COMPUTER-AIDED UNDERGROUND MINING MACHINE SEQUENCING

A Thesis Presented to
The Faculty of the College of Engineering and Technology
Ohio University

In Partial Fulfillment
of the Requirements for the degree
Master of Science

By
Ilango Sankaralingam

November 1987
ACKNOWLEDGEMENTS

The author wishes to express his gratitude to Professor Donald E. Scheck, whose valuable guidance and constructive criticisms made the completion of this study possible.

The author would also like to thank Professor P. K. Chatterjee for all the technical information and encouragement provided during this project.

Finally, the author would like to express his thanks to Professor J. Gunasekara for allowing the use of the Intergraph CAD/CAM facilities and to Professor E. Ralph Sims Jr. for agreeing to be in the thesis committee.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PROBLEM STATEMENT</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>BACKGROUND</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>Description of Problem</td>
<td>5</td>
</tr>
<tr>
<td>2.2</td>
<td>Problem Analysis</td>
<td>8</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Mine Layout</td>
<td>8</td>
</tr>
<tr>
<td>2.2.2</td>
<td>Mining Equipment and Technique</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>LITERATURE SURVEY</td>
<td>13</td>
</tr>
<tr>
<td>3.1</td>
<td>Linear Programming</td>
<td>13</td>
</tr>
<tr>
<td>3.2</td>
<td>Integer Programming</td>
<td>15</td>
</tr>
<tr>
<td>3.3</td>
<td>Dynamic Programming</td>
<td>20</td>
</tr>
<tr>
<td>3.4</td>
<td>Graph and Network Theory</td>
<td>21</td>
</tr>
<tr>
<td>3.5</td>
<td>Simulation</td>
<td>22</td>
</tr>
<tr>
<td>3.6</td>
<td>Heuristic Methods</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>MODEL DEVELOPMENT</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>Evaluation of Techniques</td>
<td>30</td>
</tr>
<tr>
<td>4.2</td>
<td>Proposed Methodology</td>
<td>31</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Scheduling Procedure</td>
<td>34</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Evaluation of Schedule</td>
<td>35</td>
</tr>
<tr>
<td>5</td>
<td>PROGRAM DESCRIPTION</td>
<td>36</td>
</tr>
<tr>
<td>5.1</td>
<td>Section Definition</td>
<td>38</td>
</tr>
<tr>
<td>5.2</td>
<td>Network Generation</td>
<td>38</td>
</tr>
</tbody>
</table>
5.3 Production data input and Tonnage Calculation 47

5.4 Schedule Generation and Timing 47

5.4.1 Machine Allocation Algorithm 47

5.4.2 Next Area to be Mined Algorithm 50

6 RESULTS AND CONCLUSIONS 57

6.1 Results 57

6.2 Conclusion 64

6.3 Recommendation for future action 65

BIBLIOGRAPHY 67

APPENDIX A A-1
(User's Manual)

APPENDIX B B-1
(Data files)

APPENDIX C
(Program Listing)

Copy on file at the Industrial and Systems Engineering Department
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Precedence constraints for longwall mining.</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Sample mine layout</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>Layout with numbered sections</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td>Flexibility in layout design</td>
<td>12</td>
</tr>
<tr>
<td>5</td>
<td>Block diagram of scheduling program</td>
<td>37</td>
</tr>
<tr>
<td>6</td>
<td>Section definition module</td>
<td>39</td>
</tr>
<tr>
<td>7</td>
<td>Network generation module</td>
<td>40</td>
</tr>
<tr>
<td>8</td>
<td>Adjacency relationships between areas</td>
<td>41</td>
</tr>
<tr>
<td>9</td>
<td>Auto-splitting of adjacent areas</td>
<td>43</td>
</tr>
<tr>
<td>10</td>
<td>Recreated area definitions</td>
<td>44</td>
</tr>
<tr>
<td>11</td>
<td>Network determination procedure</td>
<td>45</td>
</tr>
<tr>
<td>12</td>
<td>Network representation of sample layout</td>
<td>46</td>
</tr>
<tr>
<td>13</td>
<td>Production data input and tonnage calculations</td>
<td>48</td>
</tr>
<tr>
<td>14</td>
<td>Schedule generation and timing module</td>
<td>49</td>
</tr>
<tr>
<td>15</td>
<td>Flowchart of the machine allocation algorithm</td>
<td>51</td>
</tr>
<tr>
<td>16</td>
<td>Flowchart of the next area to be mined algorithm</td>
<td>55</td>
</tr>
<tr>
<td>17</td>
<td>Timing map (2 continuous miners and one longwall shear)</td>
<td>62</td>
</tr>
<tr>
<td>18</td>
<td>Timing map (3 continuous miners and one longwall shear)</td>
<td>63</td>
</tr>
</tbody>
</table>
CHAPTER 1

PROBLEM STATEMENT

In the hierarchy of performing mine planning tasks, production scheduling is naturally at the bottom. Yet good production scheduling is probably the most demanding and difficult task to achieve for various reasons. Being the last task in the mine planning hierarchy, a proper production schedule must not only conform to both the long-range and short-range mine plans but also satisfy many practical details that are unique to day-to-day operations.

Underground long-range planning and mine design is an iterative process beginning with a specified production level, quality criteria, and geology of the deposit. Given these factors, face equipment is selected, the required number of units needed and production per unit are calculated, and the mine infrastructure needs are developed. The scheduling of units then follows, with the design of the various support systems coming next.

