be included in the planning program or introduced externally (Koenigsberg, 1982).

The standard algorithm, which is used by commercial packages[1] in industry today, is Lerchs and Grossmann (1965) based on graph theory. The algorithm is rigorously optimal with due consideration to the primary assumptions. The algorithm is limited due

[1] Whittle Four X, MineMax Planner, MaxiPit Earthworks,

to (i) complexity in terms of comprehensibility and programming; (ii) long computing time; (iii) difficulty of incorporating variable slopes; (iv) optimizing criterion is total undiscounted profit, where as it should be net present value (Dowd and Onur, 1993).

Achireko and Frimpong (1996), Frimpong and Achireko (1997), Frimpong et al. (1998), Frimpong et al. (1998), Frimpong et al. (2001), Frimpong et al. (2002), and Frimpong et al. (2002) outlined the limitations of LG[2] and other common algorithms. These limitations are due to the stochastic processes governing ore reserves, commodity price, and production cost. The variables underlying mining operations are stochastic in nature, and their inherent uncertainties can significantly affect the value of an optimized layout. Any algorithm that neglects these processes has a potential to yield suboptimal results. Short learning curves and CPU times are invaluable in data processing, program execution, and interpretation of results for prompt decisions. Current algorithms must be rerun for any changes in information and database, thus may result in long CPU times.

Computational intelligent (CI) algorithms are capable of storing the submitted patterns of layout in memory after complete training of the neural agents. Thus, any changes in database can be processed at a faster rate without having to retrain the neural agents, resulting in shorter CPU times. The ease of understanding CI algorithms and their numerical procedures results in short learning curves. This is particularly significant in mining applications since mining engineers spend much time in mastering the concepts and procedures in current algorithms. An important design focus in surface mine layout is slope stability, a function of the geological structures within the formation environment. Current algorithms are blind to geological structures that may affect the pit layout. As a result, the resulting optimized layout may not be technically feasible because of the presence of major faulting, folding, jointing, or disturbed regions. The potential for the CI algorithms to store information on geological structures for use in layout optimization is significant in surface mining. Finally, current algorithms make no allowance for operating strategies to maximize the net pit values. The value derived from any mining activity is dependent on operating strategies used for extracting minerals. The sequence

[2] Lerchs and Grossmann

and schedule of material extraction, incremental haul road distances, fleet management, and other important strategies affect the value of a surface mine. These deficiencies can cost a surface mining operation severely in terms of dollars, time, material and logistics handling, and future strategic and tactical plans.

In spite of the importance of haul roads in open pit mine design, most computer-aided design software packages neglect haul roads in surface mine layout design. Haul road design is directly input into the computer model and the results of ultimate pit limits are subsequently adjusted to reflect the incorporation of the haul road. Direct input can be either through manual coding or the digitizer (Kim, 1978). The incorporation of the road forces changes in the shape and slopes in different areas of the ultimate pit limit design. The location of the haul road alternates the access to the ore that was designed by the optimization method.

Another important consideration in open pit planning and design, which is mostly neglected, is the consideration of the whole mining process as a unit. Mining operation consists of exploitation and mineral processing as a unit global entity. One should always take this into account that neither pit limit optimization nor production scheduling optimization is goal by themselves; they are parameters that are used to maximize the total value of the mining venture. Most of the existing algorithms do not consider the effects of the processing plant and stockpiles on the block extraction sequence. The basic demands of the processing plant are uniform material grade and maximum usage of the processing plant capacity. Uniformity should be both in quantity (amount) and quality (grade) to minimize idle times of the plant. The supply of material to the plant requires an optimized excavation sequence. Hence, the optimum pit configuration is not only a function of monetary values but also of technical constraints.

The variable slope deficiency associated with the LG algorithm has been solved through research (Whittle Programming Pty, 1992; Whittle Programming Pty, 1998; Khalokakaie et al., 2000; Khalokakaie et al., 2000; Khalokakaie et al., 2000). Tremendous improvements in computing science have resulted in new algorithms with acceptable running times.

Although numerous algorithms have been developed since 1965, when Lerchs and Grossmann published the landmark paper, there are still many shortcomings in the available techniques that are used in industry. There is a requirement for continuous research that will lead to the development of a robust and comprehensive algorithm, which will improve the open pit mine planning and design extensively.

1.3 Objectives of the Study

The primary goal of this research study is to develop and implement an intelligent 3-D algorithm for optimal long-term open pit planning. The elements of this primary objective include: (i) mathematical modeling and adapting the long-term open pit mine planning problem in the context of reinforcement learning (RL); (ii) analyzing and implementing "**I**ntelligent **O**pen Pit Optimal **P**roduction **S**imulator" (IOPS) with an interactive graphical user interface; and (iii) verifying and validating the models.

The entire RL design framework is broken down into the development of three major entities of reinforcement learning paradigm: (i) the environment - geostatistical ore-body block model is the entity that the agent learns from; (ii) intelligent agent - a computer system capable of autonomous action in the environment in order to meet the main objective of maximizing the NPV over time; and (iii) simulation – a simulation model, which stochastically mimics the discrete and continuous dynamic expansion of open pit layouts.

In order to achieve the objectives, this work includes development of the theoretical and conceptual architecture of IOPS. Also, the research is concerned with algorithm development, object oriented analysis, design, and implementation of the algorithm as a graphical user interface.

1.4 Scope and Limitations of the Study

The study deals with development of an intelligent 3-D open pit long-term planning and optimization algorithm. The main focus of the study is on optimal long-term open pit planning, employing reinforcement learning, which is a subtopic of artificial intelligence. The objective of the algorithm is maximizing the net present value of open pit operation subject to geological, operating, technological, financial, and marketing requirements and constraints. Following a stochastic dynamic open pit expansion model, which imitates the extraction sequence of a mining operation, constrained by variable pit slopes, and limited shovels movements' blocks are removed to optimize the material scheduling and pit layouts.

A comprehensive problem definition and mathematical modeling is carried out to define and develop the IOPS architecture, which complies with the objectives of the study. The conceptual frame work of intelligent open pit planning is based on reinforcement learning paradigm. The IOPS is developed based on an off-policy temporal difference (TD) control algorithm known as one step Q-learning (Watkins, 1989).

Object oriented analysis, design, and implementation are carried out to develop IOPS application as a graphical user interface (GUI). Java (Sun Microsystems, 1994-2006) and MATLAB (MathWorks, 2005) are employed to develop IOPS. The GUI for the IOPS engine is implemented using Java IDE IntelliJ IDEA (JetBrains, 2000-2006). For visualization of the results the commercial mining package GEMCOM and AutoCAD are employed.

Experimental design framework and model correctness are verified and validated through a case study. The optimized pit limits are designed using LG (Lerchs and Grossmann, 1965) algorithm. Subsequently the best-case and the practical annual long-term schedule generated by IOPS and the shells node in Whittle Four-X (Gemcom Software International, 1998-2005) are compared. The experiments compared the annual stripping ratio, average grade, the annual waste and ore production, and the respective NPV. The results of the experimental design are subjected to sensitivity and statistical analysis. Although optimality is not guaranteed mathematically with the parametric analysis in Whittle 4-X, it is a strong tool for identifying high grade ore clusters in the model and is the standard tool used in industry for long term planning.

1.5 Research Methodology

The first section of the study carried out a comprehensive literature survey on the open pit optimization and scheduling algorithms. This was followed by gathering borehole data from an iron ore mine in Iran for the future experimental studies. Figure 1.2 illustrates the research methodology as part of the broader context of mineral supply process. Appropriate mathematical and numerical models have been formulated to capture the inputs and outputs of the intelligent open pit planning framework. The research focuses on development, analysis, and implementation of two main models (i) intelligent open pit optimal production simulator (IOPS) based on reinforcement learning (RL) to find an optimal plan which maximizes the NPV of the mining operation; and (ii) continuous open pit simulation (COPS) based on the modified open pit geometrical model, and a system of PDE's capturing the continuous-time open pit dynamics.

Appropriate solution techniques and theory have been employed to convert these models into a finite set of procedural instructions, algorithms, and numerical methods in order to accomplish the research objectives.

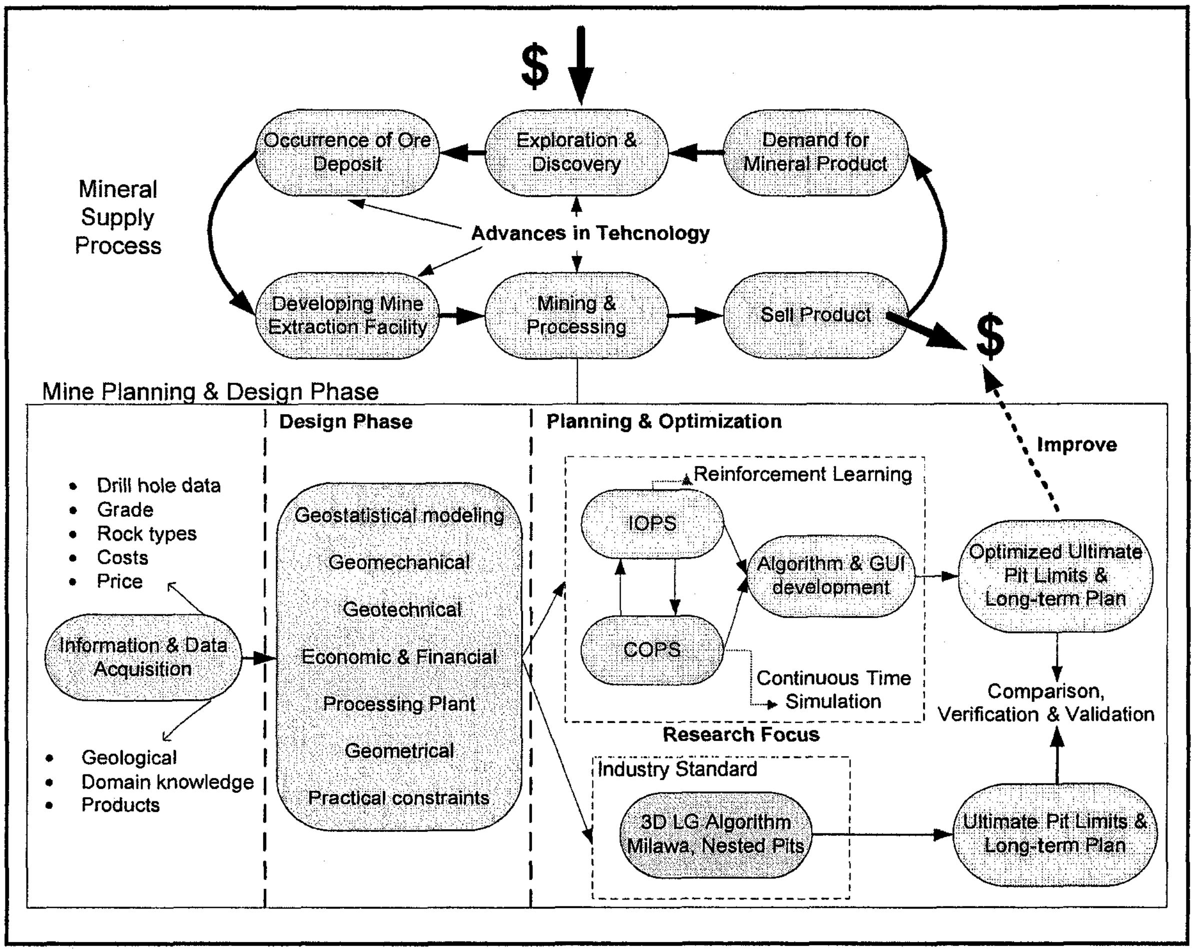


Figure 1.2 Flow diagram of research methodology

The design and implementation of IOPS application continues with class design, object identification, and implementation in Java and MATLAB. The result of this research phase is the IOPS application with a graphical user interface capable of learning the optimal long-term plan based on Q-learning algorithm. Subsequently, the models have been verified and validated with an experimental study of long-term planning of an iron ore mine. The optimized pit limits are designed using LG (Lerchs and Grossmann, 1965) algorithm. Afterward comparisons are made of the best-case and the practical annual long-term schedule generated by IOPS, COPS, and the shells node in Whittle Four-X. The experiments then compared the annual stripping ratio, waste and ore production, average grade, and the respective NPV of the venture. The results of the experimentation are subjected to sensitivity and statistics analysis. The solution of the IOPS and COPS return the trajectories of changes in major and minor axes and the depth of the open pit model, as well as, the volume of extracted materials. The results from all the experiments have been analyzed to draw relevant conclusions with appropriate recommendations.

1.6 Contributions and Industrial Significance of the Study

This research enormously contributes to the body of knowledge on intelligent open pit mine planning and design. The research has formulated robust mathematical models and comprehensive algorithms, expanding the frontiers of open pit planning and optimization using reinforcement learning. The research resulted in implementation of the IOPS framework, which has the potential to be the fundamental of the next generation of mine design software packages based on intelligent agents. The outcome of the research has resulted in the development of a novel methodology, which improves the expected monetary return on the mining investments to a great extent. The original algorithm enables step-changes in planning and management of mines.

The key limitations of current production scheduling methods namely heuristic methods, nested pits, and operations research methods (mixed integer programming, dynamic programming, linear programming, goal programming, branch and bound methods) are: (i) inability to solve large industrial problems. The extraction sequence in a predetermined final pit limits has to deal with the limitation of computing resources. These limitations include: time and space; (ii) limitation in dealing with stochastic processes governing ore reserves, commodity price, cut-off grade, and production cost; (iii) inadequacy of the current final pit limits optimization methods in taking into account the temporal nature of the exploitation; and (iv) shortcoming in defining the economics of ore with respect to the economics of the entire mining process, from ore to the finished product.

The study aimed at addressing all the abovementioned limitations. The research tackled the curse of dimensionality and the computational limitations of the mine long-term planning problem by employing reinforcement learning. Q-learning as the core of IOPS engine can learn optimal plans from experience in the form of sample simulation episodes of push-backs. Unlike the current algorithms IOPS has the capability of dealing with the mine planning parameters as stochastic variables.

1.7 Organization of Thesis

Chapter 1 of this dissertation discusses the background and the statement of the problem; this is followed by objectives of the study, scope and limitations of the study, the proposed methodology, report structure, and the contributions of the study.

Chapter 2 provides a detail literature survey of open pit planning and design. The survey comprises the final pit limit optimization and production scheduling algorithms, which have been employed in the mineral industry in the last 45 years. It also contains the review on the design and incorporation of haul roads in optimization techniques. The chapter concludes with the rationale for the PhD research.

Chapter 3 fully outlines the conceptual framework of intelligent open pit planning. The general theoretical framework, mathematical models and their interrelationships are developed and discussed for achieving research objectives. The focus of research is on development, analysis, and implementation of intelligent open pit optimal production simulator (IOPS) based on reinforcement learning (RL). This chapter focuses on formulating, modeling, and developing the components of IOPS: environment, simulation, intelligent agent, and their relationships. The chapter also contains the continuous open pit simulation (COPS) model. COPS characterizes the changes made in the geometry of the open pit and the volume of materials moved by a set of differential equations. The chapter concludes with a review on solution methods to the RL problem.

Chapter 4 deals with algorithm development, computer modeling, and IOPS application implementation. A graphical user interface (GUI) with an object oriented design has been implemented. The IOPS application is implemented in Java (Sun Microsystems, 1994-2006) and MATLAB (MathWorks, 2005) environment. This exercise consisted of class and object identification based on the Java Reinforcement Learning Library, JavaRL, (Kerr et al., 2003).

Chapter 5 is mainly concerned with experimental design and experimentation, which consists of validation, verification, and model correctness. A case study of an iron ore deposit was carried out to verify and validate the model. The optimized pit limit was designed using LG algorithm. Subsequently the best-case and the practical annual long-term schedule generated by IOPS, COPS, and the shells node in Whittle Four-X are compared. The experiments compare the annual stripping ratio, average grade, the annual waste and ore production, and the respective NPV. The results from all the experiments have been analyzed to draw relevant conclusions and make necessary recommendations.

Chapter 6 contains the study summary, concluding statement, research contributions, and recommendations for future work in this research paradigm.

CHAPTER 2
ANALYTICAL LITERATURE SURVEY

An extensive literature survey is the focus of this chapter. This literature survey is being employed in evaluating past and current developments in the optimization and scheduling algorithms in open pit mining. Open pit mine design and optimization can be divided into open pit optimization algorithms and open pit scheduling algorithms. A short review of haul road design is included because it has a great influence on the projects economics.

2.1 Open Pit Optimization Algorithms

Kim (1978) classified optimization techniques as either "rigorous" or "heuristic". Thomas (1996) expanded the definitions to 'rigorous', 'heuristic', 'stochastic', 'static', 'dynamic'. The existing methods today could be categorized as:

Rigorous – A term applied to algorithms which, given enough time, will always find the optimum solution to the problem for the supplied data and constraints, and for which a formal proof has been developed for the optimality of their solutions.

Heuristic – A term applied to techniques which lack rigorous mathematical proof to the optimality of their solutions. These methods either work in nearly all cases or find an approximate solution to the problem, which may or may not be close to the true optimum.

Stochastic – A term applied to techniques which base their analysis on probabilistic sampling of the range of possible solutions.

Static – A term applied to analyses which neglect the effect of time on monetary values.

Dynamic – A term applied to analyses which consider the effect of time on monetary values.

Artificial Intelligence – A term applied to methods which use the computations that make it possible to perceive, reason and act (Winston, 1991).

Open pit mining is defined as the process of development and superposition of a geometric surface called a pit onto the mineral reserve. The mineable reserve is the component within the pit boundaries. The size and shape of the pit depends upon economic factors and design/production constraints. The pit existing at the end of the mining is called the final or the ultimate pit. In between the startup and the closure of an open-pit mine, there are a series of 'intermediate' pits (Hustrulid and Kuchta, 1998). An open-pit mining operation can be viewed as a process by which the surface topography of a mine is continuously extracted using a series of box cuts. The planning of a mining program involves the design of the final shape of this open surface (Lerchs and Grossmann, 1965).

The ultimate pit limit defines the size and shape of an open pit at the end of its life based on the operating, technical, economic and ground stability constraints. It also determines the extent of the mineable reserves and the waste materials to be moved in the mining process. Pit limits on the surface mark the boundaries for locating surface structures, such as processing plants and mine offices. The pit limits would normally delineate the limiting boundary beyond which the open pit mining of a given deposit will be uneconomic. As such the pit limits are commonly referred to as the economic pit limits (EPL), ultimate pit limits (UPL) or ultimate pit design (UPD) (Asa, 2002).The optimum ultimate pit limit of a mine is defined to be that contour which is the result of extracting the volume of material which provides the total maximum profit whilst satisfying the underlying field constraints. The ultimate pit limit gives the shape of the mine at the end of its life (Caccetta and Giannini, 1990).

Since the development of the first computer models in the late 1950's, much research has been carried out and many papers have been written in the quest for good solutions to the optimum ultimate pit limit problem. Lerchs and Grossmann (1965) introduced two modeling approaches, namely, the graph theory and the dynamic programming algorithm. Both methods make use of the economic block model to yield an optimum pit limit where the effect of time on monetary values is not taken into consideration. Additionally, the two methods don not consider the effect of extraction time of ore blocks. This assumes that, from the point of view of mine economics, there is no difference or significance attached to the extraction time of a more valuable block to that of a less valuable block. It can be said that under these assumptions, graph theory is a powerful and reliable tool for the determination of the optimum pit limit.

Several researchers including (Picard, 1976; Whittle, 1988; Zhao and Kim, 1992; Khalokakaie et al., 2000; Khalokakaie et al., 2000) have applied this theory to solve several mining optimization problems. Underwood and Tolwinski (1998) developed a network flow algorithm based on the dual of Lerchs and Grossmann's (1965) primal model. This new model provides an interpretation of the graph theory when applied in a mathematical programming setting to analyze convergence and other mathematical implementation issues. Dynamic programming optimization is typically performed on two-dimensional block sections and integrated to obtain a three dimensional pit limit. This approach tends to yield inconsistent shapes necessitating the use of smoothing routines.

Johnson and Sharp (1971) and Koenigsberg (1982) have tried to eliminate the shortcoming of this algorithm but these attempts have not been totally successful in finding the mathematically true optimum. Pana and Davey (1965) introduced an algorithm called moving cone that works on the same assumptions as graph theory and the dynamic programming model. However, the shortcoming of this model is that it creates overlapping cones, and it is incapable of examining all combinations of adjacent blocks. For this reason, the algorithm fails to consistently give realistic results. Lemieux (1979) has tried to solve this problem by adding some heuristic approaches. Lizotte (1988) proposed yet another solution but he was candid enough to admit that his approach can't give the true optimum pit limit in all cases. Meyer (1969) has brought an approach to the optimization problem, namely a linear programming algorithm. However as the author implies, the algorithm does not guarantee the optimum, and interest rate is not taken into consideration. Two other algorithms for pit limit optimization have been introduced. These are the maximum flow algorithm (Yegulalp and Arias, 1992) and the transportation algorithm (Huttagosol and Cameron, 1992). Maximum flow algorithms are widely applied in operations research (Ford and Fulkerson, 1956; Ahuja and Orlin, 1989). Yegulalp and Arias (1992) successfully adapted this algorithm for solving mining problems. The algorithm finds optimum limits quickly, and the authors were able to greatly increase the processing speed. However, the algorithm does not consider either the time value of money or the economical significance of the block extraction sequence. The transportation algorithm was introduced by Huttagosol and Cameron (1992). The algorithm gives true optimum limits.

Frimpong and Achireko (1997) introduced a set of algorithms that address the random filed properties of the various parameters involved in open pit optimization. The modified conditional simulation and multilayer feed-forward error-back propagation algorithm were developed to capture the stochastic behavior of ore grades and reserves on one hand and to utilize neural network architecture in the optimization process. Aside from the fact that the open pit optimization problem is a 3-D problem. The use of a back propagation could result in a local optimum instead of global optimum. Their approach took into consideration the stochastic nature of the open pit optimization process. Asa (2002) used neuro-genetic algorithms for slope design, grade and block modeling. Simulated annealing was used to optimize the pit for 30 realizations and the results were fed into a neuro-genetic algorithm for true optimization.

Besides the algorithms described above, there are two other optimization models that may be used to solve open pit mine problems. These are bounding techniques (Kim, 1978; Barnes and Johnson, 1982; Whittle, 1988) and parameterization (Bongarcon and Marechal, 1976; Bongarcon and Guibal, 1982; Dagdelen and Bongarcon, 1982; Barnes and Bertrand, 1990; Wang and Sevim, 1992; Zhao and Kim, 1992) According to Lizotte (1988), bounding techniques are not optimization techniques, but they provide a methodology for limiting the search for optimum pit by defining upper and lower bounds, thereby reducing the processing time. The parameterization approach seeks to simplify optimization by initially identifying problem-bearing variables and solving the optimization problem using convenient parameters and dimensions. The approach is generally a geostatistical one wherein starting block dimensions are relatively large, but once the pit outline has been determined, the block dimensions are reduced to the normal size. Parameterization routines also use cost/price ratio as a more stable parameter in place of monetary value, and the maximum metal content in place of the economic block model.

These models also impose geometric constraints on the pit. One of the arguments being made by researchers who favor the use of this optimization technique is that economic parameters are dynamic, and that economic block models are not reliable. Therefore, a search for the pit with maximum metal content eliminates most of the aforementioned problems. One of the problems is that the choice of maximum metal content as the optimization parameter allows the researcher to focus only on the grades of blocks while ignoring such factors as price, cost, and interest rate. However, one should always remember that the location of an ore block is as crucial as its grade. For instance, two blocks that have the same grade and metal content, but with one block located on a shallow part of pit, while the other block is located on a deeper part, will not have the same value from an economic point of view. The block located at the deeper part of the pit may not be extracted due to stripping cost of the cone over it. In other words, even if the block located at the deeper parts of the pit contributes significantly to the overall metal content, it may not provide the same contribution economically because its mining cost may exceed its potential revenue. So, the pit with maximum metal content may not necessarily mean the pit with maximum profit. Thus, a block by block production plan should be included in optimization calculations.

Wang and Sevim (1992) have also emphasized the necessity of production planning in parameterization studies. Their approach is based on the estimation of initial pits with maximum metal content (long-term planning), followed by mining through those pits block by block (short-term planning). By using this approach, one avoids the misleading effects of monetary values while allowing further processing with the economic block model. Processing plant requirements should be factored into the optimization routines as they have a major influence on the extraction sequence, and subsequently on the final pit contours. In other words a complete optimization system should be able to reflect the stages; in a mine so that before mining starts, the proposed methodology could be simulated to its future impact should be predicted. Such a system would be a powerful decision making tool for mining operations.

2.1.1 The Manual Method

Manual approach of open pit design is the traditional design method which is based on trail and error, in which the designer uses subjective and objective analyses to define the pit limits. The data required are: (i) vertical sections showing ore boundaries, grade distribution within the ore, overburden and waste rock; (ii) plans for proposed mining levels showing ore/waste; (iii) allowable maximum slope for different rock types and different areas of the pit; (iv) minimum pit width at the suggested pit bottom; (v) stripping ratio-ore grade curves and (vi) net value–ore grade curves representing the alterations of the stripping-ratio and net value of a section with the ore grades and possible product selling price.

With the aid of stripping ratio-grade curves and net value-grade curves the pit limits are designed as follows: (i) generate vertical sections and level plans of the deposit (cross-sections, longitudinal sections, radial-sections) and select a slice; (ii) from these sections and their intervals, generate volumes (waste, ore) tonnage and stripping-ratios; (iii) generate the economic striping-ratio which defines the limit of mining that ensures the achievement of this cost minimum profit margin; (iv) generate economic limit stripping-ratio versus ore grade and economic limit stripping-ratio versus selling price; (v) using the economic limit stripping-ratio, generate the pit limits for each section and (vi) using planimeter, estimate the volumes and tonnage of ore and waste on each section.

This method is very time consuming and tedious, and it is only possible for small deposits with simple geology. The method is prone to errors from judgment and tiredness and it is impossible to carry out many design options using sensitivity and risk analysis.

2.1.2 The Moving Cone Algorithm

Moving or floating-cone algorithm and its variants are the most popular heuristic or nearly optimizing techniques (Pana and Davey, 1965; Williams, 1970; Gauthier and Gray, 1971; Robinson and Prenn, 1973; Lemieux, 1976; Lemieux, 1979). The method is a simulation technique in which cones or frustums are established using positive economic blocks as basis. If the sum of economic block values for all the blocks in the cone is positive, the cone is included in the mine-able set of cones. Negative economic blocks and cones with negative values established with positive economic blocks are not considered as part of the optimum layout. The basic element for the optimization process is the minimum removal cone. A summary of the algorithm can be stated as follows: (i) the cone is floated from left to right along the top row of blocks in the section if there is a positive block it is removed; (ii) after traversing the first row, the apex of the cone is moved to the second row. Starting from the left hand side it floats from left to right stopping when it encounters the first positive block; (iii) if the sum of the economic block values (EBV) of all blocks contained in a given cone is positive, consider the cone removed (extracted); (iv) the floating cone process moving from left to right and top to bottom of the section continues until no more blocks can be removed; (v) the profitability of this section is found by summing the values of the blocks removed; (vi) the ultimate pit is formed by the shape left after the removal of all positive blocks; Figure 2.1 illustrates the moving cone algorithm flowchart.

The 3D positive moving cone technique, the search commences at the north-west corner (top left) of the block model and proceeds from west to east (left to right) along each level. All blocks with positive value on the first level are examined, before proceeding to the second level and then to the third level until the ultimate pit depth. Finally the 3D block representation of the optimum pit is a combination of the optimum 2D cross-sections and longitudinal sections. Smoothing is employed to fit the sections into 3D pits after the completing the optimum pit limits on 2D sections (Asa, 2002).

According to Wright (1990) the floating cone method can be an expensive process, especially for a large open pit mine. Furthermore, the technique can miss the optimum pit limit under certain unusual conditions, because it cannot execute the pit for a joint contribution by two blocks that are laterally some distance apart. Lemieux (1968) stated that his heuristic algorithm does overcome this short coming of the moving cone technique. However, the geometric requirements of slope stability, in combination with certain ore blocks, can lead to situations where the technique will fail to yield the pit with the maximum value (Kim, 1978). Barnes (1982) has presented three problems representing floating cone situations not leading to an optimum pit. (i) missing combinations of profitable blocks; (ii) extending the ultimate pit beyond the optimal pit limits; and (iii) combination of problems (i) and (ii).

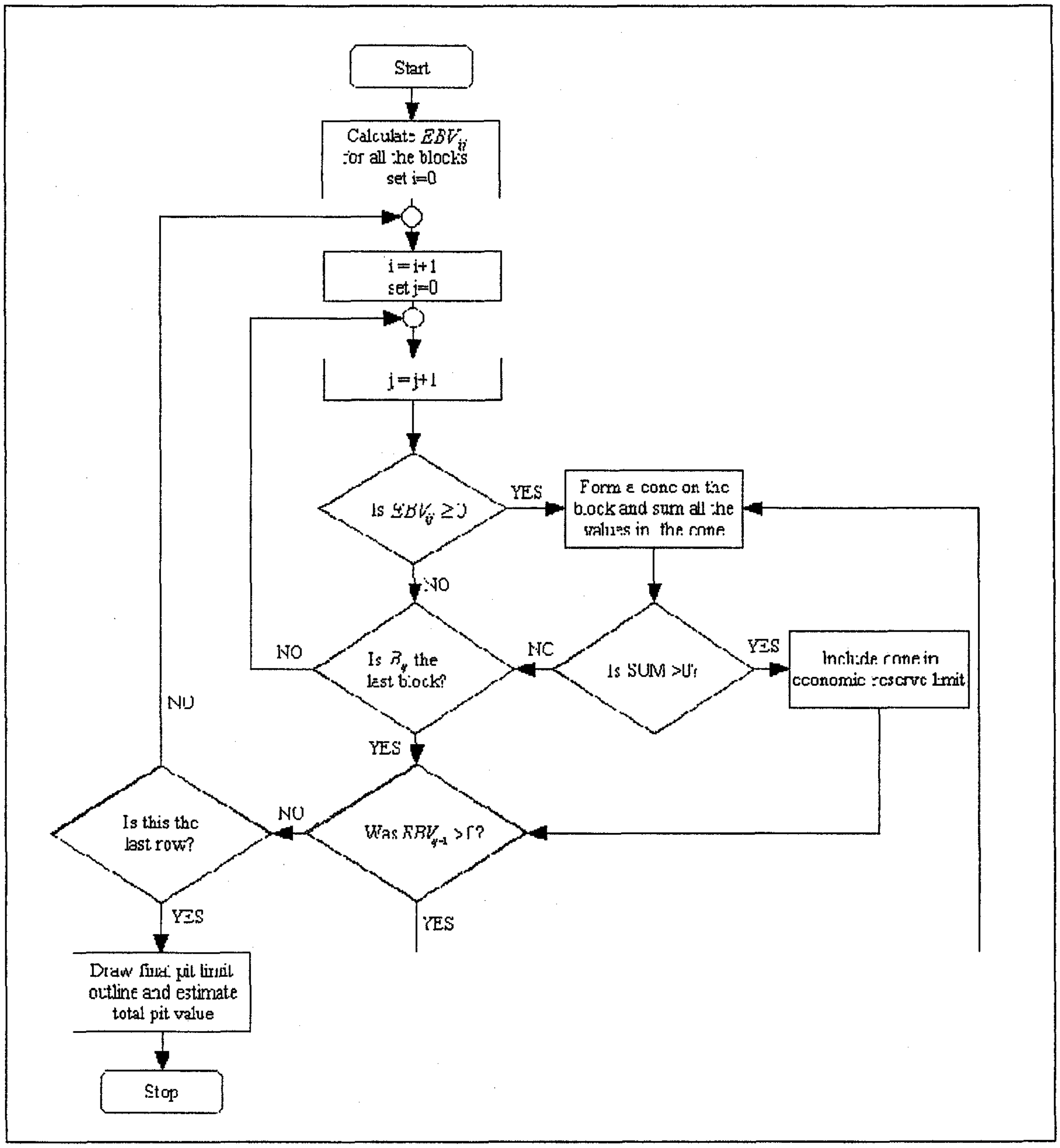

Start
Calculate EBV_{ij} for all the blocks set i=0
i=i+1 set j=0
j=j+1
Is $EBV_{ij} \geq 0$
YES
Form a cone on the block and sum all the values in the cone
NO
Is B_{ij} the last block?
NO
NO
Is SUM >0?
YES
Include cone in economic reserve limit
NO
YES
Is this the last row?
NO
Was EBV_{ij-1} > 0?
YES
YES
Draw final pit limit outline and estimate total pit value
Stop

Figure 2.1 Moving cone algorithm flowchart

In spite of these problems, there are however a number of very positive aspects of the technique, which accounted for its widespread use and popularity especially during the 80's including: (i) The method is easy to understand and use by mining engineers; (ii) computationally, the algorithm is quite simple. Development and implementation of a moving cone computer program does not require sophisticated knowledge in operations research or computer science. The computer code could be developed in-house rather than purchased from a software company. Thus, a more custom-fitted product can be provided at an operating mine site; (iii) the moving cone technique can be used with generalized pit slopes. The single requirement is an unambiguous rule for determining which blocks overlie individual ore blocks; (iv) it provides highly useable and sufficiently accurate results for engineering planning (Hustrulid and Kuchta, 1998).

2.1.3 Lerchs and Grossmann Algorithm

Lerchs and Grossmann (1965) used dynamic programming and graph theory for optimum design of open pit mines. The objective of their studies was defined as the design of the contour of a pit so as to maximize economic value of the material contained in the layout. Their model was based on the following assumptions: (i) the type of material, its mine value and its extraction cost are given for each point; (ii) restrictions on the geometry of the pit are specified (surface boundaries and maximum allowable wall slopes); (iii) the objective is to maximize total profit: total mine value of material extracted minus total extraction cost. The 3-D algorithm is mathematically proven to generate a rigorous optimum solution, based on its assumptions. The algorithm is the basis of commercial packages used in industry.

2.1.3.1 Lerchs and Grossmann 2-D Algorithm

By this method, the economic block values, cumulative block values, and temporary block values, are estimated column wise. Arcs are provided to indicate the direction of optimum pit layout using dynamic programming based on the following procedure: (i) divide the 2-D section of the ore body into blocks, $B_{ij}(i=1...n; j=1..m)$. Provide a dummy row of blocks at the top; (ii) estimate the economic block values, EBV_{ij}, for all the blocks using the following formula:

$$EBV_{ij} = \mathrm{Re}\,v_{ij} - TC_{ij}\ \forall \qquad \text{waste or ore block} \qquad (2.1)$$

$$EBV_{ij} = 0 \qquad \forall \text{ dummy blocks} \qquad (2.2)$$

(iii) compute the cumulative column values for each block from top to bottom. Designate this cumulative column figure as the second value in the respective block;

$$CBV_{ij} = \sum_{r=0}^{i} EBV_{ij} \qquad r \leq i \qquad (2.3)$$

(iv) from the northwest corner or the section (on level zero), compute the temporary block values as follows and designate TBV_{ij}, as the third number in the respective blocks.

$$TBV_{ij} = CBV_{ij} + \max \begin{cases} TBV_{i+1,j-1} \\ TBV_{i,j-1} \\ TBV_{i-1,j-1} \end{cases} \tag{2.4}$$

(v) compute the TBV_{ij} values for all the blocks, column by column. (vi) The maximum value at the surface, P_{0j}, is the total value of the pit; (vii) The path followed in back tracing from this maximum P_{0j}, defines the optimal contour of the pit, which further defines the pit outline that maximizes the net value of the ore body.

2.1.3.2 Lerchs and Grossmann 3-D Algorithm

The open pit optimization can be formally represented as a graph problem defined on a directed graph *G=(V,E)*. Each block corresponds to a node in *V* with a weight EBV_i representing the net value of the individual block. This weight can either be positive, negative or zero. There is a directed arc in *E* from node i to node j if block i can not be extracted before block j. Thus to decide which block to extract in order to maximize profit is equivalent to finding a maximum weight set of nodes in the graph such that all successors of the nodes in the set are included in the set. Such a set is called a maximum closure of *G* (Hochbaum and Chen, 2000).

Some Important terms and concepts are as follows: *Path:* A path is a sequence of arcs such that the terminal vertex of each arc corresponds to the initial vertex of the succeeding arc. *Circuit:* A circuit is a path in which the initial vertex coincides with the terminal vertex. *Directed Graph:* A directed graph is defined by a set of nodes connected by ordered pairs of elements called arcs. *Closure:* Closure from the view point of a mining engineer is simply a sub-graph yielding a feasible pit. *Tree:* A tree T is a connected and directed graph containing no cycles. *Root:* A root is one node selected from a tree. A tree may have only one root.

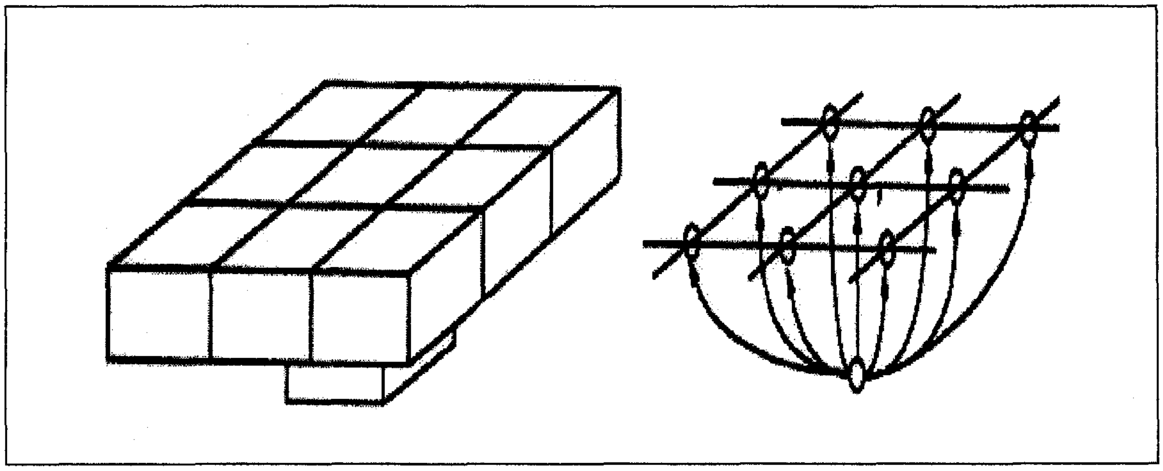


Figure 2.2 Representation of the 1-9 block constraints (Hustrulid and Kuchta, 1998)

Formally the optimization problem is defined as the determination of the maximum closure graph. Given a directed graph *G=(V,E),* and node weights (positive or negative) EBV_i for all $i \in V$. The algorithm finds a closed subset of nodes $V' \subset V$ such that $\sum_{i \in V'} EBV_i$ is maximum.

An edge of the tree either points towards the root or points away from the root. To distinguish the orientation of the edges with respect to the root in a particular tree, the following definitions are used: (i) an edge pointing away from the root is a *p-edge* and an edge pointing toward the root is a *m-edge*; (ii) a p-edge is **strong** if it supports a mass that is strictly positive and an m-edge is **strong** if it support a mass that it is null or negative. Edges that are not strong said to be weak Table 2.1 and Figure 2.3; (iii) a branch is strong if the edge that links it to the root is strong, otherwise it is weak. The nodes of a strong (weak) branch are strong (weak) nodes; (iv) a tree in a graph is **normalized** if only strong edges it contains are adjacent to the root.

The graph *G* is first augmented with a dummy node. The algorithm starts with the construction of a tree T_0 in *G.* T_0 is then transformed into successive trees $T^1, T^2, \ldots.., T^n$ following given rules, until no further transformation is possible. The maximum closure is then given by the vertices of a set of well identified branches of the final tree. Each iteration involves three operations: (i) **Merger:** adding an arc from a strong branch S to a weak branch W and removing the arc from the root of the sub-tree S to the root of S; (ii) **Mass update:** updating total value of all nodes; and (iii) **Renormalization:** removal of strong edges and there sub-trees and reattaching them to root as separate branches, see Figure 2.4. The flow chart of the algorithm is shown in Figure 2.5.

Table 2.1 Labeling guide for arcs

Case	Direction	Cumulative weight	Label
1	Plus	Positive	Strong
2	Plus	Null or negative	Weak
3	Minus	Positive	Weak
4	Minus	Null or negative	Strong

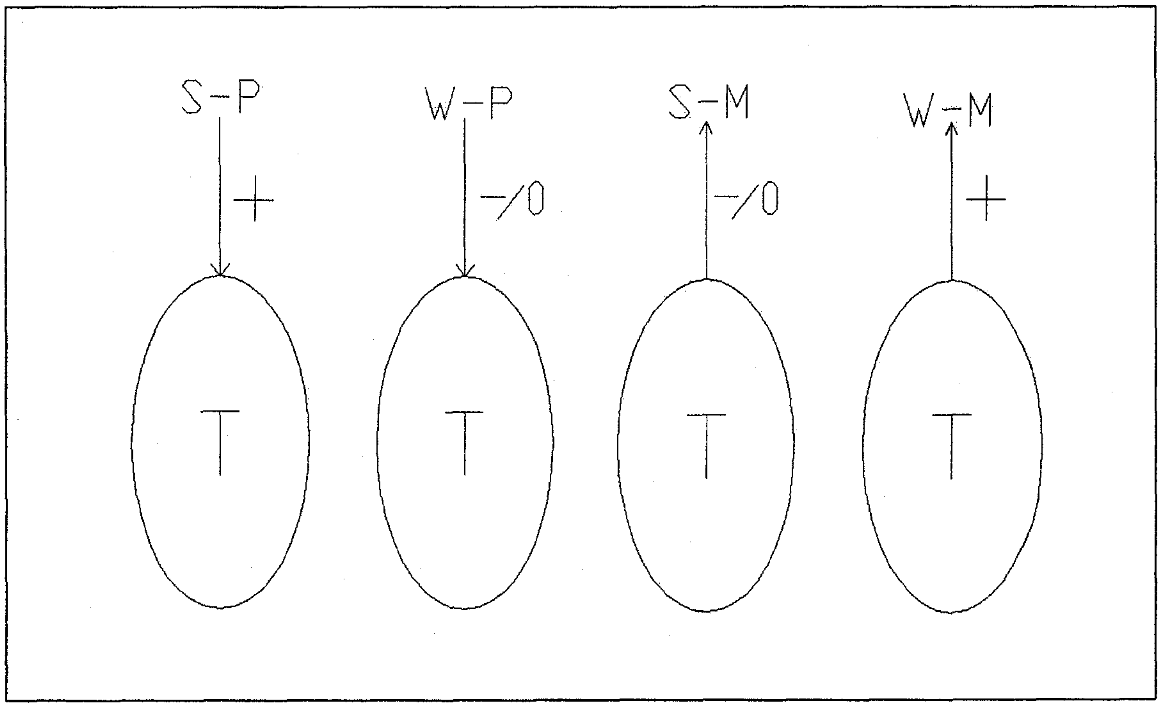


Figure 2.3 Labeling guide for arcs

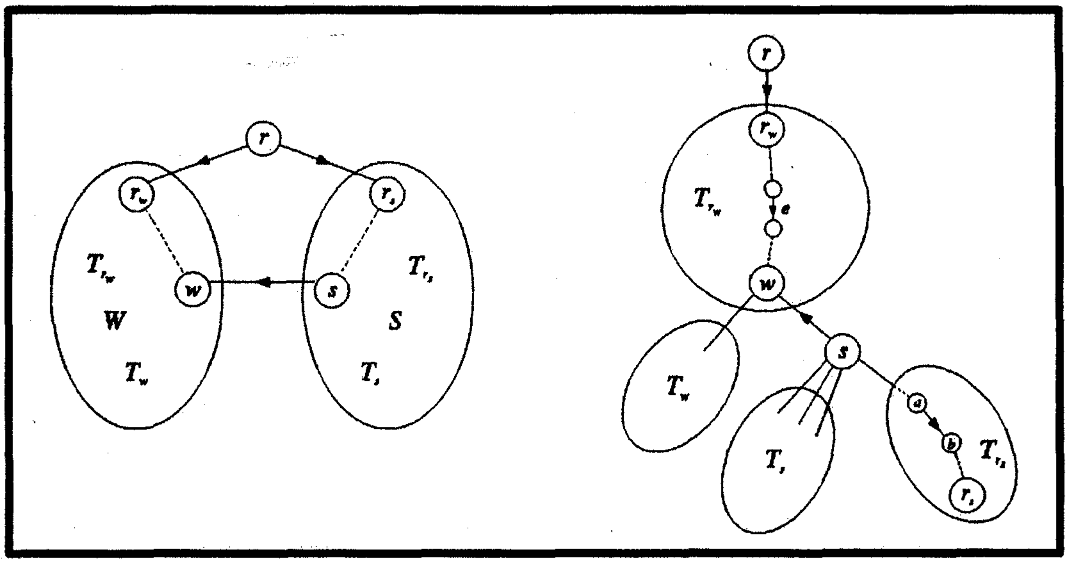


Figure 2.4 Illustration of the steps in LG algorithm (Hochbaum and Chen, 1996)

2.1.4 Neural Network (MCS/MFNN) Algorithm for Open Pit Optimization

Frimpong and Achireko (1997) introduced a set of algorithms that address the random field properties of the various parameters involved in open pit optimization. The modified conditional simulation and multilayer feed-forward error back propagation algorithm was developed to capture the stochastic behavior of ore grades and reserves on one hand and to utilize neural network architecture in the optimization process. The modified conditional simulation (MCS) algorithm involves the use of linear average subdivision technique (LAS) and best linear unbiased estimation (BLUE) to model the stochastic nature of ore reserves and grade. In this case conditional simulation is employed to model the grade using the mean and the standard deviation. The MFNN is then used for classifying and partitioning the conditionally simulated blocks. The 2-D model as presented by this algorithm does not represent reality. There is a possibility of the existence of both local and global maxima in a maximization problem. The learning algorithm can converge to a solution that does not minimize the mean square error. The back propagation algorithm can converge to a local maximum with one set of initial conditions and converge to a global maximum with another set of initial conditions. To ensure that a global optimum point has been achieved, several different initial conditions have to be evaluated. The situation may even get worse in multi dimensional problem with multiple parameters. Therefore an optimal solution may not be achieved when a back propagation algorithm is used to solve this problem (Asa, 2002).

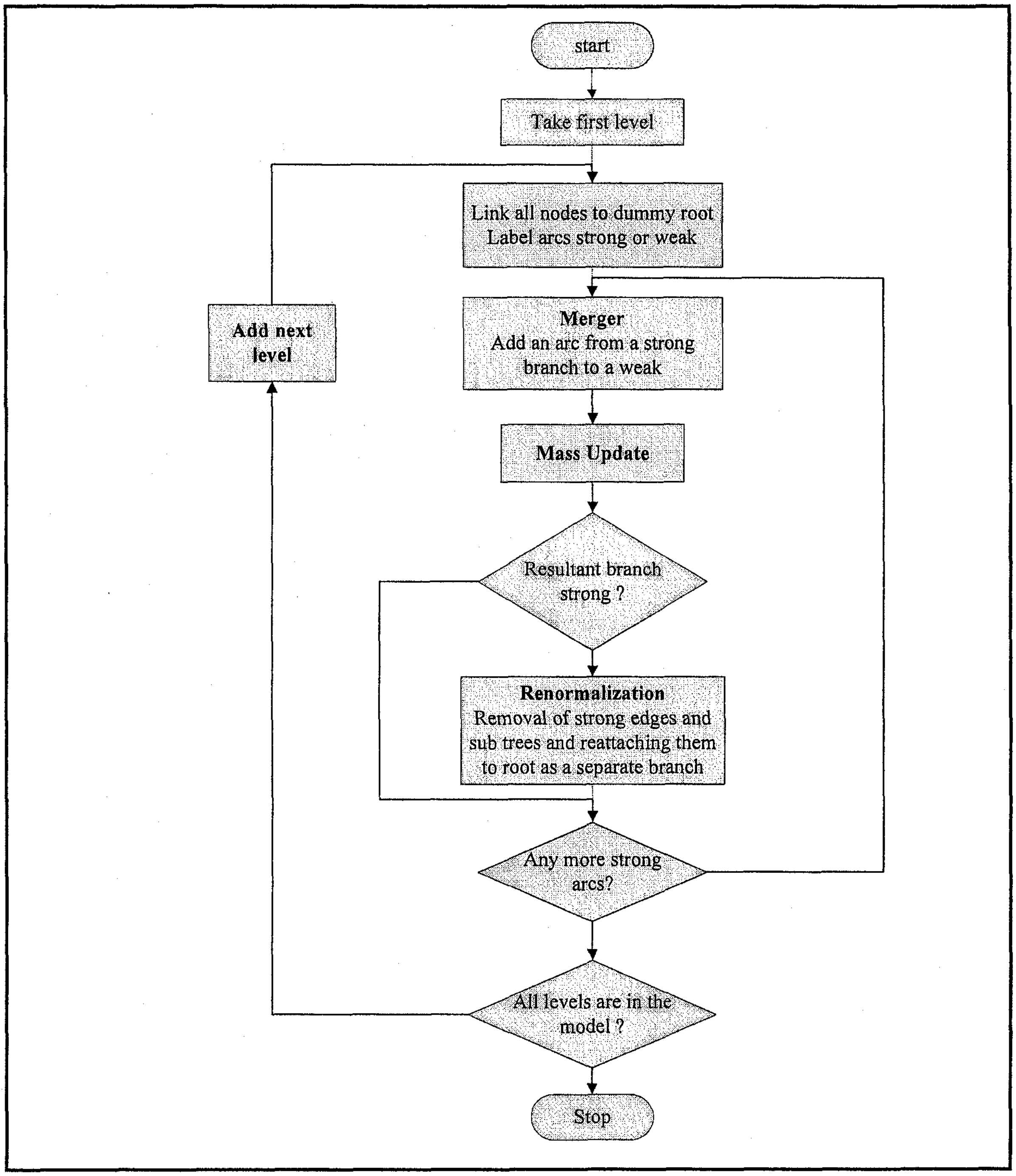

start
Take first level
Link all nodes to dummy root
Label arcs strong or weak
Add next level
Merger
Add an arc from a strong branch to a weak
Mass Update
Resultant branch strong ?
Renormalization
Removal of strong edges and sub trees and reattaching them to root as a separate branch
Any more strong arcs?
All levels are in the model ?
Stop

Figure 2.5 Lerchs and Grossmann algorithm flow chart

2.2 Production Scheduling Algorithms

Several researchers have studied and developed algorithms for long-term mine planning problem. Scheduling studies have typically been carried out for the purpose of profit maximization (Chanda and Wilke, 1992; Tolwinski and Underwood, 1992; Onur and Dowd, 1993; Dijilani and Dowd, 1994; Elveli, 1995; Frimpong et al., 1998) or for the determination of optimum plant feeding conditions (Dowd and L., 1987; Mann and Wilke, 1992; Youdi et al., 1992; Huang, 1993; Zhang et al., 1993; Chanda and Dagdelen, 1995). The algorithms use a block model as the basis for extraction scheduling to optimize an economic objective function. Initially, the excavator is located at a starting point. This excavator extracts the blocks considering either economic criteria or plant requirements. All the algorithms use slope conditions determined from the mechanical and geological properties of the rock.

In most of the profit maximization studies (Chanda and Wilke, 1992; Tolwinski and Underwood, 1992; Onur and Dowd, 1993; Dijilani and Dowd, 1994; Elveli, 1995) the block model consists of economic values which are either given to the system, or calculated from block volume, tonnage factor and grade. The aim is to sequence blocks to maximize the net present value. The general and common approach is to include scheduling in a previously determined pit boundary. This boundary is determined by the use of another optimization algorithm for open pit limit. However, as it is implied, these boundaries are not real optimum boundaries from an economic point of view, since the interest rate can never be zero, and the extraction time of a more valuable block may be significantly different from those for a less valuable block. On the other hand, algorithms that seek optimum plant feeding conditions (Dowd and L., 1987; Mann and Wilke, 1992; Youdi et al., 1992; Huang, 1993; Chanda and Dagdelen, 1995) use grade blocks to control the input to the processing plant. Generally, a stockpile is employed to provide a blend of material, if necessary, and to store a certain amount of material in the event that the mine is unable to supply the plant for a short period of time. Thus, the sequence of block extraction is dictated by the management of material stockpiles. The amount and grade of accumulated material should be adequate to provide a uniform feed for the next production phase. Accordingly, mining is directed to those blocks that have the required quality and quantity. The resulting optimization problem then becomes minimizing the difference in quality and quantity of the supplied blended material from desired feed. An application of this approach to phosphate mining was carried out by Smith (1999). The method combines mixed integer, goal and separable programming techniques to minimize quantity and quality of material transfer to plant. In the system, stockpiles are employed as regulators to prevent fluctuation in material quality. Branch and bound approach enables the system to work quicker than other similar systems. The main drawback of these approaches is the same as that identified with the profit maximization approach, namely, sequencing is optimized in only given boundaries. However, the net present value may be greater or the disparity in the quantity and quality of the supply from the required material may be less for a different pit boundary.

2.3 Haul Road Design and Incorporation

One of the first published accounts of the incorporation of roadways in optimal open-pit design was by (Taylor, 1971). Yun and Lu (1992) presented a dynamic programming approach to the optimization of roadway location within an overall expert system. Haul roads are designed to transport all the ore and waste from the bottom of the pit to the dumping point (for waste) or to the crusher (for ore) and there must, therefore, be a connection to the surface from the levels within the pit. The width of the road depends entirely on safety and mining and equipment capacities. Bench widths depend on the amount of access space that is required by equipment, but they can be altered during the life of the pit. Bench heights are function of the geology, blasting requirements, selectivity, and the existing mining equipment. Hence, they are rarely changed during the life of the pit unless there is a significant change in economic parameters that alters the selectivity criteria.

