

https://dspace.org/

An economic model of an open pit mine

Robinson, Robert Howard

2015-08-18
http://hdl.handle.net/10539/18256

Downloaded from a DSpace repository. https://dspace.org/

TOPOGRAPHIC MAP OF PIT IN YEAR 41

J

I	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50
2	0	0	0	0	0	0	0	2	2	2	4	2	2	2	2	2	2	2	2	2	2	0	0	0	
4	0	0	0	0	2	2	2	4	4	4	6	4	4	2	2	2	2	2	2	2	2	0	0	0	0
6	0	0	0	2	2	4	4	6	6	6	8	8	4	4	2	2	2	2	2	2	0	0	0	0	0
8	0	0	2	2	4	6	6	8	8	8	8	8	6	6	4	4	4	2	2	2	0	0	0	0	0
10	0	2	2	4	6	6	8	10	10	10	10	10	8	8	6	6	6	4	4	4	2	0	0	0	0
12	0	2	4	6	6	8	10	12	12	10	10	10	12	10	8	8	8	6	6	6	4	2	0	0	0
14	2	4	4	6	8	10	12	12	14	12	12	12	14	10	10	10	10	8	8	8	6	4	2	0	0
16	2	4	6	8	10	12	12	12	12	14	14	14	14	12	12	14	12	12	12	10	8	6	4	2	0
18	4	6	8	10	10	12	14	12	10	12	14	14	14	16	14	14	14	14	12	12	10	6	4	2	0
20	6	6	8	10	10	12	14	14	12	14	14	14	14	14	14	14	14	12	12	12	10	6	4	2	0
22	6	6	6	8	8	10	12	14	14	14	14	12	12	12	12	14	14	10	10	10	8	6	4	2	0
24	6	8	8	8	8	8	10	12	14	14	12	10	10	12	10	14	12	10	10	8	6	4	2	0	0
26	6	8	8	6	6	8	8	10	12	12	10	8	8	10	10	10	10	8	6	6	4	2	0	0	0
28	6	6	6	6	6	6	8	8	10	10	8	8	6	8	10	10	8	6	4	4	4	0	0	0	0
30	6	6	6	6	6	6	6	6	8	8	6	6	4	6	8	8	6	6	2	4	0	0	0	0	0

FIG. 7.4 (CONTINUED)

The capital requirements (CR) listed in the cash flow are the development expenditures for the combination of plant facilities selected from Table 7•1 to exploit the example deposit• Recall that these capital requirements were discounted to the first year to make up the entry: pit = 7m tons/yr•, concentrator = 1•7m tons/yr•, refinery = 20t tons/yr• and overall recovery efficiency = 95%•

The last column lists the loan capital repayments (LR)• All entries are in millions of cash units•

The total present value (-5•39) of the cash flow in Table 7•18 is not identical with the corresponding present value of the production targets for the preceding plant combination• (See Table 7•9•) Obviously, this must be so, if the mining conditions are such that the production targets cannot be met•

The cash flow analysis is the ultimate measure of any open pit mining solution's worth• That solution resulting from the input mining proposal which yields the greatest profitability, is the obvious choice for exploiting the user's deposit• Note that this solution for the example deposit is unprofitable with a negative total present value and a discounted cash flow yield less than the opportunity cost of investment capital• However, it is possible that the deposit could be made profitable with a different mining proposal and better scheduling and proportioning of the loan and equity capital•

7• 4 AN INVALID SOLUTION

An invalid solution of the MP Programme is illustrated in Table 7•19• This table was generated by the Programme for the example deposit with the following plant capacities:

1) Pit = 5•0m tons per year•

2) Concentrator = 3m tons per year, and

3) Refinery = 20t tons per year•

Recall from Subsection 6•7•2 that an invalid solution arises when the simulated waste plus ore production exceeds the

YR	REV	INT	TAX	OB	CFI	ECAP	CF	PV	LC	CR	LR
1	0•59	•0	0•0	2•61	-2•02	11•0	-13•02	-5•39	0•0	7•1	0•0
2	0•56	•0	0•0	2•53	-1•97	4•2	-6•22	7•19	0•0	5•6	0•0
3	0•68	•0	0•0	2•78	-2•10	0•0	-2•10	13•99	0•0	5•5	0•0
4	2•14	•56	0•0	3•33	-1•75	0•0	-1•75	17•21	7•5	6•5	0•0
5	2•93	•81	0•0	1•59	0•53	0•0	0•53	20•34	3•2	7•6	0•0
6	2•26	•81	0•0	1•42	0•03	0•0	0•03	21•43	0•0	7•5	0•0
7	2•08	•81	0•0	1•51	-0•24	0•0	-0•24	23•11	0•0	0•0	2•0
8	2•31	•66	0•0	1•28	0•37	0•0	0•37	25•20	0•0	0•0	2•0
9	3•33	•51	0•0	1•49	1•33	0•0	1•33	26•84	0•0	0•0	2•0
10	3•84	•36	0•0	1•50	1•98	0•0	1•98	27•65	0•0	0•0	2•0
11	2•52	•21	0•0	1•35	0•96	0•0	0•96	27•88	0•0	0•0	2•0
12	2•32	•06	0•0	1•32	0•94	0•0	0•94	29•15	0•0	0•0	0•7
13	2•67	•0	0•0	1•74	0•93	0•0	0•93	30•54	0•0	0•0	0•0
14	4•02	•0	1•2	1•34	1•43	0•0	1•43	32•05	0•0	0•0	0•0
15	3•68	•0	1•0	0•38	2•30	0•0	2•30	33•19	0•0	0•0	0•0
16	6•51	•0	1•9	0•21	4•41	0•0	4•41	33•54	0•0	0•0	0•0
-	-	-	-	-	-	-	-	-	-	-	-
25	4•79	•0	1•4	0•66	2•78	0•0	2•78	26•96	0•0	0•0	0•0
26	8•79	•0	2•6	0•0	6•18	0•0	6•18	26•33	0•0	0•0	0•0
27	5•89	•0	1•7	0•14	4•06	0•0	4•06	22•26	0•0	0•0	0•0
28	2•96	•0	0•8	0•40	1•79	0•0	1•79	19•98	0•0	0•0	0•0
29	3•59	•0	1•0	0•12	2•50	0•0	2•50	19•79	0•0	0•0	0•0
30	4•23	•0	1•2	0•03	3•03	0•0	3•03	18•87	0•0	0•0	0•0
31	2•97	•0	0•8	0•01	2•19	0•0	2•19	17•35	0•0	0•0	0•0
32	3•27	•0	0•9	0•0	2•40	0•0	2•40	16•55	0•0	0•0	0•0
33	3•03	•0	0•8	1•01	1•23	0•0	1•23	15•47	0•0	0•0	0•0
34	5•81	•0	1•7	0•0	4•14	0•0	4•14	15•43	0•0	0•0	0•0
35	3•70	•0	1•0	0•0	2•70	0•0	2•70	12•58	0•0	0•0	0•0
36	3•93	•0	1•1	0•01	2•84	0•0	2•84	10•89	0•0	0•0	0•0
37	3•76	•0	1•0	0•13	2•61	0•0	2•61	8•92	0•0	0•0	0•0
38	3•26	•0	0•9	0•27	2•13	0•0	2•13	7•02	0•0	0•0	0•0
39	4•79	•0	1•4	0•0	3•44	0•0	3•44	5•46	0•0	0•0	0•0
40	3•24	•0	0•9	0•0	2•38	0•0	2•38	2•45	0•0	0•0	0•0
41	0•29	•0	0•0	0•0	0•29	0•0	0•29	0•27	0•0	0•0	0•0

DISCOUNTED CASH FLOW YIELD=6•24%

TABLE 7•18

total pit capacity, as it has in the third, fourth, fifth years, etc• This is due to the fact that the production targets have been computed from an inadequate pit capacity•

SIMULATED PIT PRODUCTION SCHEDULE(FOURTH ITERATION CYCLE)

YR	OVERBURDEN M TONS	STRIPPING COSTS M CASH UNITS	ORE M TONS	REVENUE M CASH UNITS
1	2•53	1•25	1•27	0•68
2	2•96	1•40	1•27	0•51
3	6•62	3•41	1•27	1•06
4	5•49	2•69	1•27	1•66
5	2•25	1•12	1•41	2•20
6	4•64	2•45	2•25	3•30
7	1•55	0•78	2•25	2•62
8	1•55	0•79	2•25	3•19
9	1•13	0•60	2•25	5•07
10	5•49	2•65	2•11	3•05
11	6•62	3•48	1•97	1•84
12	5•49	2•95	1•97	3•09
13	5•35	2•90	1•97	2•56
14	0•28	0•15	1•97	4•53
15	1•69	1•02	1•97	7•26
16	0•84	0•53	2•53	11•35
17	0•56	0•20	2•96	6•90
18	1•27	0•77	2•96	8•01
19	0•14	0•09	2•96	5•97
20	4•22	2•85	2•96	7•85
21	0•14	0•03	2•96	12•24
22	1•13	0•57	2•96	8•79
23	1•97	1•22	2•96	6•50
24	0•99	0•43	3•04	5•99
25	0•0	0•0	2•96	6•61
26	0•0	0•0	2•96	8•93
27	0•28	0•18	2•96	7•61
28	0•56	0•27	2•96	5•43
29	2•12	1•07	2•96	7•88
30	4•85	3•26	1•18	1•00

TOTAL REVENUE=154•24 MILLION CASH UNITS

TOTAL NUMBER OF PIT INCREMENTS=6

TABLE 7•19

CHAPTER 8 THE MINE PLAN PROGRAMME

The MP Programme is made up of three modules which are needed for the different computations in the various iteration cycles of the open pit mining solution. The computations within the iterative solution are varied by calling different modules. The construction of these modules is illustrated in Table 8.1. Each module is created by including the appropriate calling programme and four subroutines. The calling programmes and subroutines are stored individually in a partitioned data set. There is one module per cycle for the execution of the first two cycles of the iterative solution. The third module is used to execute all additional cycles.

Having divided the MP Programme in this way, it became possible to exclude some time consuming operations in the first two iteration cycles. The excluded operations are not needed because they are refinements unnecessary to the initial accuracy. The large computer time per cycle also made it advantageous to segment the computations for increased control of the solution and to achieve a minimum time loss from computer failures. The results of each iteration cycle are stored on a magnetic disk. If there is a failure in the solution, it is unnecessary to restart the Programme from the beginning. The user merely is required to retrieve the results of the previous cycle and proceed.

8. 1 JOB CONTROL LANGUAGE

The MP Programme is executed in the three job steps illustrated in Fig. 8.1 through 8.3. Note that certain entries in the figures are written in lower case letters. These entries name that which is to be replaced in their position. The label of the disk pack on which the partitioned, direct access and sequential data sets reside is substituted for 'disk pack label'. The position of the source decks and data sets within the control cards is

indicated with explanatory phrases. The remaining substitutions are explained in the following description of the job steps:

JOB STEP 1:

The calling programmes and subroutines are compiled, link-edited and stored individually in a partioned data set. The control cards illustrated in Fig. 8.1 are used repetitively to load the source deck of cards for each member of the partitioned data set. The names of the calling programmes and subroutines, as listed in Table 8.1, are substituted, in turn, for the internal name of the LKED.SYSLMOD card's DSNAME parameter.

	MP PROGRAMME		
	MODULE 1	MODULE 2	MODULE 3
CALLING PROGRAMMES	MAIN1	MAIN2	MAIN3
SUBROUTINES	CCG1	COG2---|----->	
	COGSUB-|----->	-----|----->	
	PD-----|----->	-----|----->	
	SUBPD--|----->	-----|----->	

TABLE 8. 1

JOB STEP 2:

The second job step creates one of the three modules (MP1, MP2 or MP3) by including the appropriate members of the partioned data set. The modules are made up by executing the control card list in Fig. 8.2 with the following substitutions:

1) 'n' and 'nn' are replaced with '1' to create the module MP1 which computes the first iteration cycle,

2) 'n' and 'nn' are replaced with '2' to create the module

```
(job card)
// EXEC FORTHCL,PARM.LKED='MAP'
//SYSIN DD *
(source card deck for the calling programmes and subroutines, in turn)
/*
//LKED.SYSLMOD DD DSNAME=PROGMP(name),DISP=(NEW,KEEP),UNIT=SYSDA,      X
//                VOLUME =SER=disk pack label,SPACE=(CYL,(20,1,20))
//
```

The preceding control cards are used to load the first member of the partioned data set only. The 'LKED.SYSLMOD' card must be replaced by the following card to compile the remaining data set members or to recompile any of these members:

```
//LKED.SYSLMOD DD DSNAME=PROGMP(name),DISP=OLD,VOLUME=SER=disk pack label
```

FIG. 8.1 COMPILATION AND LINK-EDIT JOB STEP

```
(job card)
// EXEC PGM=IEWL,PARM='MAP',REGION=150K
//SYSPRINT DD SYSOUT=A
//SYSUT DD UNIT=SYSDA,VOLUME=SER=disk pack label,SPACE=(TRK,(50,10))
//SYSLIB DD DSNAME=SYS1.FORTLIB,DISP=SHR
//SYSLMOD DD DSNAME=PROG(MPn),DISP=(NEW,KEEP),UNIT=SYSDA,                X
//                VOLUME=SER=disk pack label,SPACE(CYL,(10,1,10))        X
//                DCB=(RECFM=U,BLKSIZE=3625)
//A DD DSNAME=PROG(P,DISP=OLD,VOLUME=SER=disk pack label,UNIT=SYSDA
//SYSLIN DD *,DCB=BLKSIZE=80
 ENTRY MAIN
 INCLUDE A(MAINn)
 INCLUDE A(COMMON)
 INCLUDE A(COSUB)
 INCLUDE A(PD)
 INCLUDE A(SUBPD)
/*
//
```

The 'DISP' parameter must be changed to 'OLD', and the 'DCB' and 'SPACE' parameters deleted on the 'SYSLMOD' card for any executions of this job step after the first, as follows:

```
//SYSLMOD DD DSNAME=PROG(MPn),DISP=OLD,UNIT=SYSDA,VOLUME=SER=disk pack label
```

FIG. 8.1 CREATION OF MODULES JOB STEP

MP2 which computes the second iteration cycle, and

3) 'n' is replaced by '3' and 'nn' by '2' to create the module MP3 which computes all iteration cycles from the third one on.

JOB STEP 3:

The module created in the second job step is executed by the job control cards listed in Fig. 8.3. The module number 'n' is replaced by the one created in the second step.

Each module required an average of thirty three minutes to execute a solution for a grid system with 13,500 grid-intersections on the International Business Machines Corporation's IBM 360/50 computer.

8.2 PREPARATION OF THE PROGRAMME FOR EXECUTION

The steps needed to implement the MP Programme are listed below. The Programme can be prepared for execution by using these steps to locate the comments within the Programme listing which describe any required modifications.

STEP 1:

The dimensions must be specified for the FORTRAN arrays, sequential data sets and direct access data sets. These dimensions are

A) defined in Section 8.8 for the specification statements of the arrays,

B) given for the sequential data sets in Section 8.1, and

C) given for the direct access data sets in Section 8.1 for the SPACE parameters and in Subsection 8.3.2 for the DEFINE FILE statement.