The introduction of powerful mining equipment and new techniques has increased the productivity of mining operations. The benefits from the improved technology and organization may not be fully achieved if production planning and scheduling introduce unnecessary delays and idle time. Computer-aided scheduling can help management achieve the highest possible productivity that is consis-
tent with the available resources and contractual requirements. Higher production rates have made mine planning and scheduling more critical. Extensive cost analyses are now performed manually to determine the economics of proposed changes in the layout.

Production scheduling in mining takes two basic forms. One of these consists essentially of making specific operating decisions to assign equipment and personnel in an effort to optimize their utilization. The other consists of studying a situation to determine the most general operating policies.

The complexity and intricacies involved in production scheduling have made the use of sophisticated operations research techniques and computers imperative. Developing a schedule manually for many different alternatives would be inconceivable. The sheer magnitude of the computations involved in validating a schedule would make the evaluation of many alternative schedules impossible. The mining engineer faces a major task of scheduling the actual working force while taking into account a multitude of constraints such as ore quality, ventilation, power and production requirements.

The usual objective is to meet the production goal and the short range plan at a minimum operating expense. The optimum production schedule must be able to maintain the required quality of finished ore. Other consider-
ations such as ventilation, power, material handling and other production support systems complicate the scheduling problem. Generally speaking, the constraint on production schedules is the capacity and capability of the plant which converts the ore into a marketable mineral or commodity. Over the planning horizon, the demand for the commodity can be regarded as constant, and thus the variance of the sales forecast is not a major factor in the scheduling decision.

Typically the cost of mining tends to increase and the quality of ore tends to decrease over the life of a mine. The primary objective is to mine in such a way so as to maximize the total present worth of the mine. This objective may frequently also mean that the recovery of the mineable resources has to be maximized.

The difficulty in finding an optimum schedule is not so much to meet the production requirement in the near future but to insure that production can be sustained at the required level throughout the life of the mine. The variability of the material and of the difficulty of estimating the variability in advance of mining adds to the complexity of the problem. The initial schedule may be optimal or near optimal for the conditions at the time of scheduling but as more information becomes available, the schedule may need to be revised. The dynamic nature of the mine scheduling problem requires that the schedule
have inherent flexibility. The development work should be timed so that the mine planner has many practical options for deviating from the original schedule and reoptimizing future plans.
CHAPTER 2

BACKGROUND

2.1 Description of problem:

The research is confined to scheduling the mining of coal or other bedded deposits that will be mined with a combination of room-and-pillar and longwall methods. Given the layout, the problem is to schedule the mining of $n$ sections so as to achieve the following objectives.

1. minimize the deviations from the desired production levels over a period of time.
2. minimize the time to open up longwall panels which ensures their continuous use and consequently reducing the overall production cost.
3. maximize the total production that can be obtained from the mine and increase opportunities to open up more mining sections in the future.

The objectives are listed in their hierarchical order of preference. Although all the objectives are important, consistent production is essential to meet contractual obligations and thus is the greatest concern of management. Stockpiling production to meet future demand is expensive in terms of the time value of money, cost of stockpiling facilities and risk of spontaneous combustion of some ores.
Mining sections are divided into three categories which depend upon the mining methods, that is, longwall, room-and-pillar advance and room-and-pillar retreat mining. The advance room-and-pillar mining creates entries and pillars and all or part of the pillars are removed during the retreat phase. Normally the procedure is to do advance and retreat mining in opposite directions.

The time required to mine a section (duration) is dependent on the size of the section and on the mining rate. The mining rate, on the other hand, depends on the geologic and environmental conditions and on the mining method. In this analysis the following are assumed:

1) the mining rates, and therefore the durations, are fixed and known in advance for each section.

2) the sections are assumed to be available whenever they are scheduled for mining. This condition means that all support facilities such as ventilation and power are already in place for the sections scheduled to be mined.

3) there are no due dates by which any or all sections must be mined.

4) the mining equipment can operate in "parallel" which means that all the mining machines can work at the same time, independently of each other.

5) a machine can mine at most one section at a time, and a section can be mined by at most two mining
machines at the same time.

The sections are to be scheduled subject to precedence constraints in the form of a partial order induced by a given acyclic graph of sections (nodes) constructed from the mine layout. Every room-and-pillar section, except the first section, must be preceded by another room-and-pillar section. Also, every longwall section must be preceded by all four or more development sections surrounding the longwall panel as shown in Figure 1. The development sections numbered from 1 to 4 must be completed before starting to mine the longwall panel.

![Diagram](image)

**Figure 1**: Precedence constraints for Longwall mining.
Advance mining of a given room-and-pillar area obviously must be done before retreat mining. A schedule is termed feasible if all the sections are completed and it satisfies all the precedence constraints. A schedule is termed optimal if it is feasible and if the objective is met. In general, underground mining can be reduced to the problem of scheduling of precedence constrained jobs on parallel processors.

Very real constraints such as the availability of ventilation, power, ore quality and other requirements have been assumed out of the problem. The final schedule may not be practical since some of these restrictions have not been considered in the scheduling process. Hence, manual adjustments to the schedule may be required.

2.2 PROBLEM ANALYSIS:

2.2.1 Mine Layout:

In present practice the mine layout forms the basis of the scheduling function. An interesting question is whether the determination of this layout could be fully computerized, that is, whether a feasible or optimal layout can be determined by modelling the layout procedure and implementing it on the computer. This study follows present practice, that is, the mine layout is an input to the scheduling process. Since the layout must be in a
machine readable form, the layout was generated by ICAMPS
(Interactive Computer Aided Mine Planning System, Chatterjee, Scheck and Sridhar (1986)).