Original bench widths may not always be large enough to accommodate roads and some additional excavation will usually be necessary. The roads are specified by their slopes and widths. The slopes depend on the materials on which the roads are built, traction factors, types of trucks and weather conditions. Slopes define the distance of travel from level *n* to level *n-1*. It is assumed that an algorithm has been used to obtain an optimum pit design in the form of a block model. With this design several alternative roads can be created by starting them from different points at the bottom of the pit and changing the direction in which they are allowed to develop from clockwise to anticlockwise. The user specifies a dumping location on the surface, which may be the treatment plant, a stockpile or a loading point. The algorithm starts with a block on the pit bottom and, moving in clockwise direction, generates a ramp with the specified slope and width to the level above. A ramp is then generated in the same direction to the next level until the surface is reached. The length of the road is measured from the starting point at the pit bottom to the specified dumping location on the surface. This procedure is repeated, starting from the same block but moving in an anticlockwise direction. When the two roadways have been generated for this block the procedure is repeated for all the other blocks on the pit bottom. A removal cone is defined for each block that is added to the road and all blocks within this cone, if not already removed, are added to the pit. The computing time can be reduced by specifying a minimum increment from one starting point to the next (e.g. every second or third block on the pit bottom). The road is constructed in terms of blocks that are removed from the block model. The use of a block model prevents the algorithm from producing precise road dimensions because it is always restricted by those of the blocks: to obtain a proper width a whole block must be removed rather than the required portion of it.

The most significant advantage of the block model approach is rapid computing time. When the algorithm completes the process, the user is given a summary of all roads that have been generated together with three criteria for assessing each road. These include: (i) the number of additional blocks to be removed; (ii) the reduction in the overall profit that would result from removing these blocks; and (iii) the distance from the bottom of the pit to the dumping point. The latter is regarded as a measure of the haulage cost. All three selection criteria are then used to choose a road from the list of alternatives. Although this haul-road design method is not based on a truly optimizing algorithm, it does provide information that allows near optimal choices to be made in a very short time (Onur and Dowd, 1993).

2.4 Rationale for PhD Research

The current production scheduling methods are not just limited to, but can be divided into: (i) heuristic methods; (ii) nested pits; (iii) operations research methods; and (iv) artificial intelligence methods. The main OR methods include mixed integer programming (Gershon, 1983; Dagdelen, 1985; Ramazan and Dimitrakopoulos, 2004), dynamic programming (Dowd, 1976; Dowd and Onur, 1993), linear programming (Gershon, 1982), goal programming (Chanda and Dagdelen, 1995; Esfandiari et al., 2004), integer programming (Kim, 1990), and branch and bound methods (Caccetta and Hill, 2003). Different AI methods (Tolwinski and Underwood, 1992; Elveli, 1995; Askari-Nasab et al., 2005) with explicitly genetic algorithms (Denby and Schofield, 1994; Denby et al., 1996; Wageningen et al., 2005) have also been used by researchers to find a solution to the mine long-term scheduling problem.

In practice, the optimized solution cannot be attained without examining all possible combinations and permutations of the extraction sequence. Whittle (1989) outlined the complexity of the problem in three areas, including: (i) the pit outline with the highest value cannot be determined until the block values are known; (ii) the block values are not known until the mining sequence is determined; and (iii) the mining sequence cannot be determined unless a pit outline is available. Such an objective results in a computationally intractable problem.

Most of the aforementioned techniques guarantee the optimality of the solution by enumerating all possible combinations, with mathematical rigor for solving material transfer problems. However, the key limitations of these methods are: (i) inability to solve large industrial problems. The extraction sequence in a predetermined final pit limits has to deal with the limitation of computing resources. These limitations include: time (how many steps does it take to solve a problem), and space (how much memory does it take to solve a problem); (ii) limitation in dealing with stochastic processes governing ore reserves, commodity price, cut-off grade, and production cost. The treatment of stochastic variables as deterministic variables results in over or under-estimated layouts; (iii) inadequacy of the current final pit limits optimization methods in taking into account the temporal nature of the exploitation. The ultimate pit limits optimization algorithms assume that the final contour is dug at once without considering the time aspect of the problem; and (iv) shortcoming in defining the economics of ore with respect to the economics of the entire mining process, from ore to finished product.

These deficiencies can cause an open pit mining operation severely in terms of dollars, time, material logistics handling, and strategic and tactical plans. Mining industry, as a dynamic uncertain system with a number of underlying stochastic variables, needs optimization tools to be kept profitable. In complex mining operations, small deviations from the optimal strategic plan could result in the loss of millions of dollars.

2.5 Summary and Remarks

The literature relevant to this research has been reviewed. In summary, open pit mine planning and optimization has a great direct influence on the mining economics. Various attempts have been made to address the problem during the last 40 years. A non exhaustive list of final pit limits optimization techniques categories can be classified as: (i) rigorous; (ii) heuristic; (iii) stochastic; (iv) static; (v) dynamic; and (vi) artificial intelligence methods. Lerchs and Grossman (1965) demonstrated a groundbreaking 3D algorithm based on graph theory for optimum ultimate pit limits problem. Some of the researchers used different variants of the original algorithm using graph theory (Picard, 1976; Whittle, 1988; Zhao and Kim, 1992). Other categories of algorithms namely heuristic methods including different versions of floating or moving cone (Pana and Davey, 1965; Williams, 1970; Gauthier and Gray, 1971; Robinson and Prenn, 1973; Lemieux, 1976; Lemieux, 1979) the Korobov algorithm (Korobov, 1974), the corrected form of Korobov algorithm (Dowd and Onur, 1993), dynamic programming (Johnson and Sharp, 1971; Koenigsberg, 1982; Wilke and Wright, 1984; Yamatomi et al., 1995). Rigorous methods including network or maximum flow techniques (Johnson and Barnes, 1988; Yegulalp and Arias, 1992), transportation algorithm (Huttagosol and Cameron, 1992), network flow algorithm based on dual of Lerchs and Grossmann (Underwood and Tolwinski, 1998), and parameterization techniques (Francois-Bongarcon and Marechal, 1976; Francois-Bongarcon and Guibal, 1982). Stochastic methods including genetic algorithm (Denby and Schofield, 1994; Denby et al., 1996), modified conditional simulation/multi-layer feed forward neural networks (MCS/MFNN) (Frimpong and Achireko, 1997), simulative optimization model (Erarslan and Celebi, 2001), simulated annealing and neurogentic algorithm (Asa, 2002), and open pit optimization including mineral processing criteria using non-linear goal programming (Esfandiari et al., 2004).

The objective function in a mine scheduling problem has typically focused on revenue maximization (Chanda and Wilke, 1992; Tolwinski and Underwood, 1992; Dowd and Onur, 1993; Elveli, 1995; Frimpong et al., 1998; Erarslan and Celebi, 2001; Halatchev, 2005) or the optimized plant feeding conditions (Dowd and L., 1987; Mann and Wilke, 1992; Youdi et al., 1992; Chanda and Dagdelen, 1995). Profit maximization functions focus on generating a block sequence that maximizes the net present value. Alternatively, plant feeding quality methods use grade block models to control input to the processing plant. The quantity and quality of available materials should be adequate to provide a uniform feed to the plant for the next production period.

The major shortcomings of planning and optimization techniques can be summarized as: (i) inability to solve large industrial problems; (ii) limitation in dealing with stochastic processes governing mining operation; (iii) inadequacy in referencing the temporal nature of the exploitation and the time aspect of the problem; and (iv) shortcoming in defining the economics of ore with respect to the economics of the entire mining process, from ore to finished product. Consequently there is a need for a more comprehensive algorithm which can address the aforementioned problems. The outcome of the research is expected to contribute towards the development and implementation of an intelligent long-term planning algorithm, which will enable step-changes in the manner in which open pit mine layouts are engineered and managed.

CHAPTER 3
INTELLIGENT OPEN PIT PLANNING THEORETICAL FRAMEWORK

3.1 Foreword

This chapter focuses on the conceptual framework of intelligent open pit planning as presented in this thesis. The general theoretical framework, mathematical models and their interrelationships are developed for achieving research objectives. The research focus is on development, analysis, and implementation of '**I**ntelligent **O**pen Pit **O**ptimal **P**roduction **S**imulator' (IOPS) based on reinforcement learning (RL). The RL problem can be formalized by the interaction of two basic entities: the agent and the environment. The agent is the learner and decision-maker. The agent's environment comprises everything that it cannot completely control. Thus, the environment defines the task that the agent is seeking to learn. A third entity, the simulation, mediates the interactions between the agent and the environment. Figure 3.1 illustrates the basic components of an RL problem. The agent takes sensory input from the environment, and produces as output actions that affect it. The interaction is usually an ongoing non-terminating process.

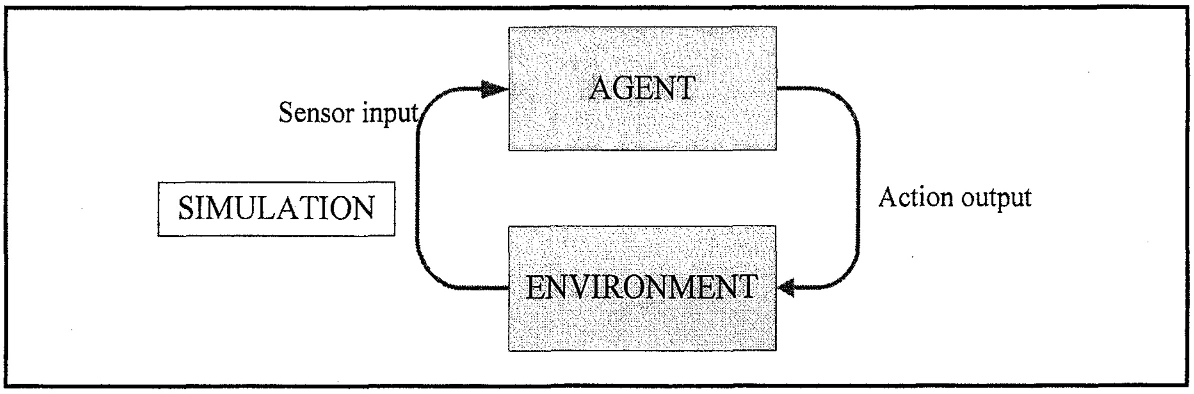
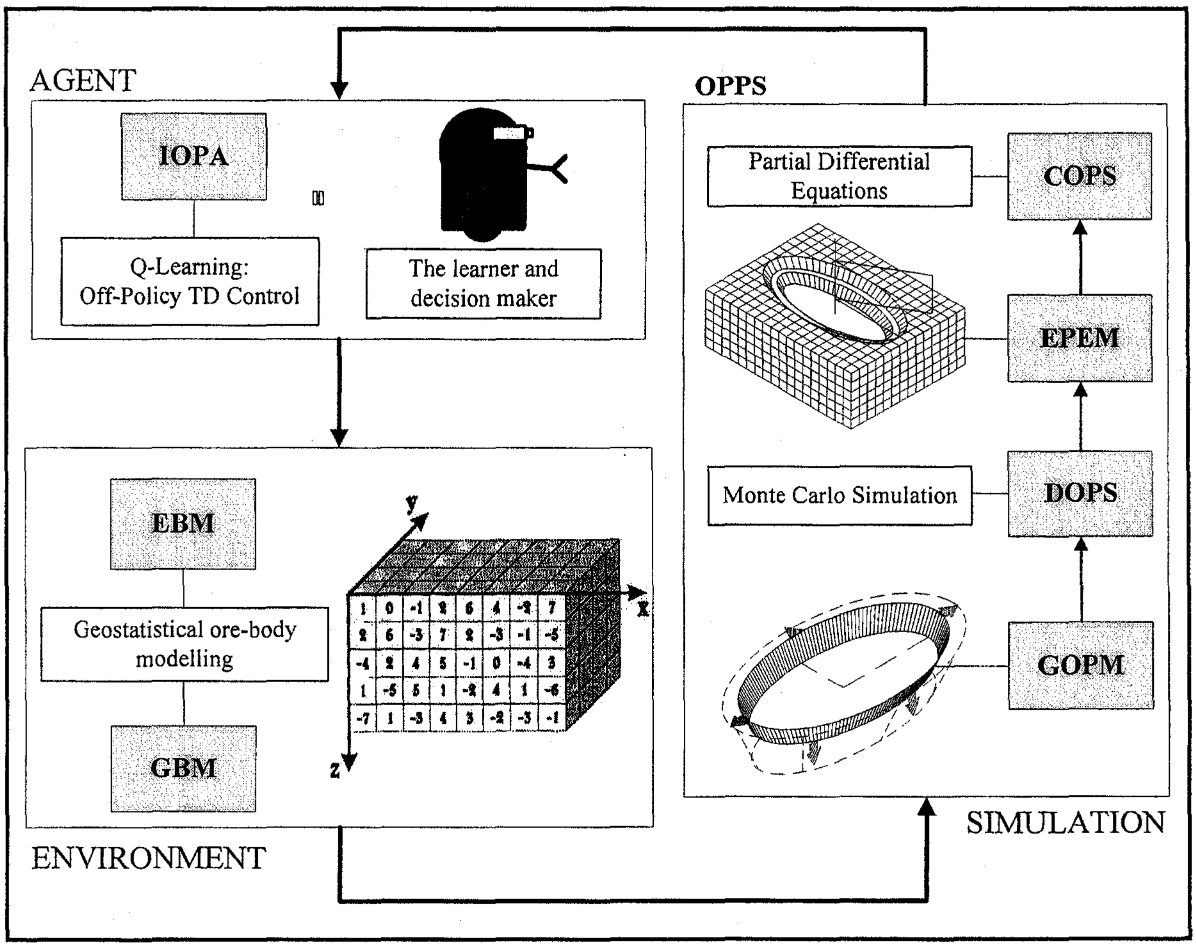


Figure 3.1 The agent-environment interaction in reinforcement learning (Sutton and Barto, 1998)



1	0	-1	2	6	4	-2	7
2	6	-3	7	2	-3	-1	-5
-4	2	4	5	-1	0	-4	3
1	-5	5	1	-2	4	1	-6
-7	1	-3	4	3	-2	-3	-1

Figure 3.2 Intelligent open pit optimal production simulator (IOPS) frame work

This chapter focuses on formulating, modeling, and developing the components of IOPS: environment, simulation, intelligent agent, and their relationships. Figure 3.2 illustrates the IOPS conceptual framework as an RL problem. IOPS comprise independent, interactive and interrelated subsystems with processes and procedures, using RL as the main engine to maximize the net present value of the mining venture. The main integral parts of IOPS are as follows: (i) environment: consists of geological block model (GBM) and economic block model (EBM); (ii) simulation: open pit production simulator (OPPS) which comprises geometrical open pit model (GOPM), discrete open pit simulation (DOPS), economic pit expansion model (EPEM), and continuous open pit simulation (COPS); (iii) agent: the output of the OPPS is the monetary value of each period of the mining operation. The simulated results are transferred to the intelligent open pit agent (IOPA) where Q-learning algorithm serves as the engine. Sections 3.2 and 3.3 discuss and develop the main RL components of the model. Section 3.5 give details about adapting open pit planning problem as an RL problem and the solution.

3.2 Environment: Geostatistical Block Model

Geostatistics, a tool for estimating spatially correlated variables is used widely in mineral reserve estimations. The variogram model, which characterizes the spatial correlation of variables, is the first and essential step for any geostatistical estimation. The variogram model is primarily dependent on number of samples or boreholes intercepts within the ore body. Figure 3.3 shows the sequence of actions in geostatistical modeling.

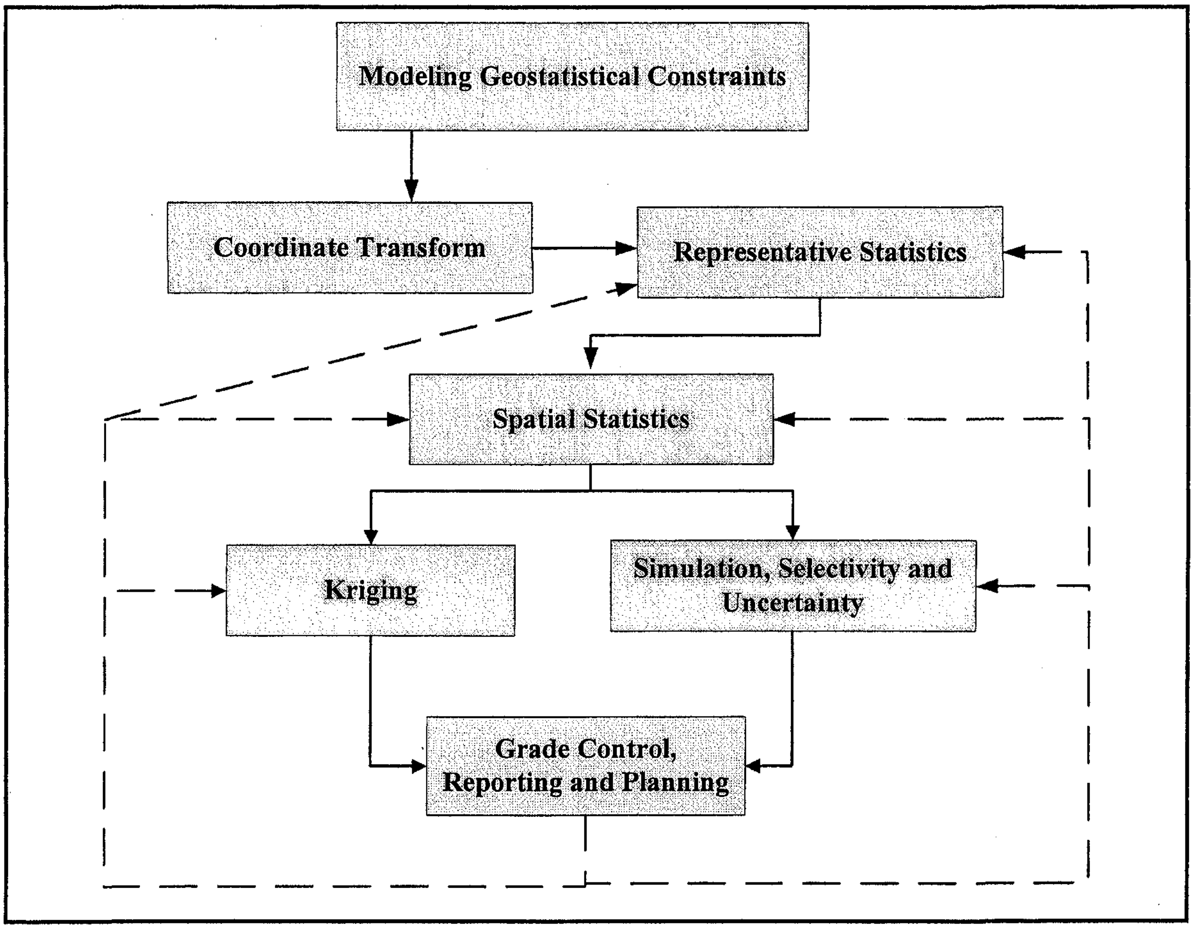

Modeling Geostatistical Constraints
Coordinate Transform
Representative Statistics
Spatial Statistics
Kriging
Simulation, Selectivity and Uncertainty
Grade Control, Reporting and Planning

Figure 3.3 Sequence of actions in geostatistical modeling

Reserve risk is an important component of the mine feasibility, optimization and operations process. The success of any optimization algorithm is dependent on estimates of reserves and grade together with the assessment of uncertainty and risk.

Geostatistical study of a mineralized zone of interest is a multiple-stage process. The first stage of an actual geostatistical study involves the modeling of the spatial variability within the study area. The variogram can be simply defined as the relationship between geological distance and Euclidean distance. The variogram quantifies the spatial variability of the contained mineral/metal by computing the variance of the grade values measured some distance h, apart. The variogram variance increases with the separation distance. Kriging techniques are estimation methods that minimize the error variance of estimation for spatially distributed data. The kriging estimation variance can be affected by errors in the calculation of the variogram. Some of the reasons for estimation errors include (i) sparse data, (ii) high variability, and (iii) incorrect assumptions in the modeling procedure.

The results of the geostatistical analysis will be the estimation of resources/reserves and classification into various categories. It is evident that the entire in-situ mineral resource identified and estimated by geostatistical analysis cannot be mined because of variable mining and treatment costs and spatial variability inherent in the deposit. In this study, sequential Gaussian simulation (Isaaks, 1990) (SGS) is used to provide reserve estimates and the associated risks. Geostatistical Software Library (GSLIB) (Deutsch and Journel, 1998; Deutsch, 2002) will be utilized to model the ore-body deposits in this study.

Briefly, the procedure of Sequential Gaussian Simulation (SGS) consists of: (i) selection of a voxel where the reserve property under investigation is unknown; (ii) identification of neighboring voxels where that property is known; (iii) assignment of weights to the neighbors, depending upon their perceived relevance at the unknown voxel; (iv) construction of a local PDF at the unknown voxel from the neighbor values; (v) extraction form the PDF of a single value to occupy the empty voxel; (vi) random selection of another empty voxel; and (vii) repetition of the process until estimations have been made at all the unknown voxels.

3.3 Simulation: Open Pit Production Simulator Components (OPPS)

An open pit production simulator, OPPS illustrated in Figure 3.2, is developed incorporating GOPM, DOPS, EPEM, and COPS models. The OPPS mimics the stochastic dynamic expansion of an open pit using discrete incremental push-backs in time and space. The interactions of economic pit expansion model, EPEM, with DOPS returns the pit's NPV following the simulated schedule. COPS models the dynamics of open pit geometry and the material movement as a continuous system described by time-dependent, differential equations. The next sections investigate the theoretical and practical framework of OPPS.

3.3.1 Geometrical Open Pit Models (GOPM)

Excavation of ore and waste in an open pit mine takes place on different elevations. The pit expands horizontally and vertically toward the final pit limits. Given an initial box cut, surface mining can be viewed as a process of pit layout deformation toward the final pit limits. Surface mine layout optimization therefore focuses on the optimal pit outline, whereas scheduling concentrates on the optimal extraction sequence to reach the final pit limits. The main objectives of long-term plans are to meet production quantity and quality targets to maximize the future market value of the venture. There is therefore a need for models of pit geometry and the evolution of the geometry as the pit is continuously deformed to reach the final pit limits. Frimpong et al. (1998) provided a basis for using the solid geometry of an elliptical frustum to model the open pit expansion process. Figure 3.4 shows the frustum which is defined by two ellipsoids with areas A_1 and A_2 separated by a vertical distance h. θ is the stable and economic overall pit slope resulted from geo-mechanical studies. Askari-Nasab et al. (2004) developed a production simulator based on the elliptical frustum model to capture the random field process associated with materials scheduling and sequencing in an open pit environment. The volume of material in the frustum is given by equations (3.1) and (3.2) as:

$$V = \frac{1}{3} \times [A_1 + A_2 + (A_1 \times A_2)^{1/2}] \times H \qquad (3.1)$$

$$V = \frac{1}{3} \times [2.\pi.H.a.b - \frac{2.H^3}{\tan^2\theta} + (2.\pi.H^2.a.b - \frac{2.H^4}{\tan^2\theta})^{1/2}] \tag{3.2}$$

2a and 2b are the length of the major and minor axes of the top ellipse. Assuming that equation (3.1) is valid and maintained during the production and any changes in production time, Δt, causes an increase equal to ΔV, to the volume of the frustum. The volume of the pit layout resulting from expansion due to incremental push-backs, $V' = V + \Delta V$, given by equation (3.3), at any time is a function of the changes in the ellipsoid radii and the incremental depth of the frustum.

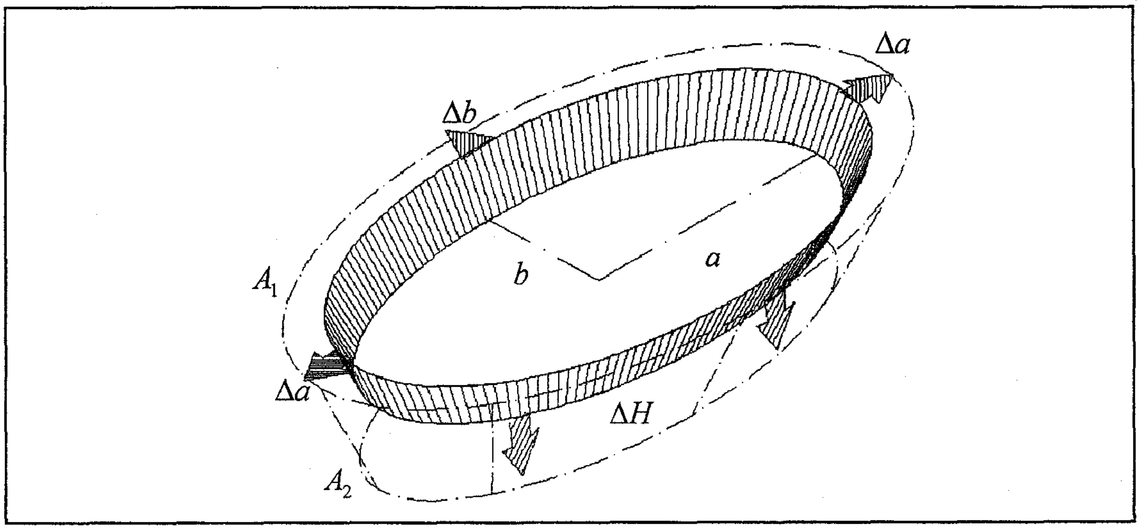


Figure 3.4 Frustum geometry of open pit

$$V' = \psi[\phi_1(a, NF_N, NE_N, A, U, L_N); \phi_2(b, NF_E, NE_E, A, U, L_E); \phi_3(\theta, H)] \tag{3.3}$$

NF_N and NF_E are respective number of concurrent active faces in the north-south and east-west directions; NE_N and NE_E are the number of loading equipment in active faces in north-south and east-west directions; A is equipment availability; U is utilization; and L_N and L_E are the length of cuts in the north-south and east-west direction ; L_{NO} and L_{EO} are the respective maximum cut lengths in the north-south and east-west directions. The amount of change in volume in equation (3.3) depends on the production planning and scheduling requirements. The goal is to design a production process to extract fractional components of equation (3.3) and to find sequential ΔV's in any changes in production time, Δt. The production process should be designed to maximize the profit and minimize the extraction costs. In equation (3.3), changes in the determinant variables like number of faces and loading equipment, will have an effect on ΔV, which could be defined only in terms of $\Delta a, \Delta b$ and ΔH. If θ is assumed to be constant over the pit, ΔV will be a function of three variables $\Delta a, \Delta b$ and ΔH, and V' can be written as equation (3.4):

$$V' = \frac{1}{3} \times [2.\pi.(H+\Delta H).(a+\Delta a).(b+\Delta b) - \frac{2.(H+\Delta H)^3}{\tan^2\theta} + \\ (2.\pi.(H+\Delta H)^2.(a+\Delta a).(b+\Delta b) - \frac{2.(H+\Delta H)^4}{\tan^2\theta})^{1/2}] \tag{3.4}$$

The simplification of reality for the purpose of building useful models initiates an adherent error in them. The assumption of modeling open pit mines with elliptical frustum causes a considerable error in volume calculations. There is hardly any open pit mine that has a completely ellipsoid cross-section. To reduce this error, a more reliable elliptical frustum model is proposed. The modified geometry consists of four concentric ellipses with various major and minor axes, as illustrated in Figure 3.5a. The modified frustum model consists of four quadrants of elliptical frustums, which are appended along the major and minor axes of the top ellipsoid as in Figure 3.5b. This figure illustrates the modified elliptical frustum, which is defined by two ellipsoids with areas A_1 and A_2 separated by a vertical distance h. Figure 3.6 illustrates the model, which divides the open pit into four sections, north-west, north-east, south-west, and south-east. Each area is defined by the major and minor axes of the respective top ellipsoid as in Table 3.1. The overall stable pit slope θ is defined for each region according to the rock slope stability and geo-mechanical studies. The volume of material in each area is given by a quarter of the volume in equation (3.1). The area of the top and bottom ellipsoids defining the frustum are given by A_1 and A_2 in equations (3.6) and (3.7).

$$A_1 = A_{NW} + A_{NE} + A_{SW} + A_{SE} \tag{3.5}$$

$$A_1 = \frac{\pi}{4} \times (a_W b_N + a_E b_N + a_W b_S + a_E b_S) \tag{3.6}$$

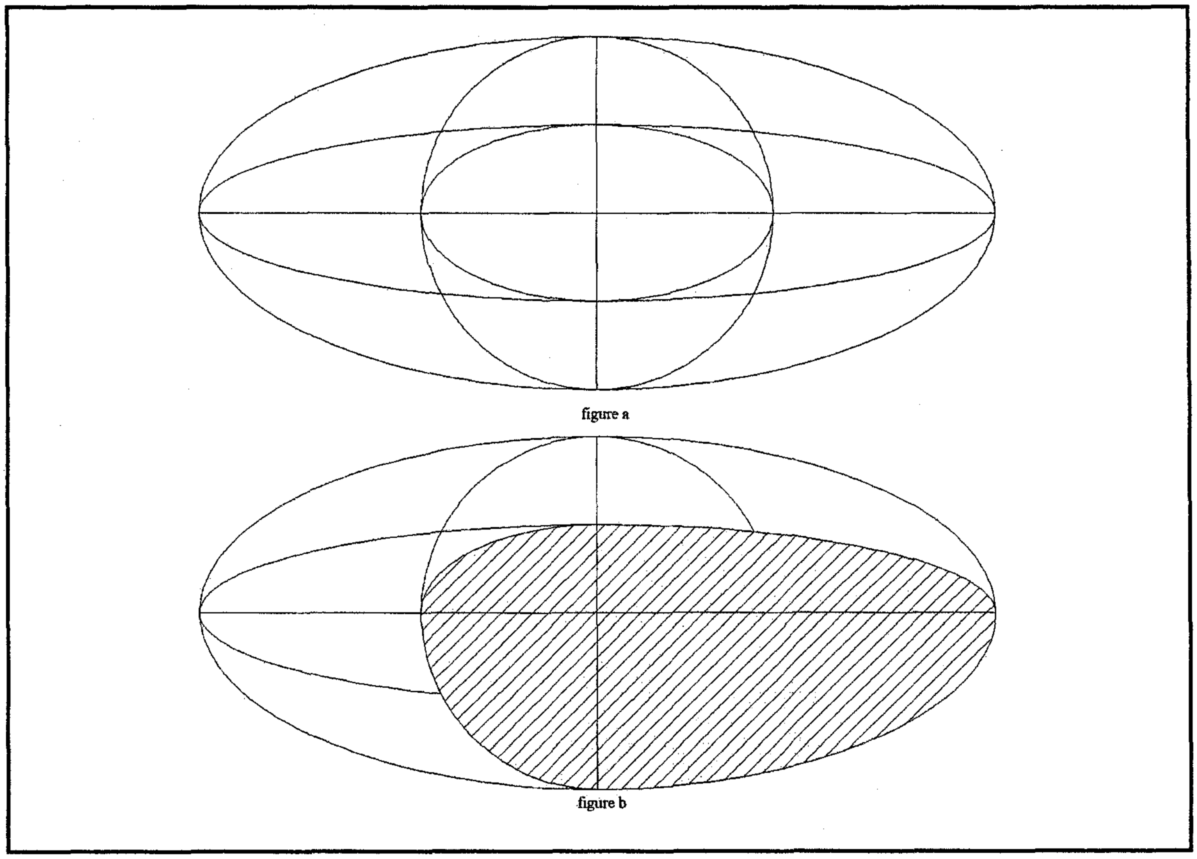


Figure 3.5 Concentric ellipses

$$A_2 = \frac{\pi}{4} \times [(a_W - \frac{h}{\tan\theta_{NW}})(b_N - \frac{h}{\tan\theta_{NW}}) + (a_E - \frac{h}{\tan\theta_{NE}})(b_N - \frac{h}{\tan\theta_{NE}}) +$$

$$(a_W - \frac{h}{\tan\theta_{SW}})(b_S - \frac{h}{\tan\theta_{SW}}) + (a_E - \frac{h}{\tan\theta_{SE}})(b_S - \frac{h}{\tan\theta_{SE}})] \qquad (3.7)$$

By substitution of A_1 and A_2 in equation (3.1) the volume of material is calculated.

Table 3.1 Variables defining frustums in each region

Region	Major Axis (m)	Minor Axis (m)	Slope (degree)	Area (m^2)
North-West	a_W	b_N	θ_{NW}	A_{NW}
North-East	a_E	b_N	θ_{NE}	A_{NE}
South-West	a_W	b_S	θ_{SW}	A_{SW}
South-East	a_E	b_S	θ_{SE}	A_{SE}

The geometry of the modified frustum is used to calculate the volume of material transferred during mining operation. Two different models are used in this thesis: (i) each open pit bench is simulated using an elliptical frustum in the discrete simulation model; and (ii) the whole open pit geometry is modeled with one modified elliptical frustum in the continuous simulation model, where the dynamics of the open pit are captured by a set of differential equations. Figure 3.6 demonstrates the model used in COPS and Figure 3.7 illustrates the model used in DOPS.

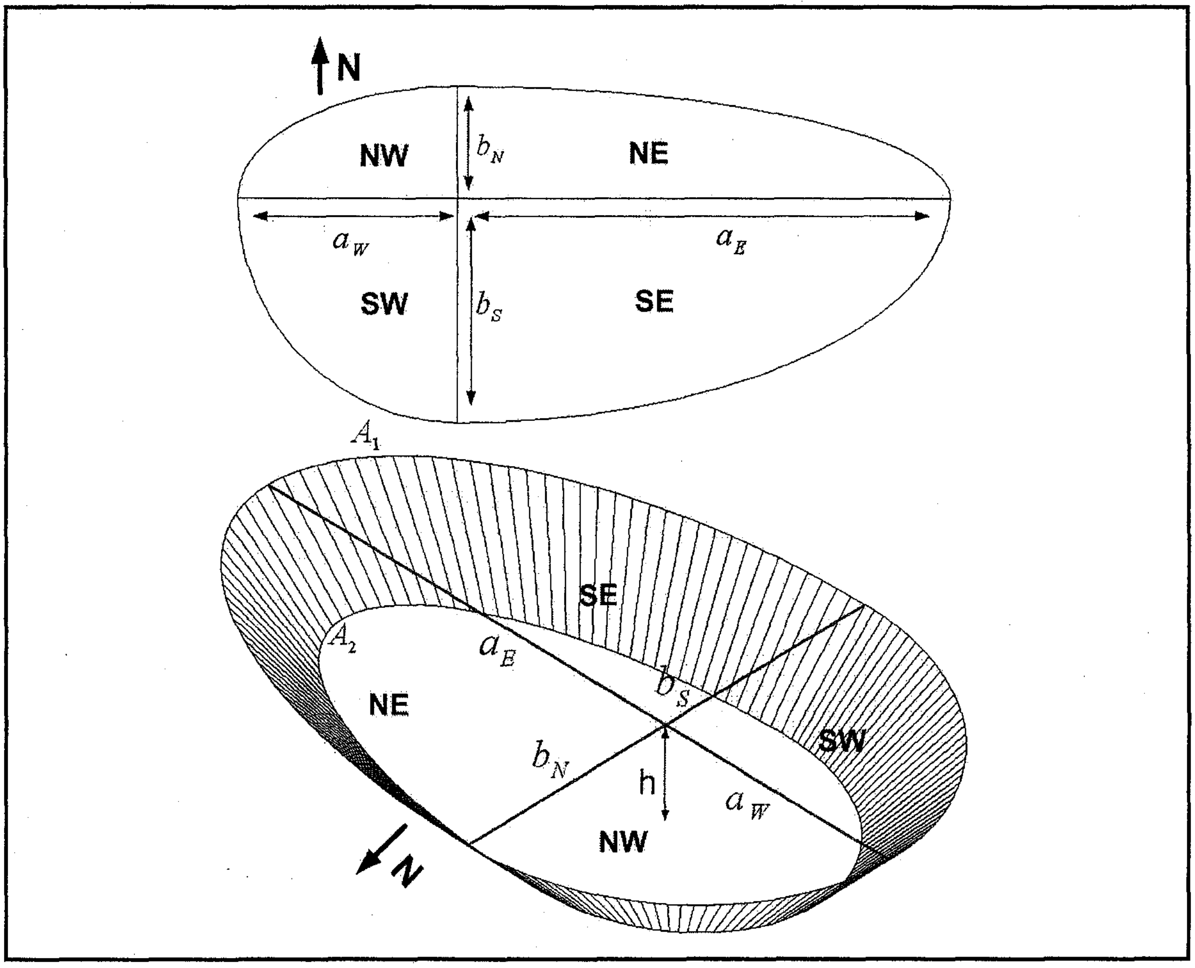


Figure 3.6 Modified geometrical frustum model

3.3.2 Dynamics of Open Pit Expansion

The mechanics of open pit layout expansion depends on the production rate, which is a function of several variables. These variables include: (i) volume of proven reserve; (ii)

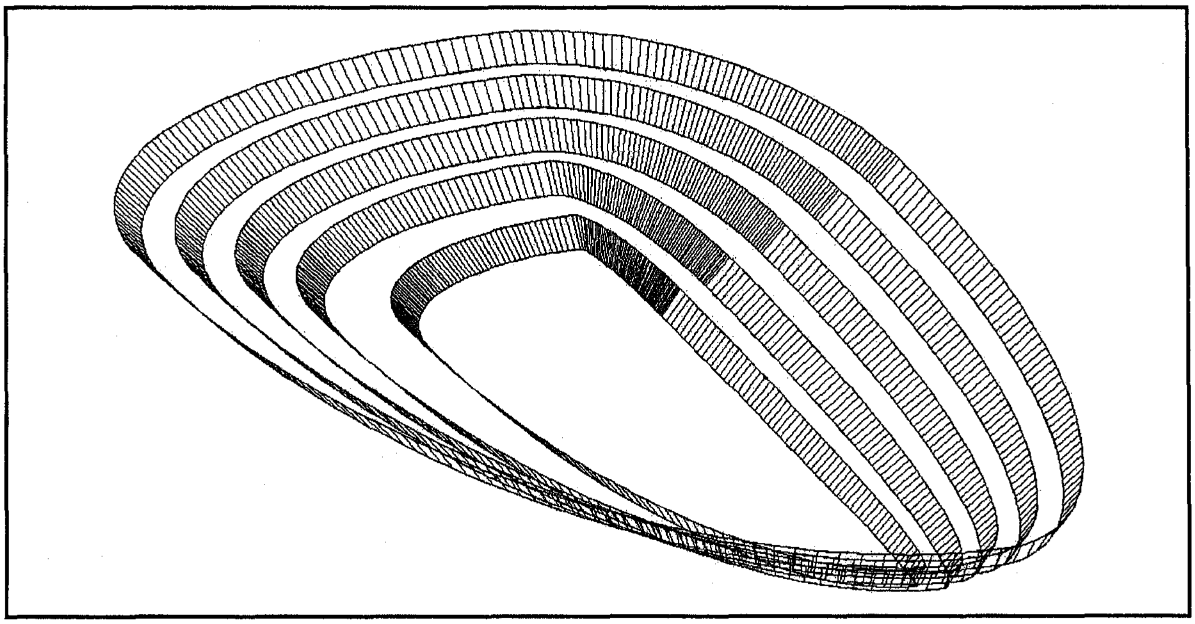


Figure 3.7 Open pit geometry, each bench is modeled by a modified frustum

reserve geometry; (iii) proposed mine life; (iv) capacity of loading and hauling fleet; (v) number of active loading equipments; (vi) number of active working faces; (vii) milling and processing plant capacity; (viii) mill head grade; and (ix) site and working conditions. To ensure optimal plant operations it needs a uniform feed during the life of mine. Stable annual ore production within an acceptable range is required to achieve this objective. The dynamics of the open pit expansion are captured by two different methods (i) discrete modeling as a guide for strategic mine plans and (ii) continuous modeling as a guide for tactical mine plans.

3.3.2.1 Discrete Open Pit Expansion Model

Discrete modeling involves the modeling of discrete functions, which represent a finite number of attributes of a system in a finite number of states, typically described by algebraic equations. In this study, the pit geometry evolution is viewed as series of snapshots over period of time, $i = 1,...,n$. Each of these snapshots represents the state of the open pit at a particular time. The pit expansion process is initiated by a box cut in the form of a frustum, $i = 1$, as in Figure 3.8. The pit expands in all directions toward reaching the next state of the open pit, $i+1$. Each pushback represents a periodic increment of production in strategic mine plans. The annual production capacity, P_c, is a function of the fleet size, F_s; fleet capacity, F_c; equipment availability, A; utilization, U; cycle time, C_T; and the loose density of materials, γ_l. The annual production capacity given by equation (3.8) is constrained by a minimum, $P_{\min}$, and maximum, $P_{\max}$, annual production targets. For practical purposes, the variability between these two constraints must be controlled significantly to ensure a steady flow of materials to meet the quantity and quality requirements of the processing plant.

The volume of material, $V_{i+1} - V_i$, given by equation (3.9), results from pit expansion from i^{th} to $(i+1)^{th}$ production period. The volume must be equal or fall within the acceptable production range. The volume of material moved in each pushback is a function of production capacity and changes of the ellipsoid radii and the geometry of the layout. The volume of material in the pit at the i^{th} period is a function of the variables in equation (3.9). The sequential discrete open pit expansion model is the theoretical framework for capturing the long-term mine plane as an RL problem. The problem is viewed as a series of open pit geometry states in time, where the intelligent agent attempts to schedule the operation to maximize the NPV.

$$P_c = \phi_1(F_s, F_C, A, U, C_T, \gamma_l)\,; \;\; P_{\min} \leq P_C \leq P_{\max} \tag{3.8}$$

$$V_{i+1} - V_i = \psi(\,\phi_1(F_s, F_C, A, U, C_T, \gamma_l)\,; \phi_2(a_W, a_E, b_N, b_S, h, \theta_{NW}, \theta_{NE}, \theta_{SW}, \theta_{SE})) \tag{3.9}$$

The discrete open pit expansion model is the basis for DOPS. There are many different scenarios of push-back designs, which deplete the ore body to the final pit limits. The goal is to find the sequence of extraction, which maximizes the net present value, subject to all the underlying constraints and satisfies the management objectives. The Monte Carlo simulation technique is used in DOPS to randomly generate the discrete changes of Δa_W, Δa_E, Δb_N, Δb_S, and Δh, which will satisfy the annual production targets of $P_{\min} \leq P_C \leq P_{\max}$. The simulation results are a series of equally probable realizations of the long-term mine plan and the respective changes in Δa_W, Δa_E, Δb_N, Δb_S, and Δh over time.

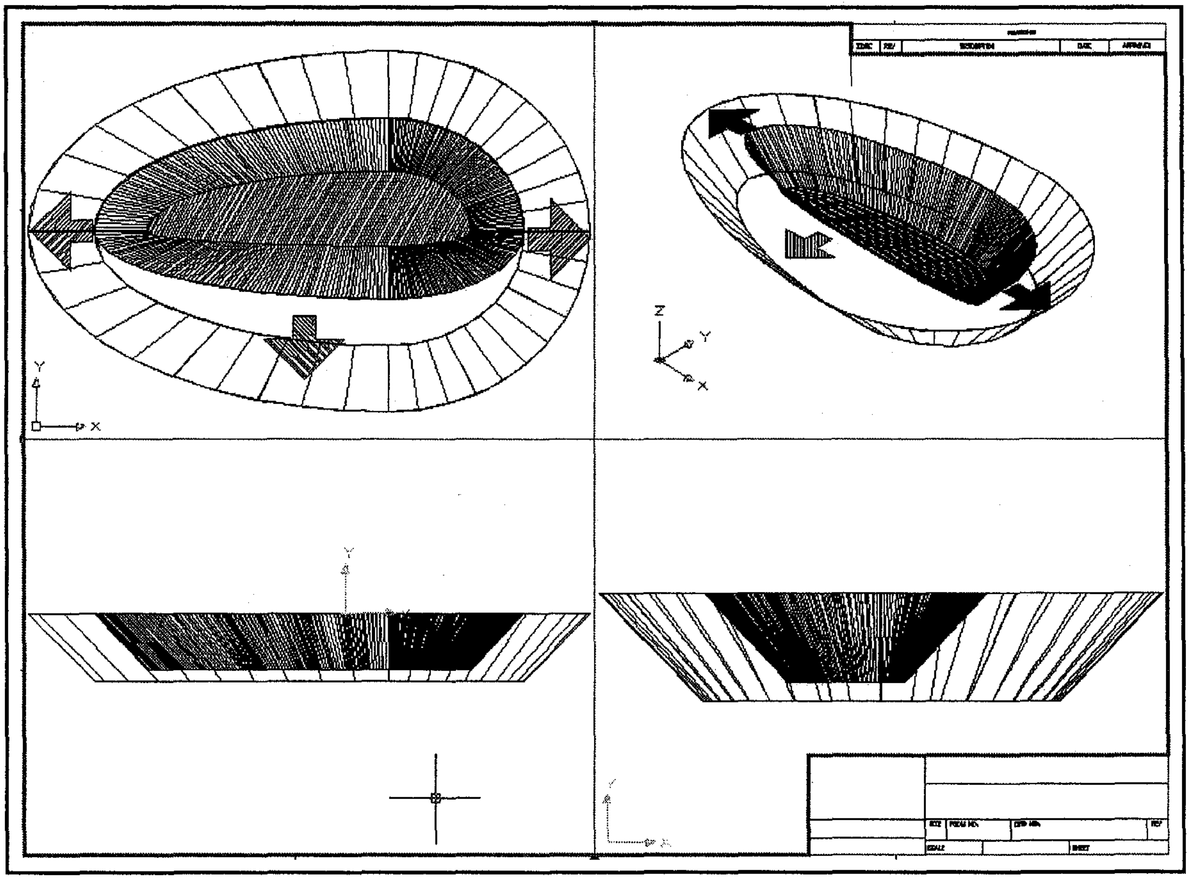


Figure 3.8 Discrete open pit expansion model

The Monte Carlo simulation technique estimates the value of a complex function by statistical sampling from a cumulative density function(s) of the input random variable(s). The design, efficiency and convergence characteristics of a Monte Carlo simulation experiment depend on the choice of variance reduction technique, random variate generation method and the input sample size (Narsing, 1997). A relationship exists between the sample size and the variance reduction technique. Even though the variance can be stabilized by choosing an appropriately large sample size, choosing a good variance reduction technique can reduce the sample size required to achieve the same results. The trade-off is between the cost to develop and program a variance reduction technique and the computational time associated with large sample sizes (Smith and Wagner, 2003).

The main components of a typical Monte Carlo simulation are: (i) a set of PDF's to represent the physical or mathematical system; (ii) a uniformly distributed set of random numbers on the unit interval; (iii) some sampling rule utilizing the set of random numbers and the PDF's; (iv) the outcome of the quantities; (v) determination of the statistical error (variance) as a function of the number of trails and other quantities; (vi) the use of a variance reduction technique (for reduction of the variance in the estimated solution) to reduce the computational time of the Monte Carlo simulation; and (vii) the use of parallelization or vectorization algorithms to enable efficient implementation.

3.3.2.2 Continuous Open Pit Expansion Model

Continuous modeling of open pit dynamics involves the characterization of the changes in the geometry of the open pit and the volume of materials moved by a set of differential equations. In general, the model does not have analytic solution and requires approximate solution methods such as finite elements, finite differences or boundary elements. Discretization of the open pit dynamics focuses on the process of transferring the continuous models and equations into discrete components. Typically discretization involves splitting the region of interest into a set of small elements, producing a discrete approximation of the differential equations in each element, and solving all of the discrete approximations simultaneously. The volume of material transferred throughout the pit expansion process, in equation (3.1), can be represented as the differential changes in the volume of material transferred by equation (3.4) for the original elliptical frustum model, and by equation (3.9) for the modified elliptical frustum model. The differential changes in volume of material transferred, ΔV, is a function of ϕ_1 and ϕ_2 in equation (3.3) for the original elliptical frustum model and in equation (3.9) for the modified elliptical frustum model. ϕ_1 is a function of F_s, F_c, A, U, C_T, and γ_l. The other variable ϕ_2 is a function of frustum variables, $a_W, a_E, b_N, b_S, h, \theta_{NW}, \theta_{NE}, \theta_{SW}$, and θ_{SE}.

First, the frame work for the continuous model was developed based on the original elliptical frustum with three variables. Considering equations (3.3) and (3.4), a Taylor series expansion of function $V(a,b,H)$ in three variables is given by equation (3.10):

$$V(a+\Delta a,b+\Delta b,H+VH)=V(a,b,H)+\frac{\partial V}{\partial a}\Delta a+\frac{\partial V}{\partial b}\Delta b+\frac{\partial V}{\partial H}\Delta H+\frac{1}{2}\frac{\partial^2 V}{\partial a^2}\Delta a^2+\frac{1}{2}\frac{\partial^2 V}{\partial b^2}\Delta b^2+$$
$$\frac{1}{2}\frac{\partial^2 V}{\partial H^2}\Delta H^2+\frac{\partial^2 V}{\partial a\partial b}\Delta a\Delta b+\frac{\partial^2 V}{\partial a\partial H}\Delta a\Delta H+\frac{\partial^2 V}{\partial b\partial H}\Delta b\Delta H+O(h^3) \qquad (3.10)$$

Since the assumption of modeling open pit mines with the original elliptical frustum model given by equation (3.2) caused a considerable error in volume calculations, the modified elliptical frustum represented in section 3.3.1 is used for open pit geometry modeling. Subsequently, the modified geometrical open pit model was developed as a function with five variables. At this stage the goal is to design a production process to make infinitesimal changes in equations (3.9) with respect to changes in dependent variables of the function ϕ_2. The result will be sequential $\Delta V's$ with respect to any changes in production time, Δt. The overall slopes in different regions $\theta_{NW},\theta_{NE},\theta_{SW},\theta_{SE}$ are assumed to be constant over each region. The changes in parameters of the function $\phi_1(F_s,F_C,A,U,C_T,\gamma_l)$, will have a direct effect on ΔV, which are defined in terms of $\Delta a_W,\Delta a_E,\Delta b_N,\Delta b_S$ and Δh. Thus, V will be a function of five parameters a_W,a_E,b_N,b_S and h. It is assumed that $V(a_W,a_{E,}b_N,b_S,h)$ is an analytic function, so that all of its derivatives exist. As a result, V can be represented by its corresponding Taylor series expansion in five variables. The generalized Taylor series expansion for a real function in n variables is given by Abramowitz (1972) in equation (3.11).

$$f(x_1+a_1,\ldots,x_n+a_n)=\sum_{j=0}^{\infty}\left\{\frac{1}{j!}\left(\sum_{k=1}^{n}a_k\frac{\partial}{\partial x'_k}\right)^j f(x'_1,\ldots,x'_n)\right\}_{x'_1=x_1,\ldots,x'_n=x_n} \qquad (3.11)$$

Accordingly, a Taylor series expansion of function $V(a_W,a_{E,}b_N,b_S,h)$ in five variables is given by equation (3.12) as:

$$V(a_W+\Delta a_W,a_E+\Delta a_E,b_N+\Delta b_N,b_S+\Delta b_S,h+\Delta h)= \qquad (3.12)$$
$$\sum_{j=0}^{\infty}\left\{\frac{1}{j!}\left(\Delta a_W\frac{\partial}{\partial x'_1}+\Delta a_E\frac{\partial}{\partial x'_2}+\Delta b_N\frac{\partial}{\partial x'_3}+\Delta b_S\frac{\partial}{\partial x'_4}+\Delta h\frac{\partial}{\partial x'_5}\right)^j f(x'_1,x'_2,x'_3,x'_4,x'_5)\right\}_{x'_1=a_W,x'_2=a_E,x'_3=b_N,x'_4=b_S,x'_5=h}$$

Using the second order approximation in equation (3.12) will yield to equation (3.13).

$$
\begin{aligned}
&V(a_W+\Delta a_W, a_E+\Delta a_E, b_N+\Delta b_N, b_S+\Delta b_S, h+\Delta h)= \\
&V(a_W,a_E,b_N,b_S,h)+\frac{\partial V}{\partial a_W}\Delta a_W+\frac{\partial V}{\partial a_E}\Delta a_E+\frac{\partial V}{\partial b_N}\Delta b_N+\frac{\partial V}{\partial b_S}\Delta b_S+\frac{\partial V}{\partial h}\Delta h+\frac{1}{2}\frac{\partial^2 V}{\partial a_W^2}(\Delta a_W)^2 \\
&+\frac{1}{2}\frac{\partial^2 V}{\partial a_E^2}(\Delta a_E)^2+\frac{1}{2}\frac{\partial^2 V}{\partial b_N^2}(\Delta b_N)^2+\frac{1}{2}\frac{\partial^2 V}{\partial b_S^2}(\Delta b_S)^2+\frac{1}{2}\frac{\partial^2 V}{\partial h^2}(\Delta h)^2+\frac{\partial^2 V}{\partial a_E \partial a_W}\Delta a_E \Delta a_W \\
&+\frac{\partial^2 V}{\partial a_E \partial b_N}\Delta a_E \Delta b_N+\frac{\partial^2 V}{\partial a_E \partial b_S}\Delta a_E \Delta b_S+\frac{\partial^2 V}{\partial a_E \partial h}\Delta a_E \Delta h+\frac{\partial^2 V}{\partial a_W \partial b_N}\Delta a_W \Delta b_N+\frac{\partial^2 V}{\partial a_W \partial b_S}\Delta a_W \Delta b_S \\
&+\frac{\partial^2 V}{\partial a_W \partial h}\Delta a_W \Delta h+\frac{\partial^2 V}{\partial b_N \partial b_S}\Delta b_N \Delta b_S+\frac{\partial^2 V}{\partial b_N \partial h}\Delta b_N \Delta h+\frac{\partial^2 V}{\partial b_S \partial h}\Delta b_S \Delta h+O(h^3) \qquad (3.13)
\end{aligned}
$$

According to equation (3.13), the changes in volume at any specific period of time, Δt, could be captured as a set of partial differential equations. The boundary conditions underlying equation (3.13) are the initial box cut and final pit limits dimensions. The first and second orders and the cross term partial derivatives in equation (3.13) are derived from equation (3.1) and presented in Appendix A. There is the need for functions of time representing $\Delta a_W, \Delta a_E, \Delta b_N, \Delta b_S$, and Δh. Functional approximation of the simulation results of DOPS is used to generate the additional conditions required to convert the set of PDEs to a system of ODEs. DOPS returns the feasible mining increments $\Delta a_W, \Delta a_E, \Delta b_S, \Delta b_N$ and Δh at the end of each period of production. As a result there will be annual tabular data for the increments over the life of mine. Trend analysis and curve fitting techniques are employed to approximate the periodic functions for increments $\Delta a_W, \Delta a_E, \Delta b_S, \Delta b_N$ and Δh over the mine life. The increments are represented as functions of time. Substitution of the approximated incremental functions of $\Delta a_W, \Delta a_E, \Delta b_S, \Delta b_N$ and Δh in equation (3.13) and substituting the results of differentiation of partial derivatives represented in Appendix A in equation (3.13), converts the set of PDEs into a system of ODEs with time as the only independent variable. Thus, the changes in the volume of the frustum, as well as, the changes in depth, major, and minor axes of the frustum will be represented as a function of time in the form of equations (3.14) to (3.19)

$$\frac{d\underline{V}}{dt} = f_1(\underline{V},t); \qquad V(t_0)=V_0,\ V(t_n)=V_{pit\_\lim it} \tag{3.14}$$

$$\frac{d\underline{a_W}}{dt} = f_2(\underline{a_W},t); \quad a_W(t_0)=a_{W0},\ a_W(t_n)=a_{W-\max} \tag{3.15}$$

$$\frac{d\underline{a_E}}{dt} = f_3(\underline{a_E},t); \quad a_E(t_0)=a_{E0},\ a_E(t_n)=a_{E-\max} \tag{3.16}$$

$$\frac{d\underline{b_S}}{dt} = f_4(\underline{b_S},t); \quad b_S(t_0)=b_{S0},\ b_S(t_n)=b_{S-\max} \tag{3.17}$$

$$\frac{d\underline{b_N}}{dt} = f_5(\underline{b_N},t); \quad b_N(t_0)=b_{N0},\ b_N(t_n)=b_{N-\max} \tag{3.18}$$

$$\frac{d\underline{h}}{dt} = f_6(\underline{h},t); \qquad h(t_0)=h,\ h(t_n)=h_{\max} \tag{3.19}$$

The set of equations (3.14) to (3.19) with the respective boundary conditions can be illustrated as equation (3.20):

$$\frac{d\underline{U}}{dt} = \underline{\underline{A}}.\underline{U} + \underline{B} \tag{3.20}$$

Where:

$$\underline{U} = \begin{bmatrix} \underline{V} \\ \underline{a_W} \\ \underline{a_E} \\ \underline{b_N} \\ \underline{b_S} \\ \underline{h} \end{bmatrix} \tag{3.21}$$

$$\frac{d\underline{U}}{dt} = \begin{bmatrix} \frac{d\underline{V}}{dt} \\ \frac{d\underline{a_W}}{dt} \\ \frac{d\underline{a_E}}{dt} \\ \frac{d\underline{b_S}}{dt} \\ \frac{d\underline{b_N}}{dt} \\ \frac{d\underline{h}}{dt} \end{bmatrix} \qquad (3.22)$$

The system of ODEs illustrated by equation (3.20) is integrated over time by numerical methods to capture the behavior of the pit shell expansion and it will result in a practical guide for the short-term production plan and the trajectories of a_W, a_E, b_N, b_S and h .