STEP 2:

The grid system is already defined for the preceding Programmes in the Model. However, the maximum number of grid co-ordinates NI, NJ and NK in the I, J and K co-ordinate

```
(job card)
//JOBLIB DD DSNAME=PROG,DISP=(OLD,PASS),UNIT=2311,VOLUME=SER=disk pack label
//A EXEC PGM=MPn,TIME=30,REGION=550K
(sequential and direct access data sets--see the list below)
//GO.FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=133,BUFNO=1)
//GO.FT05F001 DD *
(punched card input data)
/*
//
```

The sequential and direct access data sets required for the three modules (MP1 , MP2, and MP3) are:

1) For the first module (MP1):

```
//GO.FT10F001 DD DSNAME=TMODEL,DISP=OLD,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT11F001 DD DSNAME=UPDATE,DISP=(NEW,KEEP),UNIT=2311,                    X
//                VOLUME=SER=disk pack label,SPACE=(CYL,(3,1)),              X
//                DCB=(RECFM=VB,LRECL=8,BLKSIZE=3204)
//GO.FT12F001 DD DSNAME=ORDERS,DISP=(NEW,KEEP),UNIT=2311                     X
//                VOLUME=SER=disk pack label,SPACE=(CYL,(3,1))               X
//                DCB=(RECFM=VB,LRECL=12,BLKSIZE=3544)
//GO.FT13F001 DD DSNAME=BOGRAD,DISP=(NEW,KEEP),UNIT=2311,                    X
//                VOLUME=SER=disk pack label,SPACE=(CYL,(1,1))               X
//                DCB=(RECFM=VB,LRECL=10,BLKSIZE=3204)
```

2) For the second module (MP2):

```
//GO.FT11F001 DD DSNAME=UPDATE,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT12F001 DD DSNAME=ORDERS,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT13F001 DD DSNAME=BOGRAD,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
```

(continued--next page)

FIG. 8.3 EXECUTION JOB STEP

```
//GO.FT14F001 DD DSNAME=UPDATE2,DISP=(NEW,KEEP),UNIT=2311,                X
//             VOLUME=SER=disk pack label,SPACE=(CYL,(2,1)),               X
//             DCB=(RECFM=VB,LRECL=12,BLKSIZE=3544)
//GO.FT15F001 DD DSNAME=ORDERS2,DISP=(NEW,KEEP),UNIT=2311                  X
//             VOLUME=SER=disk pack label,SPACE=(TRK,(2,1)),               X
//             DCB=(RECFM=VB,LRECL=12,BLKSIZE=2404)
//GO.FT16F001 DD DSNAME=PRICES,DISP=(NEW,KEEP),UNIT=2311                   X
//             VOLUME=SER=disk pack label,SPACE=(TRK,(1,1)),               X
//             DCB=(RECFM=VB,LRECL=32,BLKSIZE=3204)
```

3) *For the third module (MP3):*

```
//GO.FT13F001 DD DSNAME=DOGRAD,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT14F001 DD DSNAME=UPDATE2,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT15F001 DD DSNAME=ORDERS2,DISP=SHR,UNIT=2311,VOLUME=SER=disk pack label
//GO.FT16F001 DD DSNAME=PRICES,DISP=SHR,UNIT=2311,VOLUME=SER= disk pack label
```

FIG. 8.3 (CONTINUED)

rections, respectively, are required as part of the MP Programme's input data.

STEP 3:

The appropriate number of FORTRAN variables must be created for each recoverable mineral reported by the Grade Prediction Programme. This modification only is required for the calling programmes. The variables and their location in the programmes are described in the comments of Subsection 8.3.2.

STEP 4:

A factor 'GS' is selected to scale the mineral grades up to integer values. Recall from Section 6.3 that the integer mineral values are used for the subscript 'M' of 'ORERES'. Additionally, a mineral grade increment 'GI' is selected to limit the size of subscript 'M'. This increment groups all minerals occurring in the same grade interval into one class. A balance must be made between accuracy and the ORERES array size. The scale factor 'GS' must be large enough to avoid INTEGER truncation. In contrast, the grade increment 'GI' must be small enough that subscript 'M' of ORERES does not become excessively large. 'GI' and 'GS' are input data to the MP Programme.

STEP 5:

The initial assumption for the subscript 'M' of 'ORERES' is given by the user. (See Subsection 6.3.1 for a description of how this is done.) This assumption is located in the calling programme MAIN1. (Subsection 8.3.2)

STEP 6:

Any geologic or physical features, legal boundaries and costing situations which may affect the following operations or variables must be simulated:

A) T-S net cash value distribution,

B) design of the final pit surface,

C) the tonnage factors, and

D) pit production simulation.

There is a comment at those places in the Programme lists where the simulation of these features is required.

STEP 7:

The variable 'D' in the cone expressions for the final pit surface design and waste mining simulation routines must be dependant on the pit wall-angles, if these angles vary. The relationship between 'D' and the pit wall-angles is explained further in the comments in the lists of subroutines PD and SUBPD.

STEP 8:

The ore excavation sequence programmed in subroutine PD must be tailored to the user's requirements. The desired number of ore working faces per bench and the number of exposed ore benches, and the direction of their advance is simulated as explained in Subsection 6.7.1.

8. 3 CALLING PROGRAMMES

A verbal description of the first calling programme's computational sequence is presented in Subsection 8.3.1. It is then listed in the next Subsection with comments to explain some of its features. The last two Subsections contain the lists of the other calling programmes with comments explaining how they differ from those preceding them.

8. 3. 1 COMPUTATIONAL SEQUENCE OF MAIN1

LINES 001 - 007

Specification statements.

LINES 008 - 054

Input statements.

LINES 059 - 080

DO loop 11 computes:

A) an equivalent mineral grade 'EQGRAD' to represent all the deposit's recoverable minerals, and

B) the initial assumption for the T-S mineral distribution.

LINES 083 - 093

The first estimate of the unit production costs is computed.

LINE 094

The following values are supplied by the nested subroutines COG1 and COGSUB for the selected plant combination:

A) the annual cut-off grades 'CUTOFF' and production targets 'MIN, CON, OREF';

B) the overall recovery efficiency 'YIELD', and

C) an average 'AVEG' of the annual cut-off grades 'CUTOFF'.

LINES 095 - 158

These computations execute the first estimate of the T-S net cash value distribution 'VAL' and the second estimate of the average unit production costs 'E, C, R'. At this point in the solution, it is not know when the grid-blocks are mined. Consequently, it is assumed that they are mined in the first year 'T' and, in addition, their equivalent mineral contents are compared with the average cut-off grade 'AVEG'.

LINE 151

The following computations are executed by the nested subroutines PD and SUBPD:

A) the final pit surface is designed and

B) the pit production is simulated.

LINES 152 - 159

The annual revenue is totalled 'PNEW' and the number of pit increments computed 'MAXN'. These results are printed out to assist the user in monitoring the solution.

LINES 160 - 179

The second estimate of the T-S mineral distribution 'ORERES' is computed with the grid-block excavation times 'YR' from the pit production simulation in subroutine PD.

LINES 180 - 185

The results of the first module 'MP1' are stored on a disk pack for future use. The T-S mineral distribution 'ORERES' and the grid-block excavation times 'YR' are used to update the iterative solution. The remaining data (TOPO, EQGRAD, IGRAD1, IGRAD2) are stored for rapid access rather than being recalculated in the other modules.

LINES 186 - 209

Yearly topographic maps are printed to illustrate the pit production simulation.

8. 3. 2 A LIST OF MAIN1

----------COMMENT----------

The example deposit has two recoverable minerals 'GRAD1' and 'GRAD2' whose present selling prices are 'S1' and 'S2', respectively. The following lines must be modified, if the user's deposit contains a different number:

MAIN1 : LINES 004, 005, 021-022, 036-038, 047, 071-072, 091-092, 124-126, 142 and 184;
MAIN2 : LINES 004, 005, 016-017, 040, 049-050, 090-092, 108 and 117;
MAIN3 : LINES 004, 005, 016-017, 046-047, 049, 052, 091-093, 109 and 118.

```
001       DOUBLE PRECISION CAP(3,3,3),CONCAP(3),REFCAP(3),
         1MINCAP,ORERES(99,20),F(99),MIN(99),Q6(20),
         2Q7(65,20),Q8(65,20),Q9(65,20),Q10(65,20),
```

```
      3Q11(65,20),Q12(3,3,3),Q14(99),POLD,PNEW,DABS,FIXC,
      4Q14A(99)
002    REAL MILC(2),INFRC,INFRP,MINC
003    INTEGER CUTOFF(99,,ONE/1/,TWO/2/,GI
004    INTEGER*2 TOPO(13500),YR(13500),PITINC,SURFC(30),
      1Q33(13500),Q34(13500),Q35(13500),T,SCORK(750),
      2IGRAD1(30,25,18),IGRAD2(30,25,18),EQGRAD(30,25,18),
      3AVEG
005    DIMENSION OB(2),OF(2),ORCI(2),ORCI(2),REFC(2),
      1PITANG(2),WRKANG(2),SLOPE(2),WSLOPE(2),S1(99),
      2S2(99),TONF(2),REC(3),E(99),C(99),R(99),PROFIT(99),
      3CAPOB(99),WASTE(99) VAL(13500)
006    COMMON ORERES,CONCAP,REFCAP,MINCAP,F,Q6,Q7,Q8,Q9,
      1Q10,Q11,Q12,MIN,Q14,Q14A,CAP,PITANG,WRKANG,SLOPE,
      2WSLOPE,S1,D,C,R,E,TONF,PROFIT, CAPOB,WASTE,REC,
      3VAL,GS,NI,NJ,NK,NBLKS,MAXN,MAXG,CUTOFF,L,Q33,Q34,
      4Q35,SURFC,TOPO.YR
```

----------COMMENT----------

ii = NK*NJ*NK; jj = 4*(number of recoverable minerals)*

```
007    DEFINE FILE 10( ii   , jj ,L,IJ)
```

----------COMMENT----------

The scale factor 'GS' and grade increment 'GI' (Line 011) are given by the user to set up the subscript 'N' of 'ORERES' in lines 037, 076 and 177 as explained in Step 4, Section 8•2•

```
008      READ (5,15) NI,NJ,NK,GS
009   15 FORMAT (3I3,F7•1)
010      NBLKS=NI*NJ*NK
011      GI=5
012      READ (5,1051) (SCORK(N),N=1,750)
013 1051 FORMAT (30I2)
014      DO 210 LOC=1,NBLKS
015      TOPO(LOC)=0•0
016      VAL(LOC)=0•0
017  210 YR(LOC)=0
018      DO 260 K=1,NK
019      DO 260 J=1,NJ
020      DO 260 I=1,NI
021      IGRAD1(I,J,K)=0
022      IGRAD2(I,J,K)=0
```

```
023   260 EQGRAD(I,J,K)=0
024       DO 1054 K=1,NK
025       N=0
026       DO 1054 J=1,NJ
027       DO 1054 I=1,NI
028       N=N+ONE
029       LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
030       FIND (10'LOC)
031       IF (SCORK(N)-K) 1055,1053,1053
032  1055 II=I
033       GO TO 1056
034  1053 II=0
035  1056 TOPO(LOC)=II
036       READ (10'LOC) GRAD1,GRAD2
037       IGRAD1(I,J,K)=GRAD1*GS
038       IGRAD2(I,J,K)=GRAD2*GS
039  1054 CONTINUE
040       READ (5,60) (((CAP(I,J,K),I=1,3),J=1,3),K=1,3)
041    60 FORMAT (3F12.1)
042       READ (5,61) (OB(I),OR(I),OBCI(I),ORCI(I),MILC(I),
         1REFC(I),PITANG(I),WRKANG(I),I=1,2)
043    61 FORMAT (5F6.3,F7.3,2F7.4)
044       DO 63 I=1,2
045       SLOPE(I)=(SIN(PITANG(I))/COS(PITANG(I)))**2.
046    63 WSLOPE(I)=(SIN(WRKANG(I))/COS(WRKANG(I)))**2.
047       READ (5,1) MAXN,S1(1),S2(1),D,INFRC,INFRP,TONF(1),
         1TONF(2)
048     1 FORMAT (I2,2F8.2,F5.2,2F6.3,2F9.1)
049       READ (5,2) (CONCAP(I),I=1,3),(REFCAP(J),J=1,3),
         1(REC(K),K=1,3)
050     2 FORMAT (3F11.1/3F9.1/3F5.2)
051       READ (5,3) MINCAP
052     3 FORMAT (F11.1)
053       READ (5,12) MINC,CONC,FIXC
054    12 FORMAT (2F5.2,F10.1)
055       DO 4 NN=1,20
056       DO 4 M=1,99
057     4 ORDRES(M,NN)=0.0
058       MAXG=0
```

----------COMMENT----------

Among other things, DO loop 11 executes the initial assumption for 'ORDRES', as explained in Subsection 6.3.1. and Step 5, Section 8.2.

```
059       DO 11 N=1,MAXN
060       K1=4*N
061       K2=K1-TWO
062    73 DO 11 K=K2,K1
063       DO 11 J=1,NJ
064       DO 11 I=1,NI
0065       LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
066       IF (TOPO(LOC)) 11,11,5
```

----------COMMENT----------

The example deposit has two tonnage factors 'TONF' one for each side of the tenth grid-level. The following four statements select the appropriate tonnage factor for the 'ORERES' computations. These statements are modified as explained in Subsection 6.3.1, if the user's deposit requires different tonnage factors. Additionally, the modification must be repeated each time tonnage is involved in a computation.

(SEE: MAIN1: LINES 110, 120 & 173;
MAIN2: LINES 076, 086 & 144;
MAIN3: LINES 077, 087 & 145;
PD: LINES 118 & 143; AND
SUBPD: LINE 050.)

```
067     5 IF (K-10) 6,6,7
068     6 MM=ONE
069       GO TO 8
070     7 MM=2
```

----------COMMENT----------

The following two statements must be modified to calculate an equivalent mineral grade for each secondary mineral reported in the user's deposit, as explained in Subsection 6.3.2.

```
071     8 X=(S2(1)/S1(1))*IGRAD2(I,J,K)
072       EQGRAD(I,J,K)=IGRAD1(I,J,K)+IFIX(X)
073       IF (EQGRAD(I,J,K)-GI) 9,9,10
074     9 M=ONE
075       GO TO 70
076    10 M=EQGRAD(I,J,K)/GI
077    31 IF (MAXG-M) 80,80,70
078    80 MAXG=M
```

```
079    70 ORERES(M,N)=ORERES(M,N)+TONF(MM)
080    11 CONTINUE
081       WRITE (6,220) MAXG
082   220 FORMAT (1X,' MAXG=',I4)
083       DO 13 N=1,99
084       T=N
085       X=(1.+INFRC)**T
086       Y=(1.+INFRP)**T
087       E(N)=X*MINC
088       C(N)=Y*CONC
089       R(N)=X*REFC(1)
090       F(N)=Y*FIXC
091       S2(N)=Y*S2(1)
092       S1(N)=Y*S1(1)
093    13 CONTINUE
094       CALL COG (AVEG,YIELD)
095       T=ONE
096       LC=0
097       MC=0
098       SUME=0
099       SUMC=0
100       SUMR=0
----------COMMENT----------

DO loop 34 computes the T-5 net cash value distribution.
This routine, and its duplicates in MAIN1 and MAIN2 must be
tailored to the user's desired cost structure, as explained
in Subsection 6.5.2.

----------

101       DO 34 K=1,NK
102       DO 34 J=1,NJ
103       DO 34 I=1,NI
104       LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
105       IF (TOPO(LOC)) 34,34,14
106    14 M=ONE
107    16 IF (EQGRAD(I,J,K)-AVEG) 19,23,23
108    19 LC=LC+ONE
109       REV=0.0
110       IF (K-10) 20,20,21
111    20 N=ONE
112       GO TO 22
113    21 N=TWO
114    22 X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**N)
115       SUME=SUME+X
116       COST=(X+(F(M)/MIN(N)))*TONF(N)
```

```
      GO TO 33
   23 LC=LC+ONE
      MC=MC+ONE
      IF (K-10) 24,24,25
   24 N=ONE
      GO TO 26
   25 N=TWO
   26 I1=IGRAD1(I,J,K)
      I2=IGRAD2(I,J,K)
      REV=((FLOAT(I1)/GS*S1(M))+(FLOAT(I2)/GS*S2(M)))*
     1TONF(N)*YIELD
      X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**M)
      Y=(OR(N)+(K*ORCI(N)))*((1.+INFRC)**M)
      SUME=SUME+X
      IF (IGRAD2(I,J,K)) 27,27,28
   27 NN=ONE
      GO TO 29
   28 NN=TWO
   29 Z=(MILC(NN))*((1.+INFRC)**M)
      SUMC=SUMC+Y-X+Z
      IF (ABS(((66.3*I)-(277.5*J)-(37.525*K)+3274.075)
     1/288)-3.0) 30,30,31
   30 NN=ONE
      GO TO 32
   31 NN=TWO
   32 W=(REIC(NN))*((1.+INFRC)**M)
      SUMR=SUMR+W
      COST=(Y+Z+(W*YIELD*((FLOAT(I1+I2))/GS))+(F(M)/
     1MIN(M)))*TONF(N)
   33 VAL(LOC)=REV-COST
   34 CONTINUE
      IF (SUME) 36,36,35
   35 E(T)=SUME/LC
   36 IF (SUMC) 38,38,37
   37 C(T)=SUMC/MC
      R(T)=SUMR/MC
   38 CONTINUE
      CALL DD
      PNEW=0.0
      DO 39 N=1,L
   39 PNEW=PROFIT(N)+PNEW
      WRITE ( 6,222) PNEW
  222 FORMAT (1X,' TOTAL REVENUE=',F14.1)
      MAXN=(L-ONE)/5+ONE
      WRITE (6,221) MAXN,L
  221 FORMAT (1X,' MAXN=',I4,'L=',I4)
```

```
      DO 41 M=1,99
      DO 41 N=1,20
   41 ORERES(M,N)=0.0
      DO 48 K=1,NK
      DO 48 J=1,NJ
      DO 48 I=1,NI
      LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
      IF (TOPO(LOC)) 48,48,42
   42 IF (YR(LOC)) 48,48,51
   51 PITINC=(YR(LOC)-ONE)/5+ONE
  231 IF (K-10) 43,43,44
   43 MM=ONE
      GO TO 45
   44 MM=TWO
   45 IF (EQGRAD(I,J,K)-GI) 46,46,71
   46 M=ONE
      GO TO 47
   71 M=EQGRAD(I,J,K)/GI
   47 ORERES(M,PITINC)=ORERES(M,PITINC)+TONF(MM)
   48 CONTINUE
      WRITE (11) (TOPO(N),YR(N),N=1,NBLKS)
      END FILE 11
      WRITE (12) ((ORERES(M,N),M=1,MAXG),N=1,MAXN)
      END FILE 12
      WRITE (13) (((EQGRAD(I,J,K),IGRAD1(I,J,K),
     1IGRAD2(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
      END FILE 13
   50 WRITE (6,201)
  201 FORMAT (' ANNUAL PIT OUTLINE')
      DO 211 N=1,NJ
  211 SURFC(N)=0
      DO 100 N=1,L
      WRITE (6,202) N
  202 FORMAT ('1',' TOPOGRAPHIC MAP OF MINE IN YEAR',I3)
      DO 100 I=1,NI
      DO 104 J=1,NJ
      M=0
      K=ONE
  151 IF (K-NK) 154,154,175
  154 LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
      IF (TOPO(LOC)) 152,152,105
  105 IF (YR(LOC)) 175,175,190
  190 IF (YR(LOC)-N) 152,152,175
  152 M=K
      K=K+ONE
      GO TO 151
```

```
205   175 SURFC(J)=M
206   104 CONTINUE
207*      WRITE (6,203) (SURFC(LEVEL),LEVEL=1,NJ)
208   100 CONTINUE
209   203 FORMAT ('0',3013)
210    49 STOP
211       DEBUG TRACE,SUBTRACE,SUBCHK
212       END
```