For the purpose of simplicity only underground mines that use longwall mining were considered. This limitation is valid because longwall mining is extremely capital intensive. Downtime due to scheduling conflicts is much more costly in longwall mining than in room-and-pillar mining. In addition, room-and-pillar equipment is much more flexible and work stoppages due to poor scheduling are less likely to occur.

A detailed layout is not essential for long-term scheduling purposes. The mining sections have been assumed to have three or four sides due to ease of representation and computation effort. Only the coordinates of the main room-and-pillar sections and longwall panels are necessary for the network determination. The boundary of each main roadway, development section and room-and-pillar panel, if any, must be determined. The interior details such as entry, pillar and crosscut dimensions can be based upon average design parameters. A sample layout and the section definitions are shown in Figures 2 and 3. The layout can be represented by a directed network, in which the nodes represent sections and the directed arcs represent the precedence constraints.
Figure 2: Sample Mine Layout
Figure 3: Section Definition with numbered areas
A general rule is that sections should be designed such that the number of immediately succeeding sections to any section are minimized. The above rule can be justified by the example given in Figure 4 in which the design in Figure 4(b) is more flexible than the layout in Figure 4(a). The layout in Figure 4(b) also has a fewer succeeding section alternatives.

2.2.2 Mining Equipment & Technique:

Since the room-and-pillar & longwall mining equipment are extremely expensive, their utilization has to be maximized. Conveyor belts represent another significant equipment investment. As a by-product of developing a schedule, the minimum number of belt heads that are required to meet the schedule could also be determined.
CHAPTER 3

LITERATURE SURVEY

A large array of techniques have been applied to solve production scheduling problems for underground mining operations. The techniques range from rigid operations research (OR) methods to heuristic procedures which are not mathematically based. The following sections cover the application of linear programming (LP), integer programming, dynamic programming, graph and network theory, simulation, and heuristic methods for mine scheduling. For an in-depth discussion of these techniques, the reader can read any of the available textbooks on these subjects.

3.1 LINEAR PROGRAMMING

Linear Programming is the most widely used OR method for production scheduling problems. An LP model consists of a linear objective function, a set of linear constraints and a set of non-negativity constraints. In production scheduling the objective function is often to optimize total profit, total cost, tonnage, cost/ton or other pertinent operating variables. Most mining operations attempt to maximize the total profits, but other variables such as shift tonnage or longwall tonnage could
be maximized. Since the available resources are finite, the constraint expressions mathematically define the resource limitations.

The final component, the non-negativity constraints, insures that the variables do not take on negative values. Since the variables normally represent production levels in a scheduling problem, this constraint simply matches the practical limitation that an operation cannot produce negative tonnages.

Wilke and Reimer (1979) have proposed a method to optimize the short-term production schedule for an open-pit iron ore mining operation. Short-term production scheduling has to be a well balanced compromise, taking into account a great number of factors that influence future development. They used an LP algorithm to obtain optimal short term production schedules.

The Wilke and Reimer model is oriented toward an actual shift-to-shift production scheduling problem and consequently is a very simple formulation. Production scheduling models which consider long range effects can become quite complicated.

Vinas (1971) illustrated the use of linear programming for scheduling cut and fill, square set, and shrinkage stoping operations. He used a LP model with an objective function formulation for maximizing profit. His work was purely theoretical and no application appears in
the literature.

Dessureault and Galibolis (1973) formulated an interesting LP model for scheduling trucks in an open-pit mine. The objective of their model was to assign all trucks within the pit so that the cost of the operation is minimized. Gershon (1982) has described a very general linear programming (LP) formulation for scheduling mining operations.

The assumption of linearity may appear to be a serious impediment that limits the use of the LP model. In some cases, if the linearity constraint is relaxed, the problem can still be solved by using one of the several non-linear techniques such as quadratic programming or separable programming.

3.2 INTEGER PROGRAMMING

Some production scheduling problems can be formulated as integer programming and solved with branch-and-bound techniques. Zero-one problems are often solved by an enumeration procedures. The following example illustrates the use of a zero-one programming model for the underground mine scheduling problem.

Objective: Maximize the total net present value of the sections as they are being mined:
\[
\max \sum_{j,k} v_{jk}/(1+r)^t \\
j, k \\
j \in N, k \in M
\]

where

\(t_j\) = starting time of section \(j\)

\[
= \sum_{i,k,t} t x_{ijkt} \\
i,k,t \\
i \in A^*(j), k \in M
\]

\(x_{ijkt} = 1\), if section \(j\) is scheduled, at time \(t\), as the first section to follow section \(i\) on miner \(k\)

\(= 0\), otherwise.

Note: The mining of section \(j\) need not immediately follow the mining of section \(i\).

\(d_{ik}\) = time to mine section \(i\) on miner \(k\) (known).

\((=\infty\) if the section cannot be mined by miner \(k\).\)

\(v_{ik}\) = net present value of section \(i\), if mined by miner \(k\), discounted to the beginning of the section (known).

\(r\) = discount rate (known).

\(N\) = set of all sections.

\(R\) = subset of \(N\) with \(1 \in R \geq 2\).

\(NW\) = set of sections without predecessors.