The first step in investigating the dynamics of the continuous-time system described by the system of ODEs by equation (3.20) is integration to obtain trajectories. Since most ordinary differential equations are not soluble analytically, numerical integration is the choice to obtain information about the trajectory. Many different methods have been proposed and used to solve accurately various types of ordinary differential equations. However, only a handful, such as the Adams-Bashforth-Moulton, Backward Differentiation Formulae and Runge-Kutta (R-K), are known and used universally (Henrici, 1962; Gear, 1971; Lambert, 1973; Stetter, 1973; Hall and Watt, 1976; Butcher, 1987). All these methods discretize the differential system to produce a difference equation or map. The objective of numerical methods are similar; they aim at corresponding the dynamics of the map to the dynamics of the differential equation (Cartwright and Piro, 1992). The R-K methods are arguably the most well known and used methods for numerical integration (Dekker and Verwer, 1984; Hairer et al., 1989; Saatdjian, 2000; Epperson, 2002; Chapra and Canale, 2006).

Stability horizons for R-K methods are known for linear and nonlinear ODEs. Therefore, if used with the right step-sizes, the R-K methods produce consistent and stable solutions and are convergent. The R-K algorithm used in this work is built on a single-step, explicit R-K (4, 5) embedded algorithm.

MATLAB uses explicit Runge-Kutta codes in its ode45 suite, which are used in integrating equation (3.20) in COPS. The solution of the continuous system returns the trajectories of changes in major, minor and depth of the frustum as well as the volume of materials transferred. The solution is passed to the economic expansion model. The scheme will return the blocks in the push-back design, net present value of the simulated schedule, and the volume of ore and waste.

3.3.3 Economic Pit Expansion Model (EPEM)

A useful planning model must be able to relate the dynamics of the open pit with the geological and economic block model. Such a model must yield the amount of ore, stockpile, and waste material moved on a bench-by-bench basis for each period of the mining operation. It must also report sufficient information about the ore grade, stockpile material, contaminant elements, and economics of the pushback design. The EPEM fits the expanding frustum on the economic and geologic block model and returns the pit present value, average ore grade, waste, and stockpile material at any desired period of production. Figure 3.9 illustrates the EPEM and it fits the center of the top ellipse of the frustum on the excavation starting point. The procedure starts searching the economic block model level by level. In each level, the distance between the center of the ellipse and the current block denoted by equation (3.24), is compared to the length of the radii of the ellipse in equation (3.29). Then a decision is made as to whether or not block is inside the frustum, as in Figure 3.9. The EPEM records the ore, waste, and stockpile material as well as the economic value for that period of mining. In the following formulas: $C(x_0, y_0)$ = center of the ellipse; (x_0, y_0) = starting point of extraction; α_{ij} = the angle between the center of the ellipse, $C(x_0, y_0)$ and block (i,j) as in equation (3.23); k=0,1,...,n-1 number of levels in the block model; and h = bench height. d = the distance between the center of each block and the center of the ellipse as in equation

(3.24); r = the distance between the center of the ellipse to the perimeter of it in the direction of *d* as in equation (3.29).

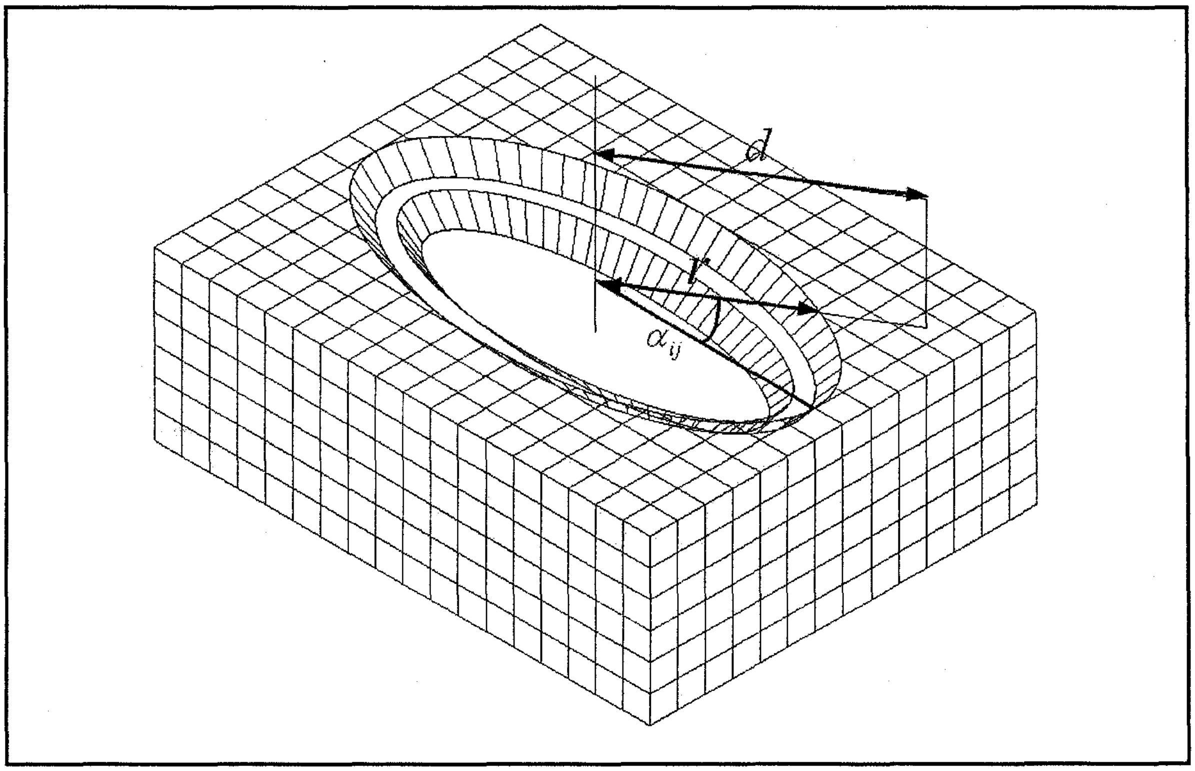


Figure 3.9 Economic pit expansion model

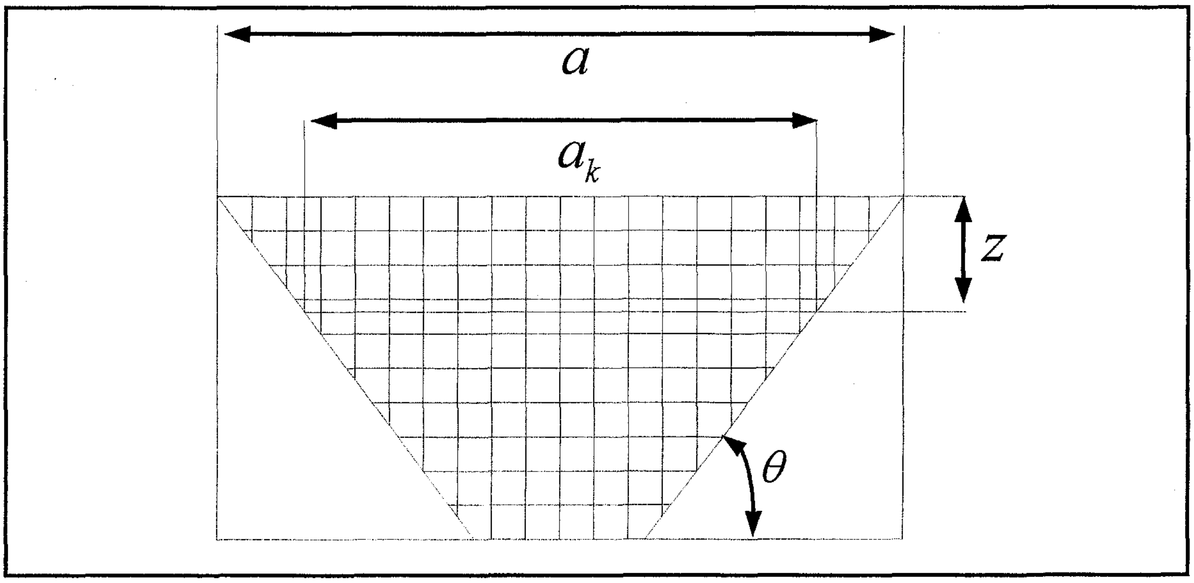


Figure 3.10 Economic pit expansion model, cross section view along frustum major axis

$$\alpha_{ij} = \arctan(\frac{|y_{ij} - y_0|}{|x_{ij} - x_0|}) \tag{3.23}$$

$$d = \left[(x_{ij} - x_0)^2 + (y_{ij} - y_0)^2\right]^{1/2} \tag{3.24}$$

The equation of the ellipse is as follows:

$$\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1 \tag{3.25}$$

In polar coordinates, the angle α_{ij} in equation (3.23) is called the eccentric angle. Substituting the polar equations into Cartesian coordinates in equation (3.25) and solving for r will yield in equation (3.29).

$$z = k \times h + (h/2) \tag{3.26}$$

$$a_k = a - \frac{z}{\tan\theta} \tag{3.27}$$

$$b_k = b - \frac{z}{\tan\theta} \tag{3.28}$$

$$r = (\frac{b_k{}^2 . a_k^2}{b_k{}^2 . \cos^2\alpha_{ij} + a_k^2 . \sin^2\alpha_{ij}})^{1/2} \tag{3.29}$$

The procedure compares ***r*** from equation (3.29) with ***d*** from equation (3.24) to decide if the block is in the current pit or not. The procedure finally returns the volume.of ore, waste, stockpile material, and their respective grades, and the economic value of the suggested pushback.

The primary implementation of EPEM in MATLAB demonstrated a reasonable run time for block models representing medium size deposits. The increase in the number of blocks in the model causes a longer search time for EPEM. Therefore, there is need for spatial data structures (Samet, 1990; Samet, 1990) to enhance the algorithm efficiency.

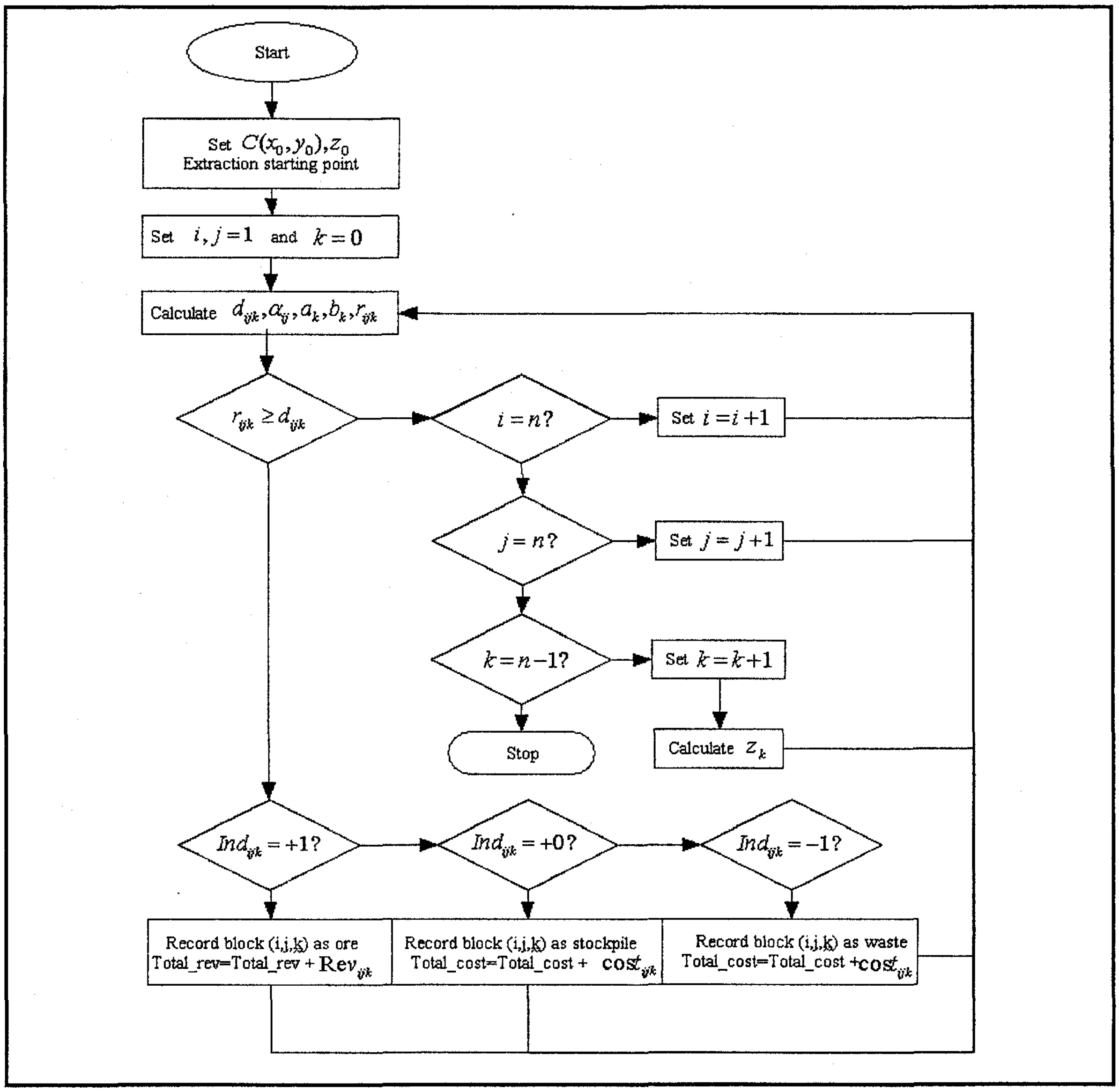

Start
Set $C(x_0,y_0),z_0$
Extraction starting point
Set $i,j=1$ and $k=0$
Calculate $d_{ijk},\alpha_{ij},a_k,b_k,r_{ijk}$
$r_{ijk}\geq d_{ijk}$
$i=n$?
Set $i=i+1$
$j=n$?
Set $j=j+1$
$k=n-1$?
Set $k=k+1$
Stop
Calculate z_k
$Ind_{ijk}=+1$?
$Ind_{ijk}=+0$?
$Ind_{ijk}=-1$?
Record block (i,j,k) as ore
Total_rev=Total_rev + Rev_{ijk}
Record block (i,j,k) as stockpile
Total_cost=Total_cost + $cost_{ijk}$
Record block (i,j,k) as waste
Total_cost=Total_cost +$cost_{ijk}$

Figure 3.11 Economic pit expansion flowchart

3.4 Agent: Intelligent Open Pit Agent (IOPA) Structure

The aim of this section is to give an understanding of how IOPA interacts with other entities and processes of the RL conceptual model, which characterizes the IOPS framework. The process of developing and building the IOPA architecture based on mathematically idealized forms of the RL problem are investigated and adapted from Wooldridge and Jennings (1995), Sutton and Barto (1998), and Wooldridge (2002).

Wooldridge and Jennings (1995) defined an agent as a computer system that is positioned in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives. Figure 3.1 gives an abstract view of an agent. In most domains of reasonable complexity, an agent will not have complete control over its environment. The agent have available a range of actions that can be executed to modify the environment, which may appear to respond non-deterministically to the execution of these actions.

The reinforcement learning problem is meant to be a straightforward framing of the problem of learning from interaction to achieve a goal. In this study the agent, IOPA is the learner and decision-maker. The phenomenon it interacts with, comprising everything outside it, is called the environment, including the economic block model and OPPS. These models interact continually as IOPA selects actions, which are defined in terms of the changes in the push-back parameters and as the result, changes in the pit geometry. The simulation and the environment respond to those actions and present new possible geometries of the open pit to the IOPA. The environment also responds with rewards, which is the cash flow of each simulated production period. The primary goal of IOPA is to maximize the NPV of the venture overtime. This means maximizing not only the immediate reward, which is the cash flow of the next production period, but also the cumulative reward in the long run, which is the NPV.

Figure 3.12 illustrates the IOPA architecture. The pit geometry evolution is viewed as a series of snapshots over of time. The IOPA and the environment interact at each sequence of discrete time steps, $t = 1, \ldots, n$. The simulation of the environment starts with the initial box cut, and the agent starts by choosing an action to perform on that state. As a result of this action, the simulation and the environment can respond with a number of possible states. However, only one state will actually result. On the basis of this second state of the environment, the agent again chooses an action to perform. The environment responds with one of a set of possible actions available, afterward the agent chooses another action. This process is repeated until the simulation of the open pit layouts reaches the optimized open pit limits.

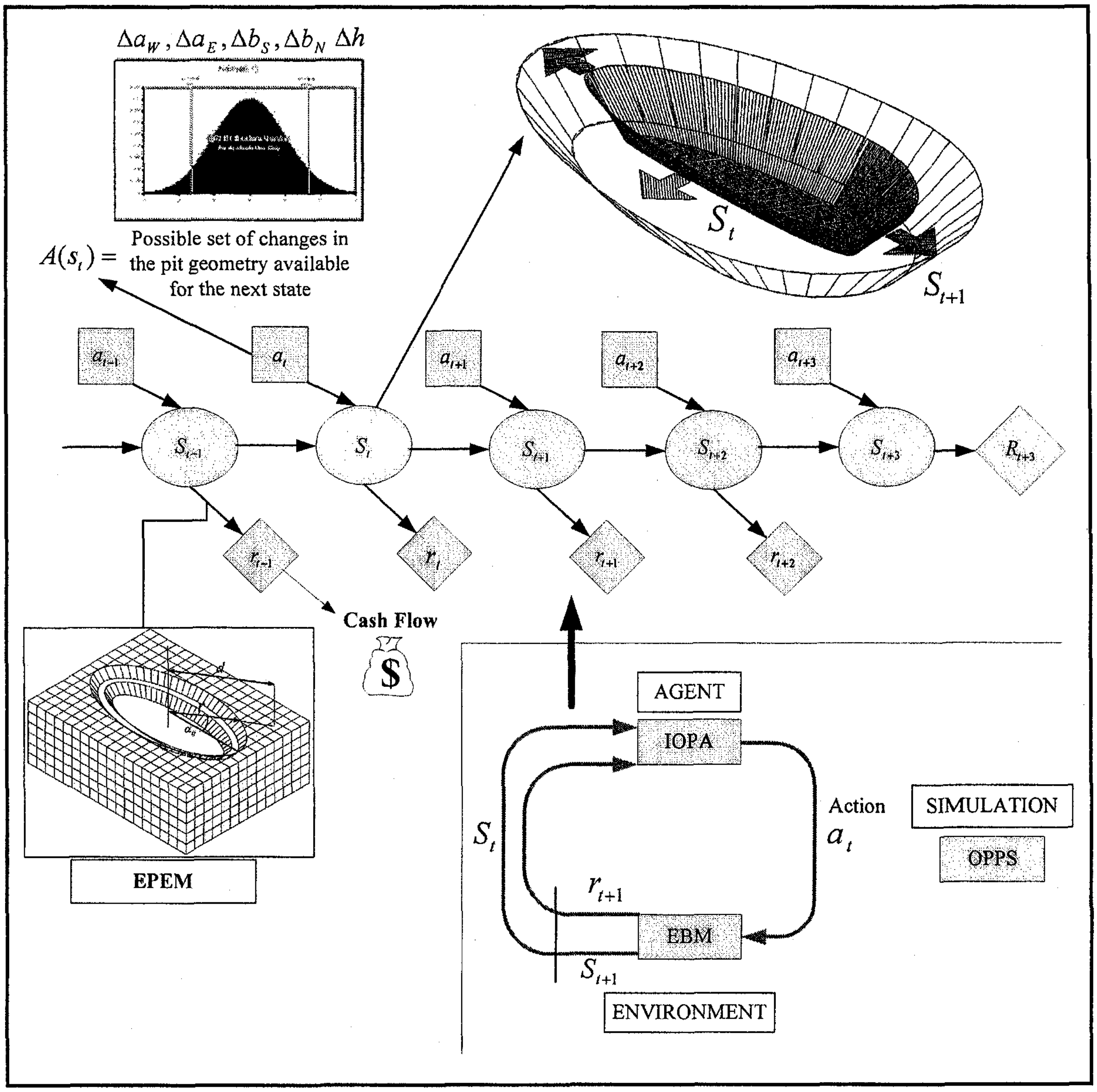


Figure 3.12 IOPS model as an RL architecture

More specifically, the IOPA and OPPS interact at each sequence of discrete time steps, $t = 1, 2, ..., n$. At each time step t, IOPA receives the geometric state of the open pit layout, $s_t \in S$, where S is the set of possible states. On the basis of S, IOPA selects an *action*, $a_t \in A(s_t)$, where $A(s_t)$ is the set of changes possible in the pit geometry in state s_t. One time step later, in part as a consequence of its action, and interaction with EPEM, the agent receives a numerical *reward*, which is the cash flow of that period of mining operation, $r_{t+1} \in R$. As a result, IOPA finds itself in a new state, s_{t+1}. At each time step, the agent implements a mapping from states to probabilities of selecting each possible action. This mapping is called the agent's *policy* and is denoted by, π_t, where $\pi_t(s,a)$ is the probability that $a_t = a$ if $s_t = s$. RL methods specify how the agent changes its policy as a result of its experience. The agent's goal, roughly speaking, is to maximize the total amount of reward it receives over the long run.

3.5 Reinforcement Learning: IOPS Theoretical Framework

Reinforcement learning (Millington, 1991; Sutton, 1992; Kaelbling, 1996; Sutton and Barto, 1998) is learning how to map situations to actions in order to maximize a numerical reward signal. The learner is not told which actions to take, as in most forms of machine learning, but instead it must discover which actions yield the most reward by trying them. The most important distinguishing features of reinforcement learning are two characteristics--trial-and-error search and delayed reward.

One of the challenges that arise in reinforcement learning and not in other kinds of learning is the trade-off between exploration and exploitation. To obtain a large reward, a reinforcement learning agent must prefer actions that it has tried in the past and found to be effective in producing reward. But to discover such actions, it has to try actions that it has not selected before. The agent has to *exploit* what it already knows in order to obtain reward, but it also has to *explore* in order to make better action selections in the future. The dilemma is that neither exploration nor exploitation can be pursued exclusively without failing at the task. The agent must try a variety of actions *and* progressively favor the best action. On a stochastic task, each action must be tried many times to gain a reliable estimate of its expected reward.

3.5.1 Elements of IOPS as an RL Problem

Figure 3.2 and Figure 3.12 demonstrate the main IOPS architecture as an RL problem. Beyond the agent and the environment, one can identify four main sub-elements of a reinforcement learning system: a ***policy***, a ***reward function***, a ***value function***, and a ***model*** of the environment.

A ***policy*** defines the learning agent's way of behaving at a given time. Roughly speaking, a policy is a mapping from perceived states of the environment to actions to be taken in those states. In IOPS, the policy is the schedule for expanding the open pit, which the agent recommends at the end of each period of mine production. The policy is the core of a reinforcement learning agent in the sense that it alone is sufficient to determine behavior. In general, the policies may be based on stochastic variables.

A ***reward function*** defines the goal in a reinforcement learning problem. Roughly speaking, it maps each perceived state of the environment to a single number, a *reward*, indicating the intrinsic desirability of that state. In IOPS, the IOPA's sole immediate objective is to maximize the cash flow of the operation at the next period of extraction. The reward function may serve as a basis for altering the policy. For example, if an action selected by the policy is followed by low reward, then the policy may be changed to select some other action in that situation in the future. In general, reward functions may be stochastic.

Whereas a reward function indicates what is good in an immediate sense, a ***value function*** specifies what is good in the long run. Roughly speaking, the *value* of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state. Whereas rewards determine the immediate, intrinsic desirability of environmental states, values indicate the *long-term* desirability of states after taking into account the states that are likely to follow, and the rewards available in those states. For example, an immediate open pit mine short-term schedule might yield a low cash flow for the next year but still have a high value because it is regularly followed by other long term prediction that yield a high NPV.

The fourth and final element of reinforcement learning systems is a ***model*** of the environment. In IOPS architecture, OPPS mimics the behavior of the open pit expansion with discrete and continuous-time simulation. Given a state and an action, OPPS predicts the resultant next state of the open pit geometry and the derived cash flow. The following sections contain a more detailed discussion on the elements of the RL problem as defined in IOPS.

3.5.1.1 Rewards and Return

In IOPS, the purpose or goal of the agent is formalized in terms of a special reward signal, which passes from the open pit expansion simulation to the IOPA. At each time step, the reward is the cash flow for that period of the mining operation, denoted by $r_t \in R$. Informally, the agent's goal is to maximize the total amount of reward it receives. This means maximizing not just the immediate reward, but also the cumulative reward in the long run, which is the NPV of the venture. So far we have been imprecise regarding the objective of learning. In general, the goal is to maximize the *expected return*, where the return, R_t by equation (3.30), is defined as a specific function of the immediate reward sequence. In equation (3.30), γ is the discount factor and is a number between 0 and 1. The discount factor describes the preferences of an agent for current rewards over future rewards. When γ is close to 0, rewards in the distant future are viewed as insignificant. i in equation (3.31) is the interest rate for time slice, t.

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots + \gamma^k r_{t+k+1} = \sum_{k=0}^{T} \gamma^k r_{t+k+1} \tag{3.30}$$

$$\gamma = \frac{1}{1+i} \tag{3.31}$$

If $\gamma = 0$, the agent is "myopic" in being concerned only with maximizing immediate rewards: its objective in this case is to learn how to choose a_t so as to maximize only r_{t+1}. But in general, acting to maximize immediate reward can reduce access to future rewards so that the return may actually be reduced. As γ approaches 1, the objective takes future rewards into account more strongly: the agent becomes more farsighted.

3.5.1.2 IOPS Action Selection Rule

The simplest action selection rule is to select the action or one of the actions with highest estimated action value, that is, to select the push-back at time step t with the highest cash flow. This method always exploits current knowledge to maximize immediate reward; it spends no time at all sampling apparently inferior actions to see if they might really be better. A simple alternative is to behave greedily most of the time, but every once in a while, say with small probability ε, instead select an action at random, independently of the cash flow estimates of the push-back. This method of using near-greedy action selection rule is called the ε*-greedy* method and it is the action selection algorithm in IOPS (Sutton and Barto, 1998).

3.5.1.3 Markov Property Assumption

IOPS is formulated as an episodic task where there is a final time step T, which is when the simulation reaches the final pit limits. The IOPA-OPPS interaction breaks naturally into sub-sequences, which are called *episodes.* Each episode ends in the special state of the open pit, the ultimate pit limits. In RL terminology this is called *terminal state*, which is followed by a reset to the initial box cut of the open pit expansion. In episodic tasks there is a need to distinguish the set of all non-terminal states, denoted S, from the set of all states plus the terminal state, denoted S^{+}.

In probability theory, a stochastic process has the Markov property if the conditional probability distribution of future states of the process, given the present state, depends only upon the current state. A reinforcement learning task that satisfies the Markov property is called a Markov decision process, or MDP (White, 1978; Hartley et al., 1980; Altman, 1999; Ratitch, 2005). If the state and action spaces are finite, then it is called a finite MDP. The IOPS architecture is based on Markov property, specifically the current state of the pit, which is the present status of the spanning frustum, and only depends on the previous state. This is called a first-order Markov process; the laws describing how the open pit state evolves over time are contained entirely within the conditional distribution $P(S_t|S_{t-1})$ in equation (3.32). A particular finite MDP is defined by its state and action sets and by the one-step dynamics of the environment. Given any state and action, s and a, the probability of each possible next state, s', is given by equation (3.32).

$$\mathrm{P}^{a}_{ss'} = \Pr\left\{s_{t+1} = s' \mid s_t = s, a_t = a\right\} \tag{3.32}$$

These quantities are called *transition probabilities*. Similarly, given any current state and action, s and a, together with any next state, s', the expected value of the next reward is given by equation (3.33). The quantities, $P^{a}_{ss'}$ and $R^{a}_{ss'}$, completely specify the most important aspects of the dynamics of a finite MDP.

$$R^{a}_{ss'} = E\left\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\right\} \tag{3.33}$$

3.5.1.4 IOPS Value Functions

Almost all reinforcement learning algorithms are based on estimating *value functions*--functions of states that estimate how good it is for the agent to be in a given state or how good it is to perform a given action in a given state. The notion of "how good" here is defined in terms of expected return. Accordingly, value functions are defined with respect to particular policies. Figure 3.13 illustrates a schematic of the OPPS simulation at a discrete time step t and the open pit current status of S. For illustration clarity purposes it is assumed that there are just three push-backs a_1, a_2, a_3 that satisfy the targets of the next production period. Following one of the push-back designs the open pit will expand to the status of s_1', s_2', or s_3'. The *value* of state s under policy π, denoted by $V^{\pi}(s)$, is the expected return or the NPV, when starting in s and following π thereafter until reaching the final pit limits. For the MDP representing the open pit dynamics in Figure 3.12, $V^{\pi}(s)$ can be defined as equation (3.34).

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\}$$

$$V^{\pi}(s) = E_{\pi}\{R_t \mid s_t = s\} = E_{\pi}\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\} \tag{3.34}$$

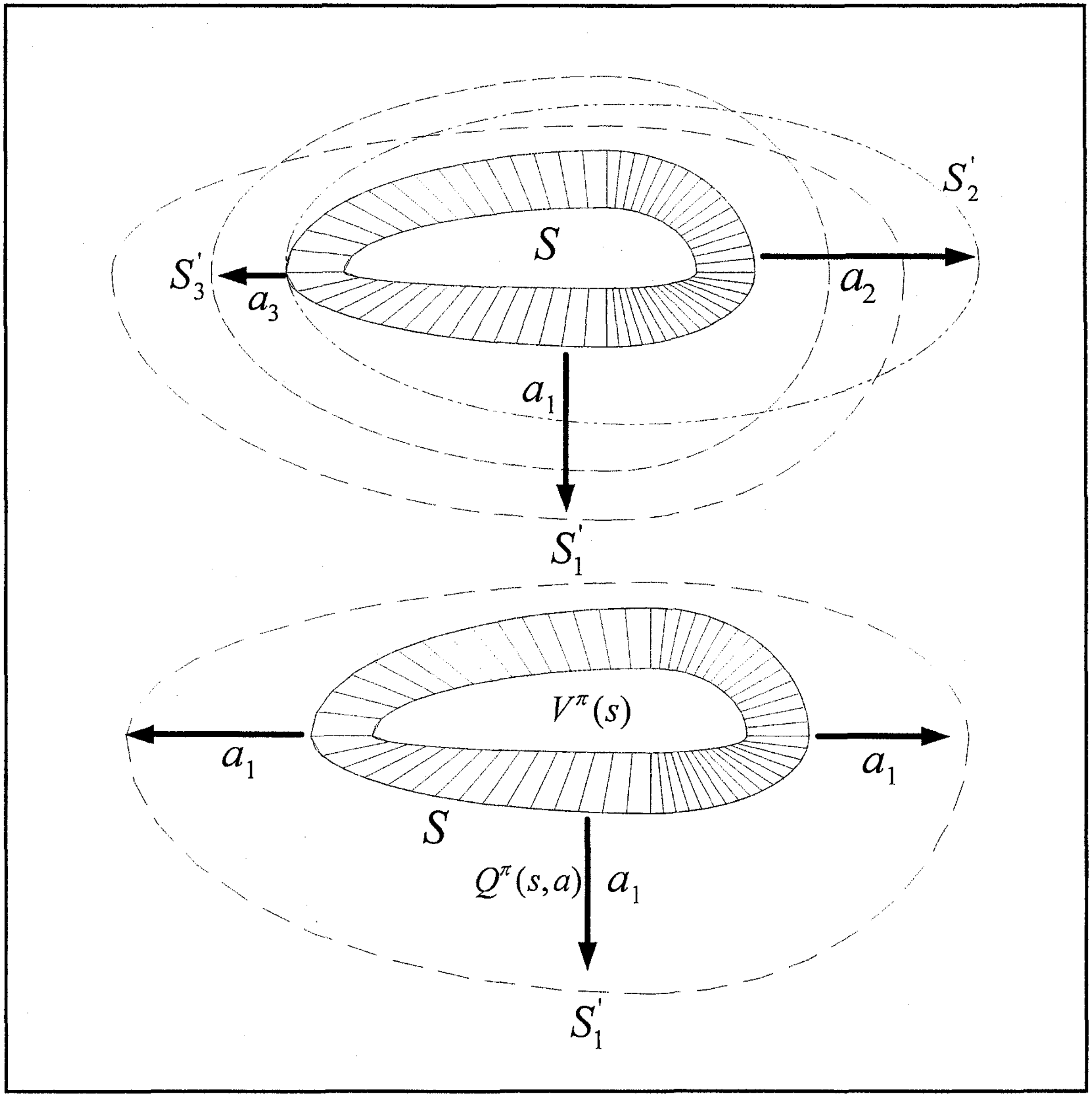


Figure 3.13 Schematic of OPPS simulation at a discrete time step t

$E_\pi\{\ \}$ denotes the expected NPV given that the agent follows policy π, and t is any time step. The function V^π is called the *state-value function for policy* π.

Similarly, the value of taking action a in state s under a policy π, denoted $Q^\pi(s,a)$ is defined as the expected NPV of the operation starting from s, taking the action a, and thereafter following policy π. Q^π is called the *action-value function for policy* π given by equation (3.35).

$$Q^{\pi}(s,a) = E_{\pi}\{R_t \mid s_t = s, a_t = a\} = E_{\pi}\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a\} \tag{3.35}$$

A fundamental property of value functions used throughout reinforcement learning and dynamic programming is that they satisfy particular recursive relationships. For any policy π and any state s, the following consistency condition in equation (3.36) holds between the value of s and the value of its possible successor states.

$$\begin{aligned} V^{\pi}(s) &= E_{\pi}\{r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_t = s\} \\ &= \sum_a \pi(s,a) \sum_{s'} P_{ss'}^a \left[R_{ss'}^a + \gamma E_{\pi}\left\{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_{t+1} = s' \right\} \right] \\ &= \sum_a \pi(s,a) \sum_{s'} P_{ss'}^a \left[R_{ss'}^a + \gamma V^{\pi}(s') \right] \end{aligned} \tag{3.36}$$

Equation (3.36) is the Bellman equation for V^{π} (Bellman, 1957; Bellman and Dreyfus, 1962; Bellman and Kalaba, 1965; Bellman, 1968). It expresses a relationship between the value of a state and the values of its successor states. Figure 3.14 demonstrates the backup diagrams for the open pit expansion model illustrated in Figure 3.13. We are looking ahead from one state to its three possible successor states, as suggested by Figure 3.14a. Each big circle represents a state and each solid circle represents a state-action pair. Starting from state s, the root node at the top, the IOPA could take any of some set of actions. From each of these, the environment could respond with one of several next states, s', along with a reward, r. The Bellman equation (3.36) averages over all the possibilities, weighting each by its probability of occurrence. It states that the value of the start state must equal the discounted value of the expected value of the next state, plus the reward expected along the way.

3.5.1.5 Optimal Value Functions

Solving a reinforcement learning task is finding a policy that achieves a lot of reward over the long run. For finite MDPs, we can precisely define an optimal policy in the following way. Value functions define a partial ordering over policies. A policy π is defined to be better than or equal to a policy π' if its expected return is greater than or

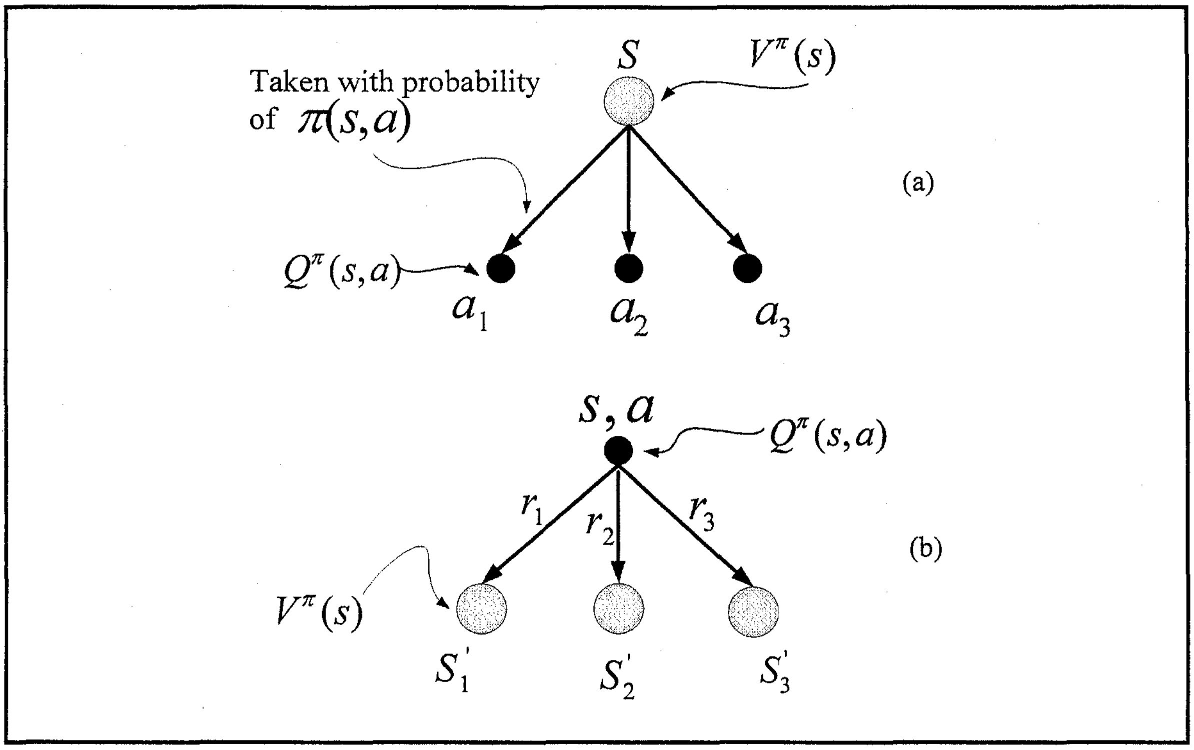


Figure 3.14 Backup diagrams for Figure 3.13

equal to that of π' for all states. In other words, $\pi > \pi'$ if and only if $V^\pi(s) \geq V^{\pi'}(s)$ for all $s \in S$. There is always at least one policy that is better than or equal to all other policies. This is an *optimal policy.* Although there may be more than one, we denote all the optimal policies by π^*. They share the same state-value function, called the *optimal state-value function*, denoted by V^*, and defined as equation (3.37) for all $s \in S$.

$$V^*(s) = \max_\pi V^\pi(s) \tag{3.37}$$

Optimal policies also share the same *optimal action-value function*, denoted by Q^* for all $s \in S$ and $a \in A(s)$, and defined as equation (3.38).

$$Q^*(s,a) = \max_\pi Q^\pi(s,a) \tag{3.38}$$

For the state-action pair (s,a), equation (3.38) gives the expected return for taking action a in state s and thereafter following an optimal policy. Thus, Q^* can be written in terms of V^* as in equation (3.39).

$$Q^*(s,a) = E\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\} \tag{3.39}$$

This is the Bellman optimality equation for V^* (Bellman, 1957; Bellman and Dreyfus, 1962; Bellman and Kalaba, 1965; Bellman, 1968). Intuitively, the Bellman optimality equation expresses the fact that the value of a state under an optimal policy must equal the expected return for the best action from that state as illustrated in equation (3.40).

$$\begin{aligned}
V^*(s) &= \max_{a\in A(s)} Q^{\pi^*}(s,a) \\
&= \max_a E_{\pi^*}\{R_t \mid s_t = s, a_t = a\} \\
&= \max_a E_{\pi^*}\left\{\sum_{k=0}^{T} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a\right\} \\
&= \max_a E_{\pi^*}\left\{r_{t+1} + \gamma \sum_{k=0}^{T} \gamma^k r_{t+k+2} \mid s_t = s, a_t = a\right\} \\
&= \max_a E\left\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\right\} \\
&= \max_{a\in A(s)} \sum_{s'} P^a_{ss'}\left[R^a_{ss'} + \gamma V^*(s')\right]
\end{aligned} \tag{3.40}$$

Equation (3.40) is a form of the Bellman optimality equation for V^*. The Bellman optimality equation for Q^* is given by equation (3.41).

$$\begin{aligned}
Q^*(s,a) &= E\left\{r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1},a') \mid s_t = s, a_t = a\right\} \\
&= \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma \max_{a'} Q^*(s',a')]
\end{aligned} \tag{3.41}$$

For finite MDPs, equation (3.40) has a unique solution independent of the policy. The Bellman optimality equation is actually a system of equations, one for each state. Thus, if there are N states, then there are N equations in N unknowns. If the dynamics of the environment are known $\mathrm{P}^a_{ss'}$ and $R^a_{ss'}$, then in principle one can solve this system of equations for V^* using any of the methods used for solving systems of nonlinear equations.

Explicitly solving the Bellman optimality equation represented by equation (3.40) provides one route to finding an optimal policy for solving the reinforcement learning problem. However, this solution is rarely directly useful. Sutton and Barto (1998) explain that this solution relies on at least three assumptions that are rarely true in practice. These assumptions are: (i) accurate knowledge of the dynamics of the environment; (ii) enough computational resources to complete the computation of the solution; and (iii) the Markov property. In the IOPS framework, we are not able to implement the direct solution because of the dimension of the problem. For instance, assume that at the end of each production period, there are 30 different possible scenarios of push back designs, which satisfies the production targets. Specifically, this means that at each state there are 30 possible actions to take. If we are modeling a mine with 20 years mine life then we will have 30^{20} states, it would take hundreds of years on today's fastest computers to solve the Bellman equation for V^*, and the same is true for finding Q^*.

3.5.2 Review on Solution Methods to the RL Problem

Sutton and Barto (1998) presented three fundamental classes of methods for solving the reinforcement learning problem: (i) dynamic programming (DP); (ii) Monte Carlo (MC) methods; and (ii) temporal-difference (TD) learning. All of these methods solve the full version of the RL problem. Each class of methods has its strengths and weaknesses. Dynamic programming methods are well developed mathematically, but require a complete and accurate model of the environment. Monte Carlo methods do not require a model and are conceptually simple, but are not suited for step-by-step incremental computation. Finally, temporal-difference methods do not require any model and are fully incremental, but they are more complex to analyze.

The term dynamic programming (DP) refers to a collection of algorithms that can be used to compute optimal policies given a perfect model of the environment as a Markov decision process (Bellman and Kalaba, 1965; Bertsekas, 1976; Puterman, 1978; Denardo, 1982; Puterman, 1994; Bertsekas, 1995; Bather, 2000). Classical DP algorithms are of limited utility in reinforcement learning both because of their assumption of a perfect model and because of their great computational expense. If we ignore a few technical details, then the (worst case) time DP methods take to find an optimal policy is polynomial in the number of states and actions. If n and m denote the number of states and actions, a DP method is guaranteed to find an optimal policy in polynomial time even though the total number of (deterministic) policies is m^n. DP is sometimes thought to be of limited applicability because of the *curse of dimensionality* (Bellman, 1957). The fact that the number of states often grows exponentially with the number of state variables. Large state sets do create difficulties, but these are inherent difficulties of the problem, not of DP as a solution method.

Monte Carlo methods (Michie and Chambers, 1968; Narendra and L.Thathachar, 1974; Barto and Duff, 1994; Singh and Sutton, 1996) require only experiences of the model. They sample sequences of states, actions, and rewards from on-line or simulated interaction with an environment. Learning from *on-line* experience is striking because it requires no prior knowledge of the environment's dynamics, yet it can still attain optimal behavior. Learning from simulated experience is also powerful. Although a model is required, the model need only generate sample transitions, not the complete probability distributions of all possible transitions that is required by dynamic programming DP methods. Monte Carlo methods are based on averaging sample returns. They are used only for episodic tasks. That is, we assume experience is divided into episodes, and that all episodes eventually terminate no matter what actions are selected. It is only upon the completion of an episode that value estimates and policies are changed. Monte Carlo methods are thus incremental in an episode-by-episode sense, but not in a step-by-step sense. The Monte Carlo methods learn value functions and optimal policies from experience in the form of sample episodes.

Temporal-difference (TD) learning (Werbos, 1987; Sutton, 1988; Watkins, 1989; Pellegrini, 1991; Sutton, 1992; Watkins and Dayan, 1992; Tsitsiklis, 1997) is a combination of Monte Carlo ideas and dynamic programming (DP) ideas. Like Monte Carlo methods, TD methods can learn directly from raw experience without a model of the environment's dynamics. Like DP, TD methods update estimates based in part on other learned estimates, without waiting for a final outcome (they bootstrap). TD methods are the core of IOPS architecture. The theory is discussed in the next section.

3.5.3 Temporal-Difference Learning

Both TD and Monte Carlo methods use experience to solve the prediction problem. Given some experience following a policy π, both methods update their estimate V of V^{π}. If a non-terminal state s_t is visited at time t, then both methods update their estimate $V(s_t)$ based on what happens after that visit. Roughly speaking, Monte Carlo methods wait until the return following the visit is known, then use that return as a target for $V(s_t)$. A simple every-visit Monte Carlo method suitable for non-stationary environments is given by equation (3.42).

$$V(s_t) \leftarrow V(s_t) + \alpha\left[R_t - V(s_t)\right] \tag{3.42}$$

R_t is the actual return following time t and α is a constant step-size parameter. Whereas Monte Carlo methods must wait until the end of the episode to determine the increment to $V(s_t)$ (only then is R_t known), TD methods need wait only until the next time step. At time $t+1$ they immediately form a target and make a useful update using the observed reward r_{t+1} and the estimate $V(s_{t+1})$. The simplest TD method, known as *TD(0)*, is:

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \tag{3.43}$$

```
Initialize $V(s)$ arbitrarily
Repeat (for each episode):
   Initialize $s$
   Repeat (for each step of episode):
           $a \leftarrow$ action given by $\pi$ for $s$
           Take action,a: observe $r$, and next state, $s'$
        $a \leftarrow$ action given by $\pi$ for $s$
        Take action a; observe, $r$, and next state, $s'$
        $V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)]$
   Until $s$ is terminal
```

Figure 3.15 Tabular TD(0) for estimating V^{π}

In effect, the target for the Monte Carlo update is R_t, whereas the target for the TD update is $r_{t+1}+\gamma V_t(s_{t+1})$. Because the TD method bases its update in part on an existing estimate, it is a bootstrapping method, like DP. Figure 3.15 specifies TD(0) in procedural form.

3.5.3.1 Q-Learning: Off-Policy TD Control Algorithm Core of IOPS

One of the most important breakthroughs in reinforcement learning was the development of an off-policy TD control algorithm known as *Q-learning* (Watkins, 1989). Its simplest form, *one-step Q-learning*, is defined by equation (3.44).

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t)+\alpha[r_{t+1}+\gamma \max_{a'} Q(s_{t+1},a_{t+1})-Q(s_t,a_t)] \tag{3.44}$$

In this case, the learned action-value function, Q, directly approximates Q^*, the optimal action-value function, independent of the policy being followed. This dramatically simplifies the analysis of the algorithm and enabled early convergence proofs. The policy still has an effect in that it determines which state-action pairs are visited and updated. However, all that is required for correct convergence is that all pairs continue to be updated. The Q-learning algorithm is shown in procedural form in Figure 3.16.

```
Initialize $Q(s,a)$ arbitrarily
Repeat (for each episode):
  Initialize S
  Repeat (for each step of episode):
       Choose $a$ from $s$ using policy derived from $Q(\varepsilon - greedy)$
       Take action a, observe $r, s_{t+1}$
       $Q(s_t,a_t) \leftarrow Q(s_t,a_t)+\alpha[r_{t+1}+\gamma \max_{a'} Q(s_{t+1},a_{t+1})-Q(s_t,a_t)]$
       $s_t \leftarrow s_{t+1}$
  Until $s$ is terminal
```

Figure 3.16 Q-Learning an off-policy TD control algorithm, the basis of IOPS framework

3.6 Summary and Remarks

The modeling effort and theoretical architecture required for '**I**ntelligent **O**pen **P**it **O**ptimal **P**roduction **S**imulator' (IOPS) has been presented in this chapter. The long-term open pit planning is formalized by reinforcement learning (RL). The interaction of three fundamental entities of RL problem—environment, intelligent agent, and simulation captures the stochastic dynamic expansion of open pit layouts in time and space. IOPS include independent and interrelated models, using off-policy TD control algorithm known as *Q-learning* to maximize the net present value of the mining venture.

The environment consists of **g**eological **b**lock **m**odel (GBM) and **e**conomic **b**lock **m**odel (EBM). The **i**ntelligent **o**pen **p**it **a**gent (IOPA) interacts with the environment by means of the **o**pen **p**it **p**roduction **s**imulator (OPPS). OPPS comprises **g**eometrical **o**pen **p**it **m**odel (GOPM); **d**iscrete **o**pen **p**it **s**imulator (DOPS); **e**conomic **p**it **e**xpansion **m**odel (EPEM); and **c**ontinuous **o**pen **p**it **s**imulator (COPS). The simulator mimics the stochastic dynamic expansion of open pit layouts using discrete incremental push-backs in different directions. The interactions of EPEM with DOPS returns the pit's NPV following the simulated schedule. The COPS models the dynamics of open pit geometry and the subsequent material movement as a continuous system described by time-dependent differential equations. A set of PDEs capture the time-related behavior of the open pit mining systems. Functional approximation of the discrete simulated push-backs, generated by DOPS, provides the means to convert the set of PDEs to a system of ODEs. Numerical integration with Runge-Kutta scheme yields the trajectory of the pit geometry over time with the respective volume of materials transferred and the resulting NPV.

In order to achieve the objectives, a theoretical modeling framework was established comprising the necessary assumptions and limitations based on reinforcement learning. This framework establishes the essential components of the IOPS architecture. These models are combined into appropriate algorithms for further application analysis requirements, classes and objects identification, and UML diagram design.

CHAPTER 4
ALGORITHM DEVELOPMENT, APPLICATION DESIGN, AND IMPLEMENTATION

4.1 Foreword

The mathematical modeling and theoretical architecture development resulted in the IOPS mathematical models, discussed in Chapter 3. This chapter deals with utilizing the models in a finite set of procedural instructions, algorithms, and numerical methods in order to accomplish the research objectives. The design and implementation of IOPS starts with class design and object identification. There are three basic objects: agents, environments, and simulations. The intelligent learning agent interacts with the economic block model as the environment.'The simulation manages the interaction between the agent and the environment and collects data. The numerical reward is a number representing a push-back's cash flow, the actions and states are instances of classes derived from the abstract classes respectively.

MATLAB (MathWorks, 2005) uses explicit Runge-Kutta codes in its ode45 suite, which are used in integrating ODE equations in the COPS. The solution of the continuous system returns the trajectory of changes in the major, minor and depth axes of the frustum model, as well as, the volume of materials. The solution is passed to the economic expansion model. The scheme will return the blocks in the push-back design, net present value of the simulated schedule, and the volume of ore and waste.

The Java Reinforcement Learning Library, JavaRL, (Kerr et al., 2003) was chosen as the core of the IOPS application implementation. Java programming language version 1.4.2 (Sun Microsystems, 1994-2006) and MATLAB version 7.04 (MathWorks, 2005) were selected as the platform for programming. The IOPS graphical user interface (GUI) was implemented using IntelliJ IDE (JetBrains, 2000-2006). The user interface facilitates the process of setting the block model, pit, production, and simulation parameters.

4.2 Algorithm Development

The modeling effort in Chapter 3 has culminated in the development of '**I**ntelligent Open Pit **O**ptimal **P**roduction **S**imulator' (IOPS) architecture based on reinforcement learning (RL). These models have to be combined in appropriate algorithms for further application analysis requirements.