8. 3. 3 A LIST OF MAIN2

The comments within the list describe how MAIN2 differs from MAIN1.

```
001       DOUBLE PRECISION CAP(3,3,3),CONCAP(3),REFCAP(3),
         1MINCAP,OPERES(99,20),F(99),MIN(99),Q6(20),
         2Q7(65,20),Q8(65,20),Q9(65,20),Q10(65,20),
         3Q11(65,20),Q12(3,3,3),Q14(99),POLD,PNEW,OABS,FIXC,
         4Q14A(99)
002       REAL MILC(2),INFRC,INFRP,MINC
003       INTEGER CUTOFF(99),ONE/1/,TWO/2/,GI
004       INTEGER*2 TOPO(13500),YR(13500),PITINC,SURFC(30),
         1Q33(13500),Q34(13500),Q35(13500),P,SCORK(750),
         2IGRAD1(30,25,18),IGRAD2(30,25,18),EQGRAD(30,25,18),
         3AVEG
005       DIMENSION OB(2),OR(2),OBCI(2),ORCI(2),REFC(2),
         1PITANG(2),WRKANG(2),SLOPE(2),WSLOPE(2),S1(99),
         2S2(99),TONP(2),REC(3),E(99),C(99),R(99),PROFIT(99),
         3CAPOB(99),WASTE(99),VAL(13500)
006       COMMON OPERES,CONCAP,REFCAP,MINCAP,F,Q6,Q7,Q8,Q9,
         1Q10,Q11,Q12,MIN,Q14,Q14A,CAP,PITANG,WRKANG,SLOPE,
         2WSLOPE,S1,D,C,R,E,TONP,PROFIT, CAPOB,WASTE,REC,
         3VAL,G5,NI,NJ,NK,NBLKS,MAXN,MAXG,CUTOFF,L,Q33,Q34,
         4Q35,SURFC,TOPO,YR
```

----------COMMENT----------

The input data of line 007 are required from the results of the first module MP1:

A) the maximum equivalent mineral grade 'MAXG' reported in the user's deposit by MAIN1 and

B) the mine's life span 'L' reported by the subroutine COG1.

```
007         READ (5,62) MAXG,L
008      62 FORMAT (2I3)
009 :       READ (5,4) NI,NJ,NK,GS
010         NBLKS=NI*NJ*NK
011       4 FORMAT (3I3,F7.1)
012         GI=5
013         DO 260 K=1,NK
014         DO 260 J=1,NJ
015         DO 260 I=1,NI
016         IGRAD1(I,J,K)=0
017         IGRAD2(I,J,K)=0
018      260 EQGRAD(I,J,K)=0
019         DO 210 LOC=1,NBLKS
020         VAL(LOC)=0.0
021         YR(LOC)=0
022      210 CONTINUE
023         READ (5,60) (((CAP(I,J,K),I=1,3),J=1,3),K=1,3)
024       60 FORMAT (3F12.1)
02          READ (5,61) (OB(I),OR(I),OBCI(I),ORCI(I),MILC(I),
           1REFC(I),PITANG(I),WPKANG(I),I=1,2)
026       61 FORMAT (5F6.3,F7.3,2F7.4)
027         DO 63 I=1,2
028         SLOPE(I)=(SIN(PITANG(I))/COS(PITANG(I)))**2.
029       63 WSLOPE(I)=(SIN(WPKANG(I))/COS(WPKANG(I)))**2.
```

----------COMMENT----------

The total number of pit increments 'MAXN' reported by the preceding iteration cycle is required to modify the input of the following 'READ' statement.

```
030         READ (5,1) MAXN,S1(1),S2(1),D,INFRC,INFRP,TONF(1),
           1TONF(2)
031       1 FORMAT (I2,2F8.2,F5.2,2F6.3,2F9.1)
032         READ (5,2) (CONCAP(I),I=1,3),(REFCAP(J),J=1,3),
           1REC(K),K=1,3)
033       2 FORMAT (3F11.1/3F9.1/3F5.2)
034         READ (5,3) MINCAP
035       3 FORMAT (F11.1)
036         READ (5,12) MINC,CONC,FIXC
037      12 FORMAT (2F5.2,F10.1)
```

----------COMMENT----------

The two data sets 'YR' and 'ORERES' are used to update the iterative solution. The data sets 'EQGRAD', 'IGRAD1', 'IGRAD2' and 'TOPO' are read form the disk pack where they were

stored rather than being recalculated.

```
038       READ (11) (TOPO(N),YR(N),N=1,NBLKS)
039       READ (12) ((ORESES(M,N),M=1,MAXG),N=1,MAXN)
040       READ (13) (((EQGRAD(I,J,K),IGRAD1(I,J,K),
         1IGRAD2(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
041       DO 13 N=1,99
042       T=N
043       X=(1.+INFRC)**T
044       Y=(1.+INFRP)**T
045       E(N)=X*MINC
046       C(N)=X*CONC
047       R(N)=X*REFC(1)
048       F(N)=Y*FIXC
049       S2(N)=Y*S2(1)
050       S1(N)=Y*S1(1)
051    13 CONTINUE
052       CALL COG (AVEG,YIELD)
```

COMMENT

DO loop 38 has been added to this module to make the first estimate of the time dimension for the T-S net cash value distribution 'VAL' and the average unit production costs 'E', 'C' and 'R'.

```
053    92 DO 38 T=1,L
054       LC=0
055       MC=0
056       SUME=0
057       SUMC=0
058       SUMR=0
059       DO 34 K=1,NK
060       DO 34 J=1,NJ
061       DO 34 I=1,NI
062       LCC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
063       IF (TOPO(LCC)) 34,34,14
```

COMMENT

The time dimension 'T' and grid-block excavation time 'YR' were added to DO loop 34 with the statements in Lines 064, 066 and 068 - 073. Note that the time is assumed to be the first year for any grid-blocks whose pit production has not been simulated in the first module 'MP1'. This assumption eventually becomes redundant as the solution converges.

```
064    14 IF (YR(LOC)) 15,15,17
065    15 M=ONE
066       IF (T-M) 16,16,34
067    16 IF (EQGRAD(I,J,K)-AVEG) 19,23,23
068    17 IF (YR(LOC)-L) 18,18,261
069   261 M=ONE
070       GO TO 16
071    18 IF (T-YR(LOC)) 34,262,34
072   262 M=T
073       IF (EQGRAD(I,J,K)-CUTOFF(T)) 19,23,23
074    19 LC=LC+ONE
075       REV=0.0
076       IF (K-10) 20,20,21
077    20 N=ONE
078       GO TO 22
079    21 N=TWO
080    22 X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**K)
081       SUME=SUME+X
082       COST=(X+(F(M)/MIN(M)))*TONF(N)
083       GO TO 33
084    23 LC=LC+ONE
085       MC=MC+ONE
086       IF (K-10) 24,24,25
087    24 N=ONE
088       GO TO 26
089    25 N=TWO
090    26 I1=IGRAD1(I,J,K)
091       I2=IGRAD2(I,J,K)
092       REV=((FLOAT(I1)/G5*S1(M))+(FLOAT(I2)/G5*S2(M)))*
         1TONF(N)*YIELD
093       X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**K)
094       Y=(OR(N)+(K*ORCI(N)))*((1.+INFRC)**K)
095       SUME=SUME+X
096       IF (IGRAD2(I,J,K)) 27,27,28
097    27 NN=ONE
098       GO TO 29
099    28 NN=TWO
100    29 Z=(MILC(NN))*((1.+INFRC)**K)
101       SUMC=SUMC+Y-X+Z
102       IF (ABS(((66.3*I)-(277.5*J)-(37.625*K)+3274.075)
         1/288)-3.0) 30,30,31
         1-3.0) 30,30,31
103    30 NN=ONE
104       GO TO 32
105    31 NN=TWO
```

```
106    32 W=(REFC(NN))*((1.+INFRC)**M)
107       SUMR=SUMR+W
108       COST=(Y+Z+(W*YIELD*((FLOAT(I1+I2))/GS))+(F(M)/
         1MIN(M)))*TONF(N)
109    33 VAL(LOC)=REV-COST
110    34 CONTINUE
111       IF (SUME) 36,36,35
112    35 E(T)=SUME/LC
113    36 IF (SUMC) 36,38,37
114    37 C(T)=SUMC/NC
115       R(T)=SUMR/NC
116    38 CONTINUE
```

----------COMMENT----------

The re-evaluated unit production costs are stored on a disk pack by the following statement to replace previous estimates.

```
117       WRITE (16) (E(T),C(T),R(T),S2(T),S1(T),F(T),T=1,L)
118       END FILE 16
119       WRITE (6,240)
120   240 FORMAT ('1',' AVERAGE UNIT PRODUCTION COSTS')
121       WRITE (6,241)
122   241 FORMAT ('0','  E',1X,'  C',2X,'  R',3X,'REF')
123       WRITE (6,242) (T,E(T),C(T),R(T),T=1,L)
124   242 FORMAT (I3,2F5.2,F7.3)
125       CALL PD
126       PNEW=0.0
127       DO 39 N=1,L
128    39 PNEW=PROFIT(N)+PNEW
129       WRITE ( 6,222) PNEW
130       MAXN=(L-ONE)/5+ONE
131   222 FORMAT (1X,' TOTAL REVENUE=',F14.1)
132       WRITE (6,221) MAXN,L
133   221 FORMAT (1X,' MAXN=',I4,'L=',I4)
134       DO 41 M=1,99
135       DO 41 N=1,20
136    41 OREPES(M,N)=0.0
137       DO 48 K=1,NK
138       DO 48 J=1,NJ
139       DO 48 I=1,NI
140       LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
141       IF (TOPO(LOC)) 48,48,42
142    42 IF (YR(LOC)) 48,48,51
143    51 PITINC=(YR(LOC)-ONE)/5+ONE
```

```
144   231 IF (K-10) 43,43,44
145    43 MM=ONE
146       GO TO 45
147    44 MM=TWO
148    45 IF (EQGRAD(I,J,K)-GI) 46,46,71
149    46 M=ONE
150       GO TO 47
151    71 M=EQGRAD(I,J,K)/GI
152    47 ORERES(M,PITINC)=ORERES(M,PITINC)+TONT(MM)
153    48 CONTINUE
```

----------COMMENT----------

Additional external data sets are created by the following four statements to store the updated variables 'YR' and 'ORERES' for the next iteration cycle. The T-S net cash value distribution 'VAL' is stored for informational purposes only. The 'TOPO' data set is stored rather than being recalculated for each module.

```
154       WRITE (14) (TOPO(N),YR(N),VAL(N),N=1,NBLKS)
155       END FILE 14
156       WRITE (15) ((ORERES(M,N),M=1,MAXG),N=1,MAXN)
157       END FILE 15
158    50 WRITE (6,201)
159   201 FORMAT (' ANNUAL PIT OUTLINE')
160       DO 211 N=1,NJ
161   211 SURFC(N)=0
162       DO 100 N=1,L
163       WRITE (6,202) N
164   202 FORMAT ('1',' TOPOGRAPHIC MAP OF MINE IN YEAR',I3)
165       DO 100 I=1,NI
166       DO 104 J=1,NJ
167       M=0
168       K=ONE
169   151 IF (K-MK) 154,154,175
170   154 LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
171       IF (TOPO(LOC)) 152,152,105
172   105 IF (YR(LOC)) 175,175,190
173   190 IF (YR(LOC)-N) 152,152,175
174   152 M=K
175       K=K+ONE
176       GO TO 151
177   175 SURFC(J)=M
178   104 CONTINUE
179       WRITE (6,203) (SURFC(LEVEL),LEVEL=1,NJ)
```

```
180   100 CONTINUE
181   203 FORMAT ('0',3013)
182.   49 STOP
183       DEBUG SUBTRACE,SUBCHK
184       END
```

8. 3. 4 A LIST OF MAIN 3

After the first two iteration cycles, only one module 'MP3' is used to compute the remaining cycles of the iterative solution. The comments listed in this subsection describe how MAIN2 was modified to form MAIN3 for these computations:

```
001       DOUBLE PRECISION CAP(3,3,3),CONCAP(3),REFCAP(3),
         1MINCAP,ORERES(99,20),F(99),MIN(99),Q6(20),
         2Q7(65,20),Q8(65,20),Q9(65,20),Q10(65,20),
         3Q11(65,20),Q12(3,3,3),Q14(99),POLD,PNEW,DABS,FIXC,
         4Q14A(99)
002       REAL MILC(2),INFRC,INFRP,MINC
003       INTEGER CUTOFF(99),ONE/1/,TWO/2/,GI
004       INTEGER*2 TOPO(13500),YR(13500),PITINC,SURFC(30),
         1Q33(13500),Q34(13500),Q35(13500),T,SCORK(750),
         2IGRAD1(30,25,18),IGRAD2(30,25,18),EQGRAD(30,25,18),
         3AVEG
005       DIMENSION OB(2),OR(2),OBCI(2),ORCI(2),REFC(2),
         1PITANG(2),WRKANG(2),SLOPE(2),WSLOPE(2),S1(99),
         2S2(99),TONF(2),REC(3),E(99),C(99),R(99),PROFIT(99),
         3CAPOB(99),WASTE(99),VAL(13500)
006       COMMON ORERES,CONCAP,REFCAP,MINCAP,F,Q6,Q7,Q8,Q9,
         1Q10,Q11,Q12,MIN,Q14,Q14A,CAP,PITANG,WRKANG,SLOPE,
         2WSLOPE,S1,D,C,R,E,TONF,PROFIT, CAPOB,WASTE,REC,
         3VAL,GS,NI,NJ,NK,NBLKS,MAXN,MAXG,CUTOFF,L,Q33,Q34,
         4Q35,SURFC,TOPO,YR
-------------------------COMMENT-------------------------
```