\(A(i)\) = set of sections after section \(i\) in the
precedence graph.

\[ B(i) = \text{set of sections before section } i \text{ in the precedence graph.} \]
\[ o = \text{a dummy section which must precede every section in } NW. \]
\[ 0 = \text{a dummy section which must be the last section for every miner.} \]
\[ A'(j) = (N U o) \setminus A(j). \]
\[ B'(j) = (N U 0) \setminus B(j). \]
\[ M = \text{set of all miners.} \]
\[ i,j,l = \text{section indices.} \]
\[ k = \text{miner index.} \]
\[ t = \text{time index.} \]

The dummy sections, \( o \) and \( 0 \), can be thought of as "storage space", from which the miners initially come from and to which the miners finally return. That is, \( o \) and \( 0 \) are basically identical.

The 0-1 variables, \( x_{ijkl} \), constitute the decision variables, i.e. the unknowns. All other variables are known. The scheduling problem can be formulated as follows.

1. Each section \( j \), must be scheduled, but can only be scheduled once and only once for one mining machine:

\[
\sum_{i,k,t} x_{ijkl} = 1 \quad j \in N
\]
2. A machine, \( k \), which is assigned to section \( j \) after completing section \( i \) will be assigned to another section \( l \) after completing section \( j \):

\[
\sum_{i,t} x_{ijkt} = \sum_{l,t} x_{jlkt} \quad j \in N, k \in M
\]

\( i \in A'(j), i<>j, k \in M \)

\( i \in B'(j), l \in B'(j) \)

\( i<>j \)

\( i<>j \)

3. At time zero, each machine \( k \) can be assigned to only one section:

\[
\sum_{j \in NW} x_{0jko} \leq 1 \quad k \in M
\]

4. The mining of section \( l \) can only start \( d_{jk} \) time units after machine \( k \) starts mining section \( j \), i.e. after section \( j \) is completed (the precedence requirement):

\[
\sum_{l,k,t} (t+d_{jk}) x_{ijkt} \leq \sum_{l,k,t} t x_{jlkt} \quad j \in N
\]

\( l \in B'(j), j<>l, k \in M \)

\( l \in B'(j), j<>l, k \in M \)

The left side equals the completion time for section \( j \). The right side equals the starting time of section \( l \).
5. The sequence of sections assigned to each miner \( k \) must start with section 0 and end with section 0, i.e. "subtours" are not allowed:

\[
\sum_{i,j,t} x_{ijkt} \leq 1 \quad \forall k \in M, i \in R, j \in R, i \neq j
\]

The model represented by expressions (1) to (5) is nonlinear due to the objective function. It was developed by West-Hansen, Sarin and Topuz (1986).

Lambert and Mutmansky (1973) outline a pure integer programming model which is designed to assign both trucks and shovels in an open-pit mine. The model is formulated in a zero-one fashion and solved by a branch and bound algorithm.

A hybrid model (Daud and Pariseau, 1975) couples integer programming with simulation. The general procedure is to use an integer programming model to choose a number of solutions and plugging these solutions into a simulation program so that the probabilistic nature of the situation can be evaluated. The basic model of Daud and Pariseau is an example of pure integer programming. The authors have proposed three objective functions for their model. The constraints are simple and straightforward with the truck constraint limiting the number of available trucks. The maximum and minimum tonnage constraints correspond to the production capacity of the
shovels and minimum production requirement to develop the planned pit geometry respectively. They solved the model using the same branch-and-bound algorithm as Lambert and Mutmansky.

3.3 DYNAMIC PROGRAMMING

When a decision making problem can be represented as a series of individual decisions which are inter-related, then the problem can often be formulated as a dynamic programming problem. This technique is a rather general approach based upon Bellman's principle of optimality which states, "An optimum policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

Most Dynamic programming problems will ordinarily take one of the two different patterns. The first case requires a sequence of decisions such as scheduling trucks in an open pit mine. The decisions for each shift are made one at a time but the interaction between the shifts affects the overall operation. The dynamic programming format simply puts the problem into a disciplined form that helps the decision maker find the correct solution.

The second type of dynamic programming application is used to allocate resources within a stage, for example,
day-to-day assignment of trucks to shovels within a shift.

Graph theory can be used to optimize the return at each individual stage. The dynamic programming approach insures that the overall decision process yields the optimal combination of individual stage decisions.

3.4 GRAPH AND NETWORK THEORY

Graph and network theory have found application in many areas of production scheduling. The algorithms that deal with flow through a network appear to be most useful. Many problems can be expressed in a network format such as the critical path method (CPM) and the project evaluation and review technique (PERT) for project management.

Although network analysis techniques are widely used in project control they are also suitable for repetitive production scheduling applications. Another use is the area called decision analysis. In this application decision trees are used to formalize the decision making process in which a series of decisions may have probabilistic outcomes.

Davis and Heidorn (1971) describe an algorithm for the computation of optimal (minimum duration) solutions for the resource constrained project network scheduling problem under conditions of multiple resource require-
ments per job (activity). Their approach is a form of bounded enumeration and employs techniques originally developed for the solution of assembly line balancing problems.

Collins (1973) has applied graph theory in production scheduling. He presents a work scheduling tool aimed at providing a smoothed schedule of work activities while minimizing manpower or other resources that are necessary to complete the work.

3.5 SIMULATION

In its broadest sense, simulation can be described as the use of a model to experiment with a system. In most applications of simulation to mineral operations, the system is described by a mathematical model and programmed for use on a digital computer. The model generally consists of equations or data sets that describe each element in the system such as a machine or a truck. The mine planner will use the computer model to evaluate different operating plans.