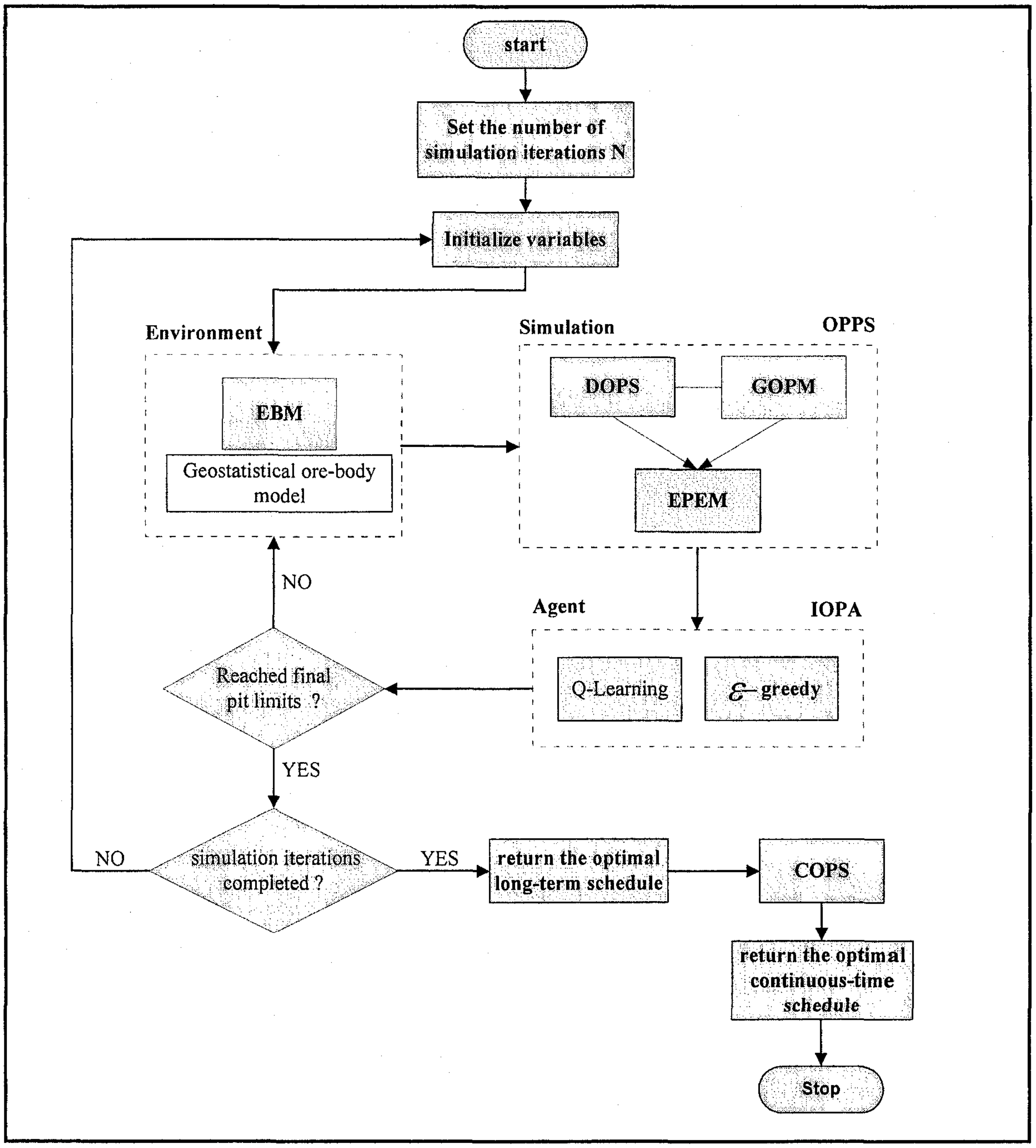

start
Set the number of simulation iterations N
Initialize variables
Environment
EBM
Geostatistical ore-body model
Simulation
OPPS
DOPS
GOPM
EPEM
Agent
IOPA
Q-Learning
ε-greedy
Reached final pit limits ?
NO
YES
simulation iterations completed ?
NO
YES
return the optimal long-term schedule
COPS
return the optimal continuous-time schedule
Stop

Figure 4.1 IOPS general flow chart

Figure 4.1 demonstrates the flow diagram of the IOPS framework in its broadest perspective. The algorithm starts with the interaction of two entities of the RL problem, the environment and the simulation. **O**pen **p**it **p**roduction **s**imulator (OPPS) employs the **g**eometrical **o**pen **p**it **m**odel (GOPM) and the **d**iscrete **o**pen **p**it **s**imulation (DOPS) framework to stochastically simulate the next practical push-backs. The cash flows of the simulated push-backs are returned by the **e**conomic **p**it **e**xpansion **m**odel (EPEM). The intelligent agent, IOPA, employs Q-learning and epsilon-greedy algorithm to choose the next suitable push-back among all the simulated results. The algorithm then updates the action value estimates$Q(s,a)$. These components interact continually, as the agent selects actions and the simulation and environment respond to those actions and present new push-backs to the agent. The simulation continues until the simulated push-back design reaches the final pit limits.

Afterward, the algorithm resets all the variables and starts a new episode of simulation from the initial pit. The IOPA objective is to maximize the NPV of the operation over time. In the Q-learning algorithm, the learned action-value function,Q, directly approximates Q^*, the optimal action-value function, independent of the chosen policy. The algorithm results in the optimal open pit long-term schedule. Subsequently, the discrete optimal push-back design is transferred to the **c**ontinuous **o**pen **p**it **s**imulator (COPS), where the specification of the continuous-time open pit dynamics is captured as a system of PDEs. The numerical solution to these set of equations yields the short range plan of the mining operation. The general perspective of the IOPS algorithm presented will be scrutinized in detailed algorithms in the following sections.

4.2.1 Intelligent Open Pit Optimal Production Simulator (IOPS)

Figure 4.2 illustrates the detailed flow chart of IOPS emphasizing on Q-learning algorithm (Watkins, 1989; Watkins and Dayan, 1992). The Q-learning algorithm is further described in a 5-step procedure. The great advantage of this algorithm is that the learned action-value function, Q, directly approximates Q^*, the optimal action-value function, independent of the followed policy. This dramatically simplifies the analysis of the algorithm. The policy still has an effect in that it determines which state-action pairs are visited and updated in the process.

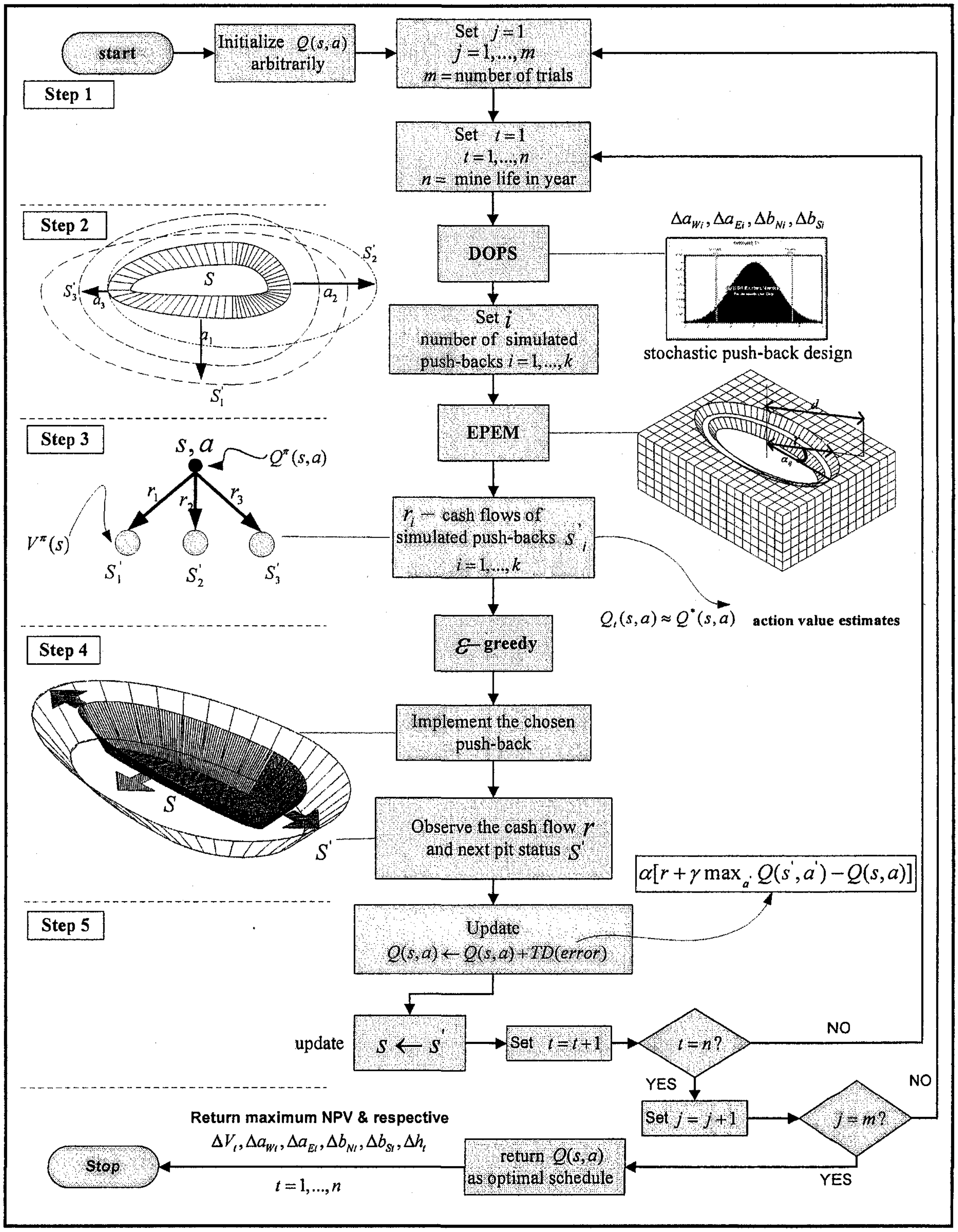

start
Step 1
Initialize $Q(s,a)$ arbitrarily
Set $j=1$ $j=1,...,m$ $m=$ number of trials
Set $t=1$ $t=1,...,n$ $n=$ mine life in year
Step 2
S'_2
S
S'_3
a_3
a_2
a_1
S'_1
DOPS
$\Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}$
stochastic push-back design
Set i number of simulated push-backs $i=1,...,k$
Step 3
s,a
$Q^{\pi}(s,a)$
r_1
r_2
r_3
$V^{\pi}(s)$
S'_1
S'_2
S'_3
EPEM
r_i – cash flows of simulated push-backs s'_i $i=1,...,k$
$Q_t(s,a) \approx Q^*(s,a)$ action value estimates
Step 4
ε-greedy
Implement the chosen push-back
S
S'
Observe the cash flow r and next pit status s'
$\alpha[r+\gamma \max_{a'} Q(s',a')-Q(s,a)]$
Step 5
Update $Q(s,a) \leftarrow Q(s,a)+TD(error)$
update
$s \leftarrow s'$
Set $t=t+1$
$t=n?$
NO
YES
Set $j=j+1$
$j=m?$
NO
YES
return $Q(s,a)$ as optimal schedule
Return maximum NPV & respective
$\Delta V_t, \Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}, \Delta h_t$
$t=1,...,n$
Stop

Figure 4.2 Q-learning algorithm flow diagram

Step 1

The algorithm starts with (i) arbitrarily initializing the $Q(s,a)$, which is the expected discounted sum of future monetary returns of expanding the open pit from status S to the $S^{'}$ by choosing the push-back a and following an optimal policy thereafter; (ii) set the number of simulation trials that the algorithm is run. In other words the number of times that the open pit dynamics are simulated from the initial box cut to the final pit limits.

Step 2

The DOPS algorithm is called to capture the open pit layout evolution as a result of the material movement throughout the mine life. The DOPS stochastically simulates a number of practical push-back designs for the next production period. The result of the simulation is k push-backs $a_1, a_2, ..., a_k$ that satisfy the next production period targets. Following one to the push-back designs the open pit will expand to the status of $s_1^{'}, s_2^{'}, ..., s_k^{'}$. The *value* of state s under policy π, denoted $V^{\pi}(s)$ is the expected return or the NPV, when starting in s and following π thereafter until reaching the final pit limits.

Step 3

Simulated push-backs $a_1, a_2, ..., a_k$ are transferred to the EPEM, where the cash-flows $r_1, r_2, ..., r_k$ of following those schedules are returned.

Step 4

The epsilon greedy algorithm demonstrated by Figure 4.3 is called. The action selection rule is to select the action or one of the actions with highest estimated action value, that is, to select the push-back at time step t with the highest cash flow. The algorithm behaves greedily most of the time, but every once in a while, say with small probability ε, instead select an action at random, uniformly, independently of the action-value estimates of the push-back. Subsequently the chosen push-back is implemented and the agent finds itself in pit status $S^{'}$ and observes the cash flow r.

Step 5

After being initialized to arbitrary numbers, Q-values are estimated on the basis of experience. Update $Q(s,a)$ based upon the previous experience as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t)+\alpha[r_{t+1}+\gamma \max_{a'} Q(s_{t+1},a_{t+1})-Q(s_t,a_t)] \tag{4.1}$$

This algorithm is guaranteed to converge to the correct Q-values with the probability one if the environment is stationary and depends on the current state and the action taken in it. Every state-action pair continues to be visited, and the learning rate is decreased appropriately over time. Once these values have been learned, the optimal action from any state is the one with the highest Q-value.

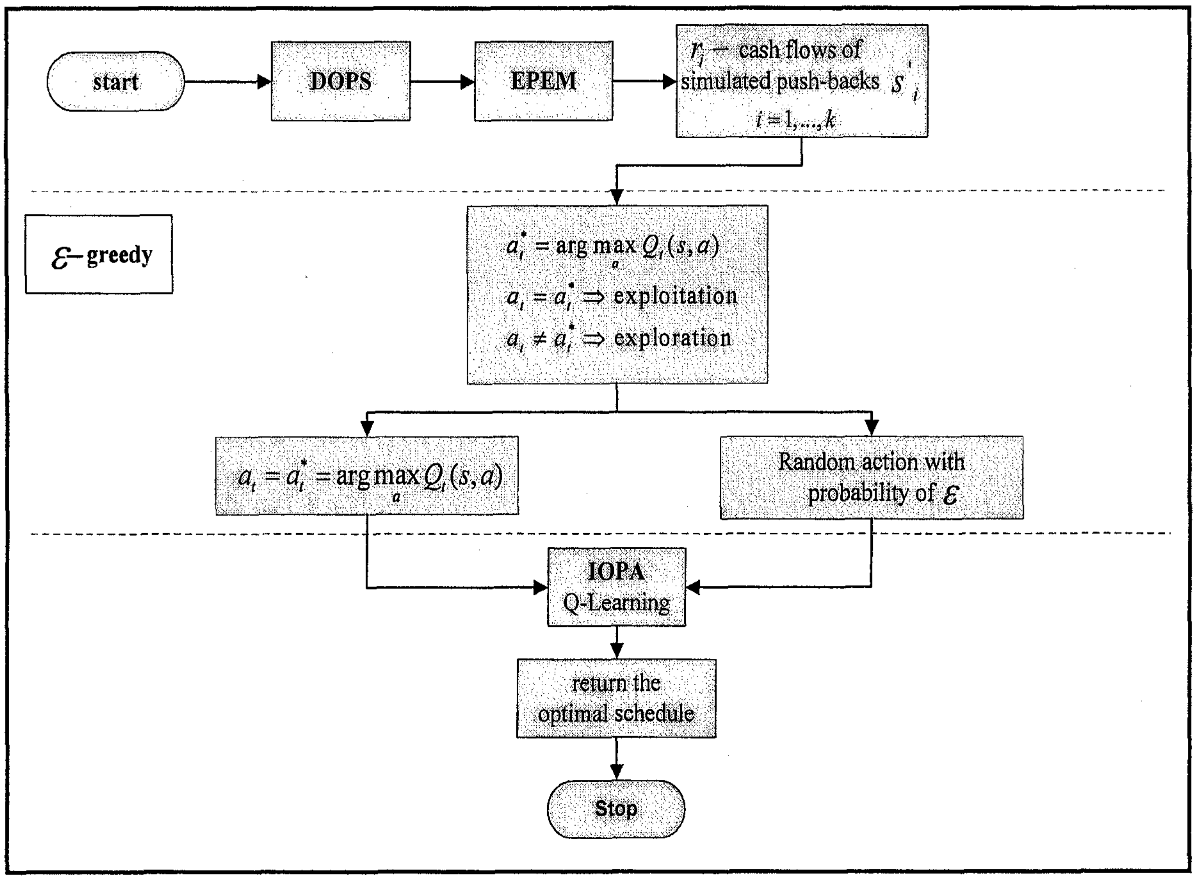

start
DOPS
EPEM
r_i – cash flows of simulated push-backs s'_i
$i=1,\dots,k$
ε–greedy
$a_t^* = \arg\max_a Q_t(s,a)$
$a_t = a_t^* \Rightarrow$ exploitation
$a_t \neq a_t^* \Rightarrow$ exploration
$a_t = a_t^* = \arg\max_a Q_t(s,a)$
Random action with probability of ε
IOPA Q-Learning
return the optimal schedule
Stop

Figure 4.3 Epsilon greedy algorithm flow chart

4.2.2 Discrete Open Pit Simulator (DOPS)

Discrete simulation is used to capture the open pit layout evolution as a result of the material movement throughout the mine life. DOPS is developed to simulate the dynamic expansion of the open pit geometry, using Monte Carlo simulation based on GOPM. The pit status is observed as snapshots over discrete time steps through mine life, $i = 1,...,n$. At each time step, the DOPS stochastically simulates several practical push-backs, which satisfy the next production targets. The simulation results are a series of equally probable realizations of changes in GOPM frustum dimensions at each time step over the mine life. Figure 4.4 illustrates the five main steps of the DOPS algorithm.

Step 1

The simulation starts with the initialization of variables: (i) setting the initial box cut dimensions $a_{W0}, a_{E0}, b_{S0}, b_{N0}$ and h_0; (ii) initializing the optimized final pit limits $a_{W-\max}, a_{E-\max}$, $b_{S-\max}$, $b_{N-\max}$ and $h_{\max}$; (iii) setting the overall pit slopes in different regions $\theta_{NW}, \theta_{NE}, \theta_{SW}, \theta_{SE}$; (iv) setting the maximum number of benches, which is practically possible to deepen the pit each year; (v) setting the mean, minimum, and maximum annual acceptable production; (vi) inputting the geological and economic block model files; and (vii) setting the number of possible push-backs at each time step. The amount of ore, waste and stockpile materials in the final pit limits are reported based on bench-by-bench and tonnage grade curves. The life of mine is estimated with due respect to the annual targeted production rate and the mining and milling constraints. The overall pit slopes, the minimum required mining width and the feasible starting points for the initial box cut are generated based on the coordinates and dimensions of the final pit limits.

Step 2

The simulation carries on with an increment distribution generator, which uses an integer random number generator to produce the number of benches that the pit will expand each year during the life of mine. The number of benches can be between zero and the maximum number of benches, which is practically possible to deepen the pit each year.

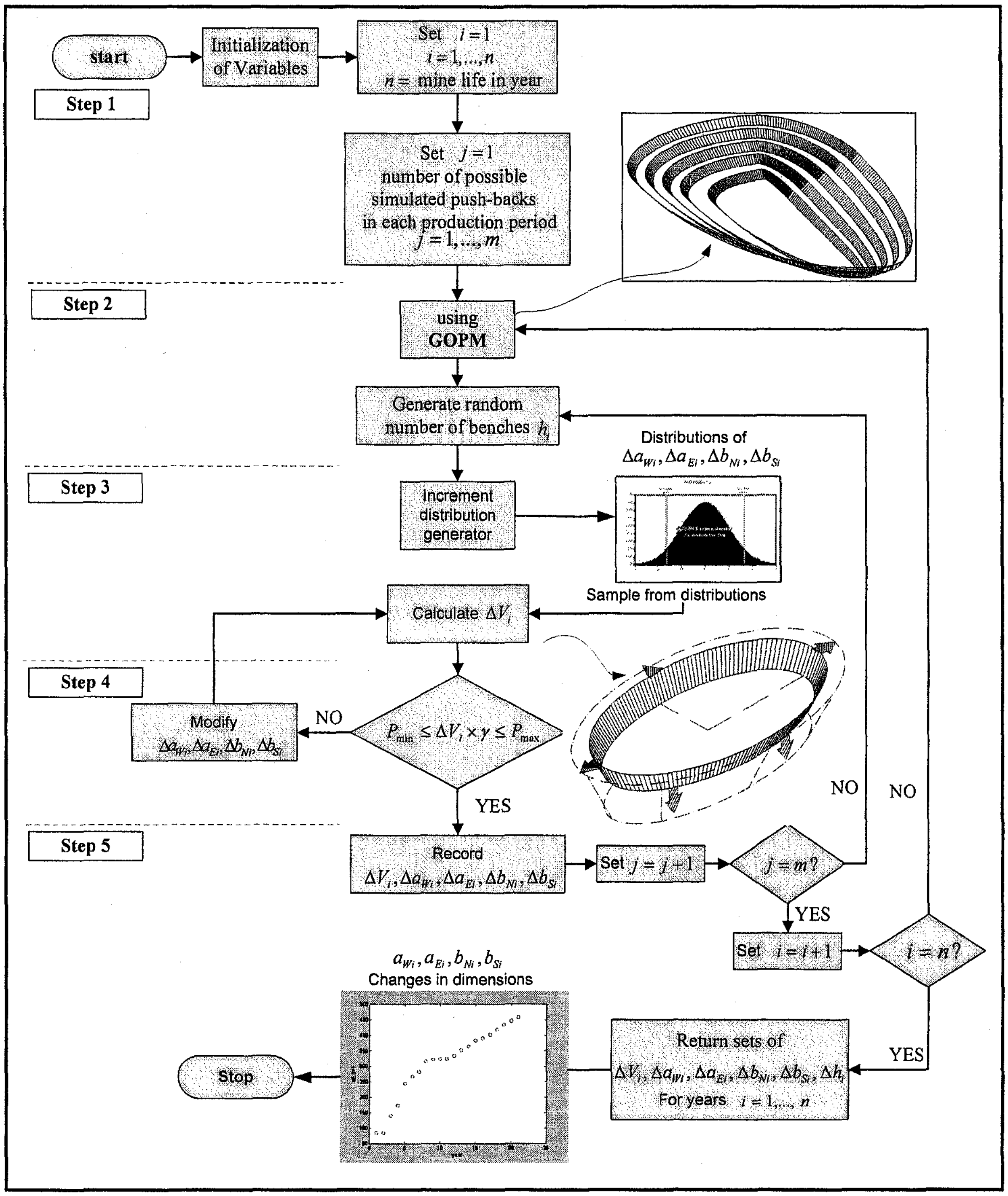

start
Initialization of Variables
Set $i=1$
$i=1,...,n$
$n=$ mine life in year
Step 1
Set $j=1$
number of possible simulated push-backs in each production period
$j=1,...,m$
Step 2
using GOPM
Generate random number of benches h_i
Distributions of
$\Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}$
Step 3
Increment distribution generator
Sample from distributions
Calculate ΔV_i
Step 4
Modify
$\Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}$
NO
$P_{\min} \le \Delta V_i \times \gamma \le P_{\max}$
YES
NO
NO
Step 5
Record
$\Delta V_i, \Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}$
Set $j=j+1$
$j=m?$
YES
Set $i=i+1$
$i=n?$
$a_{Wi}, a_{Ei}, b_{Ni}, b_{Si}$
Changes in dimensions
Return sets of
$\Delta V_i, \Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}, \Delta h_i$
For years $i=1,...,n$
YES
Stop

Figure 4.4 DOPS flowchart

Step 3

A table is generated for different values of random variables $\Delta a_W, \Delta a_E, \Delta b_N$ and Δb_S increments. It starts with the minimum increment possible and it increases with a fixed step size to the maximum possible increment in each direction. The volume of material for each possible expansion ($V_{i+1} - V_i$) is calculated and substituted in the table. The annual increments, which resulted in an acceptable production rate, are returned by the routine. A normal probability density function (PDF) is fitted on the pit incremental data. $\Delta a_W, \Delta a_E, \Delta b_N, \Delta b_S$ are sampled from the approximated probability distribution functions and the respective ΔV_i is calculated for the sampled increments.

Step 4

If the volume of material is in the acceptable production range, it will be recorded, if not $\Delta a_W, \Delta a_E, \Delta b_N, \Delta b_S$ will be modified until ΔV_i will be in the acceptable range. A simulation with sufficient number of iterations will return several scenarios of push-backs at each discrete time step.

Step 5

The results are passed on to the economic pit expansion model (EPEM). The scheme returns the cash flow of the different push-back scenarios. The monetary value of push-back designs are used as the numerical reward in IOPA to choose the best action.

4.2.3 Continuous Open Pit Simulator (COPS)

COPS is a simulation model based on the modified open pit geometrical model, and a system of PDEs capturing the continuous-time open pit dynamics. The specification of the continuous-time open pit dynamics as a system of PDEs with the respective boundary conditions represented in Chapter 3 does not allow a unique solution to the long term scheduling problem. There are possibly infinitely many solutions to the set of PDEs or in other words there are many pushback designs that can deplete the ore-body from the initial box cut to the final pit limits. Where general ODEs have solutions that are families with each solution characterized by the values of some parameters, for a PDE the solutions often are parameterized by functions, informally put, this means that the set of solutions is much larger. Therefore, there is a need for additional information and conditions to obtain a unique solution to the continuous-time open pit expansion model, which satisfies the management objectives over time.

The interaction of DOPS, EPEM, GOPM, and Q-learning algorithm returns the optimal long-term schedule. Accordingly, it records the discrete changes in the open pit geometry which maximizes the NPV. The optimal long-term plan yields the additional and auxiliary conditions needed to convert the set of PDEs to a set of ODEs. The DOPS and Q-learning output is the input for COPS where the set of PDEs are converted into a set of ODEs and are solved by Runge-Kutta integration scheme. Figure 4.5 demonstrates the continuous simulator flowchart. The optimal schedule output of the Q-learning algorithm is used as the input for the continuous simulation model.

Step 1

The best curve fit is used to approximate functions of the incremental push-backs Δa_E, Δa_W, Δb_N, Δb_S and Δh generated by Q-learning. Substitution of the approximated incremental functions in equation (3.13) and substituting the results of differentiation of partial derivatives in equation (3.13), converts the set of PDEs into a system of ODEs with time as the independent variable, in the form of equation (3.20).

Step 2

Defining the set of ODEs capturing the dynamics of the open pit following the optimal long-term plan resulted from the Q-learning algorithm.

Step 3

Numerical integration of the system of ODEs will result in continuous format of changes in dimensions of the open pit with its respective volume of ore, waste, stockpile material and the net present value of the venture. The first step in investigating the dynamics of the continuous-time system described by the system of ODEs by equation (3.20) is integration to obtain trajectories. Since most ordinary differential equations are not soluble analytically, numerical integration is the choice to obtain information about the trajectory. Many different methods have been proposed and used to solve accurately various types of ordinary differential equations. However there are a handful of methods known and used universally (i.e., Runge-Kutta, Adams-Bashforth-Moulton and

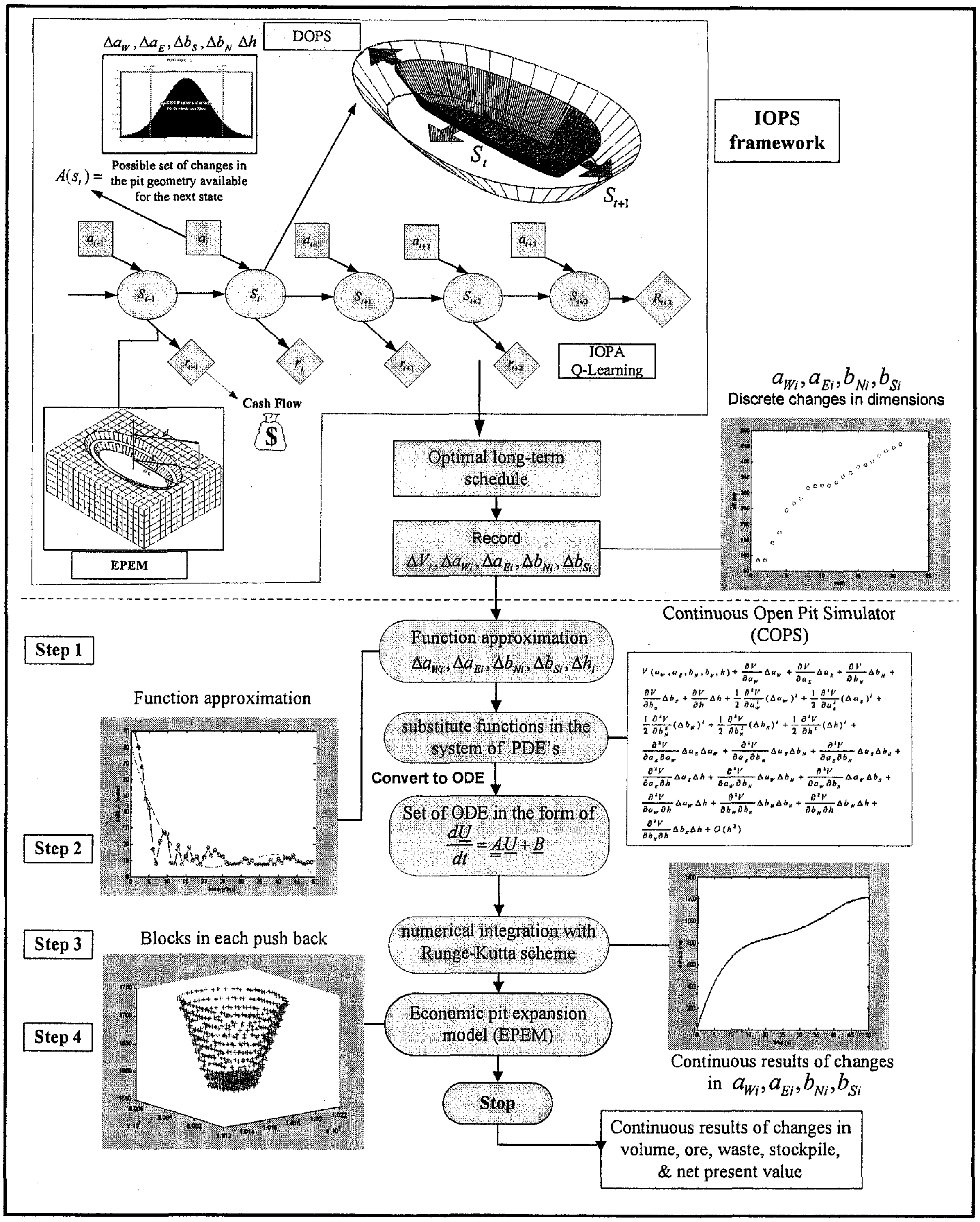

$\Delta a_W, \Delta a_E, \Delta b_S, \Delta b_N\ \Delta h$
DOPS
IOPS framework
$A(s_t) =$ Possible set of changes in the pit geometry available for the next state
S_t
S_{t+1}
a_{t-1}
a_t
a_{t+1}
a_{t+2}
a_{t+3}
S_{t-1}
S_t
S_{t+1}
S_{t+2}
S_{t+3}
R_{t+3}
r_{t-1}
r_t
r_{t+1}
r_{t+2}
IOPA Q-Learning
Cash Flow
$
EPEM
$a_{Wi}, a_{Ei}, b_{Ni}, b_{Si}$
Discrete changes in dimensions
Optimal long-term schedule
Record $\Delta V_i, \Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}$
Step 1
Function approximation $\Delta a_{Wi}, \Delta a_{Ei}, \Delta b_{Ni}, \Delta b_{Si}, \Delta h_i$
Function approximation
Continuous Open Pit Simulator (COPS)
$V(a_W, a_E, b_N, b_S, h) + \frac{\partial V}{\partial a_W}\Delta a_W + \frac{\partial V}{\partial a_E}\Delta a_E + \frac{\partial V}{\partial b_N}\Delta b_N + \frac{\partial V}{\partial b_S}\Delta b_S + \frac{\partial V}{\partial h}\Delta h + \frac{1}{2}\frac{\partial^2 V}{\partial a_W^2}(\Delta a_W)^2 + \frac{1}{2}\frac{\partial^2 V}{\partial a_E^2}(\Delta a_E)^2 + \frac{1}{2}\frac{\partial^2 V}{\partial b_N^2}(\Delta b_N)^2 + \frac{1}{2}\frac{\partial^2 V}{\partial b_S^2}(\Delta b_S)^2 + \frac{1}{2}\frac{\partial^2 V}{\partial h^2}(\Delta h)^2 + \frac{\partial^2 V}{\partial a_E \partial a_W}\Delta a_E \Delta a_W + \frac{\partial^2 V}{\partial a_E \partial b_N}\Delta a_E \Delta b_N + \frac{\partial^2 V}{\partial a_E \partial b_S}\Delta a_E \Delta b_S + \frac{\partial^2 V}{\partial a_E \partial h}\Delta a_E \Delta h + \frac{\partial^2 V}{\partial a_W \partial b_N}\Delta a_W \Delta b_N + \frac{\partial^2 V}{\partial a_W \partial b_S}\Delta a_W \Delta b_S + \frac{\partial^2 V}{\partial a_W \partial h}\Delta a_W \Delta h + \frac{\partial^2 V}{\partial b_N \partial b_S}\Delta b_N \Delta b_S + \frac{\partial^2 V}{\partial b_N \partial h}\Delta b_N \Delta h + \frac{\partial^2 V}{\partial b_S \partial h}\Delta b_S \Delta h + O(h^3)$
substitute functions in the system of PDE's
Convert to ODE
Step 2
Set of ODE in the form of $\frac{dU}{dt} = \underline{A}\underline{U} + \underline{B}$
Step 3
Blocks in each push back
numerical integration with Runge-Kutta scheme
Step 4
Economic pit expansion model (EPEM)
Continuous results of changes in $a_{Wi}, a_{Ei}, b_{Ni}, b_{Si}$
Stop
Continuous results of changes in volume, ore, waste, stockpile, & net present value

Figure 4.5 COPS flowchart

Backward Differentiation Formulae methods). All these methods discretize the differential system to produce a difference equation or map. The numerical methods have the same aim; that the dynamics of the map should correspond closely to the dynamics of the differential equation (Cartwright and Piro, 1992).

Step 4

The solution of the continuous system returns the trajectories of changes in major, minor and depth of the frustum as well as the volume of materials transferred. The solution is passed to the economic pit expansion model (EPEM). The scheme will return the blocks in the push-back design, net present value of the long-term optimal schedule, and the volume of ore and waste in the short-range plans.

4.3 IOPS Design and Implementation

The primary survey of reinforcement learning literature revealed that the Standard Interface for Reinforcement Learning in C++ (Sutton and Santamaria) is the accepted platform for implementing reinforcement learning simulations. The Java Reinforcement Learning Library, JavaRL, (Kerr et al., 2003) is an adaptation of the C++ standard. JavaRL is comprised of three core classes, which correspond to the three basic entities of the RL problem. These three classes are *Agent, Environment*, and *Simulation*. The package contains Agent subclasses that employ various TD learning-based algorithms (including SARSA and Watkin's Q-Learning) with various memory models (including neural networks and look-up tables).

The library includes valuable supplemental code as well. This code includes interfaces and classes for implementing the communication between agents and environments. The library contains *Action* and *State* interfaces for this purpose. A wrapper class, *ActionResult*, is also provided for concisely describing the environment's response to an agent's action; an *ActionResult* instance contains a double reward value and a *State* instance.

Perhaps the most useful additions to the standard interface are the provided *Agent* subclasses for programming agents that employ temporal-difference (TD) learning. These subclasses are easily adaptable to most of memory models for agents. Selected TD control algorithms are implemented in full for agents with tabular forms of memory.

The JavaRL standard interface serves as the kernel of IOPS application. Thus, we now refocus our attention on the three core classes of the interface. We briefly outline the three primary methods that appear in each of these classes. Figure 4.6 illustrates the static class UML diagram of IOPS application based on JavaRL library (Kerr et al., 2003).

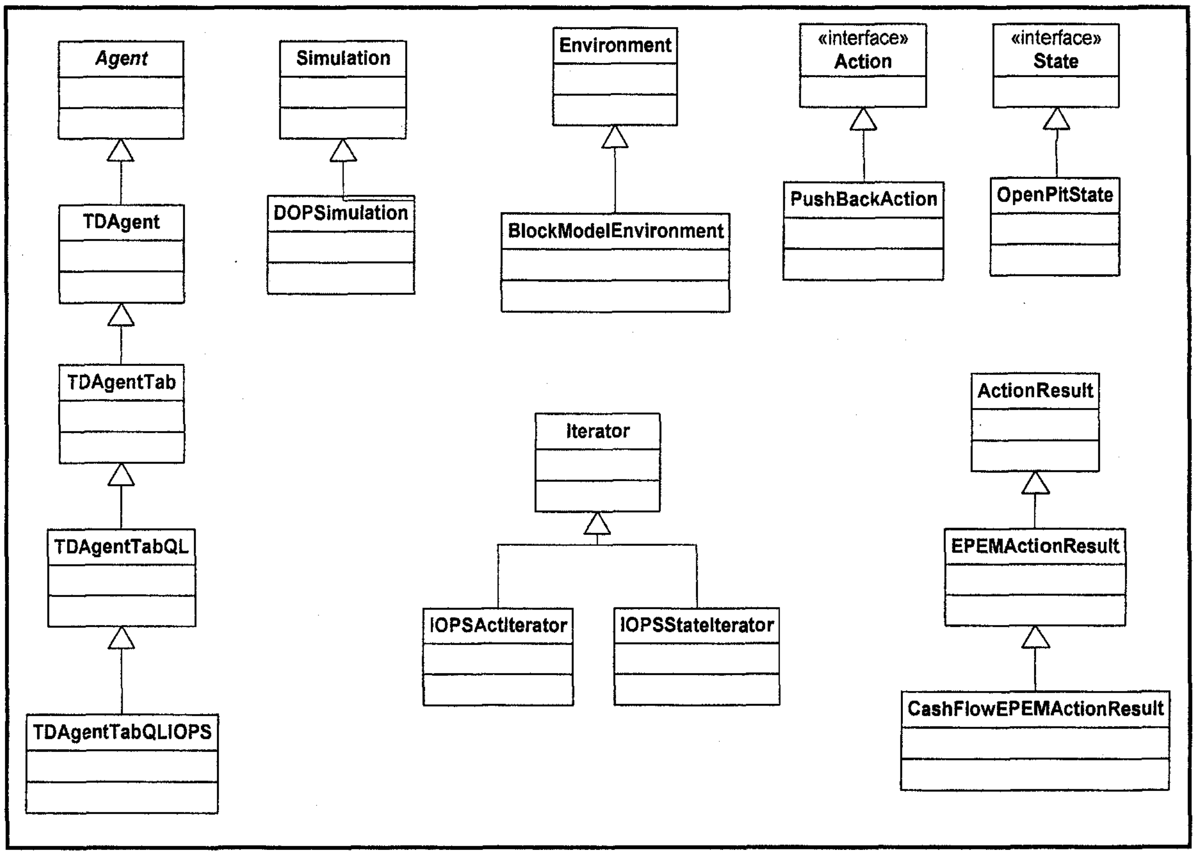


Figure 4.6 IOPS class UML diagram

4.3.1 IOPS Classes

Agent.java – This is an abstract class for implementing all agents. An agent is an entity that interacts with the environment, receiving states and selecting actions. Specific agents are instances of subclasses of Agent.

TDAgent.java – This abstract class implements some of an agent's methods for learning the optimal policy for a reinforcement learning problem using a temporal-difference control algorithm and an arbitrary memory model.

TDAgentTab.java – This class implements the agent's methods for learning using a temporal-difference control algorithm and look-up tables as memory. The agent seeks to learn the optimal policy by approximating the optimal action-value function, Q*.

TDAgentTabQL.java – This abstract class implements the agent's methods for learning the optimal policy for a reinforcement learning problem using Watkin's Q(lambda)-learning on-policy control algorithm with eligibility traces and tabular memory.

TDAgentTabQLIOPS.java – This class implements the agent's methods for learning and solving the maximization problem of the open pit planning represented by a simulation model using the Q-Learning temporal-difference control algorithm. The agent seeks to learn the optimal policy by approximating the optimal state-action function, $Q^*(state, action)$, using look-up tables as memory and an $\varepsilon - greedy$ policy.

Simulation.java – This class implements the interface, and manages the interaction, between the agent and the environment.

DOPSimulation.java – This class runs the simulations for the open pit stochastic dynamics. It also interacts with the EPEM model to return the cash flow and interacts with the agent class.

Environment.java – This is an abstract class for implementing the environment. An environment defines the problem to be solved. It determines the dynamics of the environment, the rewards, and the trial terminations.

BlockModelEnvironment.java – This class represents the economic block model. A block model object is instantiated based on the input file.

Action.java – This is an interface for the output of an agent.

PushBackAction.java – This class creates PushBackAction objects to represent the possible moves (actions) an agent can select in the planning problem.

State.java – This is an interface for implementing the input (i.e. sensory information) that an agent receives from the environment.

OpenPitState.java – This class constructs a representation of the state input that an agent receives from the environment.

ActionResult.java – This class combines a State instance and a numerical reward to form a single object that describes the results of taking a specific action in a specific state.

EPEMActionResult.java – This class is inherited from the ActionResult and utilizes the EPEM model to return the cash flow of the simulated push backs.

IOPSActIterator.java – A class for creating and using an iterator that returns all possible actions for the agent for the learning problem.

IOPSStateIterator.java – A class for creating and using an iterator that returns all possible PitStates the agent may receive for the pit reinforcement learning problem.

4.3.1.1 The Agent Class

The *Agent* class is one of the three core classes of the standard Java interface. It is the abstract class for implementing all RL agents. As mentioned, RL agent interacts with an environment by selecting actions and consequently receiving numerical rewards and state descriptions from the environment. The immediate numerical reward quantifies the short-term desirability of an action, which in IOPS is the cash flow of the simulated push-back. The *State* describes the new state of the environment, and thus, the potential for future rewards. Ideally, the agent learns from these consequences to select the actions that are most desirable in the long-term. By learning this optimal mapping, or policy, from each state of the environment to the action that yields the greatest reward in the long-term, the agent learns to optimally complete the assigned task.

The *Agent* class contains three abstract methods: *init*, *startTrial*, and *step*. We implemented an *Agent* subclass *TDAgentTabQLIOPS* and defined these three methods. The *TDAgentTabQLIOPS* subclass implements Q-learning algorithm to approximate *action-value*, or *Q*, *functions*. The *init* method initializes the IOPA agent. This initialization typically requires the construction of one or more data structures for the agent's memory. The *init* method can either preload past memory or reset the agent's memory to its original, naive condition. This method is called once when the RL simulation first begins.

The *startTrial* method prepares the agent for the start of a new episode of simulation from the initial open pit box cut, within an ongoing simulation. The method creates and returns the first action of the agent in the trial. This first push-back is determined by the agent's current policy and the first state of the trial, which is passed as a parameter to the method. The *startTrial* method is called at the beginning of each trial.

The *step* method is the third method of the *Agent* class. It is the method where all learning and decision-making occurs for the IOPA agent. As parameters, the method receives a *State* instance and a *double* reward, which describe the consequences of taking the previous action. If the agent learns in any way with experience, then this method implements this learning. The method then returns the agent's next action, as determined by the agent's current policy. This method is called once on each step of the simulation.

4.3.1.2 The Environment Class

In IOPS framework, the environment is the economic block model of ore reserve that the IOPA agent interacts with and seeks to learn about through OPPS simulation. The economic block model responds to the agent's action by outputting a numerical reward, which is the cash flow of the simulated push-back. The Environment class contains the same three abstract methods as the Agent class. We implemented an Environment subclass BlockModelEnvironment and defined these three methods.

The *init* method initializes an instance of the *Environment* subclass. The method is responsible for constructing and initializing the data structures containing the economic block model. The *startTrial* method prepares the DOPS for the start of a new trial within an ongoing simulation. The method must create and return the first *State* of the new trial. This method is called at the beginning of each trial.

The *step* method is the third method in the *Environment* class. This method receives the agent's most recent action as a parameter and it returns the immediate reward for this action, which is the cash flow of the simulated push-back.

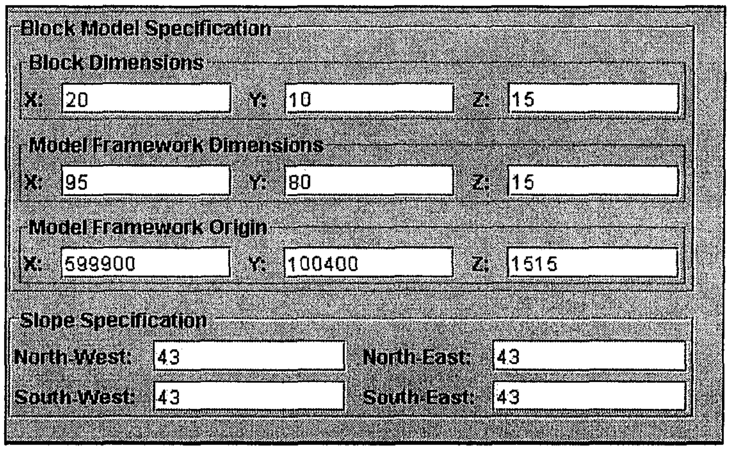
4.3.1.3 The Simulation Class

The *Simulation* class is the third and final abstract class of the standard Java interface. DOPSimulation an instance of this class manages the interactions between the IOPA agent and the block model environment. This class contains complete implementations of the *init*, *startTrial*, and *step* methods for DOPS simulations. Each of these methods calls the *Agent* and *Environment* instances' versions of the same method, often using the return value of one of the calls as the parameter for the other. The classes also contain methods that run the simulation for a set number of steps or trials. All the discrete simulation and EPEM models are defined in *collectData* method.

4.4 IOPS Graphical User Interface

The IOPS graphical user interface (GUI) was implemented using IntelliJ IDE (JetBrains, 2000-2006). The program requires the block model file as the input. The following block model parameters can be set through the block model specification tab illustrated in Figure 4.7.

Block dimensions – the dimension of small blocks in X, Y, and Z directions are specified in meters at this node.



Block Model Specification

Block Dimensions

X: 20 Y: 10 Z: 15

Model Framework Dimensions

X: 95 Y: 80 Z: 15

Model Framework Origin

X: 599900 Y: 100400 Z: 1515

Slope Specification

North-West: 43 North-East: 43

South-West: 43 South-East: 43

Figure 4.7 IOPS block specification tab

Model framework dimensions – the number of blocks in X, Y, and Z direction in the main model framework are denoted at this panel.

Model frame work origin – the coordinates of the origin of the block model

Slope specification – Overall stable slope recommended by geo-mechanical studies in each region in degrees.

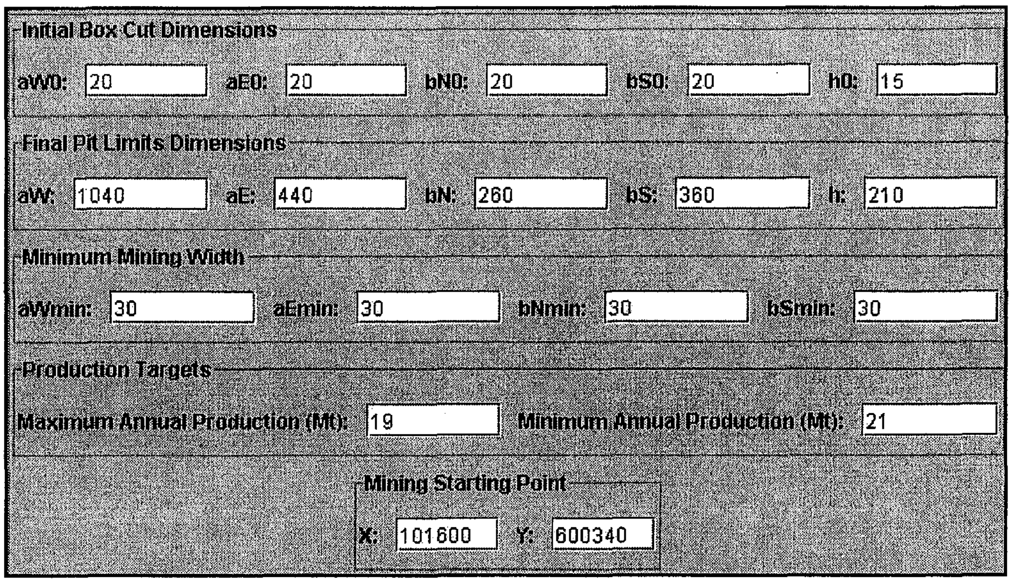
The following pit geometry and production parameters can be set through the open pit specification tab illustrated in Figure 4.8.

Initial box cut dimensions – the dimensions of the initial box cut in meter, representing the boundary condition for the differential equations.

Final pit limits dimensions – the final pit limits dimensions in meter, captured by the modified frustum geometrical model.

Minimum mining width – the minimum possible mining width at the bottom of the pit.

Production targets – maximum and minimum acceptable annual production targets in million tonnes.



Initial Box Cut Dimensions

aW0: 20 aE0: 20 bN0: 20 bS0: 20 h0: 15

Final Pit Limits Dimensions

aW: 1040 aE: 440 bN: 260 bS: 360 h: 210

Minimum Mining Width

aWmin: 30 aEmin: 30 bNmin: 30 bSmin: 30

Production Targets

Maximum Annual Production (Mt): 19 Minimum Annual Production (Mt): 21

Mining Starting Point

X: 101600 Y: 600340

Figure 4.8 open pit specification tab

Mining starting point – the approximate coordinates of the mining starting point, the program simulates different starting point in each simulation episode.

The following parameters can be set through the Q-Learning parameters tab demonstrated in

Figure 4.9.

Epsilon – The probability that the agent "explores" by choosing a random action from the current state as opposed to "exploiting" its current knowledge and greedily selecting the state of highest estimated value.

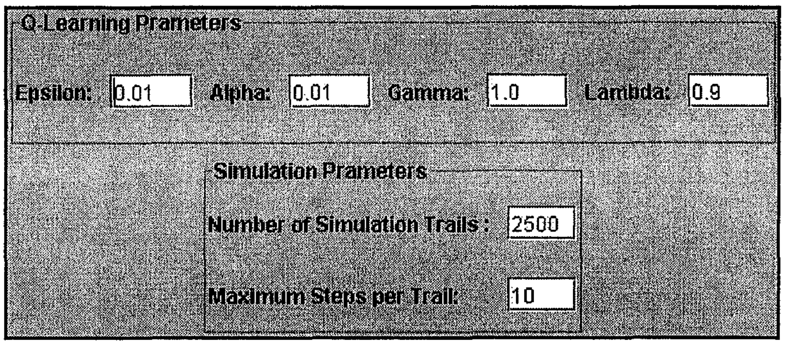
Alpha – The learning rate for a TDAgent.

Gamma – The discount rate for delayed rewards; a number in [0,1] used to quantify the current value of future rewards.

Lambda– The trace-decay parameter; a number in [0,1]. If lambda = 0, eligibility traces are not used.

Number of Trials – This defines the number of simulation iterations in a given experiment. It is actually the number of episode the agent simulates the dynamics of the open pit while learning through experience.

Maximum Steps Per Trial – This defines the maximum number of steps that will be taken in a given trial within an experiment.



Q-Learning Prameters

Epsilon: 0.01 Alpha: 0.01 Gamma: 1.0 Lambda: 0.9

Simulation Prameters

Number of Simulation Trails : 2500

Maximum Steps per Trail: 10

Figure 4.9 Q-Learning specification tab

4.5 Summary and Remarks

In summary, the models developed in Chapter 3 are used as the basis for the development of IOPS algorithm and subroutines. The algorithm starts with the interaction of two entities of the RL problem, the environment and the simulation. DOPS stochastically simulate the next feasible push-backs. The cash flows of the simulated push-backs are returned by the EPEM. The intelligent agent, IOPA, employs Q-learning and epsilon-greedy algorithm to choose the next suitable push-back among all the simulated results. The algorithm then updates the action value estimates $Q(s,a)$. The simulation continues until the simulated push-back design reaches the final pit limits. Afterward the algorithm resets all the variables and starts a new episode of simulation from the initial box cut.

The IOPS application is implemented in Java (Sun Microsystems, 1994-2006) and MATLAB (MathWorks, 2005) environment. This exercise consisted of class and object identification based on the Java Reinforcement Learning Library, JavaRL, (Kerr et al., 2003). The second half of this chapter dealt with the IOPS class UML diagrams and GUI development. The block model specifications, open pit production requirements, and the learning parameters are set through The IOPS user interface.

CHAPTER 5
EXPERIMENTAL DESIGN AND EXPERIMENTATION

5.1 Foreword

The verification and validation of the models are the next logical steps in this study. This chapter focuses on experimental design and experimentation. The mathematical and numerical models presented in Chapter 3 were transformed to algorithms and have been implemented as the IOPS application in Chapter 4. The Whittle software based on LG (Lerchs and Grossmann, 1965) algorithm was used to design the optimized final pit limits of an actual iron ore mine. The blocks within the boundary of the optimal pit were imported to the IOPS for the subsequent optimal long-term planning.

Verification and validation of the results are analyzed by comparing the results of the IOPS with Milawa algorithm used in Whittle software (Gemcom Software International, 1998-2005). The worst-case, best-case, and Milawa algorithms of Whittle are compared to the best case and the practical annual long-term schedule generated by the IOPS. The experiments compared the annual stripping ratio, average grade, annual waste and ore production, and the respective NPV.

COPS is used as a framework for experiments that are the basis for the mine tactical plans. The specification of the continuous-time open pit dynamics as a system of PDE's with the respective boundary conditions allow a unique solution to the long term schedule generated by the IOPS. COPS produces the trajectories of the changes of the open pit geometry in time and the respective amount of ore, waste, stockpile material, and the NPV of the operation over time. The results of the IOPS and COPS are compared to validate the employed numerical solutions.