The mine's life span 'L' reported by the results of the preceding iteration cycle's subroutine COG2 is required to modify the input of the following statement.

```
-----------------------------------------------------------
007       READ (5,62) MAXG,L
008    62 FORMAT (2I3)
009       READ (5,4) NI,NJ,NK,GS
010       NBLKS=NI*NJ*NK
011     4 FORMAT (3I3,F7.1)
012       GI=5
```

```
013         DO 260 K=1,NK
014         DO 260 J=1,NJ
015         DO 260 I=1,NI
016         IGRAD1(I,J,K)=0
017         IGRAD2(I,J,K)=0
018   260   EQGRAD(I,J,K)=0
019         DO 210 LOC=1,NBLKS
020         VAL(LOC)=0.0
021         YR(LOC)=0
022   210   CONTINUE
023         READ (5,60) (((CAP(I,J,K),I=1,3),J=1,3),K=1,3)
024    60   FORMAT (3F12.1)
025         READ (5,61) (OB(I),OR(I),OBCI(I),ORCI(I),MILC(I),
           1REFC(I),PITANG(I),WRKANG(I),I=1,2)
026    61   FORMAT (5F6.3,F7.3,2F7.4)
027         DO 63 I=1,2
028         SLOPE(I)=(SIN(PITANG(I))/COS(PITANG(I)))**2.
029    63   WSLOPE(I)=(SIN(WRKANG(I))/COS(WRKANG(I)))**2.
```

----------COMMENT----------

The total number of pit increments 'MAXN' reported by the preceding iteration cycle is required to modify the input of the following 'READ' statement.

```
030         READ (5,1) MAXN,S1(1),S2(1),D,INFRC,INFPP,TONF(1),
           1TONF(2)
031      1  FORMAT (I2,2F8.2,F5.2,2F6.3,2F9.1)
032         READ (5,2) (CONCAP(I),I=1,3),(REFCAP(J),J=1,3),
           1REC(K),K=1,3)
033      2  FORMAT (3F11.1/3F9.1/3F5.2)
034         READ (5,3) MINCAP
035      3  FORMAT (F11.1)
036         READ (5,12) MINC,CONC,FIXC
037     12  FORMAT (2F5.2,F10.1)
```

----------COMMENT----------

The unit production costs and selling prices estimated by DO loop 13 are replaced by overwriting them with the more accurate ones read in from the preceding iteration cycle (Line 052). However, the values read in from data set 16 have only been computed for the mine's life span 'L' of the preceding iteration cycle. Therefore, DO loop thirteen is required to estimate these costs for any years exceeding the previous cycle's mine life 'L', if the mine life of the current cycle exceeds that of the previous cycle. This

estimate becomes redundant as the solution converges.

```
038         DO 13 N=1,99
039         T=N
040         X=(1.+INFRC)**T
041         Y=(1.+INFRP)**T
042         E(N)=X*MINC
043         C(N)=Y*CONC
044         R(N)=X*REFC(1)
045         F(N)=X*FIXC
046         S2(N)=Y*S2(1)
047         S1(N)=Y*S1(1)
048      13 CONTINUE
```

COMMENT

The following external data sets (13, 14, 15, & 16) are read into internal storage. The unit production costs and prices, excavation times and the T-S mineral distribution from the preceding iteration cycle are used to update the solution. The remaining data sets are constant for all iteration cycles.

```
049         READ (13) (((EQGRAD(I,J,K),IGRAD1(I,J,K),
           1IGRAD2(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
050         READ (14) (TOPO(N),YR(N),VAL(N),N=1,NBLKS)
051         READ (15) ((ORDRES(M,N),M=1,MAXG),N=1,MAXN)
052         READ (16) (E(T),C(T),R(T),S2(T),S1(T),F(T),T=1,L)
053         CALL COG (AVEG,YIELD)
054      92 DO 38 T=1,L
055         LC=0
056         MC=0
057         SUME=0
058         SUMC=0
059         SUMR=0
060         DO 34 K=1,NK
061         DO 34 J=1,NJ
062         DO 34 I=1,NI
063         LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
064         IF (TOPO(LOC)) 34,34,14
065      14 IF (YR(LOC)) 15,15,17
066      15 M=ONE
067         IF (T-M) 16,16,34
068      16 IF (EQGRAD(I,J,K)-AVEG) 19,23,23
069      17 IF (YR(LOC)-L) 18,18,261
```

```
070   261 M=ONE
071       GO TO 16
072    18 IF (T-YR(LOC)) 34,262,34
073   262 M=T
074       IF (EOGRAD(I,J,K)-CUTOFF(T)) 19,23,23
075    19 LC=LC+ONE
076       REV=0.0
077       IF (K-10) 20,20,21
078    20 N=ONE
079       GO TO 22
080    21 N=TWO
081    22 X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**M)
082       SUME=SUME+X
083       COST=(X+(F(M)/MIN(M)))*TONF(N)
084       GO TO 33
085    23 LC=LC+ONE
086       MC=MC+ONE
087       IF (K-10) 24,24,25
088    24 N=ONE
089       GO TO 26
090    25 N=TWO
091    26 I1=IGRAD1(I,J,K)
092       I2=IGRAD2(I,J,K)
093       REV=((FLOAT(I1)/GS*S1(N))+(FLOAT(I2)/GS*S2(N)))*
          1TONF(N)*YIELD
094       X=(OB(N)+(K*OBCI(N)))*((1.+INFRC)**X)
095       Y=(OR(N)+(K*ORCI(N)))*((1.+INFRC)**M)
096       SUMB=SUMB+X
097       IF (IGRAD2(I,J,K)) 27,27,28
098    27 NN=ONE
099       GO TO 29
100    28 NN=TWO
101    29 Z=(MILC(NN))*((1.+INFRC)**M)
102       SUMC=SUMC+Y-X+Z
103       IF (ABS(((66.3*I)-(277.5*J)-(37.625*K)+3274.075)
          1/288)-3.0) 30,30,31
          1-3.0) 30,30,31
104    30 NN=ONE
105       GO TO 32
106    31 NN=TWO
107    32 W=(PRPC(NN))*((1.+INFRC)**M)
108       SUMR=SUMR+W
109       COST=(Y+Z+(W*YIELD*((FLOAT(I1+I2))/GS))*(F(M)/
          1MIN(M)))*TONF(N)
110    33 VAL(LOC)=REV-COST
111    34 CONTINUE
```

```
      IF (SUMB) 36,36,35
   35 B(T)=SUMB/NC
   36 IF (SUMC) 38,38,37
   37 C(T)=SUMC/NC
      R(T)=SUMR/NC
   38 CONTINUE
```

----------COMMENT----------

The external data sets--'YR' only of 14, 15 and 16 (Lines 118 and 155 - 158)--used to update the solution are the same for all iteration cycles from the third on. Therefore, the obsolete data is lost when the current updating information is over-written on the same data sets. The 'VAL' data set is updated for informational purposes only. The 'TOPO' data set is stored rather than being recalculated for each iteration cycle.

```
      WRITE (16) (B(T),C(T),R(T),S2(T),S1(T),F(T),T=1,L)
      END FILE 16
      WRITE (6,240)
  240 FORMAT ('1',' AVERAGE UNIT PRODUCTION COSTS')
      WRITE (6,241)
  241 FORMAT ('0','  B',1X,'  C',2X,'  R',3X,'REF')
      WRITE (6,242) (T,B(T),C(T),R(T),T=1,L)
  242 FORMAT (I3,2F5.2,F7.3)
      CALL PD
      PNEW=0.0
      DO 39 N=1,L
   39 PNEW=PROFIT(N)+PNEW
      WRITE ( 6,222) PNEW
      MAXN=(L-ONE)/5+ONE
  222 FORMAT (1X,' TOTAL REVENUE=',F14.1)
      WRITE (6,221) MAXN,L
  221 FORMAT (1X,' MAXN=',I4,'L=',I4)
      DO 41 N=1,99
      DO 41 K=1,20
   41 ORDRES(N,K)=0.0
      DO 48 K=1,NK
      DO 48 J=1,NJ
      DO 48 I=1,NI
      LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
      IF (TOPO(LOC)) 48,48,42
   42 IF (YR(LOC)) 48,48,51
   51 PITINC=(YR(LOC)-ONE)/5+ONE
  231 IF (K-10) 43,43,44
```

```
146    43 MM=ONE
147       GO TO 45
148    44 MM=TWO
149    45 IF (EQGRAD(I,J,K)-GI) 46,46,71
150    46 M=ONE
151       GO TO 47
152    71 M=EQGRAD(I,J,K)/GI
153    47 ORERES(M,PITINC)=ORERES(M,PITINC)+TONF(MM)
154    48 CONTINUE
155       WRITE (14) (TOPO(N),YR(N),VAL(N),N=1,NBLKS)
156       END FILE 14
157       WRITE (15) ((ORERES(M,N),M=1,MAXG),N=1,MAXN)
158       END FILE 15
159    50 WRITE (6,201)
160   201 FORMAT (' ANNUAL PIT OUTLINE')
161       DO 211 N=1,NJ
162   211 SURFC(N)=0
163       DO 100 N=1,L
164       WRITE (6,202) N
165   202 FORMAT ('1',' TOPOGRAPHIC MAP OF MINE IN YEAR',I3)
166       DO 100 I=1,NI
167       DO 104 J=1,NJ
168       M=0
169       K=ONE
170   151 IF (K-NK) 154,154,175
171   154 LOC=NJ*NI*(K-ONE)+NI*(J-ONE)+I
172       IF (TOPO(LOC)) 152,152,105
173   105 IF (YR(LOC)) 175,175,190
174   190 IF (YR(LOC)-N) 152,152,175
175   152 M=K
176       K=K+ONE
177       GO TO 151
178   175 SURFC(J)=M
179   104 CONTINUE
180       WRITE (6,203) (SURFC(LEVEL),LEVEL=1,NJ)
181   100 CONTINUE
182   203 FORMAT ('0',3013)
183    49 STOP
184       DEBUG SUBTRACE,SUBCHK
185       END
```

8. 4 SUBROUTINES COG1 AND COG 2

A verbal description of subroutine COG1's computational sequence follows in Subsection 8.4.1. Then, it is listed in the next Subsection. The last Subsection contains a list of COG2 with comments to explain how it differs from COG1.

8. 4. 1 COMPUTATIONAL SEQUENCE OF COG1

LINES 001 - 008

Specification and input statements.

LINES 009 - 052

Ratios of ore : pit material 'R1', refined product : pit material 'R2' and refined product : ore 'R3' are calculated for each grade increment 'M' and pit increment 'N' of 'ORERES'.

LINES 053 - 056

Subroutine 'COGSUB' is called to compute the following information for the combination of plant capacities (MINCAP, CONCAP, REFCAP, REC) named by (I,J,K):

A) the annual cut-off grades 'CUTOFF';

B) the annual production targets 'MIN', 'CON' and 'QREF'; and

C) the overall recovery efficiency 'YIELD'.

It is not necessary to compute the plant combination yielding the maximum present value for the rough computations of the first iteration cycle. Therefore, any plant combination (I,J,K) is selected. The plant combination yielding the maximum present value is selected later by replacing this subroutine with subroutine 'COG2' in subsequent iteration cycles.

LINES 057 - 067

These statements execute printer output used to monitor the

solution as described and illustrated in Section 7•2•

LINES 070 - 073

An average 'AVEG' of the annual cut-off grades 'CUTOFF' is computed•

8• 4• 2 A LIST OF COG1

```
001       SUBROUTINE COG (AVEG,YIELD)
002       DOUBLE PRECISION MAT(20),ORERES(99,20),
         WASTE(65,20),ORE(65,20),REF(20),DFLOAT,PROD(65,20),
         2R1(65,20),R2(65,20),R3(65,20),PVMAX,PV(3,3,3),
         3CAP(3,3,3),Q2(3),Q3(3),Q4,Q5(99),MIN(99),CON(99),
         4QREF(99),P(99)
003       INTEGER CUTOFF(99),ONE/1/,SUM1,SUM,GI
004       INTEGER*2 AVEG,Q33(13500),Q34(13500),Q35(13500),
         1Q37(30),Q38(13500),Q40(13500)
005       DIMENSION REC(3),Q16(2),Q17(2),Q18(2),Q19(2),
         1Q20(99),Q22(99),Q23(99),Q24(99),Q25(2),Q26(99),
         2Q27(99),Q28(99),Q29A(13500)
006       COMMON ORERES,Q2,Q3,Q4,Q5,MAT,ORE,PROD,R1,R2,R3,PV,
         1MIN,CON,QREF,CAP,Q16,Q17,Q18,Q19,Q20,Q21,Q22,Q23,
         2Q24,Q25,Q26,Q27,Q28,REC,Q29A,GS,NI,NJ,NK,NBLKS,
         3MAXN,MAXG,CUTOFF,L,Q33,Q34,Q35,Q37,Q38,Q40
007       L=0
```