Due to the simplistic, yet realistic nature of simulation, it is the most versatile of the OR methods. Although it is versatile, simulation often requires a great deal of programming work and computer time and is therefore implemented only if a more direct procedure is unavailable.
Simulation is not an optimization method per se but rather a tool which can aid in improving the operation of any portion of a mining operation. In production scheduling, the computer can model the use of alternative production plans to obtain the mineral product. If the model is properly constructed and reflects the actual operation to a satisfactory degree, the output should match the actual mine production. Frequent uses of simulation in production scheduling are to determine the correct number of haulers assigned to an excavator, choosing the best equipment capacity, or assessing the output of a given operating subsystem. Because the simulation is not capable of optimizing, it can only help the analyst find a good if not optimal solution to his problem.

Daud and Pariseau (1975) used simulation in production scheduling. They simulated a hybrid model to evaluate integer programming problems. Manula, Falkie, and Su (1973) have developed a simulator to evaluate production schedules for an entire mining system. Butler and Fouts (1975) have outlined a model to simulate the operation of a shovel-truck system. Many other shovel-truck simulators appear in the literature.

3.6 HEURISTIC METHODS

Heuristic methods are procedures which are not mathematically proven or centered but are based upon practical
or logical operating procedures. The techniques used vary from rule of thumb methods to iterative procedures that move from solution to solution to seek an improved outcome. Heuristic models are common in the solution of mining problems but seldom appear in OR literature.

Agin (1966) provides a generalized description of branch and bound algorithms and their wide applicability to combinatorial problems in general. Stinson, Davis and Khumawala (1978) reported the development of a multiple resource constrained scheduling algorithm using a branch and bound technique. Their algorithm focuses on assigning feasible start points to a set of activities making up a project under two constraint sets: (1) no activity may be started until all activities technologically established as its predecessor set have been completed, (2) the total resource requirements of all activities in process at any time must not exceed the level of availability for each of the multiple resource classes.

Mason and Moodie (1971) present a "branch and bound" algorithm for a project consisting of a set of activities partially ordered by a set of predecessors and restrictions with each activity having a fixed resource requirement and duration. The algorithm determines the activity start times which minimize the combined fluctuations in resource demand and delay of project completion.

Lindstrom (1971) and Kappes (1973) illustrate an
example of heuristic methods used in production scheduling. The computer was used for accounting purposes but was not programmed to make decisions.

West-Hansen, Sarin and Topuz (1986) have suggested a sequencing theory approach for the problem of long-term production planning in an underground coal mine. They have reduced the scheduling problem to the general problem of non preemptive scheduling of precedence constrained jobs on parallel processors. A feasible and near-optimal job schedule is determined given the objective of maximizing the total net present value of the mine. They have suggested the following approach for scheduling n non-precedence related sections that are assigned to one miner. Consider two schedules,

\[ p = (q, y, x, r) \]
\[ p' = (q, x, y, r) \]

where \( x \) and \( y \) are sections, and \( r \) and \( q \) are the subsequences of all other sections. Assuming that both sequences start at time zero, the respective net present values of these schedules are:

\[ v_p = v_q + v_y/a_q^{d_q} + v_x/a_q^{d_q+d_x} + v_r/a_q^{d_q+d_r+d_y} \]
\[ v_{p'} = v_q + v_x/a_q^{d_q} + v_y/a_q^{d_q+d_x} + v_r/a_q^{d_q+d_r+d_y} \]

where \( a = 1+r \), \( d_u \) is the duration of the subsequence, \( u \), and \( v_u \) is the net present worth of the subsequence, \( u \), discounted to the beginning of the sequence. Therefore, for \( p \) to be better than \( p' \),
\[ v_p \geq v_p', \]
\[ \iff (1-1/a^q) v_x \geq (1-1/a^q)/v_y \]
\[ \iff Q_x \geq Q_y \]

where
\[ Q_z = (1-1/a^q)/v_z \]

Consequently, the optimal sequence of the sections when using one miner, can be established by ordering the sections according to non-decreasing Q-values.

Prelaz et al (1973) have developed a computer program (COST-SCHED II) to analyze the cost variables associated with underground coal mining using a pre-determined mine schedule and production plan. This program could be used as an effective engineering tool for generating production and operating data. The earlier program COST-SCHED I was designed to provide the planning engineer with a tool to aid him in the evaluation of a mine after basic decisions concerning the capital costs and manpower requirements have been established. The COST-SCHED II program can be used to schedule the mine and determine the time when each mining unit can be put into operation.

Chatterjee, Scheck and Sridhar (1986) have reported an underground mine planning package for interactive mine design and simulation (ICAMPS - Integrated Computer Aided Mine Planning System). This package allows the user to plan a mine, simulate the mine plan and generate production summary reports and timing maps.
Turpin and Weyher (1976) have developed an interactive coal mine production scheduling and ventilation planning package using a time-shared computer. This package calculates the monthly advance of all production faces through mining panels specified by a mine planner. The results reflect the choice of a mine layout and the sequence of face operations. This package also provides information about potential scheduling conflicts within each mine plan, the monthly haulage requirements in ton-miles for each plan and a monthly coal grade estimate.

Bandopadhyay and Sundararajan (1987) have simulated a longwall development - extraction network. They have discussed the concept of criticality indexing and its application in a simulation model for a longwall development - extraction network. The approach in modelling taken by the authors is based on the Monte Carlo simulation and the model is capable of:

1. finding the criticality index of each activity.