5.2 Verification and Validation

Verification is one aspect of testing a product's fitness for a designed purpose. Whereas, verification deals with whether the design or the system has been built to meet the set of requirements or specifications, validation is concerned with determining whether the engineering process has produced the required product or design/system. Verification is the comparison of configuration items, components, subsystems, systems to their requirements to ensure correctness. Validation is the complementary aspect. Often one refers to the overall checking process as V & V.

At this stage we validate the IOPS models and implementation by answering the question that, did we build the right model? Or in other words does IOPS do what it was designed to do? To verify the IOPS implementation we would answer the question that, did we implement the model right? Or in other words does the developed application conform to the specifications?

As discussed in Chapters 3 and 4, the kernel of the IOPS architecture is the Q-learning algorithm. In this algorithm, the learned action-value function, Q, directly approximates Q^*, the optimal action-value function, independent of the policy being followed. This dramatically simplified the analysis of the algorithm and enabled early convergence proofs. The policy still has an effect in that it determines which state-action pairs are visited and updated. However, all that is required for correct convergence is that all pairs continue to be updated. Watkins and Dayan (1992) proved that the discrete case of Q-learning will converge to an optimal policy. If each pair (x,a) is visited an infinite number of times, then for lookup tables, Q-learning converges to a unique set of values $Q(x,a)=Q^*(x,a)$, which define a stationary deterministic optimal policy. Q-learning is an asynchronous algorithm. Each $Q(x,a)$ is updated one at a time, and the control policy may visit them in any order, so long as it visits them an infinite number of times. Watkins (1989) describes this algorithm as "incremental dynamic programming by a Monte Carlo method".

5.3 Experimental Design Framework for IOPS architecture

The problem solving methodology employed in IOPS framework consists of a solution paradigm based on Q-learning algorithm (Watkins, 1989), expected solution criteria and assumptions. In order to obtain reliable experimental results, the reinforcement learning paradigm employed in solving the problem must be able to capture the true definition or nature of the open pit planning problem, the data involved, some basic data structures, file management issues, tasks and subtasks involved in solving the problem as well as a generic control regime. The assumptions are based on prior knowledge of practical mining environments and the RL framework.

The general experimentation methodology employed in this study is outlined in Figure 5.1. The numerical models presented in Chapter 3 and the IOPS algorithm and application developed in Chapter 4 are validated through two case studies using data from an actual gold-copper mine and an iron ore mine.

In the study an iron ore geological block model is estimated by Kriging (Krige, 1951). Estimation is locally accurate and smooth, appropriate for visualizing trends and does not assess of global uncertainty, because there is just one model of the ore-body. If sequential Gaussian simulation was used to build the model, then there would be a number of realizations of that were equally probable. Simulation reproduces histogram, honors spatial variability, and allows an assessment of uncertainty with alternative realizations of reality.

The optimized pit limits are designed using LG (Lerchs and Grossmann, 1965) algorithm. Subsequently the best-case and the practical annual long-term schedule generated by IOPS and the shells node in Whittle Four-X are compared. The experiments will compare the annual production, stripping ratio, average grade, and the respective NPV. The results of the experiments design will be subjected to sensitivity and statistics analysis. Although optimality is not guaranteed mathematically with the parametric analysis in Whittle 4-X, but it is a strong tool for identifying high grade ore clusters in the model and is the standard tool used in industry today.

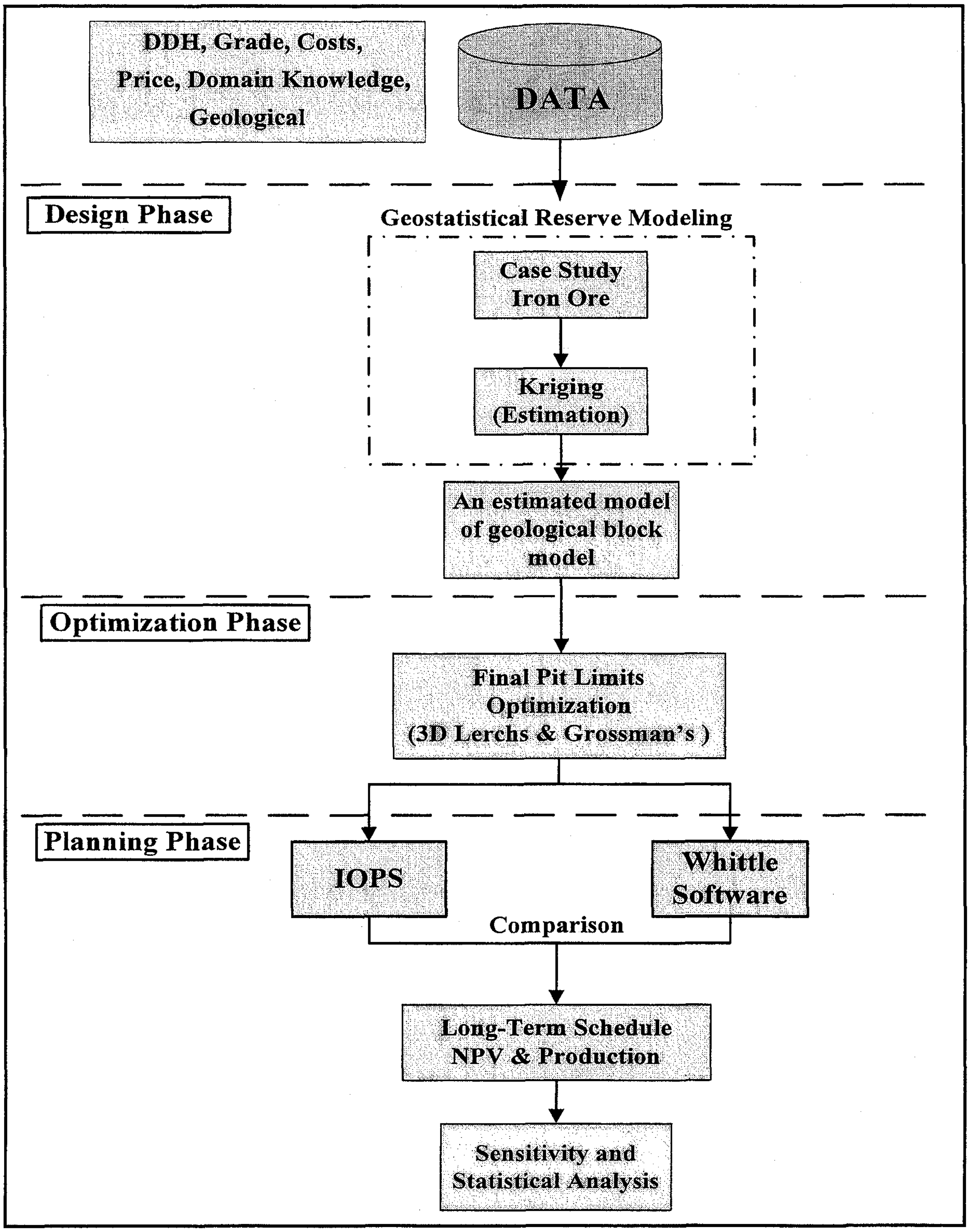


Figure 5.1 Design, optimization, and planning experiments

5.4 Case Study: Gol-E-Gohar Iron Ore Mine

Gol-E-Gohar iron ore deposit is situated in the province of Kerman, Iran, almost in the center of an imaginary triangle with its corners on the cities of Kerman, Shiraz, and Bandar Abbas. The mine is located approximately 1740 m above sea level in an area of planer topography. The aero magnetic survey carried out in 1974 indicated six separate anomalous areas in the immediate surroundings of the area of study. The iron ore deposit is explored with 159 exploration drill holes and 113 infill drill holes totaling 6000m of drilling illustrated in Figure 5.2.

The ore-body is one of several magnetite iron ore-bodies situated in an east-west trending belt of indeterminate age. The ore-bodies occur within a belt of metamorphic schist and gneisses. An idealize cross section of the deposit looking west is illustrated in Figure 5.3. Three types of ore, top magnetite, oxide and bottom magnetite are classified in the deposit. The processing plant is based on magnetic separators so the main criterion to send material from mine to the concentrator is weight recovery. It is predicted that if all the mineable reserve is extracted the total concentrates produced will have an average grade of 0.49% sulphur and 0.037% phosphorus and the average anticipated weight recovery will be about 66%.

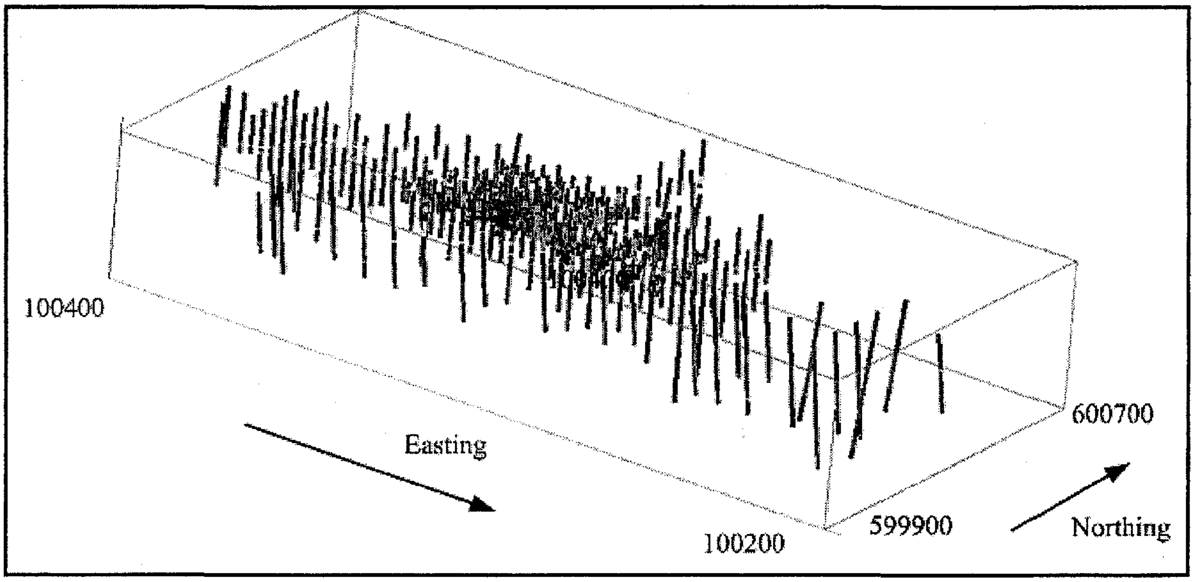


Figure 5.2 Iron ore deposit borehole locations

The deposit is divided by a 100m×100m local grid. The geological sections are referred to as profile lines (north-south cross sections through the ore-body) and reference lines (east-west cross sections through the ore-body). Profile lines are available at 100m intervals from 100400-east to 102300-east, supplemented by intermediate sections in the central pit area. Reference lines are maintained at 100m intervals from 599900-north to 600700-north, supplemented by denser coverage in the central area. A large block of 1900m×800m×225m was defined from 100400-east to 102300-east on the X axis; from 599900-north to 600700-north on the Y axis and from level 1515m to 1740m on Z axis in a way that includes all the ore-body and the related waste and overburden blocks. Figure 5.4 and Figure 5.5 demonstrate the first topography of the area of study before that any mining activity starts.

Kriging (Krige, 1951), as a procedure of constructing a minimum error variance linear estimate, has been used to build the geological block model. The initial model contains 152,000 blocks that makes an array with dimensions of 95×80×20 blocks. About 18% of these blocks are located above the first topography of the mine and are flagged as air blocks and almost 124,000 blocks are flagged as overburden, waste, and ore. The primary examination of the geological block model revealed that three top levels

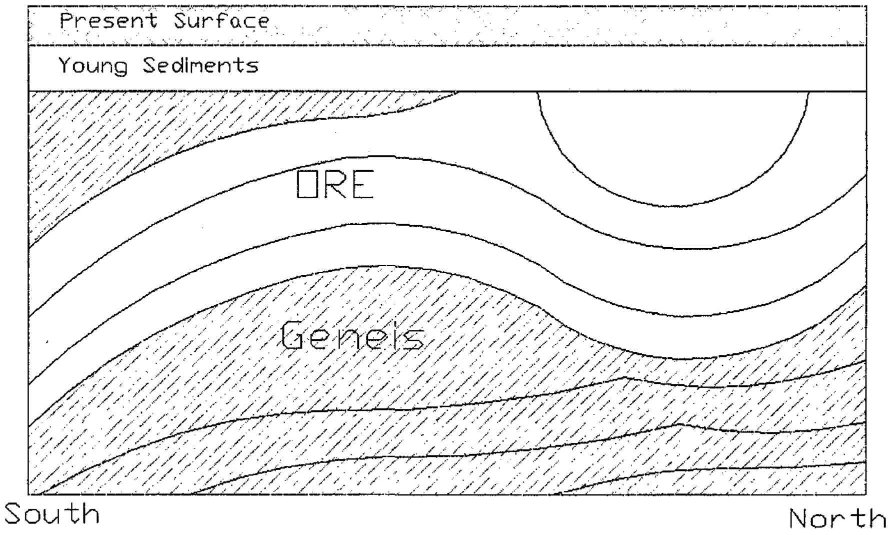


Figure 5.3 Idealize schematic cross section of the deposit looking toward west

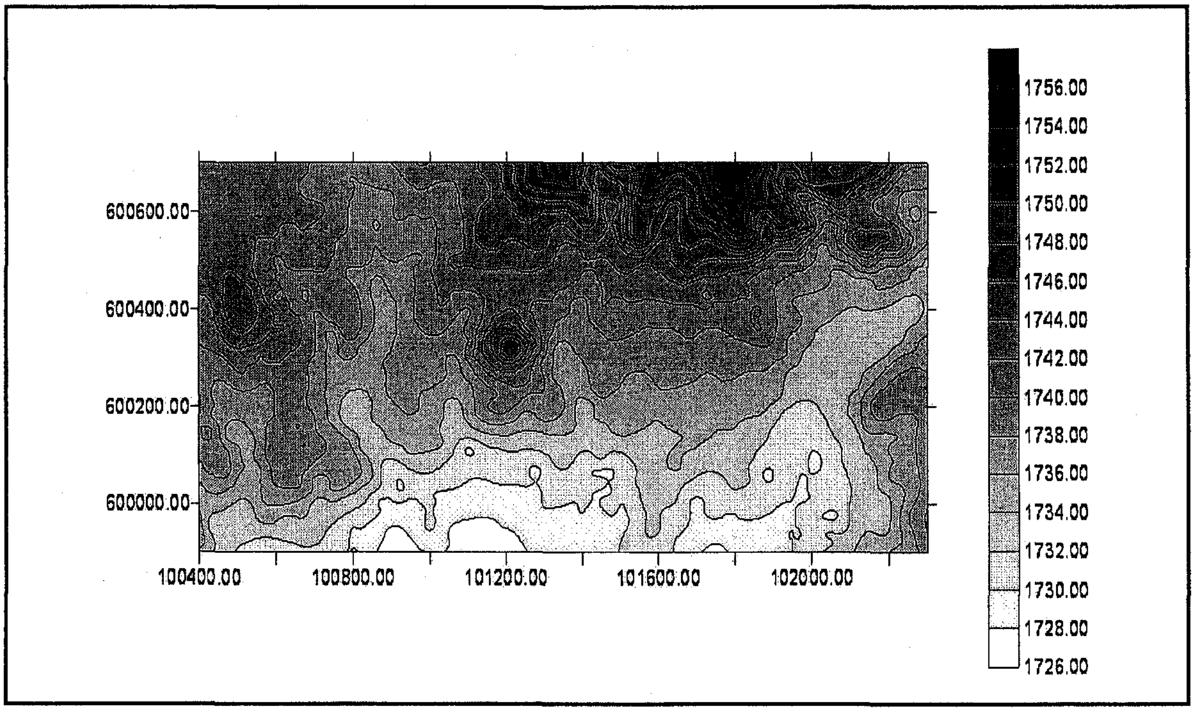


Figure 5.4 First topography of the area of study

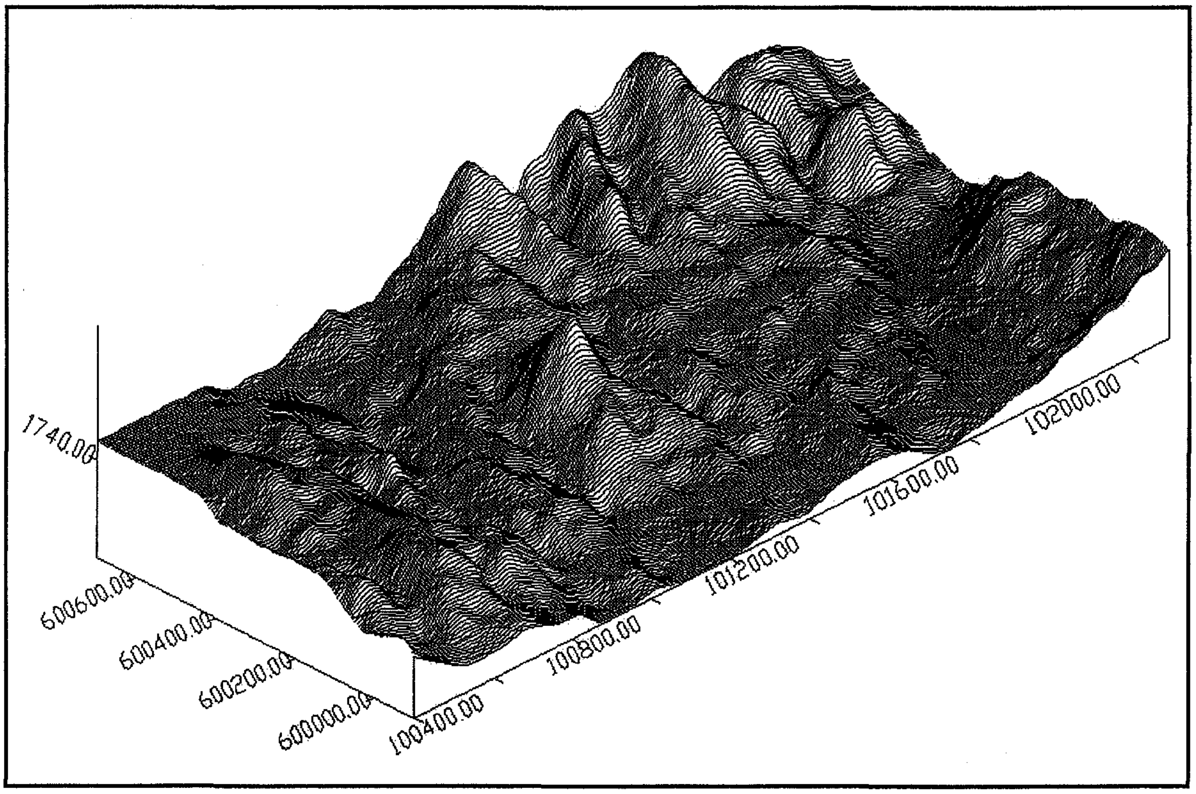


Figure 5.5 3D view of the first topography

of the

of the block model are all air blocks and two bottom levels of the data are in waste. The block model was modified and the aforementioned blocks were omitted from the model. The modified model is based on small blocks representing a volume of rock equal to 20m×10m×15m, which is approximately equal to the production of the mine per day. This model contains 114,000 blocks that makes a model framework of dimensions 95×80×15. Figure 5.6 illustrates a multi cross section of the deposit along sections 101800-east; 600170-north; and elevation of 1620m.

The block model contains almost 243 million tonnes of indicated recourse of iron ore with the average grade of 63%. Table 5.1 summarizes the ore and waste in the geological block model. Figure 5.7 illustrates the average grade, total amount of ore and total amount of Fe on a bench by bench basis.

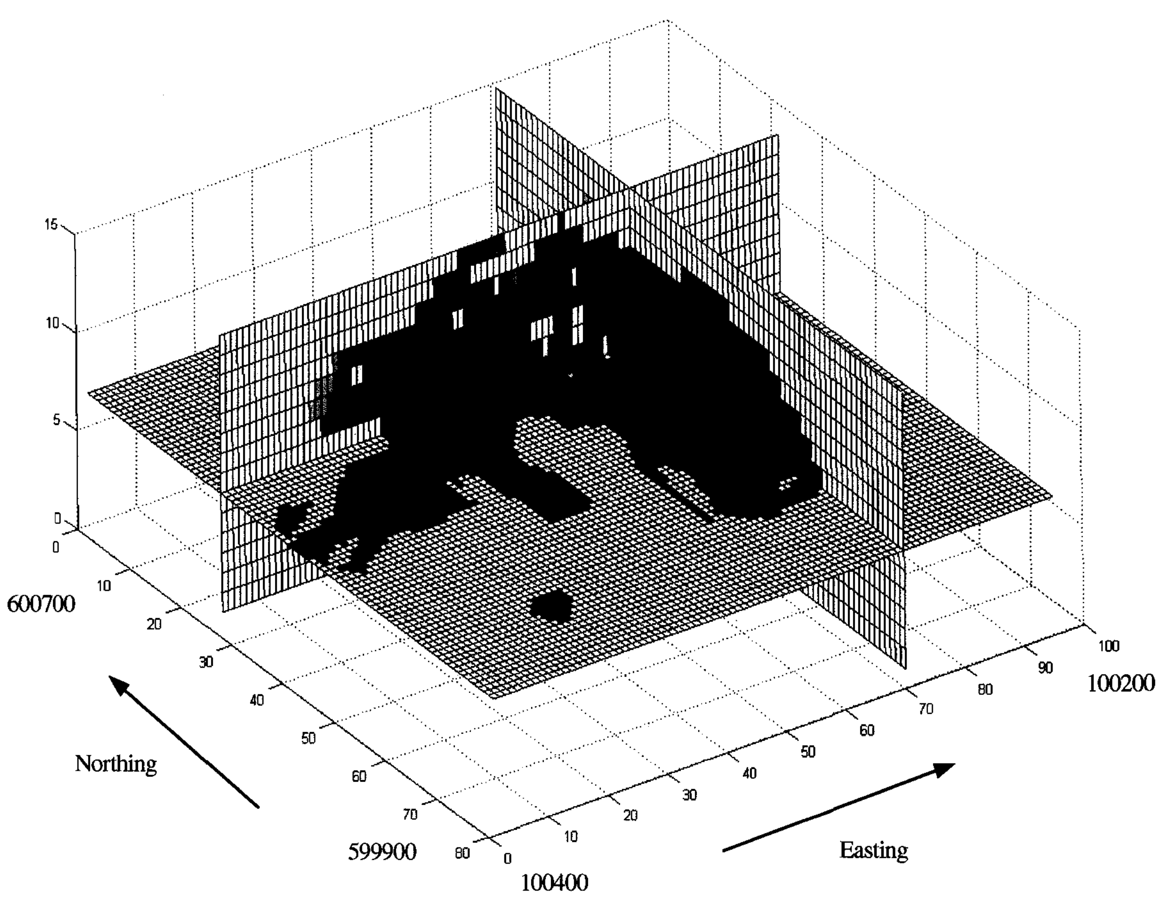


Figure 5.6 3D view of the deposit

Table 5.1 Summary of the ore and waste in the geological block model

Rock Type	Blocks in model	Total (Mt)	Total Fe element (Mt)	Grade % Min	Grade % Avg	Grade % Max
Ore	19328	243.533	159.140	13	63.5	89
Waste	94672	1192.867	-	-	-	-

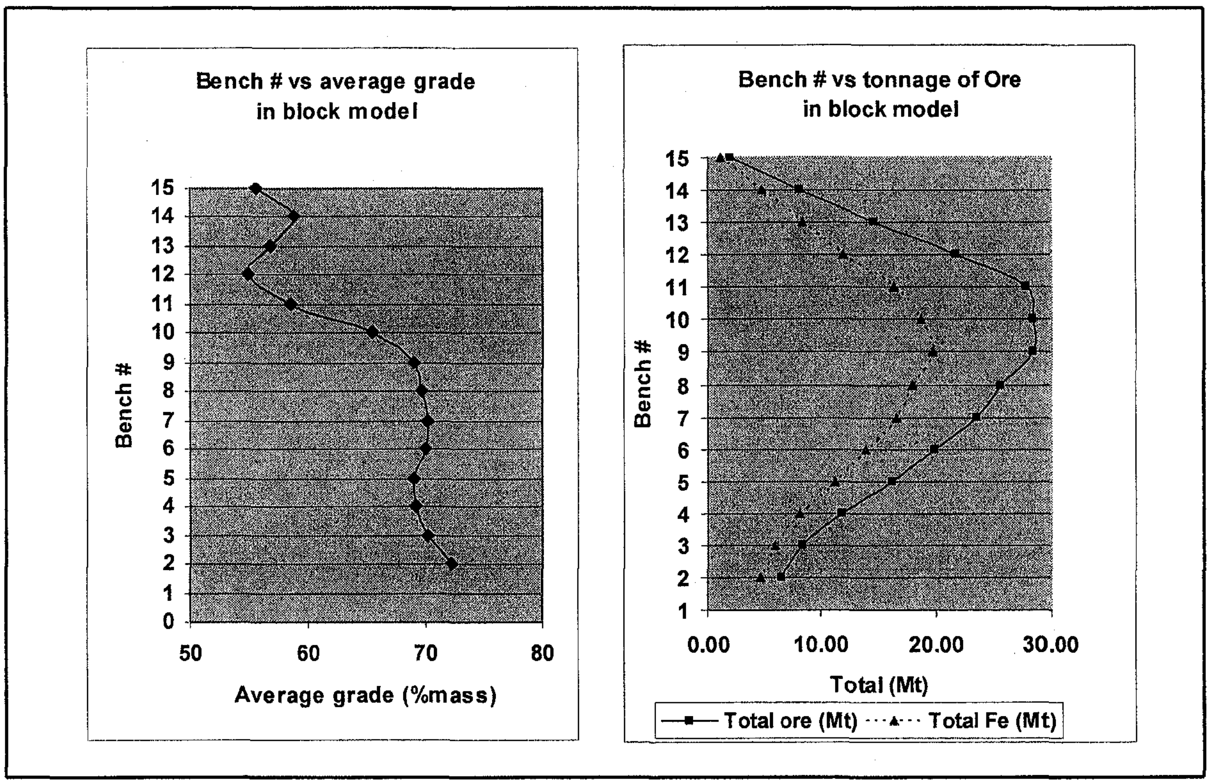


Figure 5.7 bench by bench report of the geological block model

5.4.1 Iron Ore Optimized Final Pit Limits Design

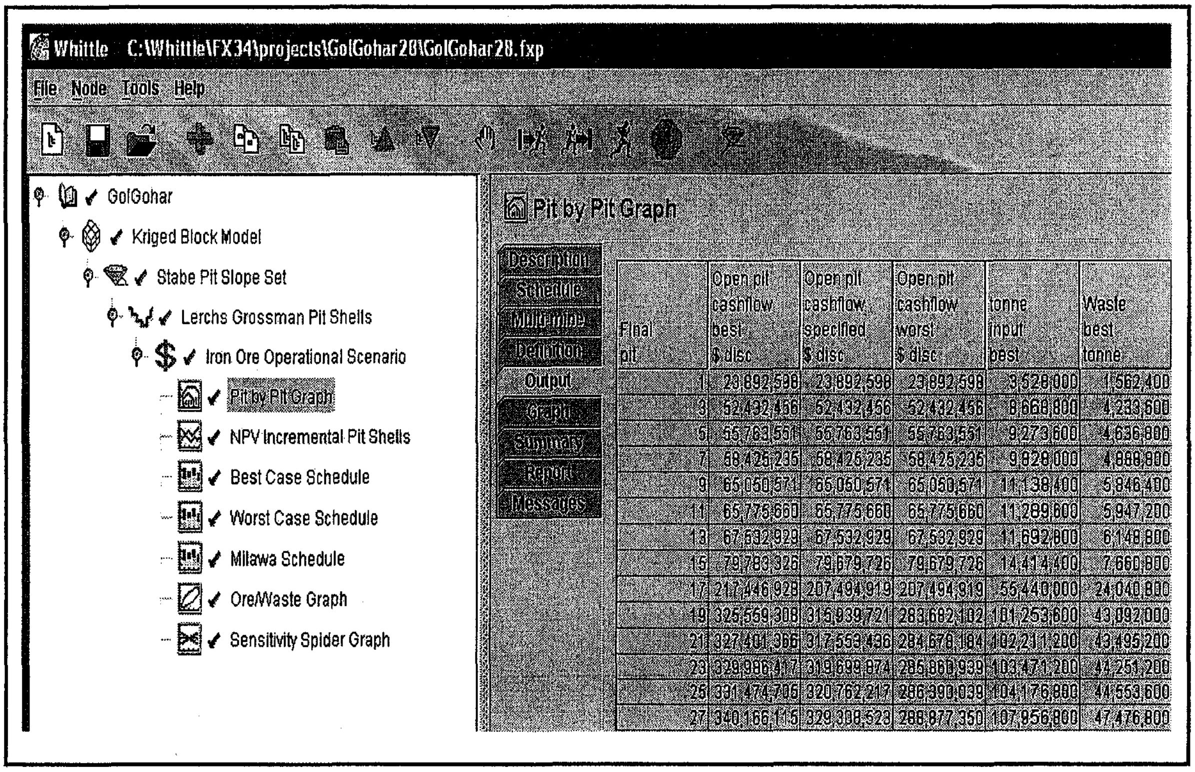
Lerchs and Grossmann (1965) have used the graph theory for optimum design of open pit mines. The objective of their studies was defined as the design of the contour of a pit so as to maximize the difference between the total mine value of ore extracted and the total extraction cost of ore and waste. The LG algorithm is mathematically proven to generate a rigorous optimum solution, based on maximum undiscounted cash flow as the criterion for optimization. The Whittle software (Gemcom Software International, 1998-2005), which have been utilized in this study is based on LG algorithm. The software is the standard current tool used in industry. The method works on a block model of the ore body, and progressively constructs lists of related blocks that should, or should not, be mined. The final lists define a pit outline that has the highest possible total value, subject to the required pit slopes.

Figure 5.8 illustrates the geological block model imported to the Whittle software environment. Slope stability and geo-mechanical studies suggested a 43 degree overall slope in all regions. The average slope error in Whittle model is 0.9 degree and there are 35 possible structure arcs per block in the model which in total makes 3,075,666 arcs or edges in the graph model. The Pit Shells (Figure 5.8) node in Whittle represents a set of pit shells generated by economic parametric analysis using the LG algorithm. This process reads in the block model from the Block Model node, pit slope constraints from Slope Set node, calculates block values using the economic and operational data contained in this node, and produces optimal pit outlines. The economic and mining parameters are based on: (i) mining cost = \$2/tonne; (ii) processing cost = \$2/tonne; (iii) selling price = \$15/tonne (Fe); (iv) maximum mining capacity = 20 Mt/year; (v) maximum milling capacity = 15 Mt/year; (vi) density of ore and waste = 4.2 $tonne/m^3$; and (vii) annual discount rate = 10%.

It is usual to produce multiple pit outlines in a single run and this process is controlled by the revenue factors in the optimization tab. The program finds a sequence of optimal push-backs based on varying the profitability of the deposit. A pushback is the incremental change in the geometry of the pit that results from a change in net block value. The underlying theory is that a sequence of push-backs can be identified as mining limits over time that will maximize the project's NPV. In the generation of the pit shells, revenue factors in the range of 0.45 to 1.4 were used with variable geometric step sizes to scale base case price up and down, in order to control what nested pits are to be produced.

It should also be considered that selection of a final pit has direct impact on the expected economic ore reserve. If the pit selected is markedly different from the optimum, it can result in less than optimal selection of mine and mill capacity. Leading to loss of capital efficiency and hence to lower project value. In general in homogenous ore-bodies there is no obvious choice of pushback and a sequence of revenue factors will produce a steadily larger set of push-backs associated with deeper or lower grade ore. Alternatively, in complex geometry, multiple ore zones and mineral concentration may result in a corresponding number of highly distinct push-backs. As a result, there will be no change in pushback geometry over a wide interval of revenue factors but once the deposit value is sufficiently great, the ore value for a more deeply covered portion of the deposit will exceed stripping costs resulting in a major expansion of the pit.

The sequences of pit expansions are the basis of mine planning with Whittle and they correspond roughly to the manner in which the pit geometry will evolve over time. In terms of maximizing NPV, the lowest revenue factor that produces a pit sufficiently large to justify mining should also be the portion of the deposit to be mined first. Likewise, by increasing the revenue factor sufficiently above that required by the first pushback, a second pushback will be found and so forth until a sequence of push-backs are identified which identify the highest NPV evolution of the pit over time.



Final pit	Open pit cashflow best $ disc	Open pit cashflow specified $ disc	Open pit cashflow worst $ disc	tonne input best	Waste best tonne
1	23,892,598	23,892,598	23,892,598	3,528,000	1,562,400
3	52,432,456	52,432,456	52,432,456	8,668,800	4,233,600
5	55,763,551	55,763,551	55,763,551	9,273,600	4,636,800
7	58,425,235	58,425,235	58,425,235	9,828,000	4,888,800
9	65,050,571	65,050,571	65,050,571	11,138,400	5,846,400
11	65,775,660	65,775,660	65,775,660	11,289,600	5,947,200
13	67,532,929	67,532,929	67,532,929	11,692,800	6,148,800
15	79,783,326	79,679,726	79,679,726	14,414,400	7,660,800
17	217,446,928	207,494,919	207,494,919	55,440,000	24,040,800
19	325,559,308	315,939,721	283,682,102	101,253,600	43,092,000
21	327,401,366	317,553,436	284,678,184	102,211,200	43,495,200
23	329,986,417	319,699,874	285,860,939	103,471,200	44,251,200
25	331,474,706	320,762,217	286,390,039	104,176,800	44,553,600
27	340,166,115	329,308,523	288,877,350	107,856,800	47,476,800

Figure 5.8 Whittle interface

The results of the Shells Node generated 77 nested pits with the respective total amount of ore, waste, and the NPV illustrated by Figure 5.9. There is a need for a general long-term push-back design to choose the final pit with the highest NPV among the 77 shells generated.

Estimation of a project's NPV requires that timing of cash flow be accurately known in order to apply an appropriate discount factor. This immediately introduces a problem for pit optimization software because the year of mining for any block of ore or waste will not be known until the mine production has been scheduled. The LG algorithm, which is the basis for Whittle software treats all mining activities as though they occur simultaneously, with no discount factor applied. This usually results in the selection of a final pit that is larger than the true maximum NPV pit.

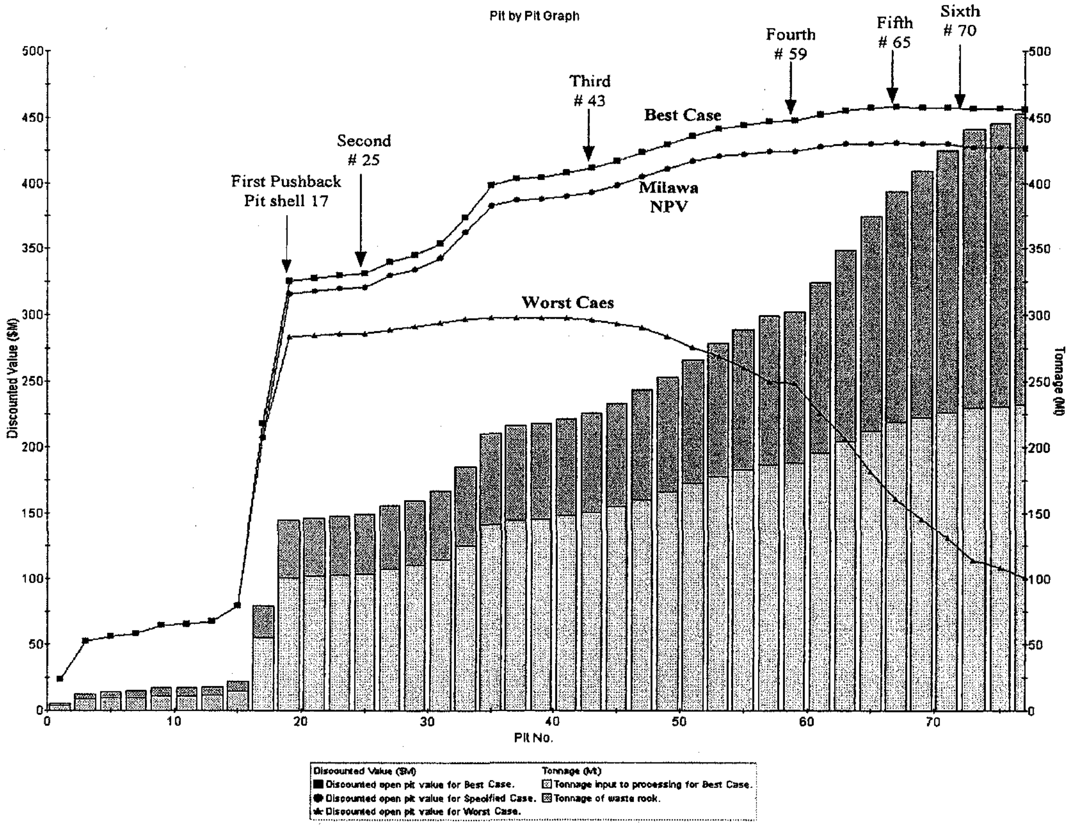


Figure 5.9 Push-backs selection in pit by pit analysis

Calculating the NPV requires knowing the relative time difference between blocks mined within a particular pit shell. This is dependent on the mill and mine capacities, practical sink rate (benches mined per year) and the equipment that can be practically operated within a specific cutback. In pits with high marginal stripping ratio and extreme changes in topography, the timing differences in final pit push-backs can be substantial. It is therefore important to have a reasonably accurate mining schedule in order to properly select the optimum pit.

In terms of production scheduling, the series of nested pits provides limits on where to mine and when, but they do not identify a production schedule in terms of the optimal extraction period for a block. Instead, the nested pits can be used to define the feasible region of production scheduling problem. Whittle provides a number of methods that work with the set of nested pits to provide a feasible production schedule: best case, worst case, and Milawa NPV.

The best case schedule is associated with completely mining a pushback before proceeding to the first bench in the subsequent pit. In this manner the highest valued ore is mined as early as possible maximizing NPV. When there is insufficient mining width between push-backs this may not be possible. Thus, the best case may not be feasible without modifying the bench width between push-backs and thereby violating the optimality of the push-backs.

In the worst case schedule the entire bench across all push-backs is mined prior to proceeding to the second bench. This amounts to pre-stripping the entire deposit, defers ore production and thereby minimizes cash flow by placing stripping cost up front while delaying positive revenue.

Milawa defines a variable bench interval between subsequent push-backs such that once a fixed number of benches have been mined out in the interior push-back then mining can commence on the next pushback. Thus, there is always a vertical lag of so many benches between push-backs. Milawa allows the lag to vary between push-backs and then searching for the combination of lags which is optimal either in the sense of cash flow or of balancing the removal of ore and waste.

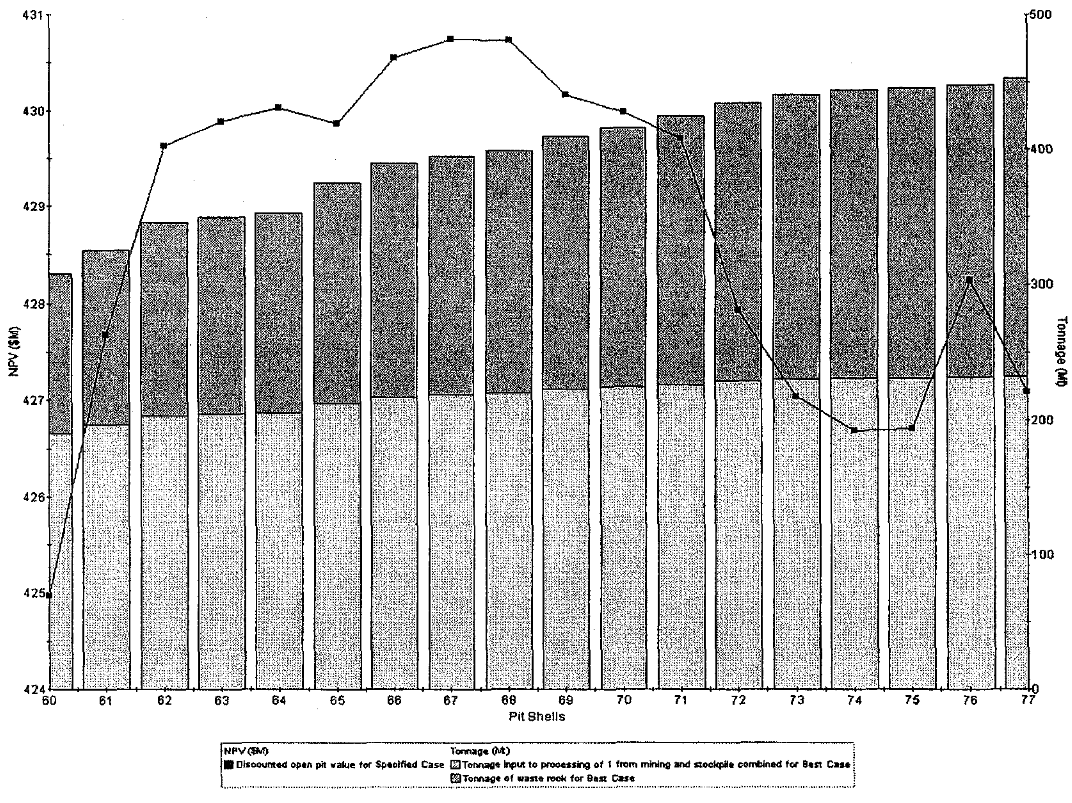


Figure 5.10 Incremental pit shells NPV

Figure 5.9 illustrates the pit by pit analysis of 77 pit shells with their respective NPV for the best case, worst case, and Milawa algorithm. The appropriate push-backs are chosen in a way that the annual production targets are met in the long-term plan. The selected phases are represented by pits 17, 25, 43, 59, 65 and the final pit expected around pit 70. Successive schedules are run to different final pits from the first push-back to the pit shell number 77 in incremental steps of 1. Figure 5.10 demonstrates the pit shells 60 to 77 and their relevant NPV. Figure 5.11 illustrates the pit shell 68 with 220 million tonnes of ore and 179 million tonnes of waste and an NPV of $430,734,000, for the Milawa schedule. The pit shell has the highest NPV among all other pit shells and is chosen as the final pit limits for the production scheduling stage.

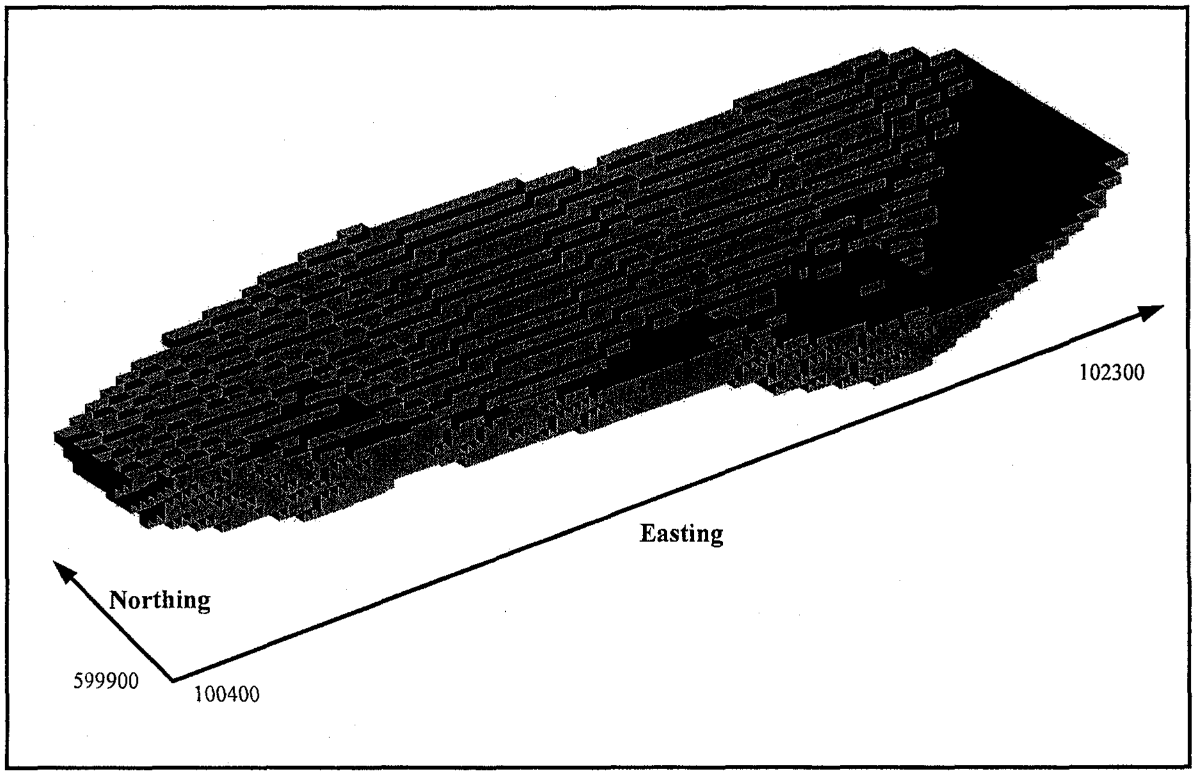
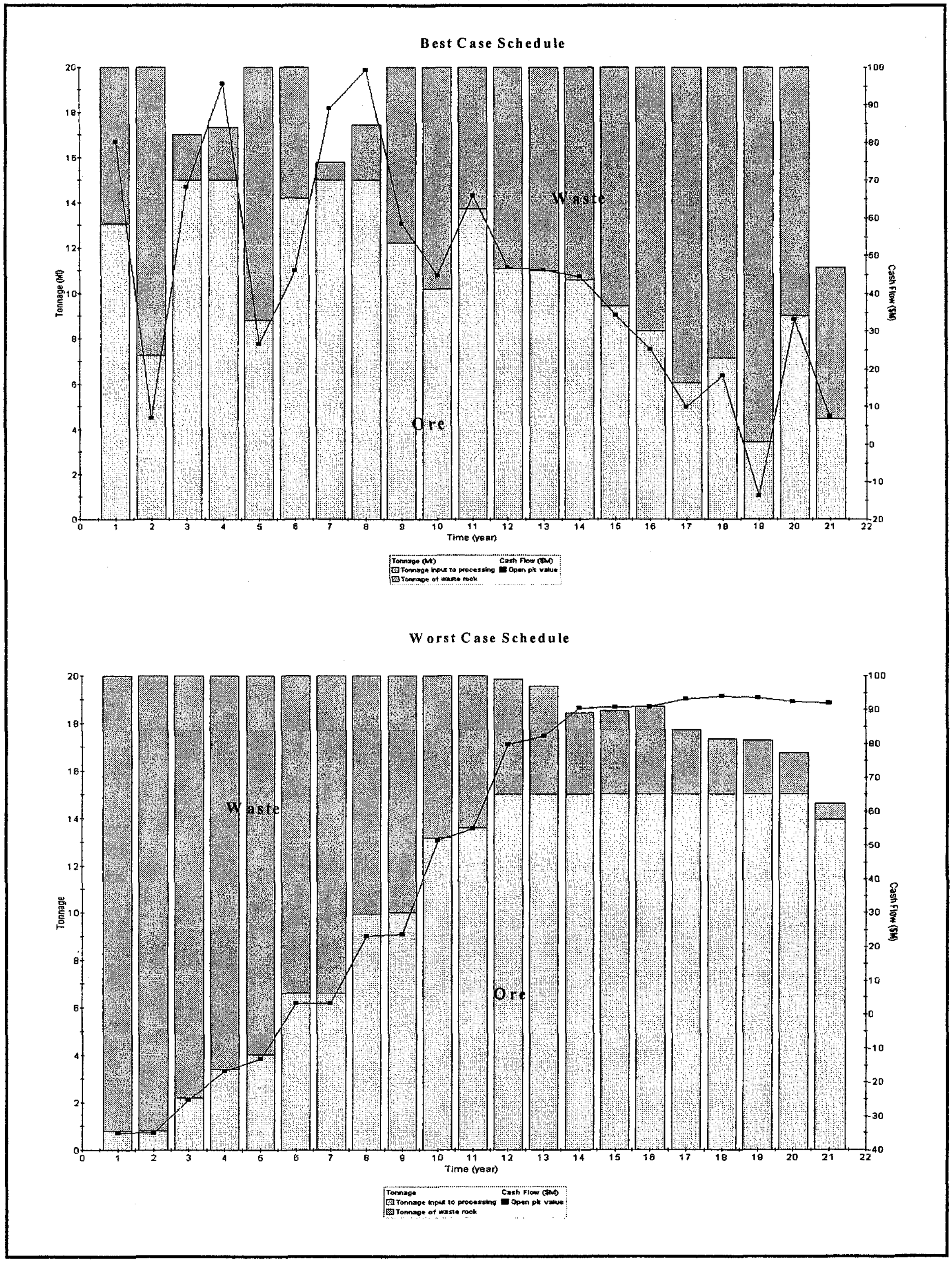


Figure 5.11 Pit shell 68, the optimized final pit limit

5.4.2 Whittle Long-Term Schedule

The optimal long term schedule is planned within the boundary of final pit limits designed by LG algorithm in the last section. For scheduling purposes, it is necessary to set limits on the throughputs for mining, and processing. In this case, the operational scenario includes the following limits: a processing limit of 15 million tonnes per annum and a total mining limit (ore and waste) of 20 million tonnes per year. The revenue factors are set ranging from 0.45 to 1.4. The increment on the revenue factor was refined in several passes to more clearly identify a set of critical push-backs having sufficient tonnage for mining. The Pit by Pit tool was used to identify the maximum valued pit as the pit shell 68 (Figure 5.11) and the sequence of critical push-backs were chosen as 17, 25, 43, 59, 65, and 68 as the basis for production scheduling.

The worst and best case schedules within the nested pits are generated. The worst case scenario demonstrates an NPV of $155,753,000, where the best case has an NPV of



Best Case Schedule

Worst Case Schedule

Figure 5.12 Best case and worst case schedule

$457,676,000. These schedules are shown in Figure 5.12 with the respective annual cash flows. In both cases, there is little control over the production stream as the amounts of ore and waste are fixed by the geometry of the push-backs in combination with the best and worst case scenario. What can clearly be seen is the pre-stripping and revenue deferment that results from the worst case and the erratic ore and waste production, typical of the best case schedule. This probably would not be acceptable even if sufficient working room is available between push-backs. Thus, the best case is not feasible without modifying the bench width between push-backs and thereby violating the optimality of the designed push-backs.

The Milawa algorithm is used to generate a feasible schedule. The algorithm does not generate and evaluate all feasible schedules, as the number of feasible schedules in any analysis is extremely large. Rather, it strategically samples the feasible domain and gradually focuses the search until it converges on its solution.

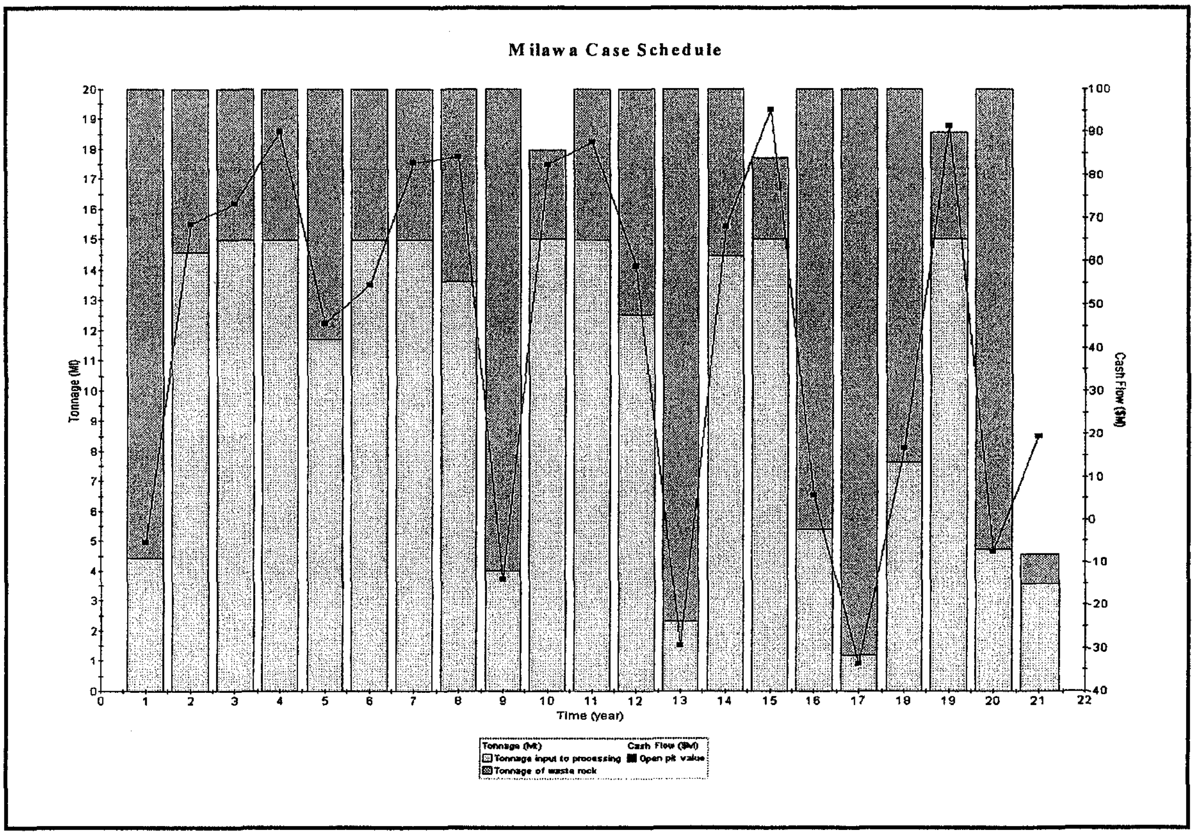


Figure 5.13 Milawa schedule

The algorithm is set to use push-backs to pits 17, 25, 43, 59, 65, and 68. This represents a solution with an NPV of $430,734,000 (Figure 5.13), which provides a compromise between practicality and the highest theoretical NPV. Figure 5.14 shows cumulative ore vs. waste by production period plotted for each of the three scheduling scenarios.