----------COMMENT----------

'GI' is given by the user and is equal to 'GI' in Line 011 of 'MAIN1'•

```
008       GI=5
009       DO 351 N=1,20
010       MAT(N)=0
011       REF(N)=0•0
012       DO 351 M=1,65
013       PROD(M,N)=0•0
014       WASTE(M,N)=0•0
015       ORE(M,N)=0•0
016       R1(M,N)=0•0
017       R2(M,N)=0•0
018       R3(M,N)=0•0
019   351 CONTINUE
020       DO 308 N=1,MAXN
021       DO 308 M=1,MAXG
```

```
022   308 MAT(N)=MAT(N)+OREBES(M,N)
023       DO 309 N=1,MAXN
024       WASTE(1,N)=OREBES(1,N)
025       IF (MAT(N)-WASTE(1,N)) 2,2,3
026     2 ORE(1,N)=OREBES(1,N)
027       GO TO 4
028     3 ORE(1,N)=MAT(N)-WASTE(1,N)
029     4 DO 309 M=2,MAXG
030       WASTE(M,N)=WASTE(M-1,N)+OREBES(M,N)
031   309 ORE(M,N)=MAT(N)-WASTE(M,N)
032       DO 310 N=1,MAXN
033       REF(N)=OREBES(1,N)*(GI/GS)
034       DO 310 M=2,MAXG
035   310 REF(N)=REF(N)+(DFLOAT(M*GI)/GS)*OREBES(M,N)
036       DO 311 N=1,MAXN
037       PROD(1,N)=REF(N)
038       DO 311 M=2,MAXG
039       IF (ORE(M,N)) 311,311,1
040     1 PROD(M,N)=PROD(M-1,N)-(DFLOAT(M*GI)/GS)*OREBES(M,N)
041   311 CONTINUE
042       DO 312 N=1,MAXN
043       DO 312 M=1,MAXG
044       R1(M,N)=ORE(M,N)/MAT(N)
045   312 R2(M,N)=PROD(M,N)/MAT(N)
046       DO 353 N=1,MAXN
047       DO 353 M=1,MAXG
048       IF (ORE(M,N)) 354,354,355
049   354 R3(M,N)=0
050       GO TO 353
051   355 R3(M,N)=PROD(M,N)/ORE(M,N)
052   353 CONTINUE
053       I=ONE
054       J=ONE
055       K=ONE
056   314 CALL COGSUB (I,J,K,P)
057       WRITE (6,6)
058     6 FORMAT ('1','DISCOUNTED PROFITS FROM THE PRODUCTION
          1TARGETS OF EACH PLANT COMBINATION')
059       WRITE (6,5) (((PV(II,JJ,KK),II=1,3),JJ=1,3),KK=1,3)
060     5 FORMAT (3F13.1)
061       YIELD=REC(K)
062       WRITE (6,7)
063     7 FORMAT ('1','CUT-OFF GRADES & TARGETS FOR THE PLANT
          1COMBINATION YIELDING THE GREATEST EXCESS OF PRESENT
          2VALUE OVER CAPITAL COST')
064       WRITE (6,8) PV(I,J,K),R2(I),R3(J),REC(K)
```

```
065      8 FORMAT ('0','PRESENT VALUE=',F13.1,'CONCENTRATOR CA
        1PACITY=',F11.1,'REFINERY CAPACITY=',F9.1,'OVERALL R
        2ECOVERY EFFICIENCY=',F5.2)
066       WRITE (6,9)
067      9 FORMAT ('0',' YR  COG',4X,'MIN',8X,'CON',8X,'REF',
        1'PROFIT')
068       WRITE (6,10)  (N,CUTOFF(N),MIN(N),CON(N),QREF(N),
        1P(N),N=1,L)
069     10 FORMAT (I3,I5,3F10.1,F13.1)
070       SUM=0
071       DO 165 N=1,L
072     165 SUM=SUM+CUTOFF(N)
073       AVEG=SUM/L
074       RETURN
075       DEBUG SUBTRACE,SUBCHK
076       END

8.4.3 A LIST OF COG2

001       SUBROUTINE COG (AVEG,YIELD)
002       DOUBLE PRECISION MAT(20),ORERES(99,20),
         WASTE(65,20),ORE(65,20),REF(20),DFLOAT,PROD(65,20),
         2R1(65,20),R2(65,20),R3(65,20),PVMAX,PV(3,3,3),
         3CAP(3,3,3),Q2(3),Q3(3),Q4,Q5(99),MIN(99),CON(99),
         4QREF(99),P(99)
003       INTEGER CUTOFF(99),ONE/1/,SUM1,SUM,GI
004       INTEGER*2 AVEG,Q33(13500),Q34(13500),Q35(13500),
         1Q37(30),Q38(13500),Q40(13500)
005       DIMENSION REC(3),Q16(2),Q17(2),Q18(2),Q19(2),
         1Q20(99),Q22(99),Q23(99),Q24(99),Q25(2),Q26(99),
         2Q27(99),Q28(99),Q29A(13500)
006       COMMON ORERES,Q2,Q3,Q4,Q5,MAT,ORE,PROD,R1,R2,R3,PV,
         1MIN,CON,QREF,CAP,Q16,Q17,Q18,Q19,Q20,Q21,Q22,Q23,
         2Q24,Q25,Q26,Q27,Q28,REC,Q29A,GS,NI,NJ,NK,NBLKS,
         3MAXN,MAXG,CUTOFF,L,Q33,Q34,Q35,Q37,Q38,Q40
007       L=0
008       GI=5
009       DO 351 N=1,20
010       MAT(N)=0
011       REF(N)=0.0
012       DO 351 M=1,65
013       PROD(M,N)=0.0
014       WASTE(M,N)=0.0
015       ORE(M,N)=0.0
016       R1(M,N)=0.0
017       R2(M,N)=0.0
```

```
018         R3(M,N)=0.0
019     351 CONTINUE
020         DO 308 N=1,MAXN
021         DO 308 M=1,MAXG
022     308 MAT(N)=MAT(N)+ORERES(M,N)
023         DO 309 N=1,MAXN
024         WASTE(1,N)=ORERES(1,N)
025         IF (MAT(N)-WASTE(1,N)) 2,2,3
026       2 ORE(1,N)=ORERES(1,N)
027         GO TO 4
028       3 ORE(1,N)=MAT(N)-WASTE(1,N)
029       4 DO 309 M=2,MAXG
030         WASTE(M,N)=WASTE(M-1,N)+ORERES(M,N)
031     309 ORE(M,N)=MAT(N)-WASTE(M,N)
032         DO 310 N=1,MAXN
033         REF(N)=ORERES(1,N)*(GI/GS)
034         DO 310 M=2,MAXG
035     310 REF(N)=REF(N)+(DFLOAT(M*GI)/GS)*ORERES(M,N)
036         DO 311 N=1,MAXN
037         PROD(1,N)=REF(N)
038         DO 311 M=2,MAXG
039         IF (ORE(M,N)) 311,311,1
040       1 PROD(M,N)=PROD(M-1,N)-(DFLOAT(M*GI)/GS)*ORERES(M,N)
041     311 CONTINUE
042         DO 312 N=1,MAXN
043         DO 312 M=1,MAXG
044         R1(M,N)=ORE(M,N)/MAT(N)
045     312 R2(M,N)=PROD(M,N)/MAT(N)
046         DO 353 N=1,MAXN
047         DO 353 M=1,MAXG
048         IF (ORE(M,N)) 354,354,355
049     354 R3(M,N)=0
050         GO TO 353
051     355 R3(M,N)=PROD(M,N)/ORE(M,N)
052     353 CONTINUE
-------------------------COMMENT-----------------------------

The annual cut-off grades 'CUTOFF' and production tarjets
'MIN' 'CON' and 'ORDF' are calculated for all plant combina-
tions (I,J,K) rather than just the single trial combination
selected in subroutine 'COG1'.

-------------------------------------------------------------
053         DO 314 I=1,3
054         DO 314 J=1,3
055         DO 314 K=1,3
```

```
056   314 CALL COGSUB (I,J,K,P)
057       WRITE (6,6)
058     6 FORMAT ('1','DISCOUNTED PROFITS FROM THE PRODUCTION
         1TARGETS OF EACH PLANT COMBINATION')
059       WRITE (6,5) (((PV(II,JJ,KK),II=1,3),JJ=1,3),KK=1,3)
060     5 FORMAT (3F13.1)
```

----------COMMENT----------

The plant combination yielding the greatest present value of the profits that would result from its production targets less its capital cost 'CAP' is chosen to exploit the given deposit between Lines 061 and 070.

```
061       PVMAX=PV(1,1,1)-CAP(1,1,1)
062       DO 162 LL=1,3
063       DO 162 M=1,3
064       DO 162 N=1,3
065       IF (PVMAX-PV(LL,M,N)+CAP(LL,M,N)) 161,162,162
066   161 PVMAX=PV(LL,M,N)-CAP(LL,M,N)
067       I=LL
068       J=M
069       K=N
070   162 CONTINUE
```

----------COMMENT----------

The cut-off grades 'CUTOFF' and production targets 'MIN', 'CON' and 'QREF' are recalculated (Line 071) for the plant combination yielding the maximum present value.

```
071       CALL COGSUB (I,J,K,P)
072       YIELD=REC(K)
073       WRITE (6,7)
074     7 FORMAT ('1','CUT-OFF GRADES & TARGETS FOR THE PLANT
         1combination yielding the greatest excess of present
         2value over capital cost')
075       WRITE (6,8) PV,I,J,K),Q2(I),Q3(J),REC(K)
076     8 FORMAT ('0','PRESENT VALUE=',F13.1,'CONCENTRATOR CA
         1pacity=',f11.1,'refinery capacity=',f9.1,'overall r
         2ecovery efficiency=',f5.2)
077       WRITE (6,9)
078     9 FORMAT ('0',' YR  COG',4X,'MIN',8X,'CON',9X,'REF',
         1'profit')
079       WRITE (6,10) (N,CUTOFF(N),MIN(N),CON(N),QREF(N),
         1p(n),n=1,1)
```

```
080    10 FORMAT (I3,I5,3F10.1,F13.1)
081       SUM=0
082       DO 165 M=1,L
083   165 SUM=SUM+CUTOFF(M)
084       AVEG=SUM/L
085       RETURN
086       DEBUG SUBTRACE,SUBCHK
087       END
```

8. 5 SUBROUTINE COGSUB

8. 5. 1 COMPUTATIONAL SEQUENCE OF COGSUB

LINES 001 - 007

Specification and input statements.

LINES 008 - 029

Computation of the balancing cut-off grades 'GMC', 'GRM' and 'GRC'.

LINES 030 - 234

An implicit loop in which the optimum cut-off grades 'CUTOFF' and production targets 'MIN' 'CON' and 'QREF' are computed. This is a trial and error computation because the present value 'V' is unknown in the economic cut-off grade expressions (LINES 044 - 046). These computations converge on 'V'.

The implicit loop is divided into two parts, Lines 047 - 113 and Lines 014 - 207, in both of which many of the computations are duplicated for two different situations. The first part computes the cut-off grades and targets for annual production coming entirely from the material within a single pit increment 'N' of 'ORERES'. The second part performs the same computations for annual production straddling two pit increments.

LINES 044 - 046

The economic cut-off grades 'GM', 'GC' and 'GR' are computed for the year 'L'.

```
080    10 FORMAT (I3,I5,3F10.1,F13.1)
081       SUM=0
082       DO 165 M=1,L
083   165 SUM=SUM+CUTOFF(M)
084       AVEG=SUM/L
085       RETURN
086       DEBUG SUBTRACE,SUBCHK
087       END
```

8. 5 SUBROUTINE COGSUB

8. 5. 1 COMPUTATIONAL SEQUENCE OF COGSUB

LINES 001 - 007

Specification and input statements.

LINES 008 - 029

Computation of the balancing cut-off grades 'GMC', 'GRM' and 'GRC'.

LINES 030 - 234

An implicit loop in which the optimum cut-off grades 'CUTOFF' and production targets 'MIN' 'CON' and 'QREF' are computed. This is a trial and error computation because the present value 'V' is unknown in the economic cut-off grade expressions (LINES 044 - 046). These computations converge on 'V'.

The implicit loop is divided into two parts, Lines 047 - 113 and Lines 014 - 207, in both of which many of the computations are duplicated for two different situations. The first part computes the cut-off grades and targets for annual production coming entirely from the material within a single pit increment 'N' of 'ORERES'. The second part performs the same computations for annual production straddling two pit increments.

LINES 044 - 046

The economic cut-off grades 'GM', 'GC' and 'GR' are computed for the year 'L'.

LINES 047 - 077

The optimum cut-off grade 'G' is chosen from the balancing cut-off grades (GMC, GRM, GRC) and the economic cut-off grades (GM, GC, GR) for the year 'L'.

LINES 078 - 111

The plant production targets 'QM', 'QC' and 'QR' are computed for the year 'L'.

LINES 112 - 113

The year 'L' and pit increment 'N' are controlled with these statements by comparing the cumulative quantity of material 'SUMQM' produced from the current pit increment 'N' to the total material 'MAT' contained in that pit increment.

LINES 114 - 117

The production targets (QM, QC, QR) are adjusted for the first part of production years 'L' that straddle two pit increments 'N'.

LINES 118 - 120

The plant capacities (MINCAP, CONCAP, REFCAP) are scaled down to compute the cut-off grades and production targets for the remaining fractional part of the production year 'L' from the next pit increment 'N'.

LINES 121 - 205

The cut-off grades 'G' and production targets 'QM', 'QC', and 'QR' are calculated for the second part of production years 'L' that straddle two pit increments 'N'. These computations are executed in a similar manner to those between Lines 047 and 111.

LINES 206 - 208

The year 'L' and pit increment 'N' are controlled with these statements.

LINES 209 - 224

The subscript 'N' is deleted from the cut-off grade 'G' and production targets 'QM', 'QC' and 'QR'. These data sets (CUTOFF, MIN, CON, QREF) are supplied to the other subroutines through the COMMON statement.

LINES 225 - 230

The annual profit 'P' and its present value 'SUMPV' are computed.

LINE 231

The trial and error computations for the cut-off grades and production targets are controlled with this statement by testing the convergence of the present value 'SUMPV' in successive cycles. If the computations have not converged, the subroutine returns for another trial; otherwise it exits.

8. 5. 2 A LIST OF COGSUB

```
001       SUBROUTINE COGSUB (I,J,K,P)
002       DOUBLE PRECISION CONCAP(3),REFCAP(3),MINCAP,V(99),
         1SUMQM(20),R1(65,20),R2(65,20),R3(65,20),P(99),
         2QM(99,20),QC(99,20),QR(99,20),F(99),MAT(20),
         3ORE(65,20),PROD(65,20),MIN(99),CON(99),QREF(99),
         4SUMPV(99),PV(3,3,3),S1,S2,S3,U1,U2,U3,OREG,PRODG
002       DOUBLE PRECISION QM2,DABS,DFLOAT,Q1(99,20),
         1Q15(3,3,3)
003       INTEGER G(99,20),ONE/1/,CUTOFF(99),GI
004       INTEGER*2 GMC(20),GRM(20),GRC(20),GM(99),GC(99),
         1GR(99),GMC1(99),GRC1(99),GRM1(99),GMC2(99),
         2GRM2(99),GRC2(99),IFIX,Q33(13500),Q34(13500),
         3Q35(13500),Q37(30),Q38(13500),Q40(13500),Y
005       DIMENSION C(99),S(99),R(99),E(99),REC(3),Q16(2),
         1Q17(2),Q18(2),Q19(2),Q25(2),Q26(99),Q27(99),
         2Q28(99),Q29A(13500)
006       COMMON Q1,CONCAP,REFCAP,MINCAP,P,MAT,ORE,PROD,R1,
         132,R3,PV,MIN,CON,QREF,Q15,Q16,Q17,Q18,Q19,S,D,C,R,
         2E,Q25,Q26,Q27,Q28,REC,Q29A,GS,NI,NJ,NK,NDLKS,MAXN,
         3MAXG,CUTOFF,L,Q33,Q34,Q35,Q37,Q38,Q40
```