2. allocating available resources to project activities in an attempt to find the shortest project schedule consistent with fixed resource limits; and

3. reducing the peak resource requirements and smoothing out the number of period-to-period assignments, within a constraint on project duration.
In the last decade the tools to handle production scheduling operations have grown in number and sophistication. The computing hardware has become faster and more versatile while the techniques for solving production scheduling problems are more numerous and practical. Further advances are expected in the OR and mathematical methods to help solve production scheduling problems. For example, the techniques of disjunctive kriging and conditional simulation of ore bodies both contribute to evaluating production quality under uncertainty.

A review of the literature suggests that application of computers and OR techniques for scheduling open-pit mines is commonplace, but little progress has been made in underground mining where the scheduling decisions are more complex. One drawback of the available techniques is that they require considerable manual input before a schedule can be generated.

All the OR methods assume that a precedence network, that is, the technological sequences in which the sections can be mined, is readily available. None of the published techniques have an automatic or computer-aided method for generating this information. But developing the precedence network itself is a major task that is very time consuming if done by manual methods. The papers also indicate that there is no single optimum technique to schedule a mine.
CHAPTER 4

MODEL DEVELOPMENT

The complexity of the underground mine scheduling problem has excluded the extensive use of any single technique. Operations research techniques have been applied to mine production problems with different degrees of success. In general, OR has been found to be successful in providing helpful and reliable production data, but the manual input and computing resources requirements discouraged their widespread use.

Most mine operators attempt to maximize the profits, but the analyst may not be able to express this objective function in terms of profit units. Also the assumption of linearity may be a serious impediment that limits the use of the well known LP techniques. Often the analyst faces a dilemma. The choices are between a more complex model which accurately describes reality or a simplistic model which is easy to implement.

Scheduling an underground mine on a computer using integer programming techniques traditionally uses a model with a non-linear objective function. Since the mine has many sections and must be scheduled over many years, the number of decision variables can be tens or hundreds of thousands.
Finally all decision variables are 0-1 variables; a model with these characteristics has been found difficult to solve and therefore impractical to be implemented on a computer.

Formulating the mine scheduling problem as a dynamic program puts the problem into a disciplined solution form that can lead the decision maker to the correct solution. However, since every dynamic program has a different format, it is difficult to apply one dynamic programming solution method to all production scheduling applications. The usefulness of a comprehensive simulation model for production scheduling is limited, due to the large number of runs and extensive computer time required to identify the best production schedule under varying mining conditions.

4.1 EVALUATION OF TECHNIQUES

A review of the current usage of operations research techniques within the mineral industry reveals that the above techniques is are rarely used.

This situation may be due to the following:

1) Lack of Trust: The potential user does not understand the methodologies underlying in the techniques.

2) Implementation problems: The logistics of obtaining the data inputs have not been adequately
addressed in literature.

3) Cost: The cost of obtaining the answer is greater than the value of this information.

4) Timeliness: The answer is frequently obtained too late to help solve the problem.

Sophisticated operations research techniques usually require a substantial investment in computing equipment or time. Because an ore deposit is a naturally occurring phenomena, many of the inputs can only be estimated. When the actual mining conditions differ appreciably from the assumptions, the model must be reformulated to incorporate the new information. This repetitive process increases the cost of using a computerized production scheduling system.

Heuristic methods are common in the solution of mining problems but seldom appear in OR literature. The reason is that these methods are often intuitive and apply to only a particular operation. Because heuristics are easy to implement and provide near optimal results most applications involve some form of a heuristic.

4.2 PROPOSED METHODOLOGY:

Although no single operations research technique has proven adequate for modeling the mine production scheduling problem, a combination of these techniques may yield a satisfactory model. Based on the work of other authors,
network theory and heuristics seem to be the most promising techniques for solving the production scheduling problem.

The scheduling of underground mines can be treated as a multiple resource constrained production scheduling problem. A heuristic technique can be applied to a network that is developed from the mine plan. The difference between a typical project management network and a mine plan is that the arcs can be bi-directional in the mine network.

The potential user of the proposed model is expected to be the mining engineer. The model is based upon the manual scheduling rules that are commonly used by mining engineers. This approach is advantageous because the model is more easily understood than one based on operations research techniques.

The implementation problems faced by earlier researchers were primarily due to the lack of machine readable data. This difficulty was overcome by using ICAMPS which generates the mine layout and sections in a machine readable format. The data input requirements have been greatly simplified by the use of ICAMPS.

The time taken by the model to generate a schedule will be minimal due to the use of heuristic rules which do not require extensive computations. Reduced computation time means that schedules can be generated at a low
cost within a reasonable amount of time.

The heuristic approach treats the problem as resource allocation rather than resource balancing. The solution routine consists of a group of scheduling rules which determine when each job can be started and which resources to assign.

The scheduling rules employed in the model can be summarized as follows:

(1) A section is scheduled provided that its technological constraints have been met, that is, all predecessor sections are already completed before the section is scheduled to start.

(2) The sequence in which the longwall panels will be mined determines the sequence of mining in the development sections.

(3) If two or more development sections are ready to be mined the development sections that are predecessors for the next longwall panel to be mined will be given preference over the other sections.

(4) Machine locations do not play a significant role in the allocation to a section since the move times for the machines are insignificant when compared to the mining time.