Figure 5.15 illustrates how sensitive the discounted pit value is to variations of product selling price and mining and processing costs. The variation of parameters is set in the range of -10 to 10 percent. The most sensitive variable in this study is the metal price causing the NPV to fluctuate between $330 million and $530 million.

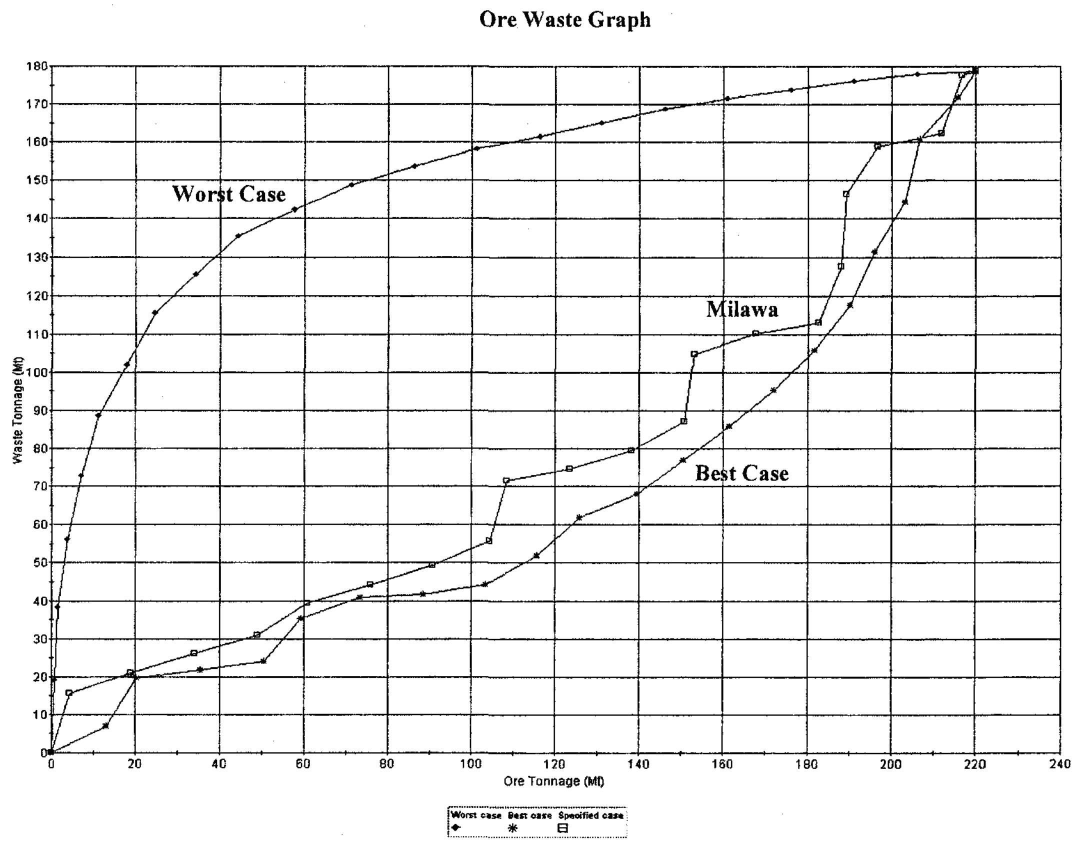


Figure 5.14 Cumulative ore vs. waste by production period

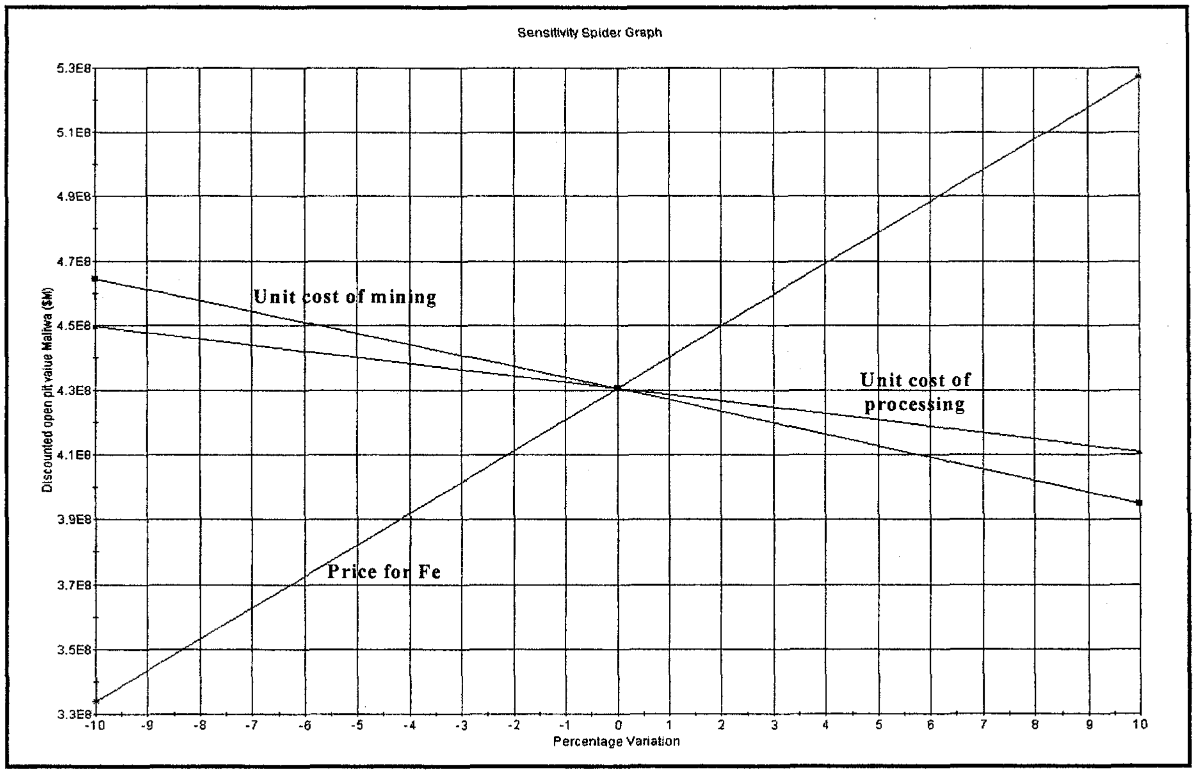


Figure 5.15 Sensitivity analysis of discounted pit value to selling price and costs

5.4.3 IOPS Long-Term Schedule

The IOPS starts with the interaction of two entities of the RL problem. The blocks within the final pit limits and the simulation of push-backs. DOPS stochastically simulates the next feasible push-backs. The cash flows of the simulated push-backs are returned by the EPEM. The intelligent agent, IOPA, employs Q-learning and epsilon-greedy algorithm to choose the next suitable push-back among all the simulated results. The algorithm then updates the action value estimates $Q(s,a)$. The simulation continues until the simulated push-back design reaches the final pit limits. Afterward the algorithm resets all the variables and starts a new episode of simulation from the initial box cut.

Figure 5.16 illustrates the blocks within the boundary of the optimized final pit limits designed by LG algorithm imported to the IOPS. The blocks within each region of the modified geometrical model are demonstrated in different colors.

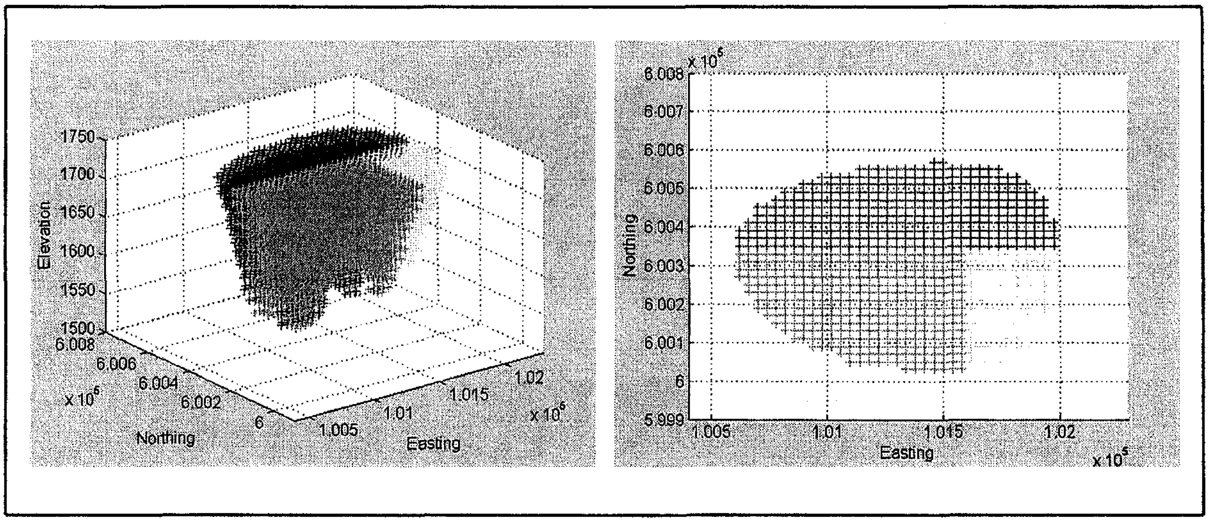


Figure 5.16 Final pit limits captured by the modified geometrical model

Through the planning phase the conditions and parameters are kept identical to the Whittle case study. The IOPS parameters are set at the initialization stage as follows (Figure 5.17):

Initial box cut dimensions – the dimensions of the initial box cut in meter, representing the boundary condition for the differential equations in COPS.

$a_{W0} = 20m; a_{E0} = 20m; b_{N0} = 20m; b_{S0} = 20m$ and $h_0 = 15m$

Final pit limits dimensions – the final pit limits dimensions in meter, captured by the modified frustum geometrical model.

$a_W = 1040m; a_E = 440m; b_{N0} = 260m; b_{S0} = 360m$ and $h_0 = 210m$

Minimum mining width – the minimum possible mining width at the bottom of the pit

$a_{W_{\min}} = 30m; a_{E_{\min}} = 30m; b_{N_{\min}} = 30m; b_{S_{\min}} = 30m$

Production targets – maximum and minimum acceptable annual production targets in million tonnes.

$mean\_production = 20Mt$; $\min\_production = 19Mt$; $\max\_production = 21Mt$

Mining starting point – the approximate coordinates of the mining starting point, the program simulates different starting point in each simulation episode based on the coordinate $X = 101600$ and $Y = 600340$

Maximum number of benches mined per year – 3 benches

Q-Learning parameters are set as follows:

Epsilon – The probability that the agent "explores" as opposed to "exploiting". $\varepsilon = 0.01$

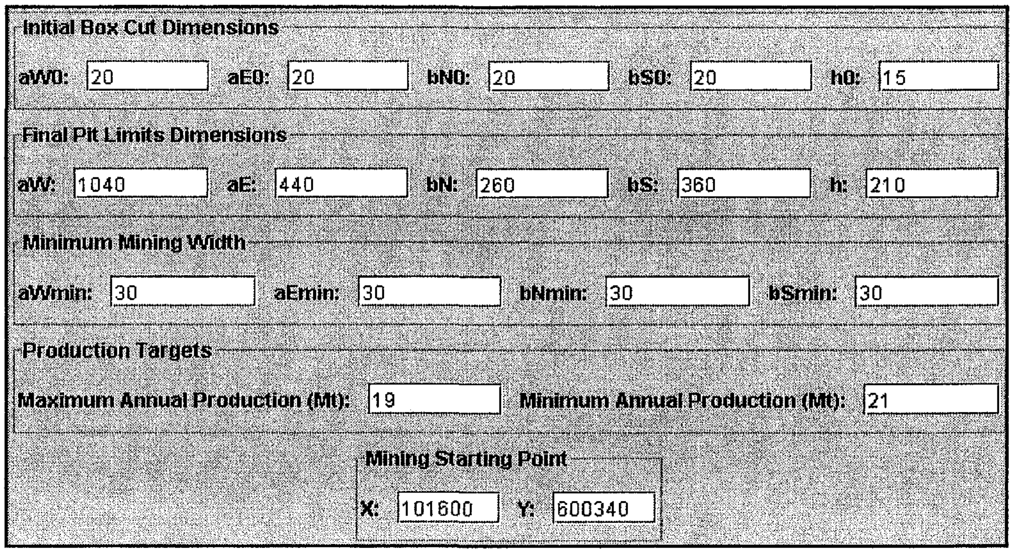
Alpha – The learning rate for a TDAgent. $\alpha = 0.01$

Gamma – The discount rate for delayed rewards. $\gamma = 0.1$

Lambda– The trace-decay parameter is a Boolean number. $\lambda = 0$ eligibility traces are not used.

Number of Trials – Defines the number of simulation iterations in a given experiment. In this experiment it is set to 3000.

Maximum Steps Per Trial – This defines the maximum number of steps that will be taken in a given trial within an experiment which is equal to 10 in this study.



Initial Box Cut Dimensions

aW0: 20 aE0: 20 bN0: 20 bS0: 20 h0: 15

Final Pit Limits Dimensions

aW: 1040 aE: 440 bN: 260 bS: 360 h: 210

Minimum Mining Width

aWmin: 30 aEmin: 30 bNmin: 30 bSmin: 30

Production Targets

Maximum Annual Production (Mt): 19 Minimum Annual Production (Mt): 21

Mining Starting Point

X: 101600 Y: 600340

Figure 5.17 Parameter initialization

IOPS was used to run Q-learning simulation with 3000 iterations with different scenarios of mining starting points. The best-case scenario proposed a starting point at 10160-east and 600340-north, which is located inside the smallest pit generated with nested pits by Whittle. Figure 5.18 illustrates the annual changes in the dimensions of the modified geometrical model a_W, a_E, b_N, b_S, and h. Figure 5.19 to Figure 5.22 demonstrate the push-backs of the optimal IOPS schedule.

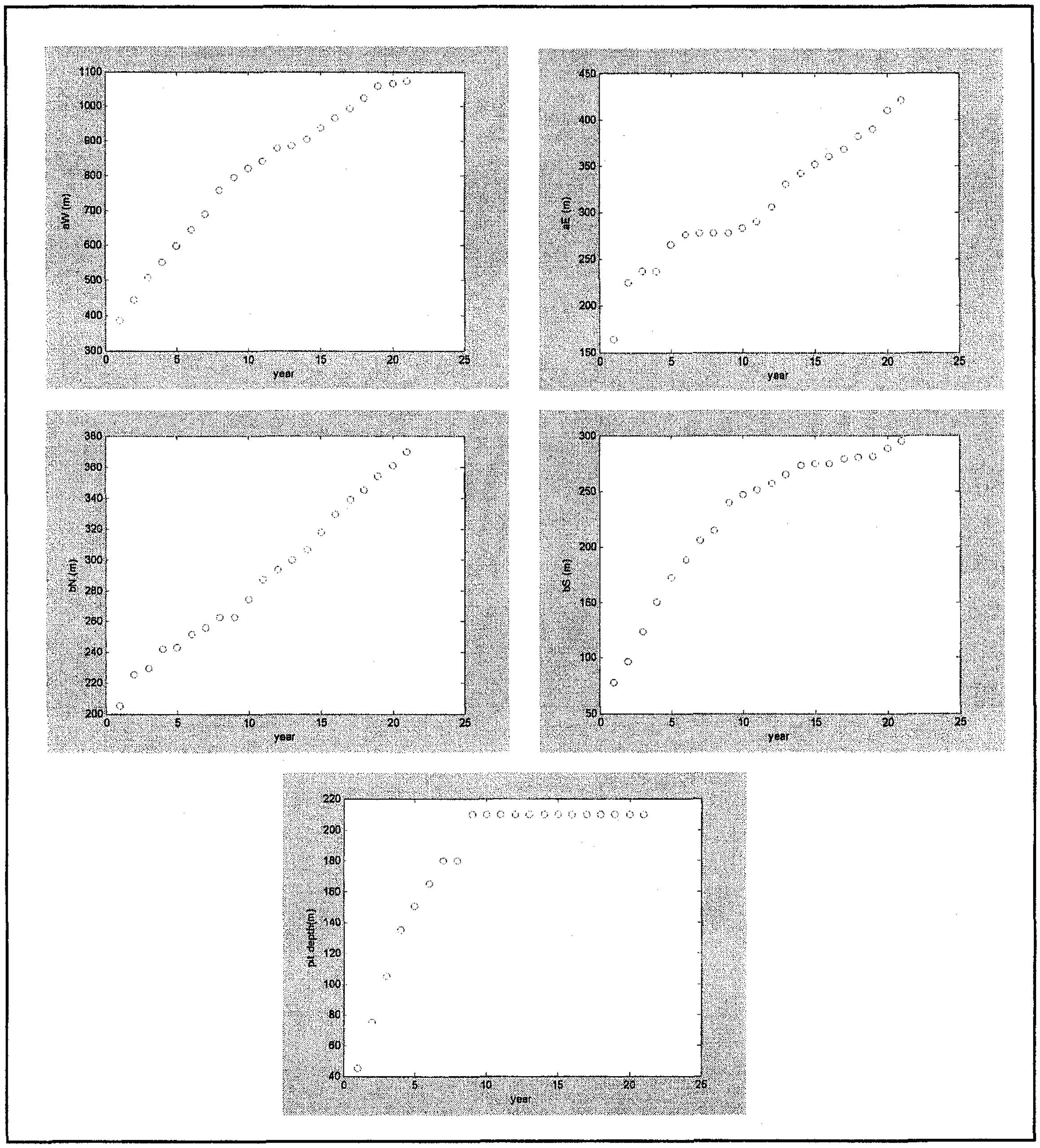


Figure 5.18 Annual changes in the dimensions of the open pit a_W, a_E, b_N, b_S and h (m)

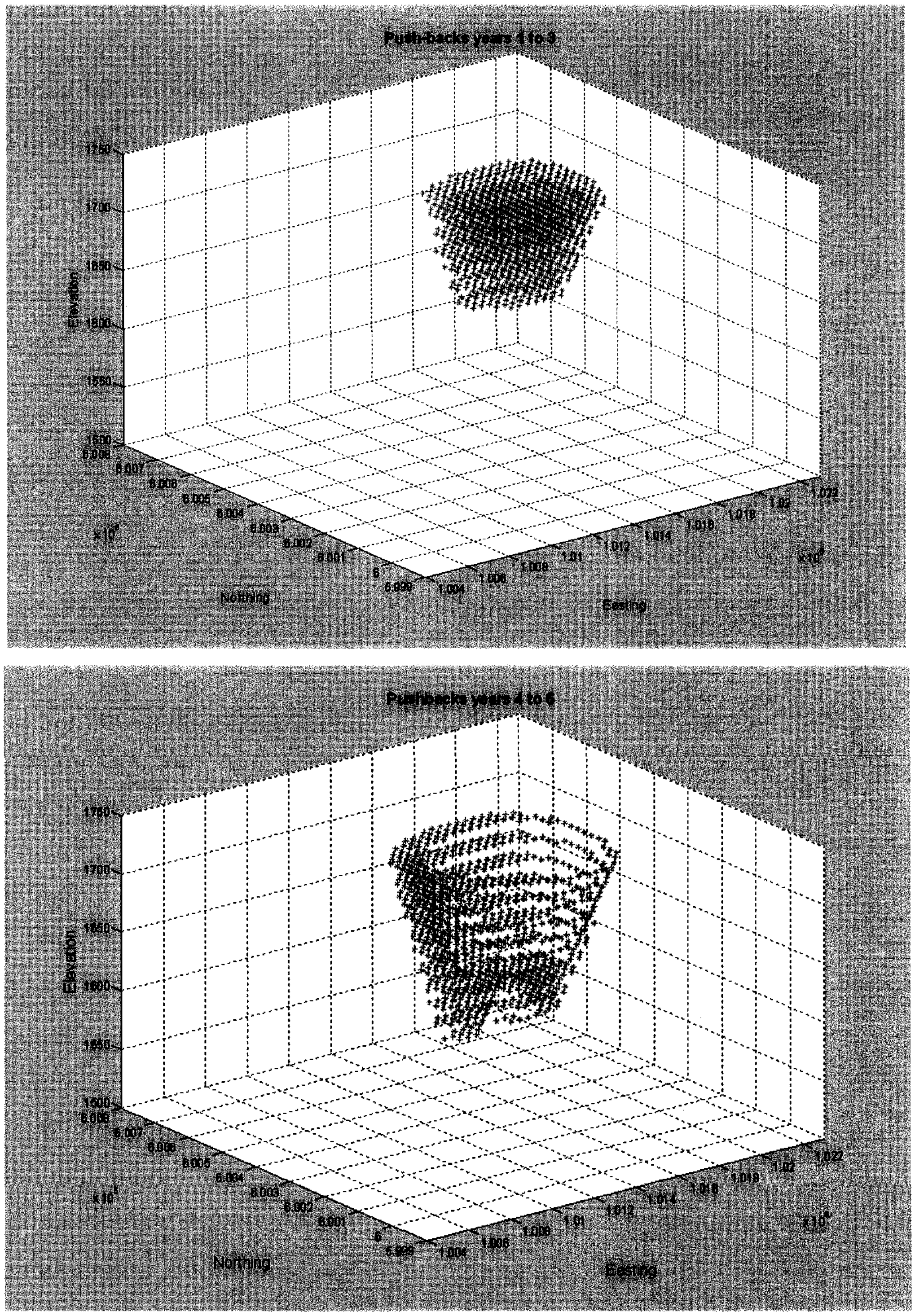


Figure 5.19 Push-backs years 1 to 6

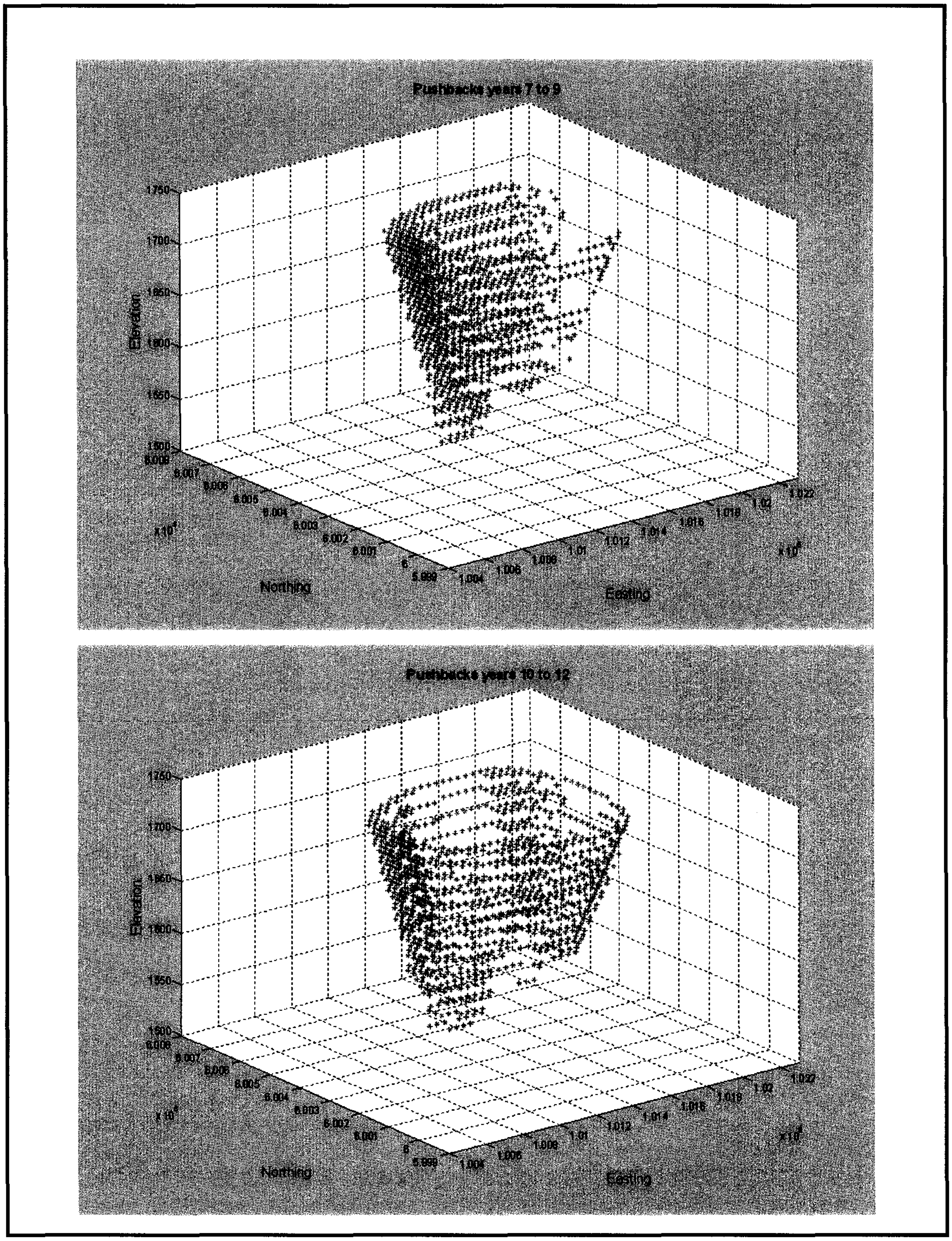


Figure 5.20 Push-backs years 7 to 12

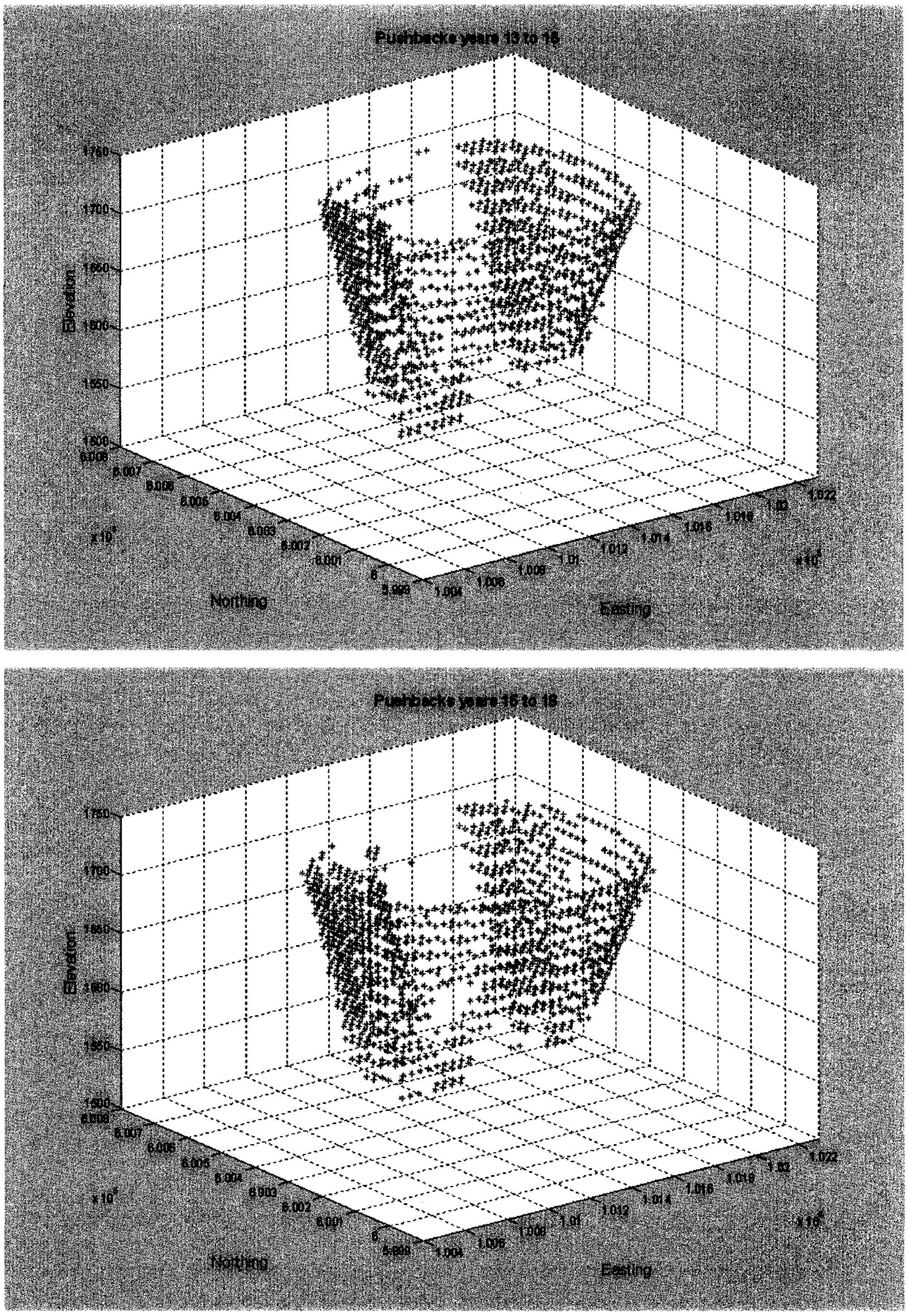


Figure 5.21 Push-backs years 13 to 18

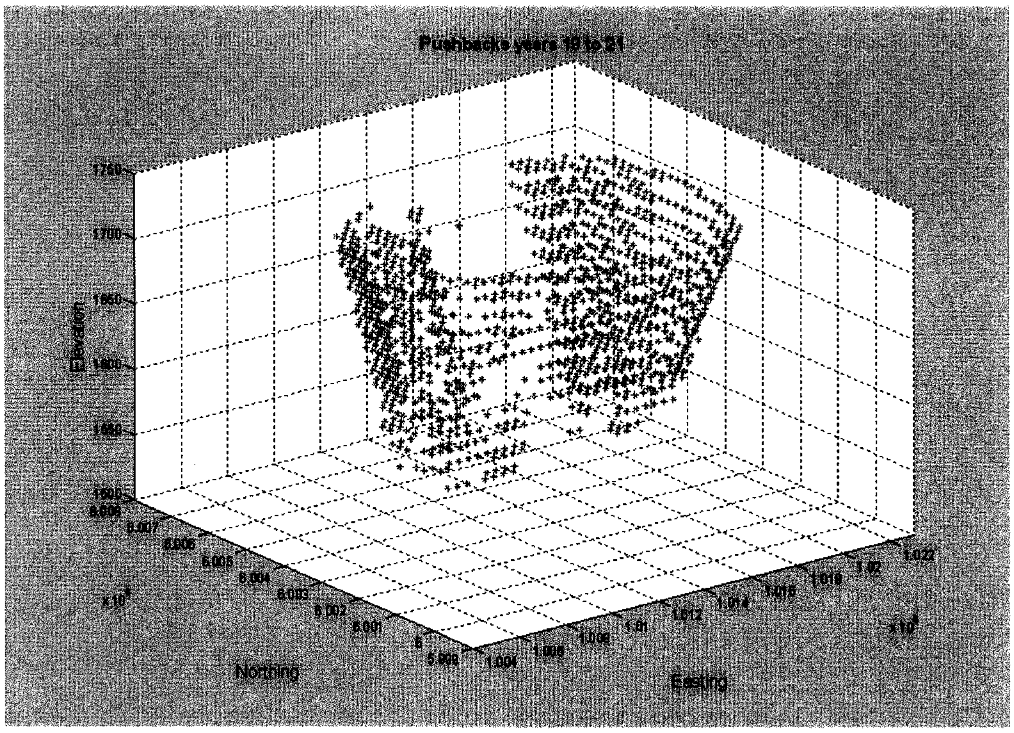


Figure 5.22 Push-backs years 19 to 21

The optimized final pit limits shows the total amount of 399 million tonnes of material consisting of 220 million tonnes of ore and 179 million tonnes of waste. The experiment was based on a maximum mining capacity of 20 Mt/year and a maximum milling capacity of 15 Mt/year. Two scenarios have been experimented with IOPS. In the first case, the minimum mining width was not considered as an input parameter. This scenario is almost equivalent to the best case schedule of Whittle software, where the push-backs generated are too narrow to mine practically. The IOPS long-term schedule for the best case yielded an NPV of $454,786,000. In the second scenario, the minimum mining width for the bottom of the pit was considered as an ellipse with major and minor axis of 60m. The IOPS long-term schedule for the practical case resulted in an NPV of $438,254,000.

The practical case is the basis of the mine strategic plan, the next step is to break it into operating and achievable targets within the framework of a tactical plan. In the next section, the COPS is used based on the same conditions underlying the strategic plan, to define periodic targets within a shorter time framework and as a continuous time solution.

5.4.4 COPS Continuous-Time Schedule

The continuous-time open pit production simulator is developed and implemented in MATLAB. The open pit geometry is captured using modified elliptical frustum. The output of IOPS is the input to COPS. The COPS models the dynamics of open pit geometry and the subsequent material movement as a continuous system described by time-dependent, nonlinear differential equations. A set of PDE's capture the time related behavior of the open pit mining systems. Function approximation of the discrete simulated push-backs generated by IOPS, provides the means to convert the set of PDE's to a system of ODEs. Numerical integration with Runge-Kutta scheme yields the trajectory of the pit geometry over time with the respective volume of materials transferred and the NPV of the mining operation.

The annual incremental push-backs generated by IOPS represents the discrete changes of Δa_W, Δa_E, Δb_N, Δb_S, and Δh. The COPS requires the abovementioned increments to be represented as functions of time to be able to capture the dynamics of the open pit expansion as a continuous system captured by time-dependent, differential equations. Function approximation of simulation results of IOPS is used to yield the additional conditions needed to convert the set of PDE's to a system of ODEs. Trend analysis and curve fitting are employed to approximate functions for increments $\Delta a_W, \Delta a_E, \Delta b_S, \Delta b_N$ and Δh over the mine life. To obtain reliable results goodness of fit statistics for all the approximations are evaluated by residuals as the differences between the response data and the fit to the response data at each predictor value. The sum of squares due to error, SSE, (Marquardt, 1963) is evaluated as well to obtain the best fit.

Table 5.2 to Table 5.5 illustrate the sum of squares due to error and R-square results of different fits. Figure 5.23 to Figure 5.27 demonstrate function approximations on the push-backs data, along axes a_W, a_E, b_N, b_S and h.

Table 5.2 aE the sum of squares due to error

Type	SSE	R-square
Exponential	1260.2	0.9859
Gaussian	1217.6	0.9864
Smoothing Spline	653.4	0.9927

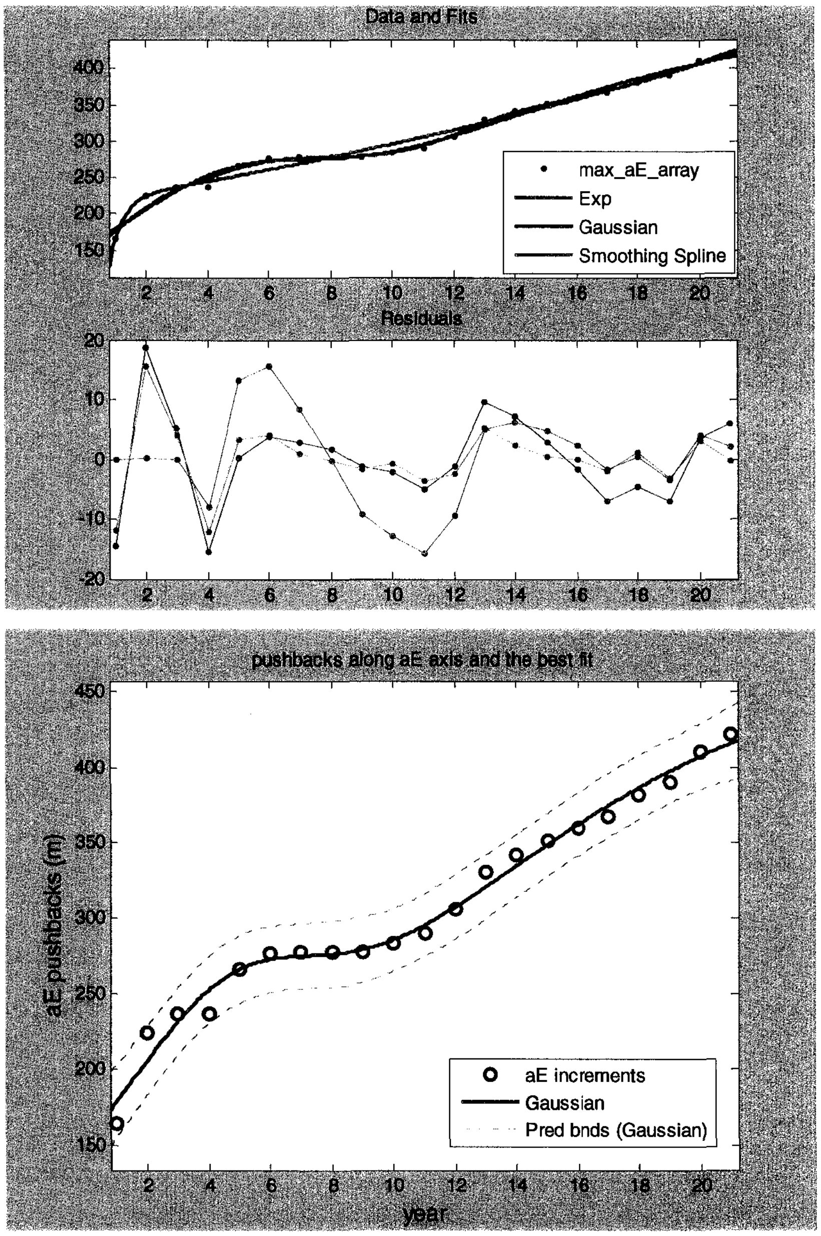


Figure 5.23 aE function approximation residuals and the best fit

Table 5.3 aW the sum of squares due to error

Type	SSE	R-square
Polynomial	2523.6	0.9971
Gaussian	853.5	0.9990
Smoothing Spline	156.3	0.9998

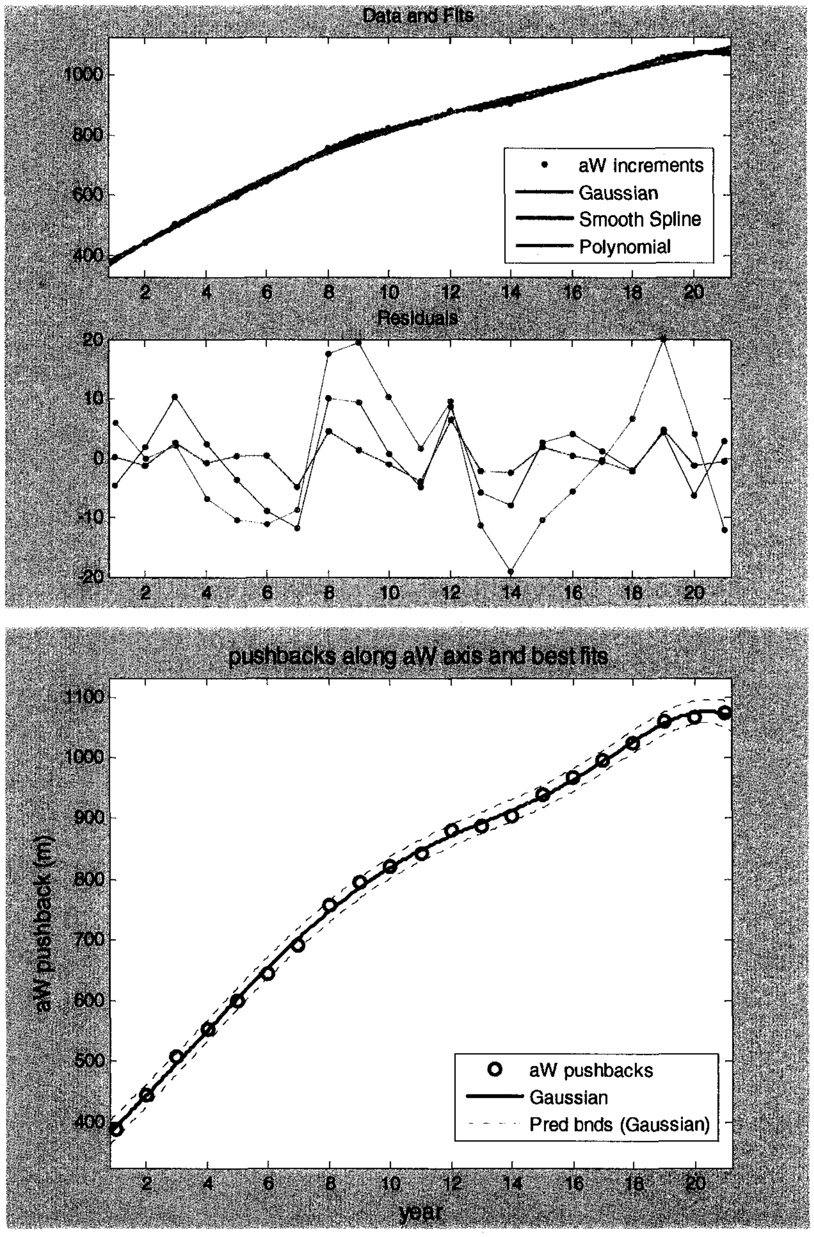


Figure 5.24 aW function approximation residuals and the best fit

Table 5.4 *bN* the sum of squares due to error

Type	SSE	R-square
Power	1642.0	0.9813
Gaussian	286.4	0.9967
Exponential	318.7	0.9963

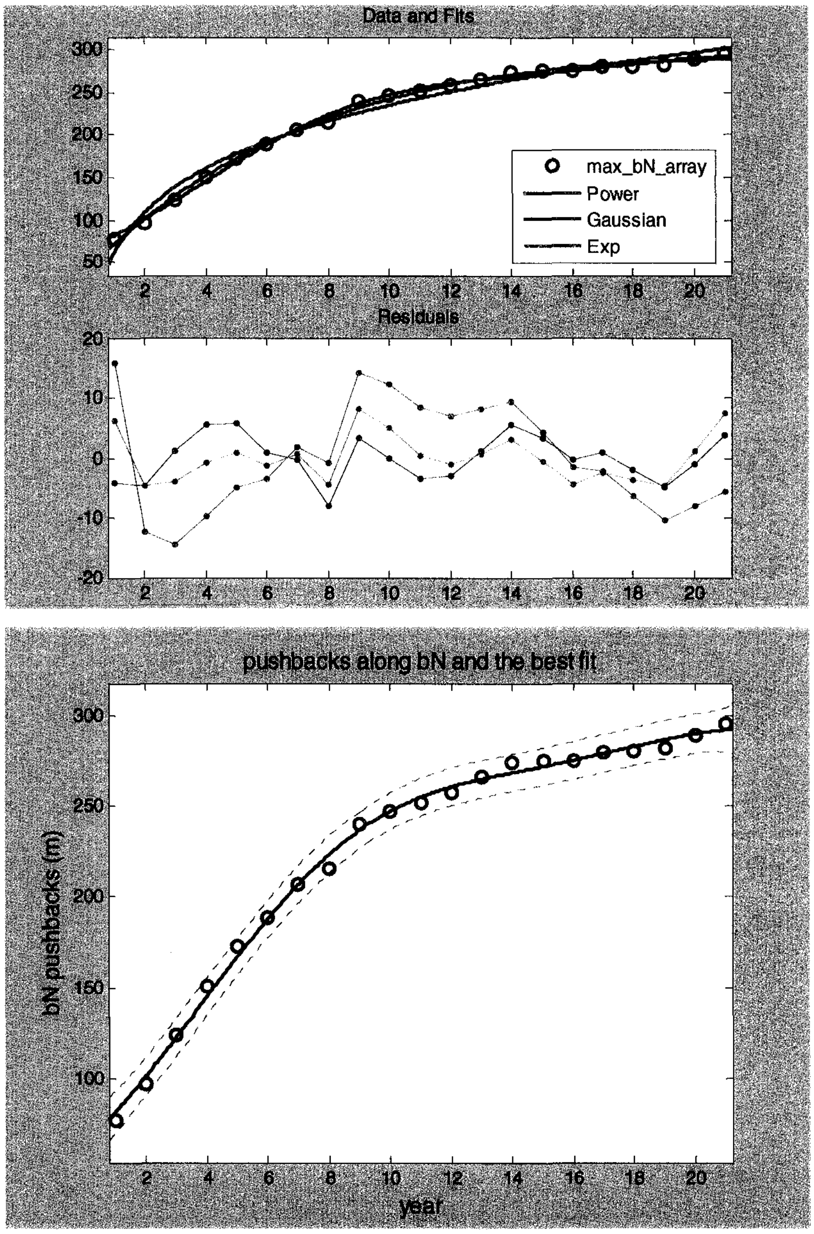


Figure 5.25 *bN* function approximation residuals and the best fit

Table 5.5 *bS* the sum of squares due to error

Type	SSE	R-square
Polynomial	1525.0	0.9815
Gaussian	297.8	0.9945
Exponential	418.6	0.9921

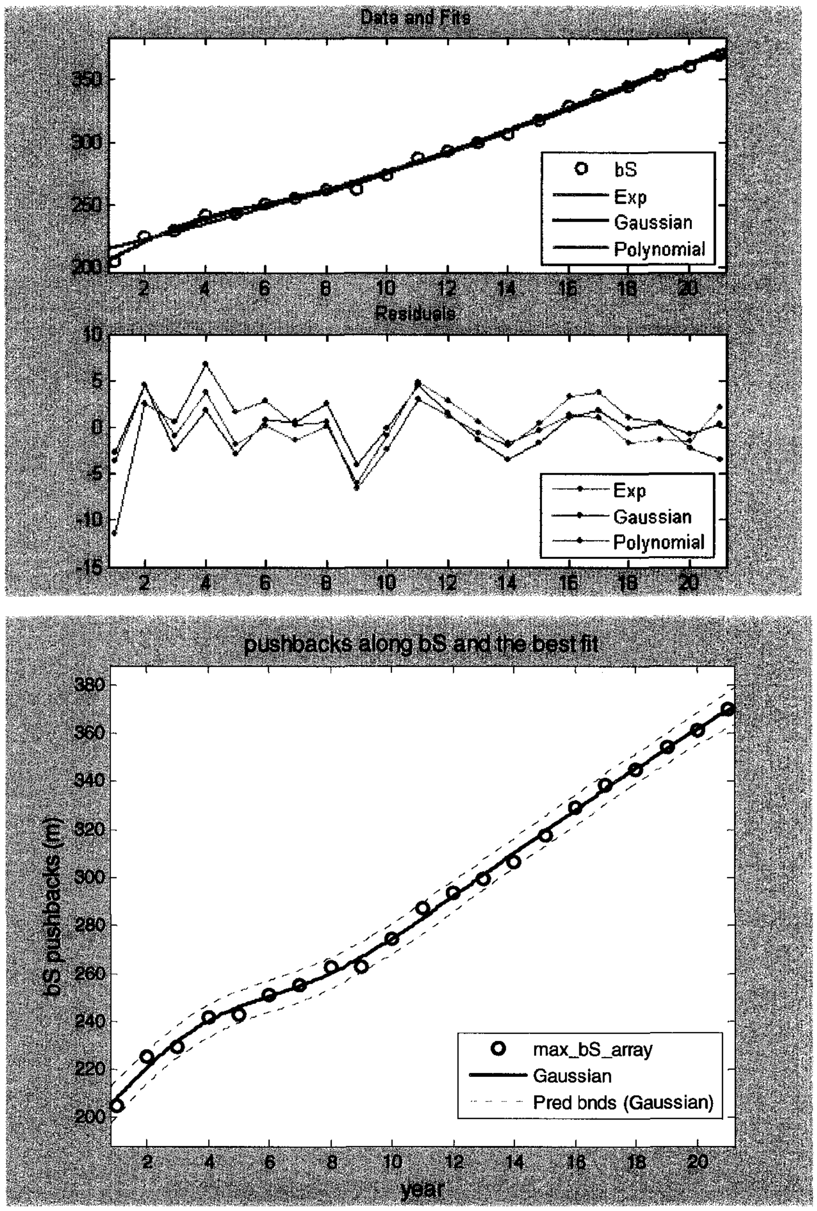


Figure 5.26 *bS* function approximation residuals and the best fit

Table 5.6 *h* the sum of squares due to error

Type	SSE	R-square
Exponential	422.6	0.9947
Power	3522.2	0.9561
Polynomial	340.8	0.9957

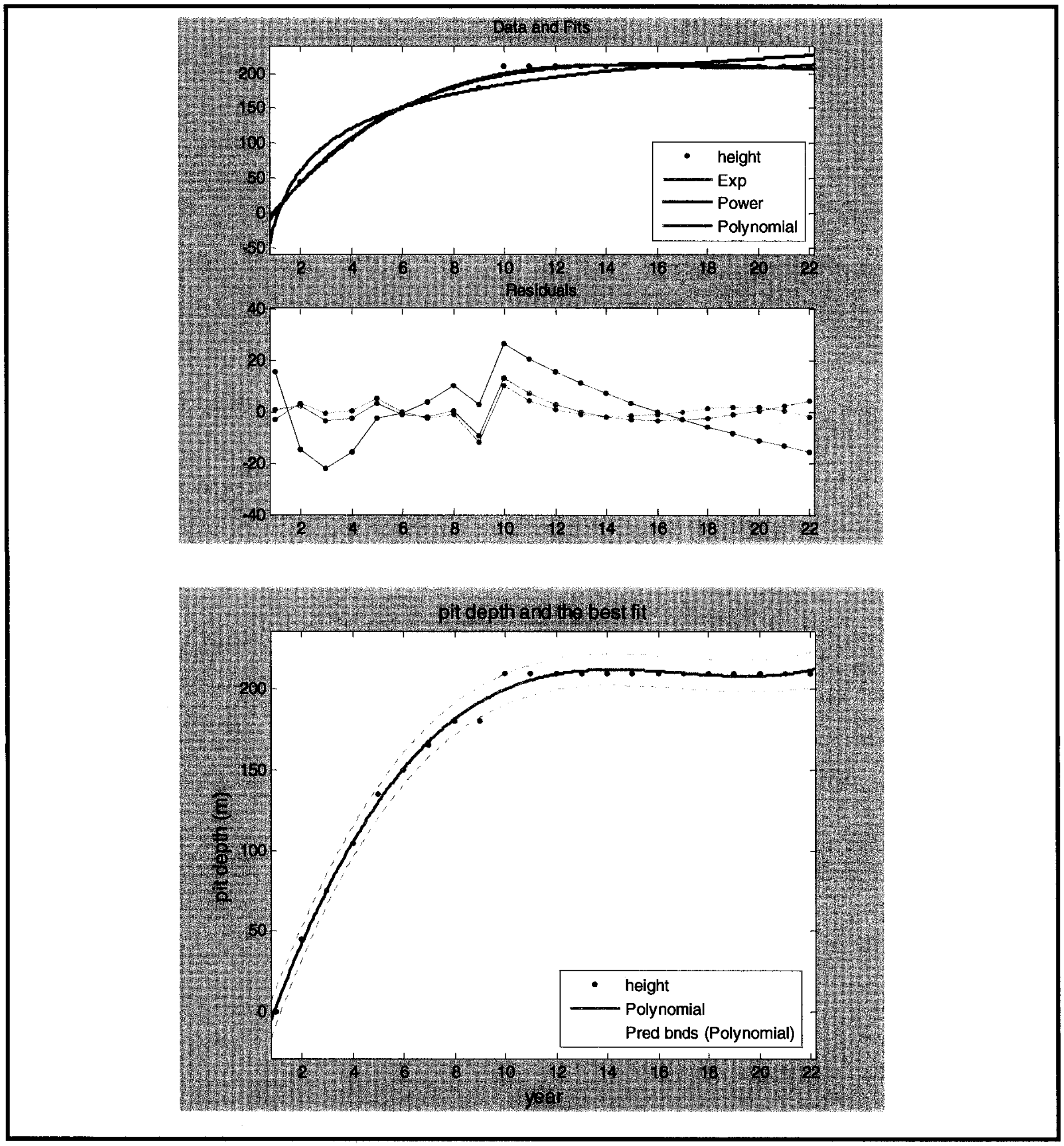


Figure 5.27 *h* function approximation residuals and the best fit

The best fit analysis recommends Gaussian functions for Δa_w, Δa_E, and Δb_s, and exponential functions for Δb_N and polynomial for Δh. The substitution of the approximated functions in equation (3.13) converts the set of PDE's to a system of ODEs.