----------COMMENT----------

'GI' is given by the user and is equal to 'GI' in Line 011 of 'MAIN1'.

```
007   888 GI=5
008       DO 160 L1=1,99
009   160 V(L1)=0.0
010   900 S1=CONCAP(I)/MINCAP
011       S2=REFCAP(J)/MINCAP
012       S3=REFCAP(J)/CONCAP(I)
013       DO 6 N=1,MAXN
014       SUMQM(N)=0
015       GMC(N)=0
016       DO 2 M=1,MAXG
017       IF (DABS(R1(M,N)-S1)-.02) 1,1,2
018     1 GMC(N)=M*GI
019     2 CONTINUE
020       GRM(N)=0
021       DO 4 M=1,MAXG
022       IF (DABS(R2(M,N)-S2)-.00010) 3,3,4
023     3 GRM(N)=M*GI
024     4 CONTINUE
025       GRC(N)=0
026       DO 6 M=1,MAXG
027       IF (DABS(R3(M,N)-S3)-.00015) 5,5,6
028     5 GRC(N)=M*GI
029     6 CONTINUE
030    52 DO 151 L1=1,99
031       CUTOFF(L1)=0
032       P(L1)=0.0
033       DO 151 N=1,20
034       QM(L1,N)=0.0
035       QC(L1,N)=0.0
036       QR(L1,N)=0.0
037       G(L1,N)=0
038   151 CONTINUE
039       L=0
040       N=0
041   998 N=N+1
042       SUMQM(N)=0
043    10 L=L+ONE
044       GM(L)=(C(L)/((S(L)-R(L))*REC(K)))*GS
045       GC(L)=((C(L)+(F(L)+(D*V(L)))/CONCAP(I))/((S(L)-
         1R(L))*REC(K)))*GS
046       GR(L)=(C(L)/(S(L)-R(L,-(F(L)+(D*V(L)))/REFCAP(J))*
         1REC(K))*GS
047       IF (GM(L)-GMC(N)) 12,11,11
048    11 GMC1(L)=GM(L)
049       GO TO 70
```

```
050    12 IF (GC(L)-GMC(N)) 13,13,14
051    13 GMC1(L)=GC(L)
052       GO TO 70
053    14 GMC1(L)=GMC(N)
054    70 IF (GM(L)-GRM(N)) 16,15,15
055    15 GRM1(L)=GM(L)
056       GO TO 71
057    16 IF (GR(L)-GRM(N)) 17,17,18
058    17 GRM1(L)=GR(L)
059       GO TO 7.
060    18 GRM1(L)=GRM(N)
061    71 IF (GR(L)-GRC(N)) 20,19,19
062    19 GRC1(L)=GR(L)
063       GO TO 72
064    20 IF (GC(L)-GRC(N)) 21,21,22
065    21 GRC1(L)=GC(L)
066       GO TO 72
067    22 GRC1(L)=GRC(N)
068    72 IF (GMC1(L)-GRM1(L)) 102,105,101
069   101 IF (GRC1(L)-GMC1(L)) 104,105,105
070   102 IF (GRC1(L)-GRC1(L)) 105,105,103
071   103 IF (GRM1(L)-GRC1(L)) 107,106,106
072   104 IF (GRM1(L)-GRC1(L)) 106,106,107
073   105 G(L,N)=GMC1(L)
074       GO TO 73
075   106 G(L,N)=GRC1(L)
076       GO TO 73
077   107 G(L,N)=GRM1(L)
078    73 T1=MAT(N)/MINCAP
079       IF (G(L,N)) 153,154,153
080   154 OREG=ORE(1,N)
081       PRODG=PROD(1,N)
082       GO TO 117
083   153 DO 118 M=1,MAXG
084       IF (G(L,N)-M*GI) 113,112,111
085   111 IF (G(L,N)-MAXG*GI) 118,112,116
086   118 CONTINUE
087   116 OREG=0
088       PRODG=0
089       GO TO 117
090   112 OREG=ORE(G(L,N)/GI,N)
091       PRODG=PROD(G(L,N)/GI,N)
092       GO TO 117
093   113 IF (G(L,N)-GI) 116,114,115
094   114 OREG=ORE(1,N)
095       PRODG=PROD(1,N)
```

```
096         GO TO 117
097     115 OREG=ORE(M,N)+(((DFLOAT(M*GI)-DFLOAT(G(L,N)))/GI)
           1*(ORE(M-1,N)-ORE(M,N)))
098         PRODG=PROD(M,N)+(((DFLOAT(M*GI)-DFLOAT(G(L,N)))/GI)
           1*(PROD(M-1,N)-PROD(M,N)))
099     117 T2=OREG/CONCAP(I)
100         T3=PRODG/REFCAP(J)
101         IF (T1-T2) 501,502,502
102     501 IF (T2-T3) 506,505,505
103     502 IF (T1-T3) 506,504,504
104     504 T=T1
105         GO TO 74
106     505 T=T2
107         GO TO 74
108     506 T=T3
109      74 QM(L,N)=MAT(N)/T
110         QC(L,N)=OREG/T
111         QR(L,N)=PRODG/T
112         SUMQM(N)=SUMQM(N)+QM(L,N)
113         IF (MAT(N)-SUMQM(N)) 23,996,16
114      23 QM2=QM(L,N)+(MAT(N)-SUMQM(N))
115         QC(L,N)=(QM2/QM(L,N))*QC(L,N)
116         QR(L,N)=(QM2/QM(L,N))*QR(L,N)
117         QM(L,N)=QM2
118         U2=MINCAP-QM(L,N)
119         C2=CONCAP(I)-QC(L,N)
120         P2=REFCAP(J)-QR(L,N)
121         U1=C2/E2
122         U2=P2/E2
123         U3=P2/C2
124         N=N+1
125         IF (MAXN-N) 43,130,130
126     130 GMC2(N)=0
127         DO 25 M=1,MAXG
128         IF (DABS(R1(M,N)-U1)-.02) 24,24,25
129      24 GMC2(N)=M*GI
130      25 CONTINUE
131         GRM2(N)=0
132         DO 27 M=1,MAXG
133         IF (DABS(R2(M,N)-U2)-.0001) 26,26,27
134      26 GRM2(N)=M*GI
135      27 CONTINUE
136         GRC2(N)=0
137         DO 29 M=1,MAXG
138         IF (DABS(R3(M,N)-U3)-.00015) 28,28,29
139      28 GRC2(N)=M*GI
```

```
      GO TO 217
213   IF (G(L,N)-GI) 214,214,215
214   OREG=ORE(1,N)
      PRODG=PROD(1,N)
      GO TO 217
215   OREG=ORE(M,N)+(((DFLOAT(M*GI)-DFLOAT(G(L,N)))/GI)
     1*(ORE(M-1,N)-ORE(M,N)))
      PRODG=PROD(M,N)+(((DFLOAT(M*GI)-DFLOAT(G(L,N)))/GI)
     1*(PROD(M-1,N)-PROD(M,N)))
217   T2=OREG/C2
      T3=PRODG/P2
      IF (T1-T2) 601,602,602
601   IF (T2-T3) 606,605,605
602   IF (T1-T3) 606,604,604
604   T=T1
      GO TO 79
605   T=T2
      GO TO 79
606   T=T3
 79   QM(L,N)=MAT(N)/T
      QC(L,N)=OREG/T
      QR(L,N)=PRODG/T
      SUMQM(N)=QM(L,N)
      IF (MAT(N)-SUMQM(N)) 23,996,10
996   IF (MAXN-N) 43,43,998
 43   DO 45 M=1,L
      MIN(M)=0.0
      CON(M)=0.0
      QREF(M)=0.0
      SUMG=0
      NUM=0
      DO 49 N=1,MAXN
      IF (G(M,N)) 49,49,44
 44   NUM=NUM+ONE
      SUMG=SUMG+G(M,N)
      MIN(M)=MIN(M)+QM(M,N)
      CON(M)=CON(M)+QC(M,N)
      QREF(M)=QREF(M)+QR(M,N)
 49   CONTINUE
      CUTOFF(M)=SUMG/NUM
 45   CONTINUE
      DO 46 M=1,L
 46   P(M)=(S(M)-R(M))*QREF(M)-C(M)*CON(M)-E(M)*MIN(M)-
     1F(M)
      DO 47 M=1,L
      SUMPV(M)=0.
```

```
229         DO 47 Y=M,L
230      47 SUMPV(M)=SUMPV(M)+P(Y)/((1.+D)**(Y-M+1))
231         IF (DABS(SUMPV(1)-V(1))-10000000.0) 53,48,48
232      48 DO 51 M=1,L
233      51 V(M)=SUMPV(M)
234         GO TO 52
235      53 PV(I,J,K)=V(1)
236         RETURN
237         DEBUG SUBTRACE
238         END
```

8. 6 SUBROUTINE PD

8. 6. 1 COMPUTATIONAL SEQUENCE OF PD

LINES 001 - 006

Specification and input statements.

LINES 007 - 067

Design of the final pit surface.

LINE 019

The grid-blocks (X,Y,Z) with a positive net cash value 'VAL' are selected, in turn, to form the base of the semi-cone closures of the Lerchs - Grossman algorithm(9).

LINES 020 - 057

The grid-blocks (I,J,K) required to expose the base grid -block (X,Y,Z) are labelled by storing their location in the array 'ICL'. Any grid-blocks (I,J,K) included in the pit 'IP' or a previous test are not included.

LINES 058 - 067

The net cash values 'VAL' of the grid-blocks within the semi-cone closure 'ICL' are totaled. If this total 'SUM' is positive, the semi-cone closure 'ICL' is included in the pit 'IP'; otherwise, it is excluded.

LINES 068 - 086

A topographic map of the final pit surface is printed by these statements.

LINES 087 - 136

An implicit loop which simulates ore production from the pit.

LINES 128 & 132

Subroutine SUBPD is called to simulate the pit's waste production.

LINES 137 - 169

Schedules for the annual revenue, costs, ore production and waste production are computed and printed. Additionally the excavation times for ore 'IOR' and waste 'IWA' are combined into one array 'YR'.

8. 6. 2 A LIST OF PD

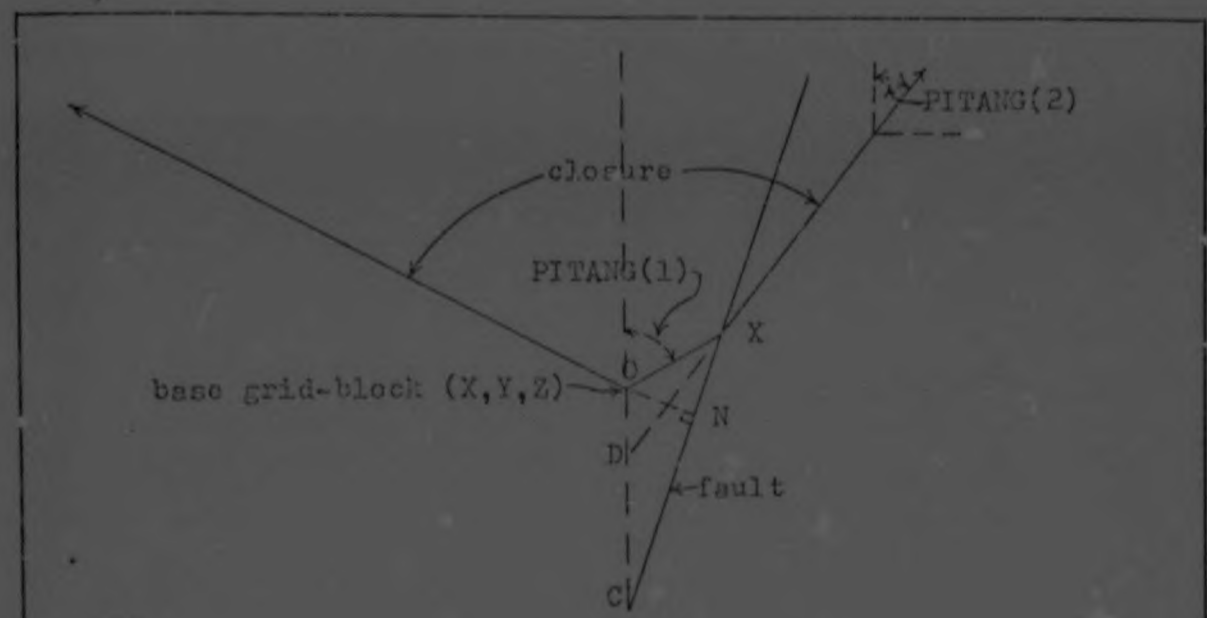
```
001         SUBROUTINE PD
002         DOUBLE PRECISION MAT(99),ORE(99),SUM,SORE,SWASTE,
           1DNEW,DOLD,DABS,Q1(99,20),Q2(3),Q3(3),Q4,Q5(99),
           2Q6(20),Q7(65,20),Q8(65,20),Q9(65,20),Q10(65,20),
           3Q11(65,20),Q12(3,3,3),Q15(3,3,3),Q14A(99)
003         INTEGER Q31(99),Q30,ONE/1/,TWO/2/
004         INTEGER*2 IP(13500),IWA(13500),ICL(13500),
           1IOR(13500),TOPO(13500),YR(13500),SURFC(30),X,Y,Z,
           2A,B
005         DIMENSION VAL(13500),PITANG(2),SLOPE(2),TONF(2),
           1WSLOPE(2),PROFIT(99),CAPOB(99),WASTE(99),Q17(2),
           2Q20(99),Q22(99),Q23(99),Q24(99),Q29(3)
006         COMMON Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,Q10,Q11,Q12,MAT,
           1ORE,Q14A,Q15,PITANG,Q17,SLOPE,WSLOPE,Q20,Q21,Q22,
           2Q23,Q24,TONF,PROFIT,CAPOB,WASTE,Q29,VAL,GS,NI,NJ,
           3NN,NBKLS,Q30,MAXG,Q31,L,IWA,IP,IOR,SURFC,TOPO,YR
007         DO 4 N=1,NBLKS
008         YR(N)=0
009         IP(N)=0
010         IWA(N)=0
011       4 IOR(N)=0
012         DO 200 N=1,NBLKS
013     200 ICL(N)=0
014         IC=0
```

```
015       DO 20 Z=1,NK
016       DO 20 Y=1,NJ
017       DO 20 X=1,NI
018       LOC=NJ*NI*(Z-ONE)+NI*(Y-ONE)+X
019       IF (VAL(LOC)) 20,20,2
```

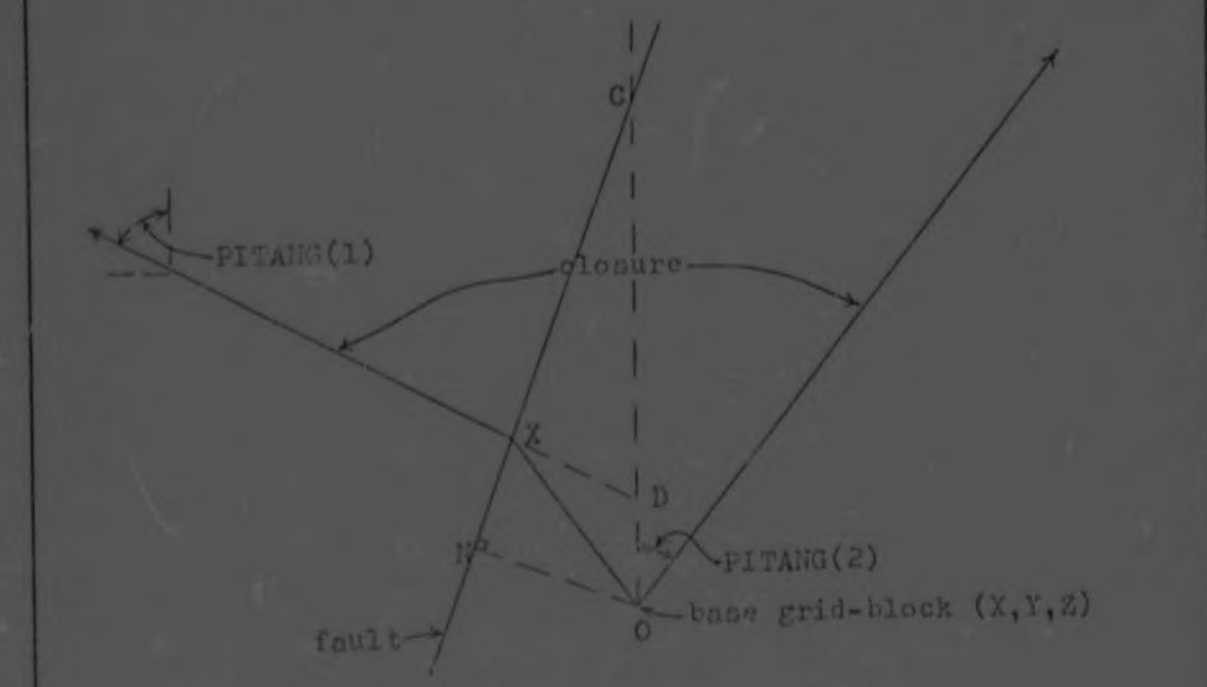
----------COMMENT----------

The constants (X,Y,D) of the cone expression in Line 51 are given by the grid co-ordinates of the base grid-block (X,Y,Z)• However, the constant 'D' must respond to variable safe wall-angles• For example, the given deposit (See Section 7•1) is divided into two zones by a fault (Line 020) either side of which has a different safe wall-angle 'PITANG'• The constant 'D' is adjusted to construct the overlapping cones required for the closures• The following statements (Lines 020 - 026 and 032 - 050), along with Fig• 8•4, demonstrate how this is accomplished for the example deposit• These statements must be tailored to the user's requirements•

```
020     2 FAULT=(66•3*X)-(277•5*Y)-(37•625*Z)+3274•075
021       ID=0
022       OC=ABS(Z-(((66•3*X)-(277•5*Y)+3274•075)/37•625))
023       IF (CC) 210,210,211
024   211 ON=ABS(FAULT/2885)
025       C=(ON/OC)
026       NOC=ATAN(SQRT(1•-(C*C))/C)
027   210 DO 240 K=1,NK
028       IF (K-Z) 8,8,288
029     8 DO 14 J=1,NJ
030       DO 14 I=1,NI
031       M=NJ*NI*(K-ONE)+NI*(J-ONE)+I
032       IF ((66•3*I)-(277•5*J)-(37•625*K)
          1+3274•075) 101,101,102
033   101 IS=TWO
034       IF (FAULT) 103,103,104
035   103 D=Z
036       GO TO 108
037   104 XOC=3•14159-PITANG(1)
039       OX=ON/COS(XOC-NOC)
039       CD=(CX*SIN(PITANG(1)-PITANG(2)))/SIN(PITANG(2))
040       D=Z+CD
041       GO TO 108
042   102 IS=ONE
043       IF (FAULT) 106,105,105
```