(5) As many sections are scheduled as available resources permit. The number of sections scheduled is only limited by the constraint that
development work is undertaken for only two panels ahead of a longwall panel that is currently being mined. Even if machines are available, the sections are not mined due to the large investment involved in the development work. This condition would justify the objective of maximizing the present worth of the mine.

4.2.1 Scheduling Procedure:

The algorithm can be summarized in the following steps:

1) Sections are scheduled, starting with the first one in the precedence network.

2) The most critical sections are the predecessors for the next longwall panel and have the highest probability of being scheduled next.

3) In case of ties, the sections are ordered according to the second criteria - minimizing deviations from the desired production levels.

4) After all planned longwall panels have been scheduled the remaining sections are scheduled based on the third criteria - maximizing future production from the unscheduled areas.
4.2.2 Evaluation of Schedule:

The schedule can be evaluated in terms of its effect upon the related costs. The relevant costs depend upon the type of project, but in general they include:

1. Resource costs (e.g., cost of machinery and idle resources);
2. Overhead costs (auxiliary equipment and the supporting infrastructure);
3. Penalty costs associated with deviations from prescribed production levels.

After the system generates a schedule, the following information is provided:

1. machine utilization.
2. deviation from planned production levels.

Based on this information the user can change the number of machines and re-run the scheduling procedure. Since the penalty costs of not meeting production requirements have a high priority, the production summary will be useful in deciding upon the number of machines. Insufficient production would necessitate the use of more machines. Excess production would require a reduction in the number of machines.

Achieving an optimum schedule using this model would be an iterative process. The user can alter the machine characteristics and production rates to generate a good if not optimal schedule.
A block diagram of the scheduling program is shown in Figure 5. The user must have a working knowledge of ICAMPS to execute the scheduling program because ICAMPS generates the layout and defines the sections in a machine readable form. The individual modules will be explained in the following sections. The program generates the schedule in the form of a timing map and also as a data file. The belt head locations are also displayed graphically while the schedule is being developed. All the user input and data calculated by the program are stored in indexed data files.

The program has been designed to be executed sequentially in separate modules. The smaller modules within the program can be executed independently so that the user can run the program in stages. This approach does not tie up the computer for extended periods of time. Considerable attention has been given to user friendliness and minimization of user input.

There are other utility options in the program, such as drawing the layout and resetting the boundary, which do not appear in the block diagram. A more detailed explanation of these functions appears in Appendix A.
Figure 5: Block diagram of the scheduling program
5.1 Section Definition

The various components of the section definition module are illustrated in Figure 6. The inputs to this module is generated by ICAMPS. The section definition module is a basic step and must be accomplished prior to the other options. Two data files are created by this module; one contains the mining section data and the other containing calendar data. The menu options available within this module are shown in the Figure 6. The module is extremely flexible; it allows the user to add, delete or change records with ease.

5.2 Network Generation:

A flow chart of the network generation module is given in Figure 7. Since the scheduling system uses the output from ICAMPS, the process must follow the logical steps in ICAMPS. Thus a three stage sequential procedure is required.

In the first stage, the program reads the section definition files and creates an area definition file in which each area is assigned a sequential number. Adjacent areas are then identified for each area and details of adjacency are written on the '.ADJ' and '.SID' data files. The steps involved in the identification of adjacent areas for each area are:

1) all areas are sequentially scanned to determine
Figure 6: Section and Calendar definition module
Figure 7: Network Generation module
whether the coordinates of the area are within limits of the area under consideration.

2) the corner points of the areas within limits are then tested to check if they lie on the sides of the area under consideration.

3) the areas which satisfy both the above conditions are treated as adjacent areas.

The sections created at this stage are shown in Figure 3. The adjacency relationships between areas fall into one of the three categories as shown in Figure 8, that is, from a large area to a small area, between equal areas or from a small area to a large area.

![Diagram of adjacency relationships](image)

**Figure 8**: Adjacency relationships between sections

This information is used to determine the network and also to split the areas. There is a path in the network from area 1 to area 2 only if the code is either 0 or 2. Based on user input, the adjacent areas are split to
allow for the precedence constraints at the time of scheduling. Figure 9 (b) is an example of splitting area 3 into three parts numbered 6, 7 and 8. In Figure 9 (b) the areas 1 or 4 can be mined only after sub-area 6 has been mined. All the above information is stored in the three data files which are shown as the network generation module outputs.

The second stage recreates the area definition file based on the areas that were split in the previous stage. The new area definitions for Figure 3 are shown in Figure 10.

The last stage reads the adjacent area information and generates the network. Followers and predecessors for each area or sub-area in the network are also determined. The network of mining sections is determined by a sequential search of the adjacent areas. The network determination procedure is shown in Figure 11. The network starts from the first area, that is, area number 2 in Figure 11. The adjacent areas are then retrieved for this area and the possible branches are shown in stage number 2. The branches for each area in stage 2 are then determined and they form the areas in stage 3. This procedure is continued until all areas have been included in the layout. The network is stored as \((i, j)\) labels in the ".NET" data file.

The values of the arrow \((i, j)\) labels are utilized
Figure 9 (a) : Sample layout before splitting

Figure 9 (b) : Sample layout after splitting
Figure 10: Recreated section definitions
to trace the followers of a particular activity. Sections which have an i label that corresponds to the j label of the section under evaluation are logical followers. The transformation routine recursively traces all possible nodes and determines the sequence of followers. The determination of predecessors is similar to the above mentioned procedure. Figure 12 is a network representation of the sample layout.