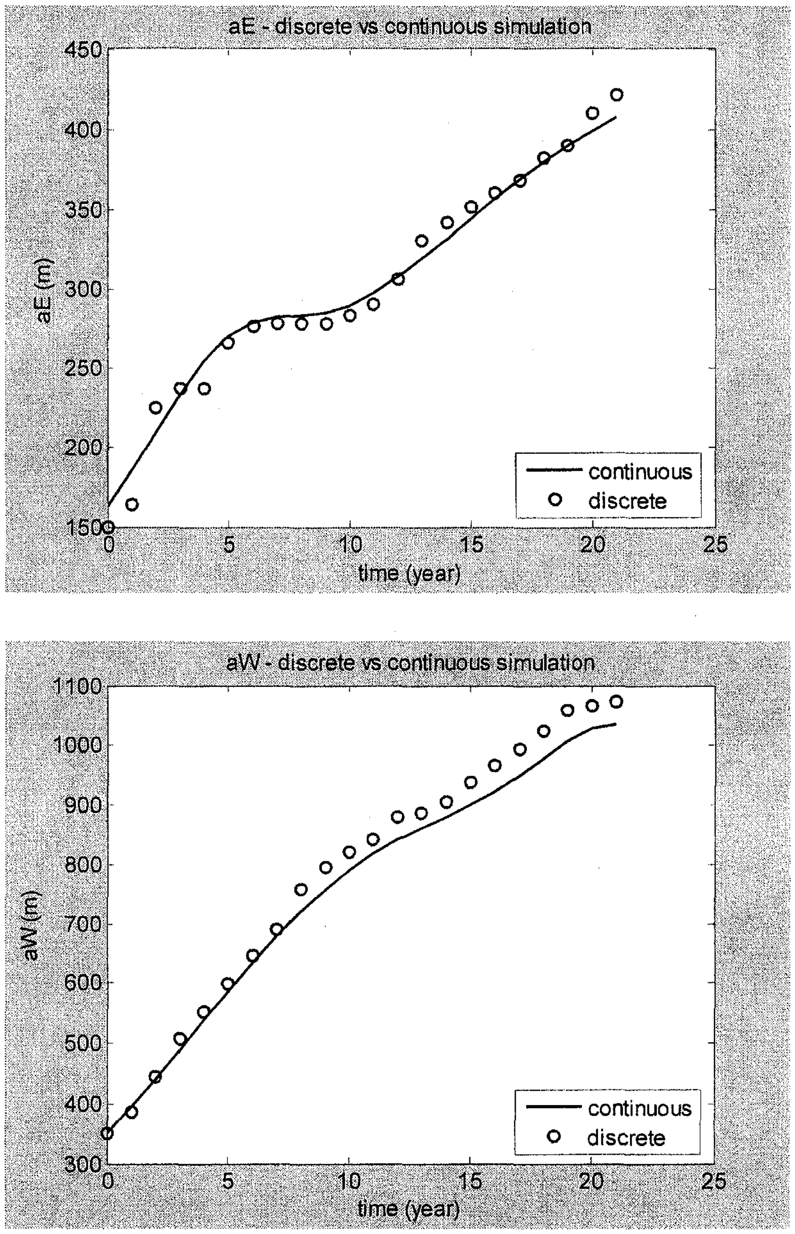


Figure 5.28 changes in a_E and a_W, COPS vs. IOPS

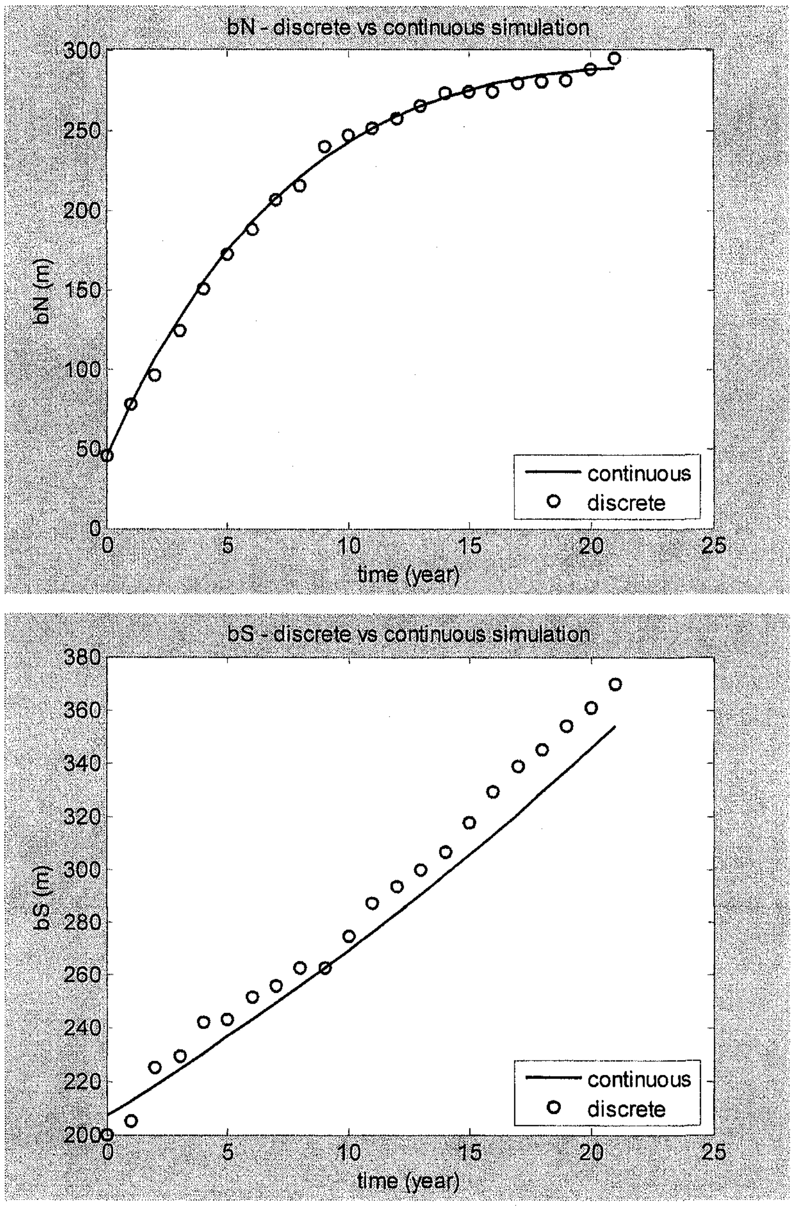


Figure 5.29 changes in b_N and b_S COPS vs. IOPS

COPS uses the MATLAB's standard solver for ordinary differential equations *ode45* suite. This function implements explicit Runge-Kutta (Cartwright and Piro, 1992) method with variable time step for efficient computation. Numerical integration with Runge-Kutta scheme yields the trajectory of changes in open pit geometry and the volume of material transferred over mine life.

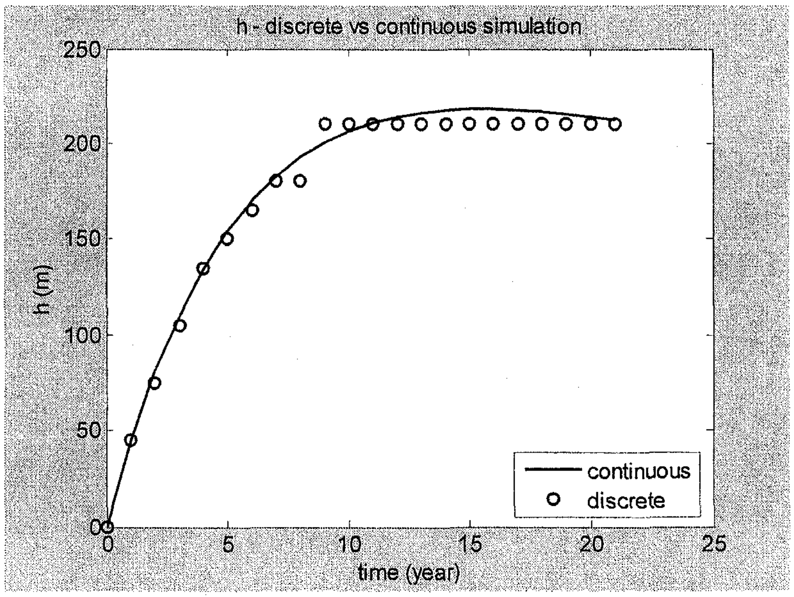


Figure 5.30 changes in h, COPS vs. IOPS

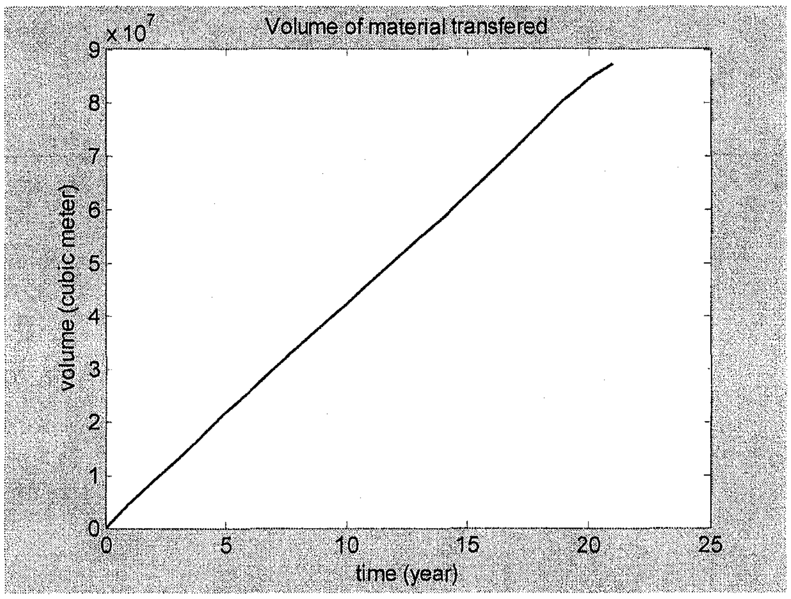


Figure 5.31 COPS volume of material transferred over mine life

Figure 5.28, Figure 5.29, and Figure 5.30 illustrate the trajectory of changes in the open pit geometry compared to the results of the optimal discrete plan generated by IOPS. The volume of rock in the frustum that captures the final pit limits are equal to $8.2056\times10^{7}m^{3}$ whereas, the results of the solution to the differential equations demonstrates a volume of $8.3203\times10^{7}m^{3}$ over the mine life. The linear format of Figure 5.31 demonstrates that the solution of the differential equations produces a schedule with a constant annual rock excavation rate. The interaction of the continuous changes in dimensions of the open pit with EPEM returns the volume of ore, waste and the net present value of the mining operation over the mine life. The COPS schedule under the similar circumstances with IOPS and Whittle resulted in an NPV of $432,350,000 over the 21 year time span.

5.4.5 Comparison of IOPS vs. Whittle Schedule

A comparative analysis of the extraction sequence is carried out to validate the results. The comparison is made between Milawa algorithm using Whittle Four-X and the IOPS. Figure 5.32 illustrates the annual production plan of IOPS and Whittle. The objective was to find a subset of pits that will approximate the best case scenario while providing sufficient working space and material for scheduling between push-backs.

In Whittle, the feasible solution space is the envelope bounded by the worst and best case scenario. The comparison of the annual schedule shows that the IOPS schedule has yielded a higher NPV than the Whittle schedule. Although balancing the feed to the processing plant was not an objective in this study but the IOPS also demonstrated a more steady flow of ore to the plant than Milawa (Figure 5.33). Whittle includes the Milawa balance mode to find a schedule with improved throughput balance. In this mode, the Milawa algorithm seeks to maximize the usage of production facilities early in the life of the mine instead of maximizing NPV.

Figure 5.34 illustrates the average annual grade of plant head grade. The IOPS schedule shows a more uniform grade to the mill than the Whittle schedule. The results drawn proves that the IOPS and the intelligent agent framework, in conjunction with stochastic simulation models presented provide a powerful tool for optimizing the long term planning of open pit mines while being able to address the random field and dynamic processes in mining environments.

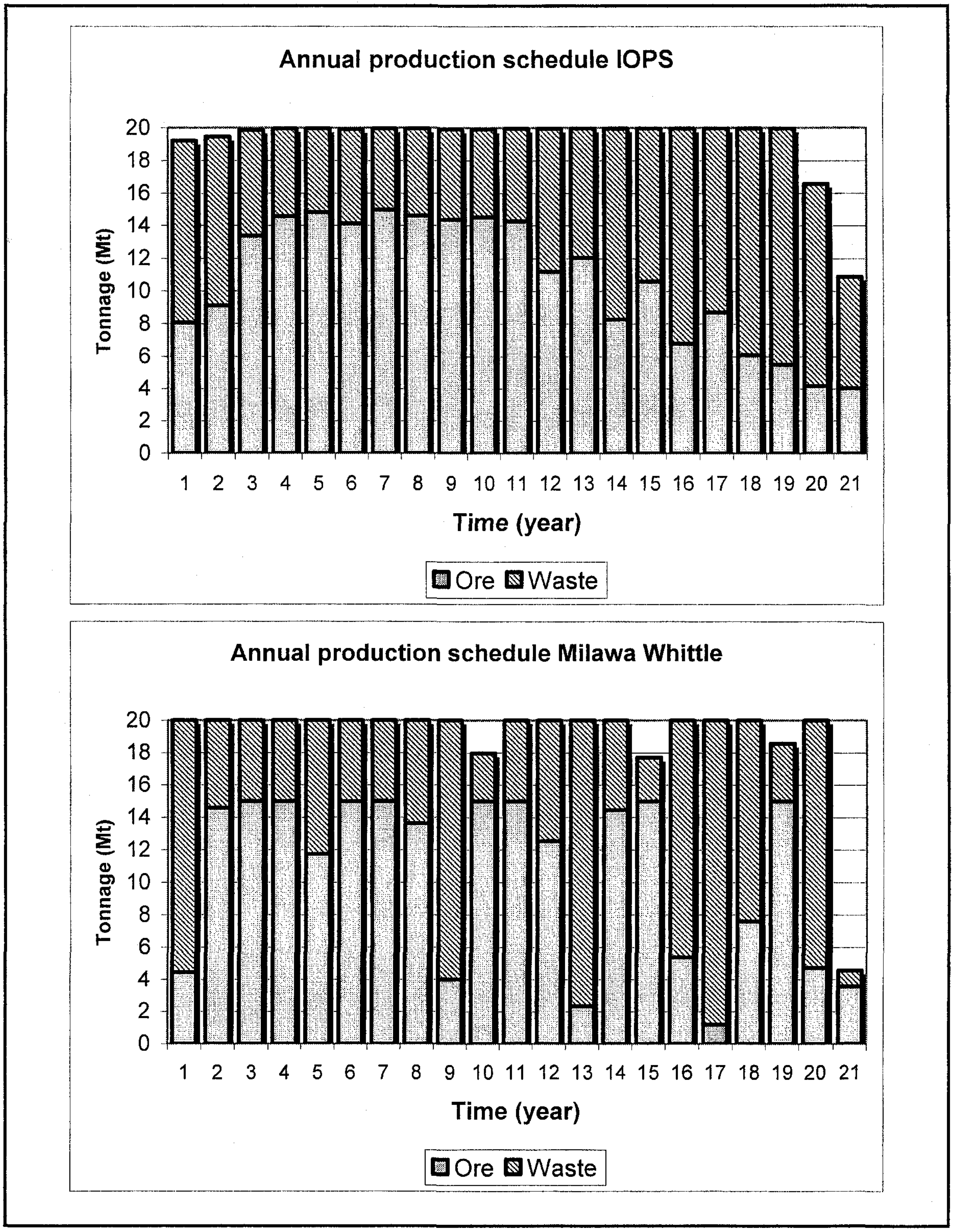
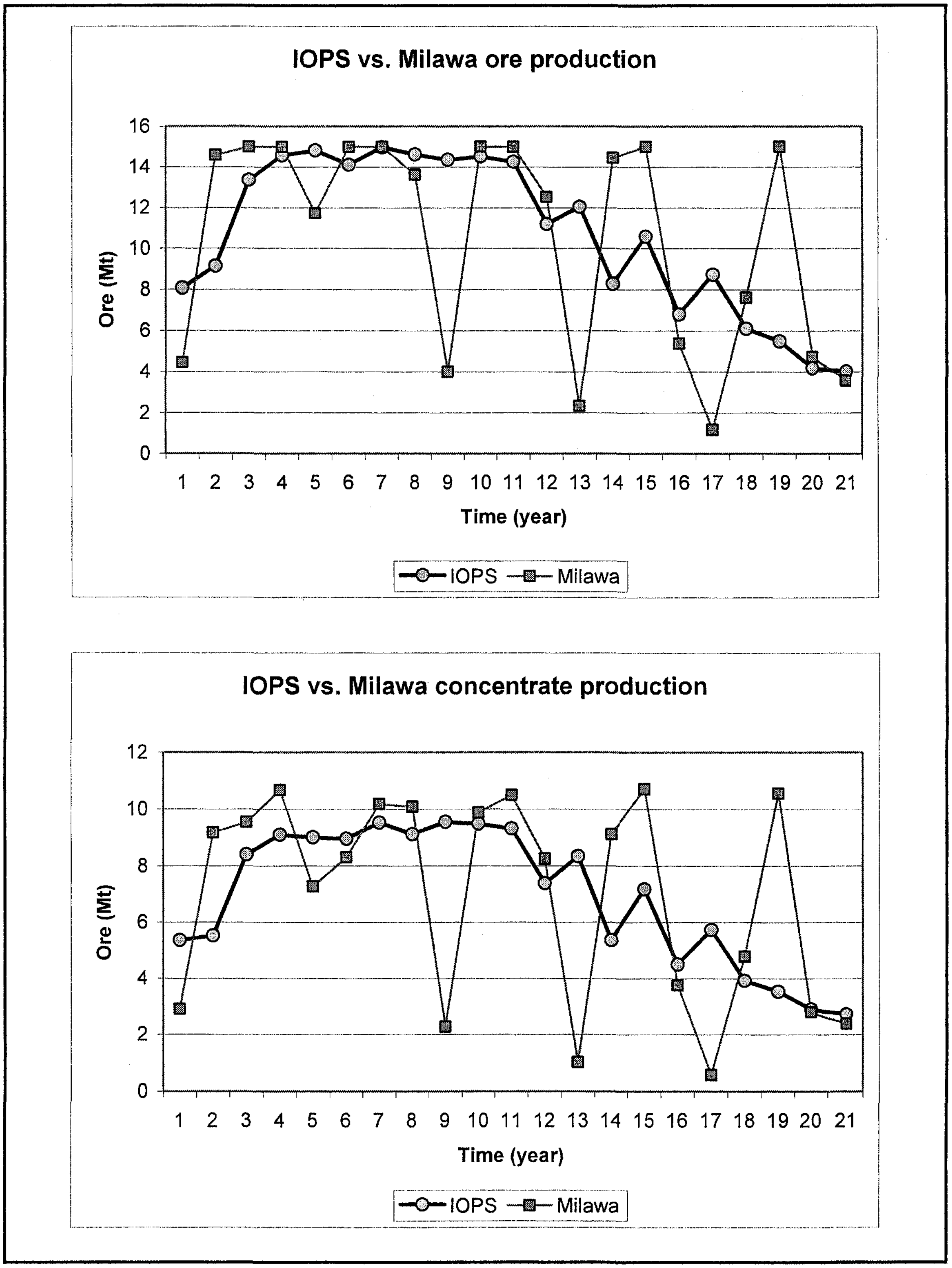


Figure 5.32 Annual production IOPS vs. Whittle



IOPS vs. Milawa ore production

IOPS vs. Milawa concentrate production

Figure 5.33 Annual ore and concentrate production, IOPS vs. Whittle

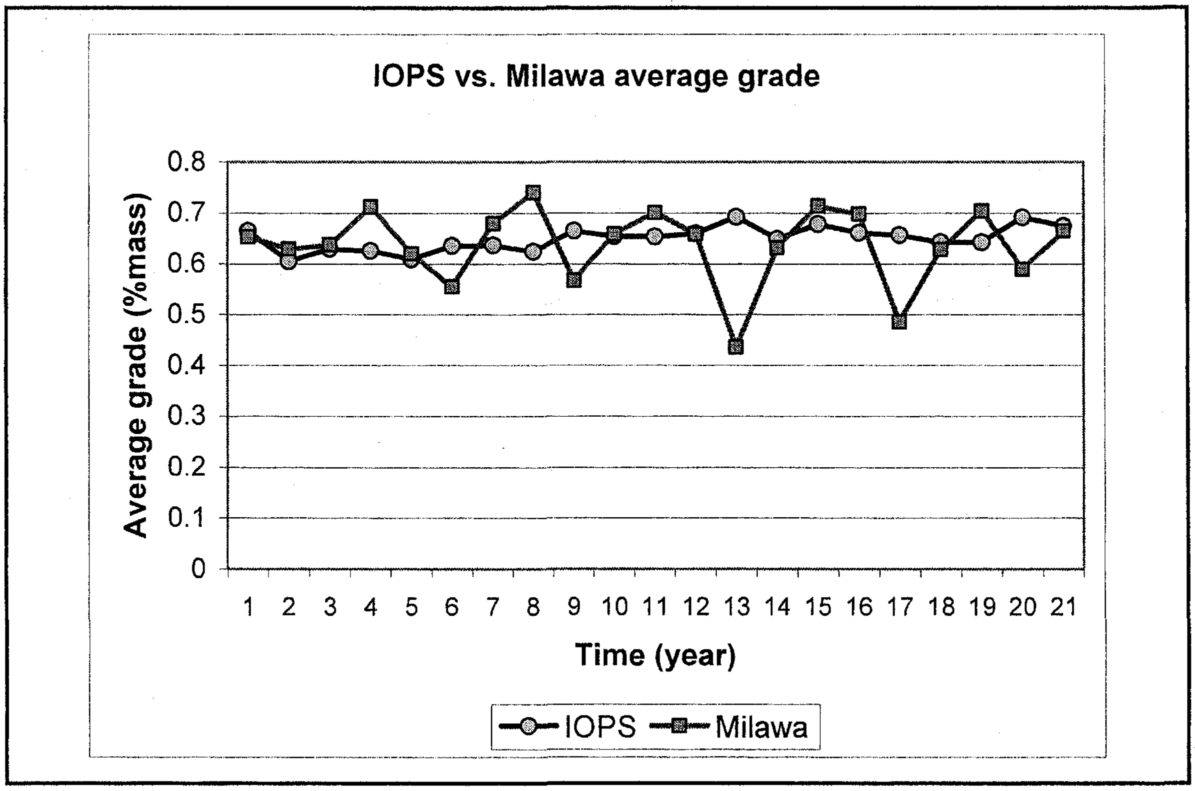


Figure 5.34 average annual grade IOPS vs. Whittle

5.4.6 Comparison of IOPS vs. COPS Schedule

COPS was used to capture the continuous-time related behavior of the open pit mine planning system by a set of PDE's. Merely the specification of the PDE's did not allow a unique solution to the problem because we must integrate an indefinite integral. The IOPS schedule was used to yield the additional and auxiliary conditions needed to convert the set of PDEs into a set of ODEs. IOPS uses the Q-learning algorithm to find a discrete optimal solution to the long-term mine planning problem. Subsequently COPS is used to capture the long-term plan as a continuous-time schedule for tactical purposes. Figure 5.35 demonstrates the annual production schedule of IOPS and COPS. Figure 5.36 compares the yearly ore and concentrate production of the two methods. Figure 5.37 shows the annual cash flow and average grade of IOSP and COPS. These results prove that the numerical solution to the set of PDE's was successful in capturing the long term schedule generated by IOPS.

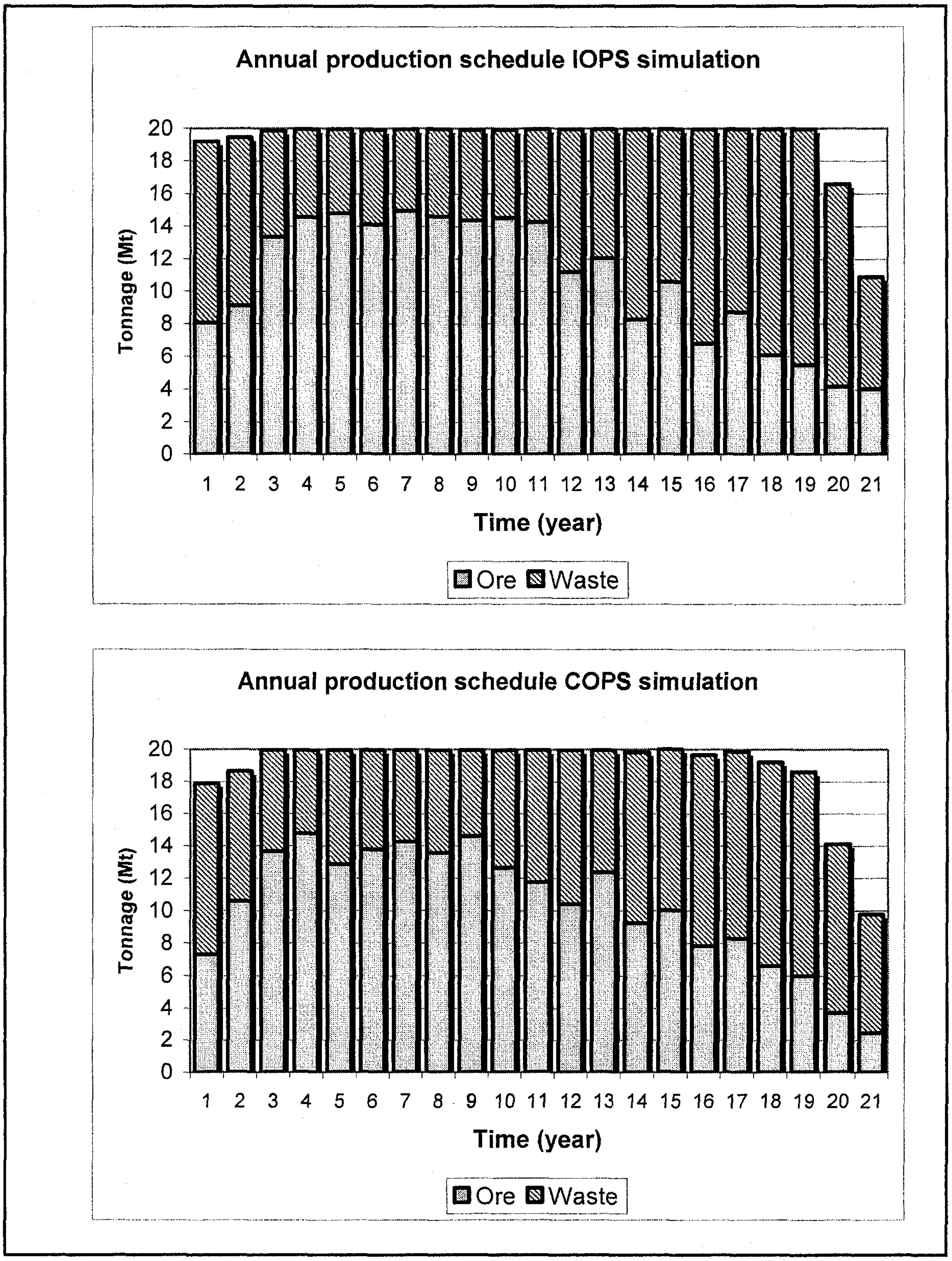


Figure 5.35 Annual production schedule, IOPS vs. COPS

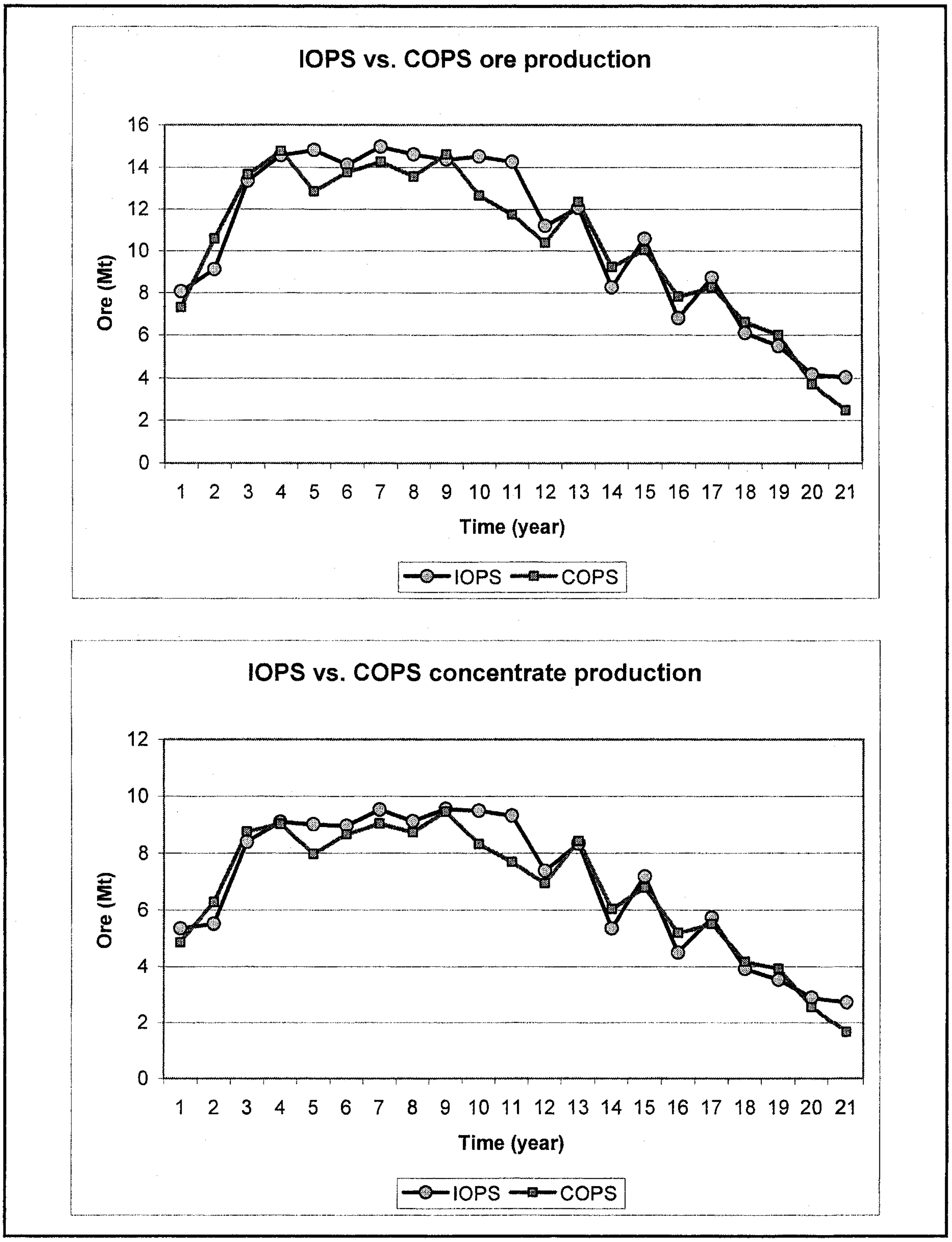
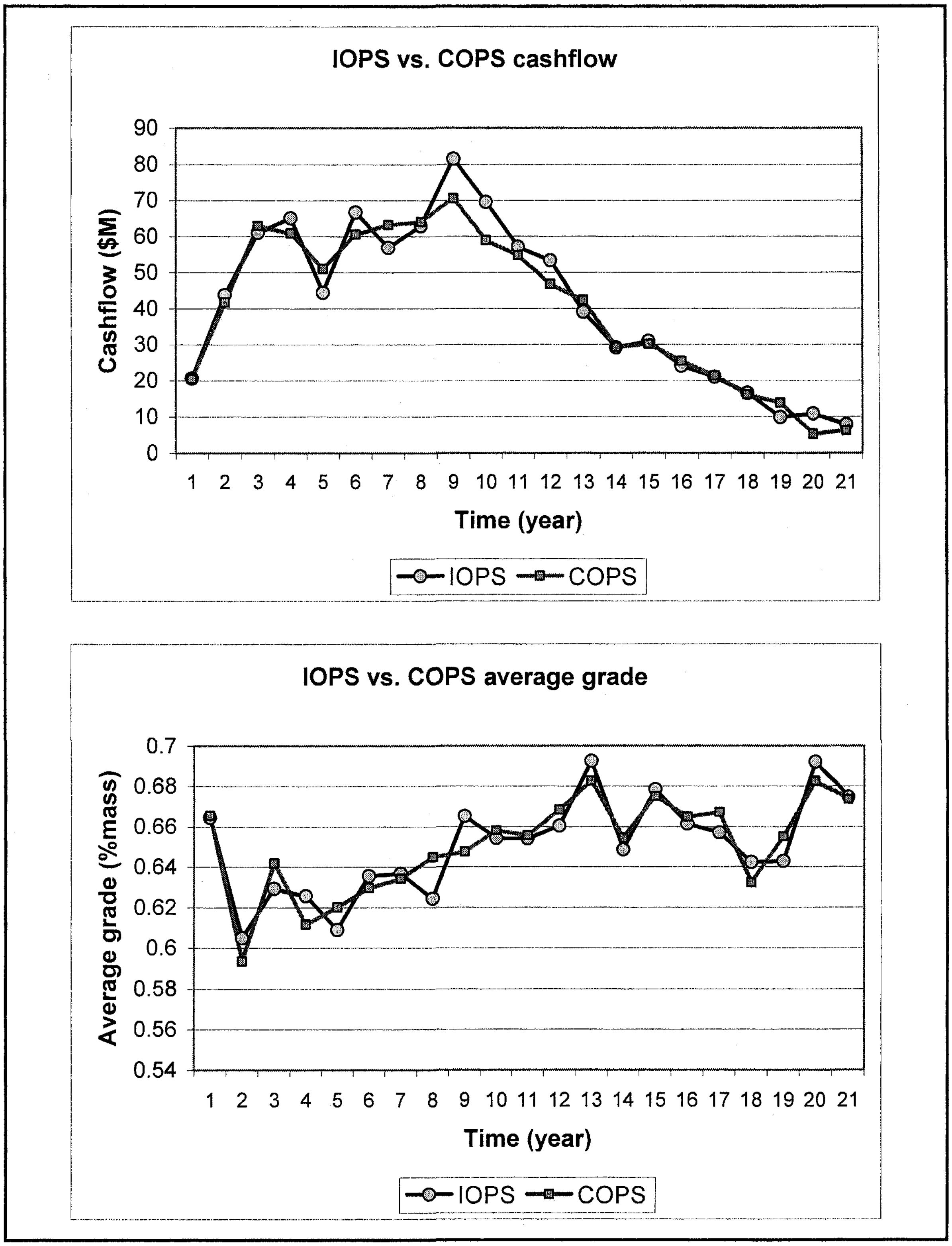


Figure 5.36 Ore and concentrate production, IOPS vs. COPS



IOPS vs. COPS cashflow

IOPS vs. COPS average grade

Figure 5.37 Cash flow and average grade, IOPS vs. COPS

5.5 Summary and Remarks

The focal point of this chapter was verification and validation of the models, algorithms, and the IOPS application through a comparative analysis among Whittle, IOPS, and COPS. A case study involving an iron ore deposit with 114,000 blocks was carried out to verify and validate the model. Table 5.7 summarizes the block model information. The final pit limits were determined using LG algorithm using the shells node in Whittle software. There were 77 pit shells generated by varying the product price using a revenue factor ranging from 0.4 to 1.4. The pit shell 68 was chosen as the optimized final pit limits. The optimized final pit limits show the total amount of 399 million tonnes of material consisting of 220 million tonnes of ore and 179 million tonnes of waste. IOPS deployed the modified geometrical open pit frustum model to capture the final pit with the dimensions of $a_W = 1040m; a_E = 440m; b_{N0} = 260m; b_{S0} = 360m$ and $h_0 = 210m$. Table 5.8 summarizes the final pit limits specifications designed by LG algorithm and exported to the IOPS.

Table 5.7 Block model information

Variable	Quantity
Number of blocks in model	114,000
Block dimensions (m)	20*10*15
model framework	95*80*15
Number of ore blocks in model	19328
Number of waste blocks in model	94672
Amount of ore in model (Mt)	243.53
Amount of waste in model (Mt)	4771.46
Total Fe in model (Mt)	154.14
Minimum grade of Fe (%)	13
Maximum grade of Fe (%)	89
Average grade of Fe (%)	63
Ore and waste density (tonne / cubic meter)	4.2

Table 5.8 Final pit limits specifications, LG and IOPS

Variable	Quantity
Revenue factor range (geometrical increments)	0.4 to 1.4
Number of pit shells generated by LG algorithm	77
Final pit limits pit shell	68
Amount of ore in the final pit (Mt)	220
Amount of waste in the final pit (Mt)	179
IOPS geometrical final pit dimensions (m)	$a_W = 1040m; a_E = 440m; b_N = 260m;$ $b_S = 360m; h = 210m$
Epsilon in IOPS (Q-learning parameter)	0.01
IOPS simulation iterations	3000

The best case and the practical extraction sequence from IOPS were compared to the worst-case, best-case, and Milawa algorithm using Whittle Four-X. Milawa algorithm employed in Whittle Four-X yielded an NPV of $430 million over a 21-year of mine life at a discount rate of 10% per annum; whereas IOPS yielded in an NPV of $438 million under the same circumstances and over the same mine life. Table 5.9 compares the NPV of the IOPS, Whittle, and COPS schedules.

Although optimality is not guaranteed with the parametric analysis in Whittle 4-X, it is a strong heuristic tool for identifying high grade ore clusters in the model and for maximizing the NPV. Whittle is the current standard and dominant software used in industry in open pit mine planning and design. The IOPS schedule shows a more uniform grade to the mill than the Whittle schedule. Although balancing the feed to the processing plant was not an objective in this study, IOPS also demonstrated a more steady flow of ore to the plant than Milawa. The results proves that IOPS and the intelligent agent framework, in conjunction with stochastic simulation models provide a powerful tool for optimizing the long term planning of open pit mines and it address the random field and dynamic processes in mining environments.

Table 5.9 NPV comparison

Method	**NPV ($)**
Worst case scenario Whittle	155,753,000
Best case scenario Whittle (impractical)	457,676,000
Best case scenario IOPS (impractical)	454,786,000
Feasible Whittle schedule (Milawa algorithm)	430,734,000
Practical IOPS schedule (Q-learning algorithm)	438,254,000
COPS continuous time schedule	432,350,000
Variable	**Quantity**
Volume of final pit by geometrical model	$8.2056\times10^7 m^3$
Volume of final pit by differential equations	$8.3203\times10^7 m^3$

CHAPTER 6
SUMMARY, CONCLUSIONS, AND RECOMMANDATIONS

6.1 Summary of Research

Numerous attempts have been made to address the open pit optimization and long-term planning problem during the last 40 years. The major shortcomings of current planning and optimization techniques can be summarized as: (i) inability to solve large industrial problems; (ii) limitation in dealing with stochastic processes governing mining operation; (iii) inadequacy in referencing time aspect of the problem. These deficiencies can cause an open pit mining operation severely in terms of dollars, time, and strategic and tactical plans. In complex mining operations, small deviations from the optimal long-term plan could result in the loss of millions of dollars.

Consequently, an intelligent open pit optimal production simulator (IOPS), based on reinforcement learning, has been proposed as a comprehensive algorithm to address these problems. The outcome of the research demonstrated a strong promise towards improving the expected net present value on the mining investments. The developed algorithms can be the basis of the next generation of mine design software packages.

Figure 6.1 illustrates the mathematical models and algorithms, which have been developed and implemented. Reinforcement learning was used to capture the open pit long-term planning by the interaction of two basic entities: the intelligent open pit agent, IOPA, interacts with the block model by means of the open pit production simulator. The OPPS stochastically simulates the next feasible push-backs. The cash flows of the simulated push-backs are returned by the EPEM. The intelligent agent, IOPA, employs Q-learning and epsilon-greedy algorithm to choose the next suitable push-back among all the simulated results. The algorithm then updates the action value estimates $Q(s,a)$. The simulation continues until the simulated push-back design reaches the final pit limits.

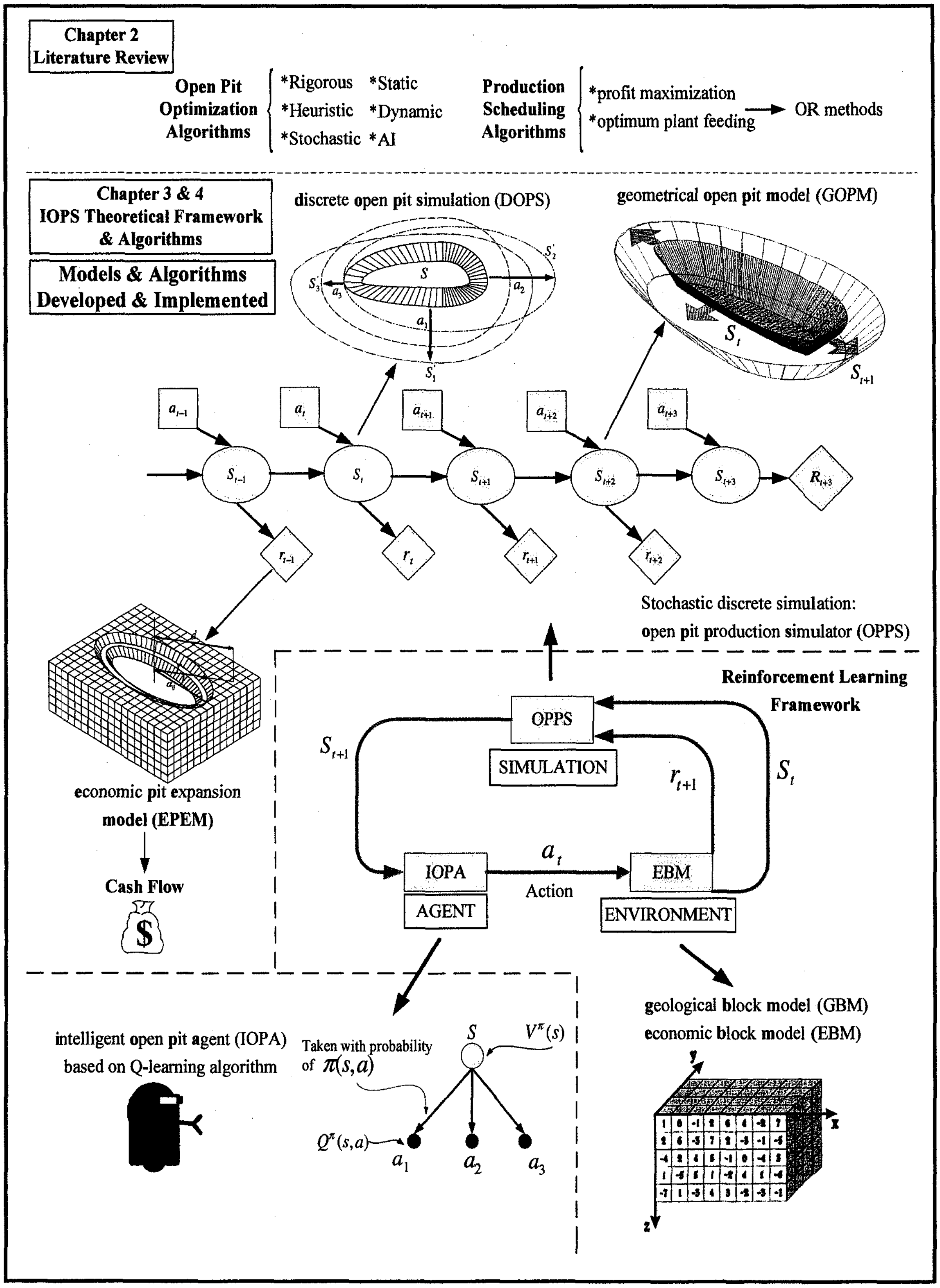

Chapter 2
Literature Review
Open Pit Optimization Algorithms
*Rigorous *Static
*Heuristic *Dynamic
*Stochastic *AI
Production Scheduling Algorithms
*profit maximization
*optimum plant feeding
OR methods
Chapter 3 & 4
IOPS Theoretical Framework & Algorithms
Models & Algorithms Developed & Implemented
discrete open pit simulation (DOPS)
geometrical open pit model (GOPM)
S
S'_1
S'_2
S'_3
a_1
a_2
a_3
S_t
S_{t+1}
a_{t-1}
a_t
a_{t+1}
a_{t+2}
a_{t+3}
S_{t-1}
S_t
S_{t+1}
S_{t+2}
S_{t+3}
R_{t+3}
r_{t-1}
r_t
r_{t+1}
r_{t+2}
Stochastic discrete simulation:
open pit production simulator (OPPS)
Reinforcement Learning Framework
OPPS
SIMULATION
S_{t+1}
r_{t+1}
S_t
a_t
Action
IOPA
AGENT
EBM
ENVIRONMENT
economic pit expansion model (EPEM)
Cash Flow
$
intelligent open pit agent (IOPA) based on Q-learning algorithm
Taken with probability of $\pi(s,a)$
S
$V^{\pi}(s)$
$Q^{\pi}(s,a)$
a_1
a_2
a_3
geological block model (GBM)
economic block model (EBM)
y
x
z

Figure 6.1 Research summary, models and algorithms developed

The IOPS finally returns the optimal long-term mine plan. Afterwards, the continuous open pit simulator, COPS, models the dynamics of the open pit geometry and the subsequent material movement as a continuous system described by time-dependent differential equations. A set of PDEs capture the time-related behavior of the open pit mining systems. Functional approximation of the discrete simulated push-backs, generated by IOPS, provides the means to convert the set of PDEs to a system of ODEs. Numerical integration with Runge-Kutta scheme yields the trajectory of the pit geometry over time with the respective volume of materials transferred and the NPV of the mining operation.

The Java Reinforcement Learning Library (JavaRL) (Kerr et al., 2003) was chosen as the core of the IOPS application implementation. Java programming language version 1.4.2 (Sun Microsystems, 1994-2006) and MATLAB version 7.04 (MathWorks, 2005) were selected as the platform for programming. The IOPS graphical user interface (GUI) was implemented using IntelliJ IDE (JetBrains, 2000-2006). The user interface facilitates setting the block model, open pit, production, and simulation parameters.

Figure 6.2 illustrates the numerical applications of IOPS and COPS to verify and validate the research outcomes. A case study involving an iron ore deposit with 114,000 blocks was carried out. The final pit limits were determined using the LG algorithm via the shells node in the Whittle software. There were 77 pit shells generated by varying the product price using a revenue factor ranging from 0.4 to 1.4. The pit shell 68 was chosen as the optimized final pit limits. The optimized final pit limits show the total amount of 399 million tonnes of material consisting of 220 million tonnes of ore and 179 million tonnes of waste.

The long-term plans are analyzed by comparing the results of the IOPS with Milawa algorithm. Table 6.1 shows the NPV of the worst-case, best-case, and Milawa algorithms of the Whittle software and the best case and the practical annual long-term schedule generated by IOPS and COPS. The analysis were run over a 21-year of mine life at a discount rate of 10% per annum. Experiments also compared the annual stripping ratio, average grade, annual waste, and the ore and concentrate production.

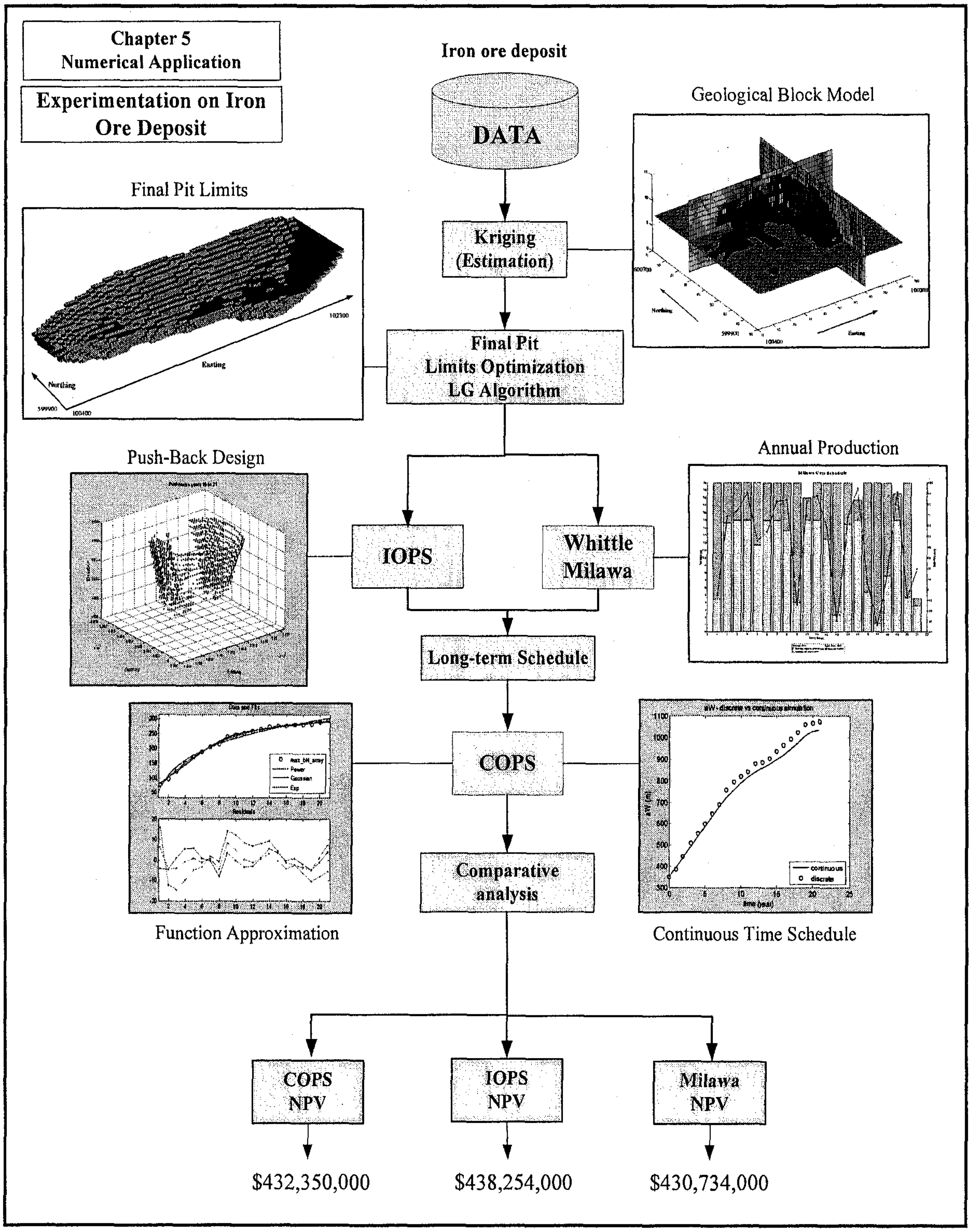


Figure 6.2 Research summary, numerical application and results

Table 6.1 Summary of the numerical application

Method	NPV ($)
Worst case scenario Whittle	155,753,000
Best case scenario Whittle (impractical)	457,676,000
Best case scenario IOPS (impractical)	454,786,000
Feasible Whittle schedule (Milawa algorithm)	430,734,000
Practical IOPS schedule (Q-learning algorithm)	438,254,000
COPS continuous time schedule	432,350,000

6.2 Conclusions

A comprehensive literature review on the current body of knowledge on open pit mine planning algorithms established the frontier in this research sphere. The results of the literature survey approved that there has never been any previous attempt for defining the open pit long-term planning problem by the reinforcement learning paradigm. Therefore, this research endeavor is a ground-breaking effort to employ Q-learning algorithm in order to provide contributions, knowledge and novel understandings into mine planning and design domain.

All the research objectives outlined in Section 1.3 have been achieved within the research scope. The following conclusions were drawn from comparative analyses of the extraction sequence generated by Whittle and IOPS:

1. The intelligent optimal open pit production simulator (IOPS) significantly improves the discounted net pit value comparing to the Milawa algorithm, which is the standard tools in industry.
2. The efficiency and the convergence rate of the IOPS algorithm is directly related to the mining starting point. IOPS randomly checks different mining initial box cuts in the learning process. A prior knowledge of high grade ore clusters in the model can speed up the computations.
3. The smallest pit generated by the LG parametric analysis is the best mining starting point because it includes the ore with highest grade.
4. The IOPS schedule illustrated a more uniform head grade than the Whittle schedule.
5. The IOPS also demonstrated a more steady flow of ore to the plant than Milawa.

6. Enforcing the practical mining constraints in mathematical models is crucial. The best case scenarios of the IOPS and Whittle, where the minimum mining widths are not enforced have resulted in optimistic results, which are impractical.
7. The sinking rate of the open pit, which is the maximum feasible number of benches deepen every year is a critical factor in long-term planning.
8. The net present value of the schedule is more sensitive to the product price than the mining and processing cost.
9. The intelligent agent framework, in conjunction with stochastic simulation models provide a powerful tool for optimizing the scheduling process while addressing the random field and dynamic processes in open pit mine planning.

The following conclusions were drawn from comparative analyses of the extraction sequence generated by COPS and IOPS:

1. Comparative analysis of the NPV estimates from IOPS and COPS illustrates that the numerical solution of the PDEs, simulating the continuous-time open pit dynamics successfully captured the optimal long-term term plan generated by the IOPS.
2. The trajectories of the changes of the open pit geometry in time captured by COPS are smoother than the discrete simulation results. Hence, the respective NPV estimates of the schedule are less than the optimal schedule generated by IOPS over time.
3. The volume of rock in the frustum capturing the final pit limit is equal to $8.2056\times10^{7}m^{3}$, whereas the results of the solution to the differential equations demonstrates a volume of $8.3203\times10^{7}m^{3}$ over the mine life.
4. The solution of the differential equations produces a schedule with an almost constant annual rock excavation rate. This is a sign that COPS is capable of generating a uniform annual production rate.
5. COPS presented the flexibility to capture the long-term plan in terms of differential equations. Any changes in the schedule can be easily modified by new functional approximations and the numerical solution to the system of ODEs. This method reduces the necessity to deal with huge data bases over time.

6.3 Contributions of PhD Research

The following constitute the major contributions of research.

1. This is the first endeavor to develop optimization models of long-term open pit mine planning problem based on reinforcement learning, intelligent agents, stochastic, and continuous time simulation. This work contributes enormously to the body of knowledge on open pit mine planning and design and can be the foundation of the next generation of mine planning software packages.
2. The research has formulated robust mathematical models and comprehensive algorithms. It expands the frontiers of open pit planning and optimization by improving the expected net discounted value of a mining venture to a great extent compared to current software packages used in industry.
3. The original algorithm enables step-changes in the manner in which mines are planned and managed. This has been made possible by providing mathematical models of the open pit geometry, stochastic and continuous-time simulation models of the open pit dynamics, and an intelligent agent – implementing Q-learning as the core of IOPS engine. The agent is able to learn optimal plans from experience in the form of sample simulation episodes of push-backs.
4. Unlike current mine planning algorithms, IOPS has the capability of dealing with mine planning parameters as stochastic variables.
5. This is a pioneering effort at modeling open pit geometry with modified elliptical frustum with five variables. The new model reduces considerable error in volume calculations. It also facilitates the process of saving the long term plan as incremental changes in the dimensions of the major and minor axes of the elliptical frustum rather than saving all the block coordinates for different scenarios.
6. A novel economic pit expansion model (EPEM), which interacts with the modified geometrical open pit model to return the monetary value of the mining operation.
7. The continuous time simulation model (COPS) is novel in various ways. First it captures the optimal long-term plan of an open pit mine as a system of PDEs. Secondly, it combines the functional approximation for capturing the optimal plan

generated by IOPS. Afterwards, the approximations are used as the auxiliary conditions to numerically solve the differential equations.

6.4 Recommendations for Further Research

Although the methods and algorithms developed in this thesis have provided a ground breaking methodology for open pit mine planning and optimization, still there should be extensive studies to expand the usefulness of the research to the mineral industry. The following areas could significantly improve and add to the body of knowledge in this research domain.