A. A BASE GRID-BLOCK ABOVE THE FAULT



B. A BASE GRID-BLOCK BELOW THE FAULT

FIG. 8.4 CROSS-SECTIONS OF OVERLAPPING CONES USED AS CLOSURES

```
044   105 D=Z
045.      GO TO 108
046 . 106 XOC=PITANG(2)
047     1 OX=ON/COS(ABS(NOC-XOC))
048     5 OD=(OX*SIN(PITANG(1)-PITANG(2)))/SIN(3.14159-
        1PITANG(1))
049       D=Z-OD
050   107 IF (D) 14,14,108
051   108 CONE=SLOPE(IS)*(K-D)*(K-D)-((2*J-2*Y)**2)-((I-X)
        1**2)
052       IF (CONE) 14,12,12
053    12 IF (IP(M)) 13,13,14
054    13 IB=IB+ONE
055       ICL(IB)=M
056    14 CONTINUE
057   240 CONTINUE
058   283 IF (IB) 20,20,174
059   174 SUM=0
060 .     DO 15 M=1,IB
061    15 SUM=SUM+VAL(ICL(M))
062 . .   IF (SUM) 18,18,16
063    16 DO 17 M=1,IB
064    17 IP(ICL(M))=ICL(M)
065    18 DO 19 M=1,IB
066    19 ICL(M)=0
067    20 CONTINUE
068       WRITE (6,99)
069    99 FORMAT ('1',' OPTIMUM PIT SURFACE')
070       DO 220 N=1,NJ
071   220 SURFC(N)=0
072       DO 97 X=1,NI
073       DO 96 Y=1,NJ
074       M=0
075       Z=ONE
076    91 IF (Z-NF) 87,87,92
077    87 LOC=NJ*NI*(Z-ONE)+NI*(Y-ONE)+X
078       IF (TOPO(LOC)) 88,88,89
079    89 IF (IP(LOC)) 92,92,88
080    88 M=Z
081       Z=Z+ONE
082       GO TO 91
083    92 SURFC(Y)=M
084    96 CONTINUE
085    97 WRITE (6,98) (SURFC(N),N=1,NJ)
086    98 FORMAT ('0',30I3)
-----------------------COMMENT-----------------------------
```

The following ore excavation sequence (Lines 088 - 114) illustrates a simulation for one working face that advances in opposite directions on alternate benches parallel to the I, J co-ordinate axes. This simulation must be tailored to the user's requirements, as described in Subsection 6.7.1.

```
087    26 L=ONE
088       J=0
089       I=0
090       K=0
091       SORE=0
092       SWASTE=0
093    31 ISIGN=ONE
094       GO TO 33
095    32 ISIGN=-ONE
096    33 K=K+ONE
097       IF (K-NK) 34,34,160
098    34 N=K-((K/2)*2)+ONE
099       GO TO (35,36),N
100    35 B=-ONE
101       GO TO 37
102    36 B=ONE
103    37 J=J+B
104       IF (J-NJ) 38,39,32
105    38 IF (J-ONE) 31,39,39
106    39 N=J-((J/2)*2)+ONE
107       GO TO (40,41),N
108    40 A=(-ONE)*ISIGN
109       GO TO 42
110    41 A=ISIGN
111    42 I=I+A
112       IF (I-NI) 43,43,37
113    43 IF (I-ONE) 37,44,44
114    44 NC=750*(K-ONE)+30*(J-ONE)+I
115       IF (IOR(NC)) 130,130,42
116   130 IF (IP(NC)) 42,42,110
117   110 IF (VAL(NC)) 42,42,45
118    45 IF (K-10) 46,46,47
119    46 M=ONE
120       GO TO 48
121    47 M=2
122    48 SORE=SORE+TONE(M)
123       IF (ORE(L)-SORE) 50,50,49
124    49 IOR(NC)=L
125       GO TO 42
```

```
126    50 IF (SORE-ORE(L)-(TONF(M)/2.)) 51,51,60
127    51 IOR(MC)=L
128       CALL SUBPD (SORE,M,K)
129       SORE=0
130       L=L+ONE
131       GO TO 42
132    60 CALL SUBPD (SORE,M,K)
133       L=L+ONE
134       SORE=TONF(M)
135       IOR(MC)=L
136       GO TO 42
137   160 DO 69 M=1,99
138       PROFIT(M)=0.0
139       CAPOB(M)=0.0
140       ORE(M)=0.0
141    69 WASTE(M)=0.0
142       DO 253 Z=1,NK
143       IF (Z-10) 71,71,72
144    71 II=ONE
145       GO TO 73
146    72 II=TWO
147    73 DO 86 Y=1,NJ
148       DO 86 X=1,NI
149       M=NJ*NI*(Z-ONE)+NI*(Y-ONE)+X
150       IF (TOPO(M)) 86,86,260
151   260 IF (IP(M)) 86,86,180
152   180 IF (IOR(M)) 181,181,83
153   181 IF (IWA(M)) 182,182,84
154   182 IWA(M)=L
155    84 YR(M)=IWA(M)
156       CAPOB(IWA(M))=CAPOB(IWA(M))+VAL(M)
157       WASTE(IWA(M))=WASTE(IWA(M))+TONF(II)
158       GO TO 86
159    83 YR(M)=IOR(M)
160       PROFIT(IOR(M))=PROFIT(IOR(M))+VAL(M)
161       ORE(IOR(M))=ORE(IOR(M))+TONF(II)
162    86 CONTINUE
163   253 CONTINUE
164       WRITE (6,250)
165   250 FORMAT ('1',' SIMULATED PIT PRODUCTION SCHEDULE')
166       WRITE (6,251)
167   251 FORMAT ('0',' YR OVERBURDEN STRIPPING COSTS ORE
      1 REVENUE')
168       WRITE (6,252) (N,WASTE(N),CAPOB(N),ORE(N),
      1PROFIT(N),N=1,I)
169   252 FORMAT (I3,4F13.1)
```

```
170       RETURN
171       END
```

8. 7 SUBROUTINE SUBPD

8. 7. 1 COMPUTATIONAL SEQUENCE OF SUBPD

LINES 001 - 006

Specification and input statements.

LINES 007 - 066

The grid-blocks required to expose the ore production of year 'L' are identified with the semi-cone set (Lines 024 - 061) and labelled by storing their location in the array 'IWA'.

LINE 008

The appropriate minimum working wall-angle 'WSLOPE' is selected.

LINES 011 - 066

DO loop 58 increases the working wall-angle 'WORKSL' in increments 'STEP' between its minimum 'WSLOPE' (Line 14) and its maximum '35.0' (Line 15) until the overburden 'SWASTE' within the waste closure approximates the waste production target 'MAT - ORE'. Then, the accuracy of the working wall-angle 'WORKSL' is increased by splitting the increment 'STEP', changing its sign and repeating the computations within the smaller tolerance. The computations are terminated when the overburden within the waste closure equals the waste production target. The termination can also occur, if the working wall-angle 'WORKSL' reaches the maximum or minimum limits without satisfying the waste production target. The maximum working wall-angle 'WORKSL=35' in Line 15 is a practical upper limit imposed to prevent the computations from going to infinity.

LINES 024 - 061

DO loop 56 identifies the ore grid-blocks 'IOR' that form

```
140    29 CONTINUE
141       IF (GM(L)-GMC2(N)) 31,30,30
142    30 GMC1(L)=GM(L)
143       GO TO 75
144    31 IF (GC(L)-GMC2(N)) 32,32,33
145    32 GMC1(L)=GC(L)
146       GO TO 75
147    33 GMC1(L)=GMC2(N)
148    75 IF (GM(L)-GRM2(N)) 35,34,34
149    34 GRM1(L)=GM(L)
150       GO TO 76
151    35 IF (GR(L)-GRM2(N)) 36,36,37
152    36 GRM1(L)=GR(L)
153       GO TO 76
154    37 GRM1(L)=GRM2(N)
155    76 IF (GR(L)-GRC2(N)) 39,38,38
156    38 GRC1(L)=GR(L)
157       GO TO 77
158    39 IF (GC(L)-GRC2(N)) 40,40,41
159    40 GRC1(L)=GC(L)
160       GO TO 77
161    41 GRC1(L)=GRC2(N)
162    77 IF (GMC1(L)-GRM1(L)) 202,205,201
163   201 IF (GRC1(L)-GMC1(L)) 204,205,205
164   202 IF (GRC1(L)-GMC1(L)) 205,205,203
165   203 IF (GRM1(L)-GRC1(L)) 207,206,206
166   204 IF (GRM1(L)-GRC1(L)) 206,206,207
167   205 G(L,N)=GMC1(L)
168       GO TO 78
169   206 G(L,N)=GRC1(L)
170       GO TO 78
171   207 G(L,N)=GRM1(L)
172    78 T1=NAT(N)/E2
173       IF (G(L,N)) 155,152,155
174   152 OREG=ORE(1,N)
175       PRODG=PROD(1,N)
176       GO TO 217
177   155 DO 210 M=1,MAXG
178       IF (G(L,N)-M*GI) 215,212,211
179   211 IF (G(L,N)-MAXG*GI) 218,212,216
180   218 CONTINUE
181   216 OREG=0
182       PRODG=0
183       GO TO 217
184   212 OREG=ORE(G(L,N)/GI,N)
185       PRODG=PROD(G(L,N)/GI,N)
```

the base of the semi-cone set. DO loops 19, 22 and 9 identify the grid-blocks that satisfy the semi-cone closure.

LINE 052

Note that any ore grid-blocks are excavated as well, if they satisfy the semi-cone set.

8. 7. 2 A LIST OF SUBPD

```
001       SUBROUTINE SUBPD (SORE,M,K)
002       DOUBLE PRECISION SORE,SWASTE,Q1(99,20),Q2(3),Q3(3),
         1Q4,Q5(99),Q6(20),Q7(65,20),Q8(65,20),Q9(65,20),
         2Q10(65,20),Q11(65,20),Q12(3,3,3),MAT(99),ORE(99),
         3Q13(3,3,3),Q14A(99),DNEW,DOLD,DABS
003       INTEGER Q31(99),ONE/1/,TWO/2/
004       INTEGER*2 IOR(13500),IP(13500),IWA(13500),
         1TOPO(13500),X,Y,Z,Q37(30),Q40(13500)
005       DIMENSION WRKANG(2),VAL(13500),TONT(2),Q16(2),
         1Q18(2),WSLOPE(2),Q20(99),Q22(99),Q23(99),Q24(99),
         2Q26(99),Q27(99),Q28(99),Q29(3)
006       COMMON Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,Q10,Q11,Q12,MAT,
         1ORE,Q14A,Q15,Q16,WRKANG,Q18,WSLOPE,Q20,Q21,Q22,
         2Q23,Q24,TONT,Q26,Q27,Q29,Q29,VAL,GS,NI,NJ,NK,
         3NBLKS,Q30,MAXG,Q31,L,IWA,IP,IOR,Q37,TOPO,Q40
077       STEP=5.0
008       WORKSL=WSLOPE(M)-STEP
009       IF (MAT(L)-ORE(L)) 121,59,121
010   121 DNEW=SORE-MAT(L)
011       DO 58 II=1,3
012    52 DOLD=DNEW
013       WORKSL=WORKSL+STEP
014       IF (WORKSL-WSLOPE(M)) 59,140,140
015   140 IF (WORKSL-35.0) 53,53,53
016    53 SWASTE=0
017       DO 171 Z=1,NK
018       DO 171 Y=1,NJ
019       DO 171 X=1,NI
020       N=NJ*NI*(Z-ONE)+NI*(Y-ONE)+X
021       IF (IWA(N)-L) 171,173,171
022   173 IWA(N)=0
023   171 CONTINUE
024       DO 56 Z=1,NK
025       IF (K-Z) 232,54,54
026    54 DO 56 Y=1,NJ
027       DO 56 X=1,NI
```

```
028         N=NJ*NI*(Z-ONE)+NI*(Y-ONE)+X
029         IF (ICR(N)-L) 56,55,56
030    55   DO 19 KK=1,NK
031         IF (KK-Z) 2,2,56
032     2   DO 22 J=1,NJ
033         DO 9 I=1,NI
034         N=NJ*NI*(KK-ONE)+NI*(J-ONE)+I
035         IF (TOPO(N)) 9,8,10
036    10   IF (IOR(N)) 1,1,9
037     1   IF (IP(N)) 9,9,3
038     3   IF (IWA(N)) 4,4,9
```

----------COMMENT----------

The constants (X,Y,D) of the cone expression in Line 47 are given by the grid co-ordinates of the base grid-blocks (X,Y,Z)• However, the constant 'D' must respond to variable working wall-angles• For example, the given deposit (See Section 7•1) is divided into two zones by the tenth grid-level (Line 039) either side of which has a different minimum working wall-angle 'WRKANG'• The constant 'D' is adjusted to construct the overlapping cones required for the closure• The following statements (Lines 039 - 046), along with Fig• 8•5, demonstrate how this is accomplished for the example deposit• These statements must be tailored to the user's requirements•

```
039     4   IF (Z-10) 14,14,15
040    14   D=Z
041         GO TO 16
042    15   IF (KK-10) 17,14,14
043    17   CO=Z-10•
044         CX=SQRT(WORKSL)*CO
045         CD=CO-CX*(COS(WRKANG(1))/SIN(WRKANG(1)))
046         D=Z-CD
047    16   IF (WORKSL*(KK-D)*(KK-D)-((2*J-2*Y)**2)-((I-X)
           1**2)) 9,5,5
048     5   IF (KK-10) 6,6,7
049     6   MM=ONE
050         GO TO 8
051     7   MM=TWO
052     8   IF (VAL(N)) 11,11,12
053    11   IWA(N)=L
054         SWASTE=SWASTE+TONP(MM)
055         GO TO 9
056    12   IOR(N)=L
```

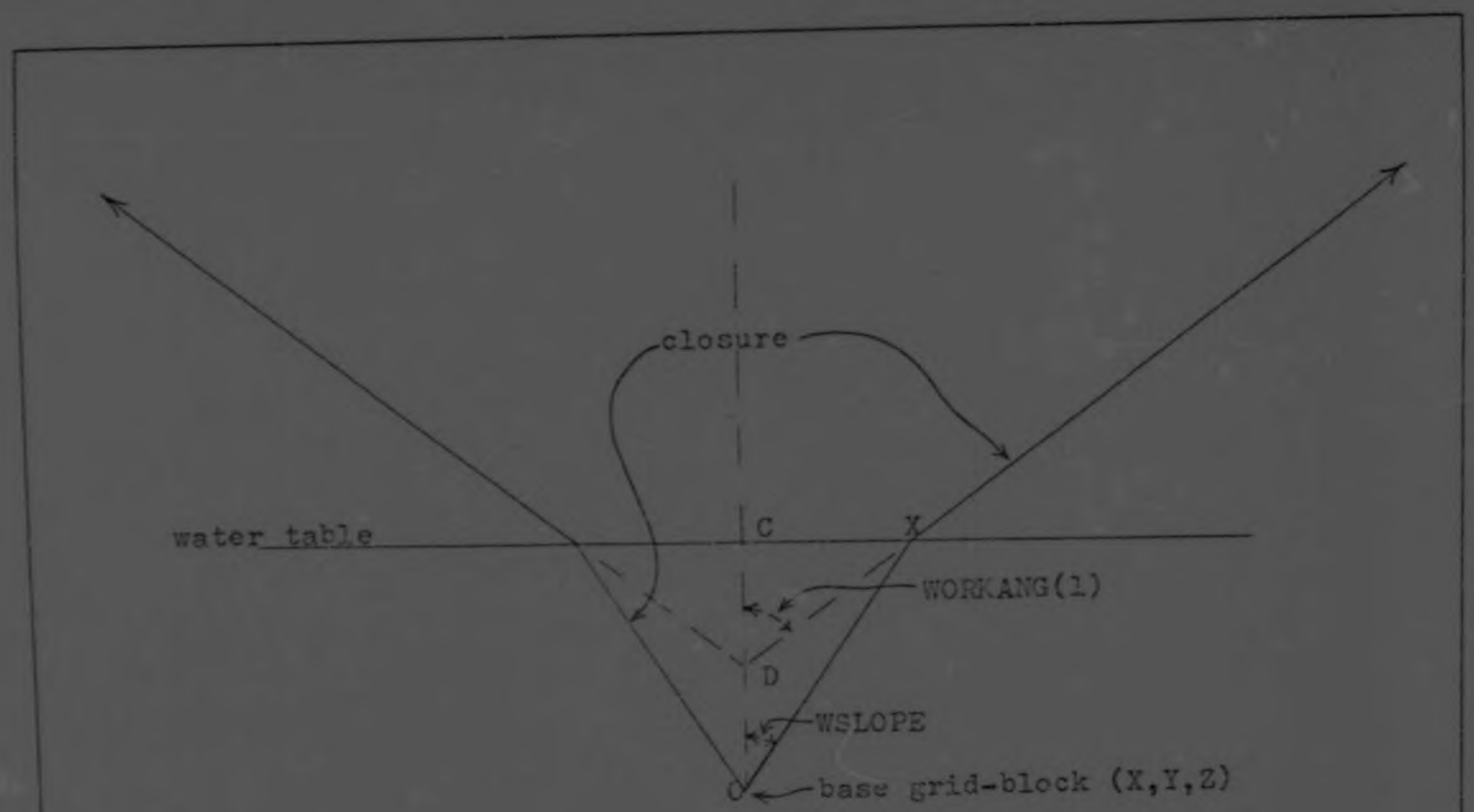


FIG. 8.5 A CROSS-SECTION OF OVERLAPPING CONES USED AS A CLOSURE

```
057        SORE=SORE+TONF(MM)
058      9 CONTINUE
059 ;   22 CONTINUE
060     19 CONTINUE
061     56 CONTINUE
062    232 DNEW=SWASTE-MAT(L)+SORE
063        IF (DABS(DNEW)-TONF(M)) 59,57,57
064     57 IF (DOLD*DNEW) 58,52,52
065     58 STEP=-0.5*STEP
066     59 SWASTE=0
067     23 RETURN
068        END
```

8. 8 GLOSSARY

The format of this glossary follows the same pattern as those of Chapters 4 and 5. Wherever a variable name appears twice in the glossary, it is the result of using the same name for two different applications in the MP Programme. The brackets which follow their definitions list the subroutines in which the given definition is applicable. Note also that any variables appearing in the specification statements of the form 'Qnn', where 'nn' is an identifying number, are only necessary to match the 'COMMON' statements. For this reason, they do not appear in the glossary. The glossary is not complete; it does not contain the variables used to facilitate programming.

AVEG - (I*2) the average cut-off grade over the pit's life span.

C(i) - (R) the annual concentration costs per input ton. See Subsection 6. 1. 2 for a rigorous definition;

i - year.

CAP(i,j,k) - (D) the total capital cost of each possible plant combination discounted to the first year at the opportunity cost of investment capital 'P';

i - concentrator identification number
j - refinery identification number
k - overall recovery efficiency identification number.

CAPOB(i) - (R) the annual cost of stripping overburden;

i - year.

CON(i) - (D) the annual production target for the concentrator in input tons;

i - year.

CONCAP(i) - (D) the plant capacity of all possible concentrators in input tons;

i - concentrator identification number.

CUTOFF(i) - (I) the annual cut-off grades scaled up by the grade scale factor 'GS' to integer value;

i - year.

D - (R) the opportunity cost of investment capital (MAIN1-3, COG1, 2 and CCGSUB).

D - (R) a constant used to define the semi-cone closures (PD & SUBPD).

E(i) - (R) the annual excavation costs per ton of pit output. See Subsection 6.1.2 for a rigorous definition;

i - year.

EQGRAD(i,j,k) - (I*2) the total equivalent mineral content at each grid-intersection;

i,j,k - I, J and K grid co-ordinates, respectively.

F(i) - (D) the fixed plus administrative costs per year;

i - year.

FIXC - (D) a variable used to transfer the fixed costs 'F' from the card reader to internal storage.

G(i,j) - (I) the annual cut-off grades in each pit increment. They are scaled up to integer values with the grade scale factor 'GS';

i - year
j - pit increment.

GM(i), GC(i), GR(i) - (I*2) the annual economic cut-off grades for the pit, concentrator and refinery, respectively. They are scaled up to integer values with the grade scale factor 'GS';

i - year.

GMC(i), GRM(i), GRC(i) - (I*2) the balancing cut-off grades for each pit increment between mine and concentrator, refinery and mine, and refinery and concentrator, respectively. They are scaled up to integer values with the grade scale factor 'GS';

i - pit increment.

GI - (I) the class interval for the increments in the T-S mineral distribution 'COARSE'.

GRAD1, GRAD2 . . . GRADn - (R) variables used to transfer the proportional mineral grades of the 'n' minerals reported in the user's deposit from external to internal storage.

GS - (R) a scale factor used for converting proportional mineral grades to integers so that they may be used as subscripts in arrays.

ICL(i) - (I*2) a pointer used to indicate the grid-blocks in a test closure of the Lerchs-Grossman graph algorithm;

i - grid-block identification number.

IGRAD1(i,j,k), IGRAD2(i,j,k) . . . IGRADn(i,j,k) - (I*2) the mineral grades of the 'n' minerals reported in the user's deposit at each grid-intersection. They are scaled up to integer values with the grade scale factor 'GS';

i,j,k - I,J and K grid co-ordinates, respectively.

INFRC, INFRP - (R) the interest rates for the time inflation of costs and selling prices, respectively.

IOR(i) - (I*2) the excavation year of the ore grid-blocks;

i - grid-block identification number.

IP(i) - (I*2) a pointer for the grid-blocks within the final pit surface;

i - grid-block identification number.

IWA(i) - (I*2) the excavation year of the overburden grid-blocks;

i - grid-block identification number.

L - (I) the mine's life span.

LOC - (I) the subscript for certain indicator arrays. It is computed to locate specific members of uni-dimensional arrays used to store three-dimensional data sets.

MAT(i) - (D) the total material in each pit increment measured in tons (COG1 - 2 and COGSUB);

i - pit increment.

MAT(i) - (D) the annual pit production target ore plus overburden, in tons (PD and SUBPD);

i - year.

MAXG - (I) the maximum equivalent mineral grade 'EQGRAD' reported in the user's deposit.

MAXN - (I) the total number of pit increments.

MILC(i) - (R) the milling cost per input ton for each recoverable mineral of the example deposit. See Section 7.1.

i - mineral identification number.

MIN(i) - (D) the annual pit production target, ore plus overburden, in tons;

i - year.

MINC - (R) a variable used to transfer the initial assumption of the excavation cost per ton 'S' from the card reader to internal storage.

MINCAP - (D) the pit capacity, ore plus overburden, per year measured in tons.

NBLKS - (I) the total number of grid-intersections or grid-blocks.

NI, NJ, NK - (I) the maximum I,J and K grid co-ordinates, respectively.

OB(i) - (R) the overburden stripping cost per ton on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

OBCI(i) - (R) the increased cost of overburden stripping per grid-level per ton on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

OR(i) - (R) the ore excavation cost per ton on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

ORCI(i) - (R) the increased cost of ore excavation per grid-level per ton on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

ORE(i,j) - (D) the available ore tonnage, in pit increments, for the range of cut-off grades from zero to the maximum reported mineral grade 'MAXG' (CCG1 - 2 and CCGSUB);

i - integer cut-off grade increment
j - pit increment.

ORE(i) - (D) this variable has two different definitions in the PD subroutine. Firstly, it is used as the production target for the concentrator. Then, it becomes the total simulated ore production in each year. Both data sets are measured in tons of concentrator input(PD);

i - year.

ORBIES(i,j) the data set for the T-S mineral distribution;

i - integer mineral grade increment
j - pit increment.

P(i) - (D) the annual profits estimated from the production targets 'MIN', 'CON' and 'OREF';

i - year.

PITANG(i) - (D) the safe pit wall-angles on each side of the fault given for the example deposit. See Section 7.1. The angles are measured in radians from vertical.

i - pit zone identification number.

PITINC - (I*2) the pit increment.

PNEW - (D) the total revenue over the mine's life span.

PROD(i,j) - (D) the tonnage, in each pit increment, of refined product available at any cut-off grade;

i - integer cut-off grade increment
j - pit increment.

PROFIT(i) - (D) the annual revenue from the simulated pit production;

i - year.

PV(i,j,k) - the total present value of the annual profits 'P' estimated from the production targets 'MIN', 'CON' and 'OREF' for each combination of the given plant capacities;

i - concentrator identification number
j - refinery identification number
k - overall recovery efficiency identification number.

PVMAX - (D) the present value of the combination of plant capacities yielding the maximum present value.

QC(i,j), QR(i,j), Q[illegible](i,j) - (D) the annual plant production targets for each pit increment in tons for the concentrator,

pit and refinery, respectively;

i - year
j - pit increment.

QREF(i) - (D) the annual production target for the refinery;

i - year.

R(i) - (R) the annual cost of refining per ton of product;

i - year.

R1(i,j), R2(i,j), R3(i,j) - (D) the ratios, in each pit increment, of ore 'ORE' to pit material 'MAT', refined product 'PROD' to pit material and refined product to ore, respectively, for the range of cut-off grades from zero to the maximum reported mineral grade 'MAXG';

i - integer cut-off grade increment
j - pit increment.

REC(i) - (R) the proportional overall recovery efficiency for each combination of plant capacities;

i - overall recovery efficiency identification number.

RLFC(i) - (R) the refining cost per ton of product for the concentrates originating from either side of a zone surrounding the fault given for the example deposit. See Section 7.1.

i - pit zone identification number.

REFCAP(i) - (D) the plant capacity of the refinery in output tons per year;

i - refinery identification number.

S(i) - (R) the annual selling price per product ton of the primary material;

i - year.

S1(i), S2(i), . . . Sn(i) - (R) the annual selling price per product ton of the 'n' minerals reported in the user's

deposit (MAIN1 - 3, PD, SUBPD);

i - year.

S1, S2, S3 - (D) the ratio of concentrator capacity 'CONCAP' to the pit capacity 'MINCAP', refinery capacity 'REFCAP' to pit capacity and refinery capacity to concentrator capacity, respectively, (COGSUB).

SLOPE(i) - (D) this variable is used to define the semi-cone closures of the final pit surface design routine. It is equal to the tanjent squared of 'PITANG'.

i - pit zone identification number.

SORE - (D) cumulative total of the simulated ore production in any given year and measured in tons.

SUMPV(i) - (D) the present value of the future profits 'P' in each year;

i - year.

SUMQM(i) - (D) the cumulative total of the annual pit production tarjets 'QM' in each pit increment;

i - pit increment.

SWASTE - (D) the cumulative total of the simulated overburden production in any given year.

T1, T2, T3, T - (D) the time periods required to process a given amount of pit material at the optimum cut-off grade 'G' for the pit, concentrator and refinery, and the maximum of the three, respectively.

TONP(i) - (D) the total weight of the grid-blocks on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

TOPO(i) - (I*2) an indicator array for the given deposit's surface topography;

i - grid-intersection identification number.

V(i) - (D) the present value of the future profits 'P' in each year;

i - year.

VAL(i) - (R) the T-S net cash value distribution;

i - grid-block identification number.

WASTE(i,j) - (D) the tonnages of overburden in each pit increment for the range of cut-off grades from zero to the maximum reported mineral grade 'MAXG' (COG1 - 2);

i - integer cut-off grade increment
j - pit increment.

WASTE(i) - (R) the total simulated overburden production in tons for each year (INT1 - 3, PD, SUBPD);

i - year.

WORKSL - (R) this variable is used to define the semi-cone sets of the overburden production simulation. It is the current value of the working wall-angle.

WORKANG(i) - (R) the minimum pit wall-angle at which the excavation equipment can work on each side of the tenth grid-level of the example deposit. See Section 7.1.

i - pit zone identification number.

WSLOPE(i) - (R) this variable is equal to the tangent squared of 'WRKANG';

i - pit zone identification number.

YR(i) - (I*2) the excavation year of each mined grid-block.

i - grid-block identification number.

YIELD - (R) the overall recovery efficiency of the combination of plant capacities yielding the maximum present value.

CHAPTER 9 CONCLUSIONS AND RECOMMENDATIONS

The economic Model of an open pit mine described in this thesis can be summarily illustrated with the following diagram:

```
MINE PLANNING--
               |
               |- +TIME FACTOR=OPEN PIT PLANNING SYSTEM
               |
FINANCIAL PLANNING--
```

Th's Model can, in fact, be considered as a "total systems" planning tool for the evaluation of open pit mining ventures. It can be adapted to the study of any proposal and any deposit by tailoring its computer programmes. These computer programmes then simulate:

1) the physical and geologic conditions of the user's deposit,

2) the particular pit production method, and

3) the venture's cost structure.

Previous contributions to the field employed in this Model are:

1) the estimation of mineral grades by Weaver (11),

2) the design of an optimum pit surface by Lerchs and Grossman (5), and

3) the economic planning decisions derived by Lane (4).

This thesis introduces simulation techniques for pit production, physical features and costing. Most importantly, it also introduces the time factor into planning and evaluation. The incorporation and synthesis of these innovations, as well as the preceding contributions makes this Model an

improvement over previous open pit planning systems. It does, in fact, represent a significant development in the field.

This development assures that there is no loss of accuracy as a result of the Model being a general planning tool. This is because its computer programmes are tailored to simulate the physio-economic differences between deposits. Additionally, equivalent cut-off grades are compared with the total mineral content of all recoverable minerals at each point in the deposit instead of comparing the traditional predetermined cut-off grades with the primary mineral only. The interpolation method used in the Model to estimate mineral grades is more adaptable than the other methods used for massive deposits. This is due to the search parameters which are redefined for each deposit.

In conclusion, certain recommendations for the further development and understanding of the Model can be made. These would be:

A) An anlysis of the Model to measure the sensitivity of its variables. A sensitivity analysis also would reveal any pathological situations not encountered in the testing of the computer programmes.

B) A study of the Model to establish whether its lengthy execution time can be cut down. For example, the Grade Prediction Programme could be made more efficient by eliminating some of the repetitious computations. The computations for the T-S net cash value distribution can be improved by eliminating its outermost DO loop which cycles with time. The time factor could be obtained solely from the indicator array 'YR' instead of from both the array and the DO loop parameter. This would eliminate the need to search through all grid -blocks for each particular year.

C) Finally, the Model could be further developed to include all costing situations. Then, rather than tailoring the programmes, the user would merely select those situations peculiar to his needs.

CHAPTER 10 REFERENCES

1) Hartman, H.J., Varma, C.G., A Three Dimensional Optimum Pit Program and a Basis for a Mining Engineering System, Proc. Symposium and Short Course on Computers and Operations Research in Mineral Industries, The Pennsylvania State University, University Park, Pa., V.2, April 1966.

2) International Business Machine Corp., IBM System/360 Operating System FORTRAN Language, IBM Corp., Poughkeepsie, N.Y., 1967.

3) International Business Machine Corp., IBM System/360 Operating System Job Control Language, IBM Corp., Poughkeepsie, N.Y., 1967.

4) Lane, K.F., Choosing the Optimum Cut-Off Grade, Quarterly of the Colorado School of Mines, Golden, Colo., V.59, no.4, p.811, Oct. 1964.

5) Lerchs, H., Grossman, I.F., Optimum Design of Open Pit Mines, Transactions of the Canadian Institute of Mining, V.118, P.17, 1965.

6) Ore, O., Graphs and their Uses, Random House, New York, N.Y., 1963.

7) Pana, M.T., ET AL, a Description of Computer Techniques Used in Mine Planning at the Utah Mine of Kennecott Copper Corporation, Proc. Symposium and Short Course on Computers and Operations Research in Mineral Industries, The Pennsylvania State University, University Park, Pa., V.2, April 1966.

8) Pana, M.T., The Simulation Approach to Open Pit Design, Transactions of the Short Course and Symposium on Computers and Computer Applications in Mining and Exploration, University of Arizona, Tuscon, Ariz., March 1965.

9) Pfleider, E.P. (editor), Surface Mining, Seely W. Mudd Series of The American Institute of Mining, Metallurgical and Petroleum Engineers, New York, N.Y., 1968.

10) Spiegel, M.R., Theory and Problems of Statistics, Schaum Publishing Co., New York, N.Y., 1961.

11) Weaver, R.C., Relative Merits of Interpolation and Approximating Functions in the Grade Prediction Problem, Stanford University Publications in the Geological Sciences, Stanford, Calif., V.9, no.1, 1964.

Author Robinson R H
Name of thesis An Economic model of an open pit mine 1969

PUBLISHER:
University of the Witwatersrand, Johannesburg
©2013

LEGAL NOTICES:

Copyright Notice: All materials on the University of the Witwatersrand, Johannesburg Library website are protected by South African copyright law and may not be distributed, transmitted, displayed, or otherwise published in any format, without the prior written permission of the copyright owner.

Disclaimer and Terms of Use: Provided that you maintain all copyright and other notices contained therein, you may download material (one machine readable copy and one print copy per page) for your personal and/or educational non-commercial use only.

The University of the Witwatersrand, Johannesburg, is not responsible for any errors or omissions and excludes any and all liability for any errors in or omissions from the information on the Library website.