1. The main objective of this research work was concerned with maximizing the NPV of the mining operation over time. Another important consideration in open pit planning is to model the exploitation process and the mineral processing criteria simultaneously. One should always take this into account that neither pit limit optimization nor production scheduling optimization is a goal by themselves; they are parameters that are used to maximize the total mining procedure profitability. The basic demands of the processing plant are uniform grade feed material and maximum usage of the processing plant capacity. Uniformity should be both in quantity, amount, and quality, grade, to minimize idle times of the plant. In order to supply the demanded material, excavation sequence should be altered, which suggests that the optimum pit configuration is not only a function of monetary values but also of technical constraints. In order to reach these goals an optimization model based on multiple criteria decision making through goal programming (Ignizio, 1976; Schniederjans, 1995; Tamiz, 1996; Trzaskalik and Michnik, 2002; Tanino et al., 2003), and multi-agent systems must be added to the IOPS framework (Weiss, 1999; Kraus, 2001; Tessier et al., 2001).
2. The primary implementation of EPEM demonstrated a reasonable run time for block models representing medium size deposits. The increase in the number of blocks in the model causes a longer search time for EPEM. Therefore, there is need for spatial data structures (Samet, 1990; Samet, 1990) to enhance the algorithm efficiency and accelerating the search.

3. The core algorithm of IOPS engine was one-step Q-learning algorithm (Watkins, 1989). To make the algorithm learn with a faster rate, eligibility traces must be considered and developed in the main algorithm (Sutton, 1992). There are two ways to view eligibility traces. The more theoretical view is that they are a bridge from TD to Monte Carlo methods. The other way to view eligibility traces is more mechanistic. From this perspective, an eligibility trace is a temporary record of the occurrence of an event, such as the visiting of a state or the taking of an action. The trace marks the memory parameters associated with the event as eligible for undergoing learning changes. Thus, eligibility traces help bridge the gap between events and training information (Sutton and Barto, 1998).

Appendix A - Partial Derivatives (Equation 3.13)

$$A1 := \frac{1}{4}\pi\,(aE\,bN + aW\,bN + aE\,bS + aW\,bS)$$

$$A2 := \frac{1}{4}\pi\left(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\right)$$

$$\frac{\partial}{\partial aE}\nu =$$

$$\frac{1}{3}h\Bigg(\frac{1}{4}\pi\,(bN + bS) + \frac{1}{4}\pi\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right) + \Bigg(\sqrt{16}\,\Bigg(\pi^2\,(bN + bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right)$$

$$+ \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg) + \pi^2\,(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+ aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial}{\partial aW}\nu =$$

$$\frac{1}{3}h\Bigg(\frac{1}{4}\pi\,(bN + bS) + \frac{1}{4}\pi\left(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}\right) + \Bigg(\sqrt{16}\,\Bigg(\pi^2\,(bN + bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right)$$

$$+ \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg) + \pi^2\,(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+ aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial}{\partial bN}\nu =$$

$$\frac{1}{3}h\Bigg(\frac{1}{4}\pi\,(aE + aW) + \frac{1}{4}\pi\left(aE - \frac{h}{\tan(\alpha)} + aW - \frac{h}{\tan(\beta)}\right) + \Bigg(\sqrt{16}\,\Bigg(\pi^2\,(aE + aW)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right)$$

$$+ \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg) + \pi^2\,(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(aE - \frac{h}{\tan(\alpha)} + aW - \frac{h}{\tan(\beta)}\right)\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+ aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial}{\partial bS}\nu =$$

$$\frac{1}{3}h\Bigg(\frac{1}{4}\pi\,(aE+aW)+\frac{1}{4}\pi\bigg(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\bigg)+\Bigg(\sqrt{16}\Bigg(\pi^2\,(aE+aW)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)$$

$$+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)+\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\bigg(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial}{\partial h}\nu =$$

$$\frac{1}{12}\pi\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)$$

$$+\frac{1}{12}\pi\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)+\frac{1}{48}\sqrt{16}\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)+\frac{1}{3}h$$

$$\Bigg(\frac{1}{4}\pi\Bigg(-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)+\Bigg(\sqrt{16}\,\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg($$

$$-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE$$

$$-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial aE^2}\nu =$$

$$-\Bigg(\sqrt{16}\Bigg(\pi^2\,(bN+bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)$$

$$+\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\bigg(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\bigg)\Bigg){}^\wedge 2\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)$$

$$+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg){}^\wedge(3/2)\Bigg)+\Bigg(\sqrt{16}\,\pi^2\,(bN+bS)\bigg(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\bigg)\Bigg)\Bigg/\Bigg($$

$$16\ \mathrm{sqrt}\Bigg(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial aW^2}\nu =$$

$$\frac{1}{3}h\Bigg(-\Bigg(\sqrt{16}\Bigg(\pi^2 (bN + bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+ \pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}\right)\Bigg)\text{^}2\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^}(3/2)\Bigg) + \left(\sqrt{16}\,\pi^2 (bN + bS)\left(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}\right)\right)\Bigg/\Bigg($$

$$16\ \mathrm{sqrt}\Bigg(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial bN^2}\nu =$$

$$\frac{1}{3}h\Bigg(-\Bigg(\sqrt{16}\Bigg(\pi^2 (aE + aW)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+ \pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(aE - \frac{h}{\tan(\alpha)} + aW - \frac{h}{\tan(\beta)}\right)\Bigg)\text{^}2\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^}(3/2)\Bigg) + \left(\sqrt{16}\,\pi^2 (aE + aW)\left(aE - \frac{h}{\tan(\alpha)} + aW - \frac{h}{\tan(\beta)}\right)\right)\Bigg/$$

$$\Bigg(16\ \mathrm{sqrt}\Bigg(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial bS^2} v =$$

$$\frac{1}{3} h (- (\sqrt{16} (\pi^2 (aE + aW) ((aE - \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)}))$$

$$+ \pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) (aE - \frac{h}{\tan(\gamma)} + aW - \frac{h}{\tan(\theta)})) \text{\textasciicircum}2) \Big/ (64 (\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) ((aE - \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)})$$

$$+ (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)}))) \text{\textasciicircum(3/2)}) + (\sqrt{16}\, \pi^2 (aE + aW) (aE - \frac{h}{\tan(\gamma)} + aW - \frac{h}{\tan(\theta)})) \Big/$$

$$(16\, \mathrm{sqrt}(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) ((aE - \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)})$$

$$+ (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)})))))$$

$$\frac{\partial^2}{\partial h^2} v =$$

$$\frac{1}{6} \pi (- \frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)}) + (\sqrt{16}\, \pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) ($$

$$- \frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)})) \Big/ (48\, \mathrm{sqrt}(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) ((aE$$

$$- \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)}))))) + \frac{1}{3} h (\frac{1}{4} \pi (\frac{2}{\tan(\alpha)^2} + \frac{2}{\tan(\beta)^2}$$

$$+ \frac{2}{\tan(\gamma)^2} + \frac{2}{\tan(\theta)^2}) - (\sqrt{16}\, \pi^4 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)^2$$

$$(- \frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)})^2) \Big/ (64 (\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS)$$

$$((aE - \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)}))) \text{\textasciicircum(3/2)}) + (\sqrt{16}\, \pi^2 (aE\,bN$$

$$aW\,bN + aE\,bS + aW\,bS) (\frac{2}{\tan(\alpha)^2} + \frac{2}{\tan(\beta)^2} + \frac{2}{\tan(\gamma)^2} + \frac{2}{\tan(\theta)^2})) \Big/ (32\, \mathrm{sqrt}(\pi^2 (aE\,bN + aW\,bN + aE\,bS + aW\,bS) ((aE - \frac{h}{\tan(\alpha)}) (bN - \frac{h}{\tan(\alpha)})$$

$$+ (aW - \frac{h}{\tan(\beta)}) (bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)}) (bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)}) (bS - \frac{h}{\tan(\theta)})))))$$

$$\frac{\partial^{2}}{\partial aE\;\partial aW}\nu \;=$$

$$\frac{1}{3}h\Bigg(-\Bigg(\sqrt{16}\Bigg(\pi^{2}(bN+bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(bN-\frac{h}{\tan(\beta)}+bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg(\pi^{2}(bN+bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)$$

$$+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^{2}(aE\,bN$$

$$aW\,bN+aE\,bS+aW\,bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg){}^\wedge(3/2)$$

$$\Bigg)+\Bigg(\sqrt{16}\Bigg(\pi^{2}(bN+bS)\left(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\right)+\pi^{2}(bN+bS)\left(bN-\frac{h}{\tan(\beta)}+bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\Bigg(aE$$

$$-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^{2}}{\partial aE\;\partial bN}\nu \;=$$

$$\frac{1}{3}h\Bigg(\frac{1}{2}\pi-\Bigg(\sqrt{16}\Bigg(\pi^{2}(aE+aW)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\right)\Bigg)\Bigg(\pi^{2}(bN+bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)$$

$$+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^{2}(aE\,bN$$

$$-aW\,bN+aE\,bS+aW\,bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg){}^\wedge(3/2)$$

$$\Bigg)+\Bigg(\sqrt{16}\Bigg(\pi^{2}\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}(aE+aW)\left(bN-\frac{h}{\tan(\alpha)}+bS-\frac{h}{\tan(\gamma)}\right)+\pi^{2}(bN+bS)\left(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\right)+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^{2}(aE\,bN$$

$$aW\,bN+aE\,bS+aW\,bS)\Bigg(\left(aE-\frac{h}{\tan(\alpha)}\right)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial aE\ \partial bS}\nu =$$

$$\frac{1}{3}h\Bigg(\frac{1}{2}\pi - \Bigg(\sqrt{16}\Bigg(\pi^2(aE + aW)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)$$

$$- \pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(aE - \frac{h}{\tan(\gamma)} + aW - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg(\pi^2(bN + bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right)$$

$$+ \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg) + \pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2(aE\,bN$$

$$\cdot\, aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^}(3/2)$$

$$\Bigg) + \Bigg(\sqrt{16}\Bigg(\pi^2\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+ \pi^2(aE + aW)\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right) + \pi^2(bN + bS)\left(aE - \frac{h}{\tan(\gamma)} + aW - \frac{h}{\tan(\theta)}\right) + \pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^2(aE\,bN$$

$$+ aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial aE\ \partial h}\nu =$$

$$\frac{1}{12}\pi(bN + bS) + \frac{1}{12}\pi\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right) + \Bigg(\sqrt{16}\Bigg(\pi^2(bN + bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg) + \pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(96\ \mathrm{sqrt}\Bigg(\pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\Bigg(aE$$

$$- \frac{h}{\tan(\alpha)}\Bigg)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg) + \frac{1}{3}h\Bigg(\frac{1}{4}\pi\left(-\frac{1}{\tan(\alpha)} - \frac{1}{\tan(\gamma)}\right) - \Bigg(\sqrt{16}$$

$$\pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(-\frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)}\right)\Bigg(\pi^2(bN + bS)$$

$$\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right) + \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+ \pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\left(bN - \frac{h}{\tan(\alpha)} + bS - \frac{h}{\tan(\gamma)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\left(bN - \frac{h}{\tan(\alpha)}\right)$$

$$+ \left(aW - \frac{h}{\tan(\beta)}\right)\left(bN - \frac{h}{\tan(\beta)}\right) + \left(aE - \frac{h}{\tan(\gamma)}\right)\left(bS - \frac{h}{\tan(\gamma)}\right) + \left(aW - \frac{h}{\tan(\theta)}\right)\left(bS - \frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^}(3/2)\Bigg) + \Bigg(\sqrt{16}\,\pi^2(bN + bS)\Bigg(-\frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)}$$

$$- \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^2(aE\,bN + aW\,bN + aE\,bS + aW\,bS)\Bigg(\left(aE - \frac{h}{\tan(\alpha)}\right)\Bigg(bN$$

$$-\frac{h}{\tan(\alpha)}\Bigg)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)+\Bigg(\sqrt{16}\,\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg($$

$$-\frac{1}{\tan(\alpha)}-\frac{1}{\tan(\gamma)}\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)$$

$$+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^{2}}{\partial aW\,\partial bN}\nu=$$

$$\frac{1}{3}h\Bigg(\frac{1}{2}\pi-\Bigg(\sqrt{16}\Bigg(\pi^{2}\,(aE+aW)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\right)\Bigg)\Bigg(\pi^{2}\,(bN+bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)$$

$$+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)+\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(bN-\frac{h}{\tan(\beta)}+bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^{2}\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^(3/2)}$$

$$\Bigg)+\Bigg(\sqrt{16}\Bigg(\pi^{2}\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}\,(aE+aW)\left(bN-\frac{h}{\tan(\beta)}+bS-\frac{h}{\tan(\theta)}\right)+\pi^{2}\,(bN+bS)\left(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\right)+\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg)\Bigg)\Bigg/\Bigg(32\ \mathrm{sqrt}\Bigg(\pi^{2}\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^{2}}{\partial aW\,\partial bS}\nu=$$

$$\frac{1}{3}h\Bigg(\frac{1}{2}\pi-\Bigg(\sqrt{16}\Bigg(\pi^{2}\,(aE+aW)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)$$

$$+\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg(\pi^{2}\,(bN+bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)$$

$$+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)+\pi^{2}\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\left(bN-\frac{h}{\tan(\beta)}+bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^{2}\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\Bigg(aE-\frac{h}{\tan(\alpha)}\Bigg)\left(bN-\frac{h}{\tan(\alpha)}\right)+\left(aW-\frac{h}{\tan(\beta)}\right)\left(bN-\frac{h}{\tan(\beta)}\right)+\left(aE-\frac{h}{\tan(\gamma)}\right)\left(bS-\frac{h}{\tan(\gamma)}\right)+\left(aW-\frac{h}{\tan(\theta)}\right)\left(bS-\frac{h}{\tan(\theta)}\right)\Bigg)\Bigg)\text{^(3/2)}$$

$$) + (\sqrt{16}(\pi^2(((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)}))$$

$$+ \pi^2 (aE + aW)(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}) + \pi^2 (bN + bS)(aE - \frac{h}{\tan(\gamma)} + aW - \frac{h}{\tan(\theta)}) + \pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS))) / (32\ \mathrm{sqrt}(\pi^2 (aE\, bN$$

$$+ aW\, bN + aE\, bS + aW\, bS)((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})))))$$

$$\frac{\partial^2}{\partial aW\, \partial h} v =$$

$$\frac{1}{12}\pi (bN + bS) + \frac{1}{12}\pi (bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}) + (\sqrt{16}(\pi^2 (bN + bS)((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)})$$

$$+ (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})) + \pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}))) / (96\ \mathrm{sqrt}(\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)((aE$$

$$- \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})))) + \frac{1}{3} h (\frac{1}{4}\pi (-\frac{1}{\tan(\beta)} - \frac{1}{\tan(\theta)}) - (\sqrt{16}$$

$$\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)(-\frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)})(\pi^2 (bN + bS)$$

$$((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)}))$$

$$+ \pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)(bN - \frac{h}{\tan(\beta)} + bS - \frac{h}{\tan(\theta)}))) / (64 (\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)})$$

$$+ (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})))^{(3/2)}) + (\sqrt{16}\,\pi^2 (bN + bS)(-\frac{bN - \frac{h}{\tan(\alpha)}}{\tan(\alpha)} - \frac{aE - \frac{h}{\tan(\alpha)}}{\tan(\alpha)}$$

$$- \frac{bN - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{aW - \frac{h}{\tan(\beta)}}{\tan(\beta)} - \frac{bS - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{aE - \frac{h}{\tan(\gamma)}}{\tan(\gamma)} - \frac{bS - \frac{h}{\tan(\theta)}}{\tan(\theta)} - \frac{aW - \frac{h}{\tan(\theta)}}{\tan(\theta)})) / (32\ \mathrm{sqrt}(\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)((aE - \frac{h}{\tan(\alpha)})(bN$$

$$- \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)}) + (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})))) + (\sqrt{16}\,\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)($$

$$- \frac{1}{\tan(\beta)} - \frac{1}{\tan(\theta)})) / (32\ \mathrm{sqrt}(\pi^2 (aE\, bN + aW\, bN + aE\, bS + aW\, bS)((aE - \frac{h}{\tan(\alpha)})(bN - \frac{h}{\tan(\alpha)}) + (aW - \frac{h}{\tan(\beta)})(bN - \frac{h}{\tan(\beta)}) + (aE - \frac{h}{\tan(\gamma)})(bS - \frac{h}{\tan(\gamma)})$$

$$+ (aW - \frac{h}{\tan(\theta)})(bS - \frac{h}{\tan(\theta)})))))$$

$$\frac{\partial^2}{\partial bN\,\partial h}\nu =$$

$$\frac{1}{12}\pi\,(aE+aW)+\frac{1}{12}\pi\left(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\right)+\Bigg(\sqrt{16}\Bigg(\pi^2\,(aE+aW)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)$$

$$+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)+\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\bigg(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\bigg)\Bigg)\Bigg)\Bigg/\Bigg(96\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)+\frac{1}{3}$$

$$h\Bigg(\frac{1}{4}\pi\bigg(-\frac{1}{\tan(\alpha)}-\frac{1}{\tan(\beta)}\bigg)-\Bigg(\sqrt{16}\,\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}$$

$$-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg(\pi^2\,(aE+aW)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)$$

$$+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)+\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\bigg(aE-\frac{h}{\tan(\alpha)}+aW-\frac{h}{\tan(\beta)}\bigg)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE$$

$$-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg){}^\wedge(3/2)\Bigg)+\Bigg(\sqrt{16}\,\pi^2\,(aE+aW)\Bigg($$

$$-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE$$

$$-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)+\Bigg(\sqrt{16}\,\pi^2\,(aE\,bN+aW\,bN+aE\,bS$$

$$+aW\,bS)\bigg(-\frac{1}{\tan(\alpha)}-\frac{1}{\tan(\beta)}\bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)$$

$$+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)\Bigg)$$

$$\frac{\partial^2}{\partial bS\,\partial h}\nu =$$

$$\frac{1}{12}\pi\,(aE+aW)+\frac{1}{12}\pi\left(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\right)+\Bigg(\sqrt{16}\Bigg(\pi^2\,(aE+aW)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)$$

$$+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)+\pi^2\,(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\bigg(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg/\Bigg(96\,\mathrm{sqrt}\Bigg(\pi^2\,(aE\,bN$$

$$+aW\,bN+aE\,bS+aW\,bS)\Bigg(\bigg(aE-\frac{h}{\tan(\alpha)}\bigg)\bigg(bN-\frac{h}{\tan(\alpha)}\bigg)+\bigg(aW-\frac{h}{\tan(\beta)}\bigg)\bigg(bN-\frac{h}{\tan(\beta)}\bigg)+\bigg(aE-\frac{h}{\tan(\gamma)}\bigg)\bigg(bS-\frac{h}{\tan(\gamma)}\bigg)+\bigg(aW-\frac{h}{\tan(\theta)}\bigg)\bigg(bS-\frac{h}{\tan(\theta)}\bigg)\Bigg)\Bigg)\Bigg)+\frac{1}{3}$$

$$h\Bigg(\frac{1}{4}\pi\Big(-\frac{1}{\tan(\gamma)}-\frac{1}{\tan(\theta)}\Big)-\Bigg(\sqrt{16}\,\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Bigg(-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}$$

$$-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg(\pi^{2}(aE+aW)\Big(\Big(aE-\frac{h}{\tan(\alpha)}\Big)\Big(bN-\frac{h}{\tan(\alpha)}\Big)+\Big(aW-\frac{h}{\tan(\beta)}\Big)\Big(bN-\frac{h}{\tan(\beta)}\Big)+\Big(aE-\frac{h}{\tan(\gamma)}\Big)\Big(bS-\frac{h}{\tan(\gamma)}\Big)$$

$$+\Big(aW-\frac{h}{\tan(\theta)}\Big)\Big(bS-\frac{h}{\tan(\theta)}\Big)\Big)+\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Big(aE-\frac{h}{\tan(\gamma)}+aW-\frac{h}{\tan(\theta)}\Big)\Bigg)\Bigg)\Bigg/\Bigg(64\Bigg(\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Big(\Big(aE$$

$$-\frac{h}{\tan(\alpha)}\Big)\Big(bN-\frac{h}{\tan(\alpha)}\Big)+\Big(aW-\frac{h}{\tan(\beta)}\Big)\Big(bN-\frac{h}{\tan(\beta)}\Big)+\Big(aE-\frac{h}{\tan(\gamma)}\Big)\Big(bS-\frac{h}{\tan(\gamma)}\Big)+\Big(aW-\frac{h}{\tan(\theta)}\Big)\Big(bS-\frac{h}{\tan(\theta)}\Big)\Big)\Bigg){}^\wedge(3/2)\Bigg)+\Bigg(\sqrt{16}\,\pi^{2}(aE+aW)\Bigg($$

$$-\frac{bN-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{aE-\frac{h}{\tan(\alpha)}}{\tan(\alpha)}-\frac{bN-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{aW-\frac{h}{\tan(\beta)}}{\tan(\beta)}-\frac{bS-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{aE-\frac{h}{\tan(\gamma)}}{\tan(\gamma)}-\frac{bS-\frac{h}{\tan(\theta)}}{\tan(\theta)}-\frac{aW-\frac{h}{\tan(\theta)}}{\tan(\theta)}\Bigg)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Big(\Big(aE$$

$$-\frac{h}{\tan(\alpha)}\Big)\Big(bN-\frac{h}{\tan(\alpha)}\Big)+\Big(aW-\frac{h}{\tan(\beta)}\Big)\Big(bN-\frac{h}{\tan(\beta)}\Big)+\Big(aE-\frac{h}{\tan(\gamma)}\Big)\Big(bS-\frac{h}{\tan(\gamma)}\Big)+\Big(aW-\frac{h}{\tan(\theta)}\Big)\Big(bS-\frac{h}{\tan(\theta)}\Big)\Big)\Bigg)\Bigg)+\Bigg(\sqrt{16}\,\pi^{2}(aE\,bN+aW\,bN+aE\,bS$$

$$+aW\,bS)\Big(-\frac{1}{\tan(\gamma)}-\frac{1}{\tan(\theta)}\Big)\Bigg)\Bigg/\Bigg(32\,\mathrm{sqrt}\Bigg(\pi^{2}(aE\,bN+aW\,bN+aE\,bS+aW\,bS)\Big(\Big(aE-\frac{h}{\tan(\alpha)}\Big)\Big(bN-\frac{h}{\tan(\alpha)}\Big)+\Big(aW-\frac{h}{\tan(\beta)}\Big)\Big(bN-\frac{h}{\tan(\beta)}\Big)$$

$$+\Big(aE-\frac{h}{\tan(\gamma)}\Big)\Big(bS-\frac{h}{\tan(\gamma)}\Big)+\Big(aW-\frac{h}{\tan(\theta)}\Big)\Big(bS-\frac{h}{\tan(\theta)}\Big)\Big)\Bigg)\Bigg)\Bigg)$$

LIST OF REFERENCES

[1] Abramowitz, M., (1972), "Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables", © New York: Dover, 9th printing. ed, Pages 880.

[2] Achireko, P. K. and Frimpong, S., (1996), "Open Pit Optimization using Neural Networks on Conditionally Simulated Blocks", *in Proceedings of 26th Applications of Computers and Operational Research in the Mineral Industry*, © SME, APCOM Pennsylvania, pp. 137-144.

[3] Ahuja, R. K. and Orlin, J. B., (1989), "A Fast and Simple Algorithm for the Maximum Flow Problem", *Operations Research*, © INFORMS, Vol. 37, pp. 748-759.

[4] Altman, E., (1999), "Constrained Markov Decision Processes", © Chapman & Hall/CRC, Boca Raton ; London, Pages 242.

[5] Asa, E., (2002), "An intelligent 3-D open pit design and optimization using machine learning : adaptive logic networks and neuro-genetic algorithms", Ph.D. Thesis, © University of Alberta. Dept. of Civil and Environmental Engineering, Pages 267.

[6] Askari-Nasab, H., Awuah-Offei, K., and Frimpong, S., (2004), "Stochastic Simulation of Open Pit Pushbacks with a Production Simulator", *in Proceedings of CIM Mining Industry Conference and Exhibition (on CD-ROM)*, © CIM, Edmonton, Alberta, Canada,

[7] Askari-Nasab, H., Frimpong, S., and Awuah-Offei, K., (2005), "Intelligent optimal production scheduling estimator", *in Proceedings of 32nd Application of Computers and Operation Research in the Mineral Industry*, © Taylor & Francis Group, London, Tucson, Arizona, USA, pp. 279-285.

[8] Barnes, R. J., (1982), "Optimizing the Ultimate Pit", MSc Thesis, © Colorado School of Mines, Colorado,

[9] Barnes, R. J. and Bertrand, L., (1990), "Operational Ore/Waste Classification at a Surface Gold Mine", *International Journal of Surface Mining. Reclamation and Environment*, © A.A.Balkema, Vol. 4, pp. 25-29.

[10] Barnes, R. J. and Johnson, T. B., (1982), "Bounding Techniques for the Ultimate Pit Limit Problems", *in Proceedings of 17th APCOM Symposium*, © AIME, New York, pp. 263-273.

[11] Barto, A. G. and Duff, M., (1994), "Monte Carlo matrix inversion and reinforcement learning", *in Proceedings of Advances in Neural Information Processing Systems*, © San Francisco, CA, pp. 687-694.

[12] Bather, J., (2000), "Decision theory : an introduction to dynamic programming and sequential decisions", © John Wiley, Chichester ; New York, Pages ix, 191.

[13] Bellman, R. E., (1957), "Dynamic programming", © Princeton University Press, Princeton,, Pages xxv, 342 p.

[14] Bellman, R. E., (1968), "Some vistas of modern mathematics; dynamic programming, invariant imbedding, and the mathematical biosciences", © University of Kentucky Press, [Lexington], Pages viii, 141 p.

[15] Bellman, R. E. and Dreyfus, S. E., (1962), "Applied dynamic programming", © Princeton University Press, Princeton, N.J.,, Pages 363 p.

[16] Bellman, R. E. and Kalaba, R. E., (1965), "Dynamic programming and modern control theory", © Academic Press, New York,, Pages xi, 112 p.

[17] Bertsekas, D. P., (1976), "Dynamic programming and stochastic control", © Academic Press, New York, Pages xv, 397.

[18] Bertsekas, D. P., (1995), "Dynamic programming and optimal control", © Athena Scientific, Belmont, Mass.,

[19] Bongarcon, D. F. and Guibal, D., (1982), "Algorithm for Parametrizing Reserves Under Different Geometric Constraints", *in Proceedings of 17th APCOM Symposium*, © SME of AIME, pp. 297 - 309.

[20] Bongarcon, D. F. and Marechal, A., (1976), "A New Method for Open Pit Design: Parameterization of the Final Pit Contour", *in Proceedings of 14th APCOM Symposium*, © AIME, pp. 573-583.

[21] Butcher, J. C., (1987), "The numerical analysis of ordinary differential equations : Runge-Kutta and general linear methods", © J. Wiley, Chichester ; New York, Pages xv, 512 p.

[22] Caccetta, L. and Giannini, L. M., (1990), "Application of Operations Research Techniques in Open Pit Mining", *in Proceedings of Asian-Pacific Operations Research : APORS'88 (Byong-Hun Ahn Ed.)*, © Elsevier Science Publishers BV, pp. 707-724.

[23] Caccetta, L. and Hill, S. P., (2003), "An Application of Branch and Cut to Open Pit Mine Scheduling", *Journal of Global Optimization*, © Springer, Vol. 27, November, pp. 349 - 365.

[24] Cartwright, J. H. E. and Piro, O., (1992), "The Dynamics of Runge-Kutta Methods", *Int. J. Bifurcations Chaos*, Vol. 2, pp. 427-449.

[25] Chanda, E. K. and Dagdelen, K., (1995), "Optimal Blending of Mine Production using Goal Programming and Interactive Graphics System", *International Journal of Surface Mining. Reclamation and Environment*, © A.A.Balkema, pp. 203 - 208.

[26] Chanda, E. K. and Wilke, F. L., (1992), "An EPD model of open pit short term production scheduling optimization for stratiform orebodies", *in Proceedings of 23rd APCOM Symposium, SME*, pp. 759-768.

[27] Chapra, S. C. and Canale, R. P., (2006), "Numerical methods for engineers", © McGraw-Hill Higher Education, Boston, 5th ed, Pages xviii, 926 p.

[28] Dagdelen, K., (1985), "Optimum multi-period open pit mine production scheduling by Lagrangian parameterization", Ph.D. Thesis, © Colorado School of Mines, Golden, CO.

[29] Dagdelen, K. and Bongarcon, D. F., (1982), "Towards the Complete Double Parametrization of Recovered Reserves in Open Pit Mining", *in Proceedings of 17th APCOM Symposium*, pp. 288 - 296.

[30] Dekker, K. and Verwer, J. G., (1984), "Stability of Runge-Kutta methods for stiff nonlinear differential equations", © Distributors for the U.S.A. and Canada, Elsevier Science Pub. Co., New York, N.Y., U.S.A., Pages ix, 307 p.

[31] Denardo, E. V., (1982), "Dynamic programming : models and applications", © Prentice-Hall, Englewood Cliffs, N.J., Pages xii, 227.

[32] Denby, B. and Schofield, D., (1994), "Open-pit Design and Scheduling by use of Genetic Algorithms", *Transactions of the IMM Section A*, © IMM, Vol. 103, January - April 1994,, pp. A21 - A26.

[33] Denby, B., Schofield, D., and Hunter, G., (1996), "Genetic Algorithms for Open Pit Scheduling - Extension into 3-Dimensions", *in Proceedings of 5th International Symposium on Mine Planning and Equipment Selection*, © A.A.Balkema/Rotterdam/Brookfield, Sao Paulo, Brazil, pp. 177-186.

[34] Deutsch, C. V., (2002), "Geostatistical reservoir modeling", in *Applied geostatistics series*, New York, © Oxford University Press, pp. 162 - 166.

[35] Deutsch, C. V. and Journel, A. G., (1998), "GSLIB geostatistical software library and user's guide", © Oxford University Press, New York, 2nd ed,

[36] Dijilani, M. C. and Dowd, P. A., (1994), "Optimal Production Scheduling in Open Pit Mines", *Leeds University Mining Association Journal*, © Leads University, pp. 133 -141.

[37] Dowd, P. A., (1976), "Application of dynamic and stochastic programming to optimize cutoff grades and production rates", *Transactions of the Institution of Mining and Metallurgy (Section A: Mining Industry)*, Vol. 85, pp. A22-A31.

[38] Dowd, P. A. and L., E., (1987), "Dynamic Programming Applied to Grade Control in Sub-Level Open Stopping", *Transactions of the Institution of Mining and Metallurgy*, © IMM, Vol. 96, pp. A171 - A178.

[39] Dowd, P. A. and Onur, A. H., (1993), "Open-pit Optimization - Part 1: Optimal Open-pit Design", *Transactions of the Institution of Mining and Metallurgy*, © IMM, Vol. 102, May - Aug, pp. A95 - A104.

[40] Elveli, B., (1995), " Open pit mine design and extraction sequencing by use OR and AI concepts", *International Journal of Surface Mining. Reclamation and Environment*, Vol. 9, pp. 149-153.

[41] Epperson, J. F., (2002), "An introduction to numerical methods and analysis", © J. Wiley, New York, Pages xv, 556 p.

[42] Erarslan, K. and Celebi, N., (2001), "A Simulative Model for Optimum Open Pit Design", *The Canadian Mining and Metallurgical Bulletin.*, © CIM, Vol. 94, October, pp. 59-68.

[43] Esfandiari, B., Aryanezhad, M. B., and Abrishamifar, S. A., (2004), "Open Pit Optimisation Including Mineral Dressing Criteria using 0–1 Non-linear Goal Programming", *Mining Technology, Transactions of the Institutions of Mining and Metallurgy:*, © Maney, London, UK, Vol. 113, January, pp. A3-A13.

[44] Ford, C. R. and Fulkerson, D. R., (1956), "Maximal Flow through a Network", *Canadian Journal of Mathematics*, © Canadian Mathematical Society, Vol. 8, pp. 399-404.

[45] Francois-Bongarcon, D. and Guibal, D., (1982), "Algorithms for Parameterizing Reserves under different Geometrical Constraints", *in Proceedings of 17th Application of Computer Methods in the Mineral Industry*, © AIME, New York, pp. 297-309.

[46] Francois-Bongarcon, D. and Marechal, A., (1976), "A New Method for Open Pit Design: Parametrization of the Final Pit Contour", *in Proceedings of 14th Application of Computer Methods in the Mineral Industry*, © SME of AIME, New York, pp. 573-583.

[47] Frimpong, S. and Achireko, P. K., (1997), "The MCS/MFNN Algorithm for Open Pit Optimization", *International Journal of Surface Mining, Reclamation & Environment*, © A.A.Balkema, Vol. 11, pp. 45-52.

[48] Frimpong, S., Asa, E., and Suglo, R. S., (2001), "Numerical Simulation of Surface Mine Production System Using Pit Shell Simulator", *Mineral Resources Engineering*, © Imperial College Press, London, Vol. 10, pp. 185-203.

[49] Frimpong, S., Asa, E., and Szymanski, J., (1998), "MULSOPS: Multivariate Optimized Pit Shells Simulator for Tactical Mine Planning", *International Journal of Surface Mining, Reclamation & Environment*, © A.A.Balkema, Rotterdam, pp. 163-169.

[50] Frimpong, S., Asa, E., and Szymanski, J., (2002), "Intelligent Modeling: Advances in Open Pit Mine Design and Optimization Research", *International Journal of Surface Mining, Reclamation and Environment*, © Taylor & Francis, Vol. 16, 2, pp. 134 -143.

[51] Frimpong, S., Szymanski, J., and Narsing, A., (2002), "A Computational Intelligent Algorithm for Surface Mine Layouts Optimization", *Simulation*, Vol. 78, October, pp. 600-611.

[52] Frimpong, S., Whiting, J. M., and Szymanski, J., (1998), "Stochastic-Optimization Annealing of an Intelligent Open Pit Mine Design." *Mineral Resource Engineering*, © Imperial College Press, London, Vol. 7, pp. 15-27.

[53] Gauthier, F. J. and Gray, R. G., (1971), "Pit Design by Computer at Gaspe Copper Mines Ltd." *CIM Bulletin*, © CIM, Vol. 64, pp. 95-102.

[54] Gear, C. W., (1971), "Numerical initial value problems in ordinary differential equations", © Prentice-Hall, Englewood Cliffs, N.J.,, Pages xvii, 253.

[55] Gemcom Software International, I., (1998-2005), "Whittle Strategic Mine Planning Software." Vancouver, B.C.

[56] Gershon, M., (1982), "A linear programming approach to mine scheduling optimizations", *in Proceedings of 17th Symposium on the Application of Computers and Operations Research in Mineral Industries (APCOM)*, © AIME, New York, pp. 483-493.

[57] Gershon, M., (1983), "Mine scheduling optimization with mixed integer programming", *Mining Engineering*, © SME, Vol. 35, pp. 351-354.

[58] Gignac, L., (1975), "Computerized Ore Evaluation and Open Pit Design", *in Proceedings of 36th Annual Mining Symposium of the Society of Mining Engineers (AIME)*, © SME, pp. 45-53.

[59] Hairer, E., Lubich, C., and Roche, M., (1989), "The numerical solution of differential-algebraic systems by Runge-Kutta methods", © Springer-Verlag, Berlin ; New York, Pages vii, 139 p.

[60] Halatchev, R. A., (2005), "A model of discounted profit variation of open pit production sequencing optimization", *in Proceedings of Application of Computers and Operations Research in the Mineral Industry*, © Taylor & Francis Group, London, Tucson, Arizona, pp. 315-323.

[61] Hall, G. and Watt, J. M., (1976), "Modern numerical methods for ordinary differential equations", © Clarendon Press, Oxford, Pages 336 p.

[62] Hartley, R., Thomas, L. C., and White, D. J., (1980), "Recent developments in Markov decision processes", © Academic Press, London ; New York, Pages xiv,334.

[63] Henrici, P., (1962), "Discrete Variable Methods in Ordinary Differential Equations", © Wiley Publishers, New York, NY, USA, Pages 407.

[64] Hochbaum, D. S. and Chen, A., (1996), "Performance analysis and best implementations of old and new algorithms for the Open - Pit Mining Problem,"

[65] Hochbaum, D. S. and Chen, A., (2000), "Performance analysis and best implementations of old and new algorithms for the Open - Pit Mining Problem", *Operations Research*, © INFORMS, Vol. 48, Nov. - Dec., pp. 894-914.

[66] Huang, S., (1993), "Computer-based Optimization of Open-Pit Mining Sequences", *Transactions of the Institution of Mining and Metallurgy*, © IMM, Vol. 102, pp. A125 - A133.

[67] Hustrulid, W. A. and Kuchta, M., (1998), "Open pit mine planning & design", Vol. 1. Fundamentals, © A.A. Balkema, Brookfield, VT, Pages 2 v. (xv, 836).

[68] Huttagosol, P. and Cameron, R. E., (1992), "A Computer Design of Ultimate Pit Limit by Using Transportation Algorithm", *in Proceedings of 23rd APCOM Symposium,*, © SME, Littleton, Colorado, Arizona, pp. 443-460.

[69] Ignizio, J. P., (1976), "Goal programming and extensions", © Lexington Books, Lexington, Mass., Pages xix, 261 p.

[70] Isaaks, E. H., (1990), "The application of Monte Carlo methods to the analysis of spatially correlated data, Ph.D. thesis, Stanford University", PhD Thesis, © Stanford University, Stanford, CA,

[71] JetBrains, (2000-2006), "IntelliJ IDEA," 4.5.4 ed. Prague, Czech Republic: JetBrains

[72] Johnson, T. B. and Barnes, R. J., (1988), "Application of the Maximal Flow Algorithm to Ultimate Pit Design", in *Engineering design : better results through operations research methods*, Vol. 8, *Publications in operations research series*, R. R. Levary, Ed. New York, © North-Holland, pp. xv, 713.

[73] Johnson, T. B. and Sharp, W. R., (1971), "A Three- Dimensional Dynamic Programming Method for Determining Ultimate Open Pit Design", © U.S. Bureau of Mines, 25.

[74] Kaelbling, L. P., (1996), "Recent advances in reinforcement learning", © Kluwer Academic, Boston, Pages 290 p.

[75] Kerr, A. J., Neller, T. W., Pilla, C. J. L., and Schompert, M. D., (2003), "Java Resources for Teaching Reinforcement Learning", *in Proceedings of International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA '03)*, © Computer Science Research, Education, & Applications (CSREA) Press, Las Vegas, Nevada, pp. 1497-1501.

[76] Khalokakaie, R., Dowd, P. A., and Fowell, R. J., (2000), "Incorporation of slope design in optimal open pit design algorithms", *Transactions of the Institution of Mining and Metallurgy*, © IMM, London, Vol. 109, pp. A70-A76.

[77] Khalokakaie, R., Dowd, P. A., and Fowell, R. J., (2000), "The Lerchs-Grossmann algorithm with variable slope angles", *Transactions of the Institution of Mining and Metallurgy*, © IMM, Vol. 109, pp. A77-A85.

[78] Khalokakaie, R., Dowd, P. A., and Fowell, R. J., (2000), "A Windows program for optimal open pit design with variable slope angles", *International Journal for Surface Mining, Reclamation and Environment*, © Swets & Zeitlinger, Vol. 14, 4, pp. 261-276.

[79] Kim, Y., (1990), "Long range sequencing with 0-1 programming", *in Proceedings of 22nd Symposium on the Application of Computers and Operations Research in the Mineral Industries (APCOM)*, © Berlin, Germany, pp. 131-144.

[80] Kim, Y. C., (1978), "Ultimate Pit limit Design Methodologies Using Computer Models - The State of the Art", *Mining Engineering*, © SME, Vol. 30, October, pp. 1454 -1459.

[81] Koenigsberg, E., (1982), "The Optimum Contours of an Open Pit Mine : An Application of Dynamic Programming", *in Proceedings of Application of Computers and Operations Research in the Mineral Industry*, © SME of AIME, New York, pp. 274-287.

[82] Korobov, S., (1974), "Method for Determining Optimal Open Pit Limits", © Dept. of Mineral Engineering, Ecole Polytechnique de Montreal, Montreal, Canada,

[83] Kraus, S., (2001), "Strategic negotiation in multiagent environments", © MIT Press, Cambridge, Mass. ; London, Pages xiv, 266 p.

[84] Krige, D. G., (1951), "A Statistical Approach to Some Basic Mine Valuation and Allied Problems at the Witwatersrand", MSc Thesis, © University of Witwatersrand, South Africa,

[85] Lambert, J. D., (1973), "Computational methods in ordinary differential equations", © Wiley, London; New York, Pages xv, 278.

[86] Lemieux, M., (1968), "A Computerized Three Dimensional Optimum Open Pit Design System", *in Proceedings of CIM Annual Meeting Montreal, Canada*, © CIM,

[87] Lemieux, M., (1976), "A Different Method of Modelling a Mineral Deposit for a Three Dimensional Open Pit Computer Design", *in Proceedings of Application of Computer Methods in the Mineral Industry*, © SME of AIME, New York, pp. 557-572.

[88] Lemieux, M., (1979), "Moving Cone Optimization Algorithm." in *Computer methods for the 80's: in the mineral industry*, A. Weiss, Ed. New York, © Society of Mining Engineers of the American Institute of Mining Metallurgical and Petroleum Engineers Inc, pp. 329-345.

[89] Lerchs, H. and Grossmann, I. F., (1965), "Optimum Design of Open-Pit Mines", *The Canadian Mining and Metallurgical Bulletin, Transactions*, © CIM, Vol. LXVIII, pp. 17-24.

[90] Lizotte, Y., (1988), "The Economics of Computerized Open-Pit Design", *International Journal for Surface Mining*, © A.A. Balkema, Rotterdam, pp. 59-78.

[91] Mann, C. and Wilke, F. L., (1992), "Open Pit Short Term Mine Planning for Grade Control - A Combination of CAD Techniques and Linear Programming", *in Proceedings of 23rd APCOM Symposium*, © SME, Colorado, pp. 487 - 497.

[92] Marquardt, D., (1963), "An algorithm for least squares estimation of nonlinear parameters", *SIAM J. Appl. Math*, Vol. 11, pp. 431-441.

[93] Matheron, G., (1962), "Traite de geostatistique appliquee", Vol. 1 (1962) , Vol. 2 (1963), © Paris,

[94] MathWorks, (2005), "MATLAB," 7.04 ed. MA, USA: MathWorks Inc.

[95] Meyer, M., (1969), "Applying linear programming to the design of ultimate pit limits", *Management Science*, © INFORMS, Vol. 16, 2, pp. 121-135.

[96] Michie, D. and Chambers, R. A., (1968), "BOXES: An experiment in adaptive control", in *Machine Intelligence 2*, E. Dale and D. Michie, Eds., © Oliver and Boyd, pp. 137-152.

[97] Millington, P. J., (1991), "Associative reinforcement learning for optimal control", Thesis M.S. --Massachusetts Institute of Technology Dept. of Aeronautics and Astronautics 1991. Thesis, © Pages xiii, 136 leaves.

[98] Narendra, K. S. and L.Thathachar, M. A., (1974), "Learning automata--A survey", *IEEE Transactions on Systems, Man, and Cybernetics*, © Institute of Electrical and Electronics Engineers, New York, 4, pp. 323-334.

[99] Narsing, R. A., (1997), "Stochastic Modeling of a Magnetic Resonance Imaging Performance and Cost in a Large Teaching Hospital", PhD Dissertation Thesis, © University of Alberta, Edmonton, Canada, Pages 255.

[100] Onur, A. H. and Dowd, P. A., (1993), "Open Pit Optimization-Part 2: Production Scheduling and Inclusion of Roadways", *Transactions of the Institution of Mining and Metallurgy*, © The Institution of Mining and Metallurgy, Vol. 102, May-August, pp. A105 - A113.

[101] Pana, M. T. and Davey, R. K., (1965), "The Simulation Approach to Open Pit Design", *in Proceedings of the 5th APCOM Symposium*, © University of Arizona, Arizona, U.S.A, pp. zz1-1124.

[102] Pellegrini, J. C., (1991), "Neural network emulation of temporal second order linear difference equations", Thesis M.S. --Massachusetts Institute of Technology Dept. of Aeronautics and Astronautics 1991. Thesis, Pages 90 leaves.

[103] Picard, J. C., (1976), "Maximum Closure of a Graph and Applications to Combinatorial Problems", *Management Science*, © INFORMS, Vol. 22, pp. 1268 - 1272.

[104] Puterman, M. L., (1978), "Dynamic programming and its applications", © Academic Press, New York, Pages xv, 410.

[105] Puterman, M. L., (1994), "Markov decision processes : discrete stochastic dynamic programming", © John Wiley & Sons, New York, Pages xvii, 649.

[106] Ramazan, S. and Dimitrakopoulos, R., (2004), "Traditional and new MIP models for production scheduling with in-situ grade variability", *International Journal of Surface Mining, Reclamation & Environment*, Vol. 18, 2, pp. 85-98.

[107] Ratitch, B., (2005), "On characteristics of Markov decision processes and reinforcement learning in large domains", Thesis Ph D Thesis, Pages xix, 286 leaves.

[108] Robinson, R. H. and Prenn, N. B., (1973), "An Optimum Pit Design Model", *in Proceedings of 10th Symposium on the Application of Computers and Operations Research in Mineral Industries (APCOM)*, © SAIMM, Johannesburg, South Africa, pp. 155-163.

[109] Saatdjian, E., (2000), "Transport phenomena : equations and numerical solutions", © John Wiley, New York, Pages xii, 414 p.

[110] Samet, H., (1990), "Applications of spatial data structures : computer graphics, image processing, and GIS", © Addison-Wesley, Reading, Mass., Pages xv, 507.

[111] Samet, H., (1990), "The design and analysis of spatial data structures", © Addison-Wesley, Reading, Mass., Pages xvii, 493.

[112] Schniederjans, M. J., (1995), "Goal programming : methodology and applications", © Kluwer Academic Publishers, Boston, Pages xiv, 219 p.

[113] Singh, S. P. and Sutton, R. S., (1996), "Reinforcement learning with replacing eligibility traces", *Machine Learning*, © Kluwer Academic Publishers, Boston, Vol. 22, pp. 123-158.

[114] Smith, H. P. and Wagner, J. C., (2003), "A Case Study in Manual and Automated Monte Carlo Variance Reduction With a Deep Penetration Reactor Shielding Problem", *in Proceedings of Nuclear Mathematical and Computational Sciences: A Century in Review, A Century Anew*, © American Nuclear Society, LaGrange Park, IL, Gatlinburg, Tennessee, pp. on CD-ROM.

[115] Smith, M. L., (1999), "Optimizing Inventory Stockpiles and Mine Production: An Application of Separable and Goal Programming to Phosphate Mining using AMPL/CPLEX", *CIM Bulletin*, © CIM, pp. 61-64.

[116] Stetter, H. J., (1973), "Analysis of discretization methods for ordinary differential equations", © Springer, Berlin New York, Pages xvi, 388.

[117] Sun Microsystems, I., (1994-2006), "Java Programming Language," 1.4.2_08 ed. 4150 Network Circle, Santa Clara, CA, USA

[118] Sutton, R. S., (1988), "Learning to predict by the method of temporal differences", *Machine Learning*, © Kluwer Academic Publishers, Boston, 3, pp. 9-44.

[119] Sutton, R. S., (1992), "Reinforcement learning", © Kluwer Academic Publishers, Boston, Pages 171.

[120] Sutton, R. S. and Barto, A. G., (1998), "Reinforcement learning : an introduction", © MIT Press, Cambridge, Mass., Pages xviii, 322.

[121] Sutton, R. S. and Santamaria, J. C., (2005), "A standard interface for reinforcement learning software in C++, version 1.1", Retrieved 2005, Dec 4 from: http://www-anw.cs.umass.edu/~rich/RLinterface/RLI-Cplusplus.html.

[122] Tamiz, M., (1996), "Multi-objective programming and goal programming : theories and applications", © Springer-Verlag, Berlin ; New York, Pages vi, 359 p.

[123] Tanino, T., Tanaka, T., and Inuiguchi, M., (2003), "Multi-objective programming and goal programming : theory and applications", © Springer, Berlin ; New York, Pages xix, 427 p.

[124] Taylor, J. B., (1971), "Incorporation of Access Roads into Computer Generated Open Pits", *in Proceedings of 9th Symposium of Computers and Operations Research in the Mineral Industries (APCOM)*, © CIM, Montreal, Canada, pp. 339-343.

[125] Tessier, C., Chaudron, L., and Müller, H. J., (2001), "Conflicting agents : conflict management in multi-agent systems", © Kluwer Academic, Boston, Pages xiii, 335 p.

[126] Thomas, G. S., (1996), "Pit Optimisation and Mine Production Scheduling - The Way Ahead", *in Proceedings of Application of Computers and Operations Research in the Mineral Industry*, © Society for Mining, Metallurgy, and Repeat, Inc, The Pennsylvania State University, University Park, Pennsylvania, pp. 221-228.

[127] Tolwinski, B. and Underwood, R., (1992), "An Algorithm to Estimate the Optimal Evolution of an Open Pit Mine", *in Proceedings of 23rd APCOM Symposium*, © SME, Littleton, Colorado, University of Arizona, pp. 399 - 409.

[128] Trzaskalik, T. and Michnik, J., (2002), "Multiple objective and goal programming : recent developments", © Physica-Verlag, Heidelberg ; New York, Pages ix, 432 p.

[129] Tsitsiklis, J. N., (1997), "An analysis of temporal-difference learning with function approximation", © Center for Intelligent Control Systems, Providence, R.I. ; Cambridge, Mass., Pages 34.

[130] Underwood, R. and Tolwinski, B., (1998), "A Mathematical Programming Viewpoint for Solving the Ultimate Pit Problem", *European Journal of Operational Research*, © Elsevier, Vol. 107, pp. 96-107.

[131] Wageningen, A. V., Dunn, P. G., and Muldowney, D. M., (2005), "sequence optimization for long-term mine planning", *in Proceedings of 32 nd Application of Computers and Operation Research in the Mineral Industry*, © Taylor & Francis Group, London, Tucson, Arizona, USA, pp. 667-673.

[132] Wang, Q. and Sevim, H., (1992), "Enhanced Production Planning in Open Pit Mining through Intelligent Dynamic Search", *in Proceedings of 23rd APCOM Symposium*, © SME, Colorado, pp. 461 - 471.

[133] Watkins, C. J., (1989), "Learning from Delayed Rewards", PhD Thesis, © Cambridge University, Cambridge, England,

[134] Watkins, C. J. C. H. and Dayan, P., (1992), "Q-learning", *Machine Learning*, © Kluwer Academic Publishers, Boston, Vol. 8, pp. 279-292.

[135] Weiss, G., (1999), "Multiagent systems : a modern approach to distributed artificial intelligence", © MIT Press, Cambridge, Mass. ; London, Pages xxiii, 619 p.

[136] Werbos, P. J., (1987), "Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research", *IEEE Transactions on Systems, Man, and Cybernetics*, © Institute of Electrical and Electronics Engineers, New York, pp. 7-20.

[137] White, D. J., (1978), "Finite dynamic programming : an approach to finite Markov decision processes", © Wiley, Chichester Eng. ; New York, Pages xiii, 204.

[138] Whittle, J., (1988), "Beyond Optimization in Open Pit Mining", *in Proceedings of First Canadian Conference on Computer Applications in Mineral Industry*, pp. 331-337.

[139] Whittle, J., (1989), "The facts and fallacies of open-pit design," in *Manuscript, Whittle Programming Pty Ltd.* North Balwyn, Victoria, Australia

[140] Whittle Programming Pty, L., (1992), "User Manual of Three-D Whittle Open Pit Optimization Software." Perth, Australia

[141] Whittle Programming Pty, L., (1998), "Reference Manual of Four -X Strategic Planning Software For Open Pit Mines." Perth, Australia

[142] Wilke, F. L. and Wright, E. A., (1984), "Determining the Optimal Ultimate Pit Design for Hard Rock Open Pit Mines using Dynamic Programming", *Erzmetall*, © Erzmetall, Vol. 37, pp. 139 -144.

[143] Williams, C. E., (1970), "Computerized Mine Planning", *in Proceedings of The American Institute of Institute of Mining, Metallurgical, and Petroleum Engineers*, Tucson, Arizona, pp. 97-104.

[144] Winston, P. H., (1991), "Artificial intelligence", © Addison-Wesley, Reading, MA, 3rd ed, Pages xxv, 737.

[145] Wooldridge, M., (2002), "An Introduction to Multi-Agent Systems", © John Wiley and Sons Limited, Chichester, UK, Pages 348.

[146] Wooldridge, M. and Jennings, N., (1995), "Intelligent Agents: Theory and Practice", *Knowledge Engineering Review*, © Cambridge University Press, Vol. 10, 2, pp. 115-152.

[147] Wright, E. A., (1990), "Open pit mine design models : an introduction with FORTRAN/77 programs", © Trans Tech Publications, Clausthal-Zellerfeld, Federal Republic of Germany, Pages xi, 187 p.

[148] Yamatomi, J., Mogi, G., Akaike, A., and Yamaguchi, U., (1995), "Selective extraction dynamic cone algorithm for three dimensional open pit designs", *in Proceedings of 25th Application of Computers and Operations Research in Mineral Industries*, © Australian Institute of Mining and Metallurgy, Brisbane, pp. 267-274.

[149] Yegulalp, T. M. and Arias, J. A., (1992), "A Fast Algorithm to Solve the Ultimate Pit Limit Problem", *in Proceedings of 23rd APCOM Symposium*, © AIME, Littleton, Colorado, pp. 391-397.

[150] Youdi, Z., Qingziang, C., and Lixin, W., (1992), "Combined Approach for Surface Mine Short Term Planning Optimization", *in Proceedings of 23rd APCOM Symposium*, © SME, Colorado, pp. 499 - 506.

[151] Yun, Q. and Lu, C. W., (1992), "Optimization for the Determination of Transportation System in Open Pit Mines", *in Proceedings of 23rd Symposium on the Application of Computers and Operations Research in the Mineral Industries (APCOM)*, © Littleton, Colorado: AIME, pp. 535-545.

[152] Zhang, Y. D., Chang, Y. P., and Su, J., (1993), "Application of Goal Programming in Open Pit Planning", *International Journal for Surface Mining and Reclamation*, © A.A.Balkema, pp. 41 - 45.

[153] Zhao, Y. and Kim, Y. C., (1992), "A New Optimum Pit Limit Design Algorithm", *in Proceedings of 23rd APCOM Symposium*, © SME, Littleton, Colorado, University of Arizona, pp. 423-434.