Figure 11: Network determination procedure
Figure 12: Network representation of layout in Figure 10.
Based on the adjacency relationships, the potential belt head location coordinates are determined and stored in the '.SID' data file.

5.3 Production Data Input and Tonnage Calculations:

Figure 13 is a flow chart of the functions performed by the production data input and tonnage calculation module. The tonnage calculation section is in ICAMPS. Consult the ICAMPS users manual for further information about the logic. This tonnage calculation module also contains procedures for entering machine specifications and production parameters. The data files created during this stage are used to compute the schedule.

5.4 Schedule Generation and Timing:

This module is the heart of the scheduling program and is illustrated in Figure 14. The majority of the inputs to this module are from ICAMPS and the remaining are from previous steps. The user provides minimal input. Two of the key algorithms within the schedule generation and timing module are described below.

5.4.1 Machine allocation algorithm:

The start date for the schedule is specified by the user. Although the final output represents the schedule in the dd/mm/yy format, the schedule "CLOCK" is expressed
Figure 13: Production data input and calculations
1) Assign mining units to mineable areas.

2) Calculate Set-up times based on retreat mining and moving times.

3) Calculate belt head locations and number of beltheads.

4) Generate timing maps.

Datafile with ".SCH" extension containing the schedule.

Timing data and maps.

Summary of machine utilization and production.

Figure 14: Schedule generation and timing
in days in the future. The method is more convenient for keeping track of data while the program executes and to produce the machine utilization summary. The production levels are checked at intervals specified by the user. If the deviation from required production is greater than the user specified tolerance level, an error message is displayed. Provision has also been made to incorporate the retreat mining time as a set-up time for the continuous miner involved in retreat mining. The user inputs the areas to be retreat mined and their preceding areas. The time taken to retreat mine will be added to the setup time of the continuous miner involved in retreat mining. This procedure will ensure that the start date of mining for each section is correct based on set-up time and retreat mining time of the continuous miner assigned to the section. The program can schedule two different continuous miners on the same area simultaneously. A flow chart of the algorithm is given in Figure 15.

5.4.2 Next Area To Be Mined Algorithm:

The key element of this algorithm is the concept of a "STACK". A stack is an array in which the data can be stored or retrieved. All the mining sections are placed on the STACK in the order of their impact on fulfilling the objectives. The stack must follow the FIFO (First In -First Out) principle, i.e., records are retrieved in
Figure 15: Machine allocation algorithm
Figure 15: Machine allocation algorithm (Continued)
Figure 15: Machine allocation algorithm (Continued)
the same order as they were stored. Predecessors for longwall panels are given the highest priority in the stack. Once the Longwall panels have been mined, the sequence of mining in the remaining sections is determined by the routine SEQ_AREA. This routine selects the next area to be mined. The area that can maximize future production and opportunities to open up new sections has the highest priority. This routine also advances the schedule CLOCK to the time of the next event which is the completion of an area. Figure 16 is a flowchart of this algorithm.

A summary of the machine utilization and deviation from the production requirements are also provided. If the results are not economically feasible, the mine planner would use this report to modify the inputs to generate another schedule. This system also generates timing maps which are used to validate the schedule and graphically check the assignment of machines. The examples are in the next chapter.
START

Are there any sections to be ready to be mined? Yes

Place all branches from this section in STACK

No

Read branches from currently mined area.

Read predecessors for the next Long Wall panel to be mined.

Is there any predecessor among the branches? Yes

Place the other branches in STACK

No

Have all longwalls been mined? Yes

No

Are there any more areas in the STACK? Yes

No
Figure 16: Next Area To Be Mined Algorithm (Continued)

Advance CLOCK to the next event. (Completion of area)

Call routine SEQ_AREA

Return
CHAPTER 6
RESULTS AND CONCLUSIONS

6.1 RESULTS:

A computer program has been developed and implemented on a MICROVAX-II. The coding was done in FORTRAN 77 with graphic and hardcopy interfaces. The graphic outputs can be displayed on Tektronix 4100 series terminals and compatibles. The program can drive Houston Instrument and Calcomp plotters. The program is menu driven to lead the user through the scheduling process. User input is minimal and the menus are self-explanatory. Appendix A describes the capabilities of the program in detail.

Prior to executing the scheduling program the time consuming functions of mine planning and ore quality calculations are done using ICAMPS. ICAMPS also has an optional quicker method of calculating the number of shifts and coal quality without sacrificing the accuracy of the results. The execution time of the scheduling program depends upon the number of sections; most of this time is spent in generating the network and other input data for the machine allocation algorithm.

The summary outputs generated by the program for machine utilizations are shown in Tables 1 and 2. The tables contain the number of machines used, the total number of days worked, the setup times and the number of
days the unit was idle during the scheduling period. Tables 3 and 4 are summaries of the simulated production from the mine and the desired production for a period of twelve months. This information helps the user to evaluate the schedule and to suggest alternative scenarios. Table 3 contains the production results for a schedule that utilizes two continuous miners and one longwall shear. The required production levels could not be obtained by using one continuous miner and one longwall shear. Increasing the number of continuous miners to three did not increase the production appreciably. As might be expected, the percentage utilization of the machines decrease with an increase in the number of mining machines. The timing map for the two continuous miners and one longwall shear alternative is shown in Figure 17. The timing map for the schedule using three continuous miners and one longwall shear appears in Figure 18.

The blocks shown on the timing maps indicate the simulated mining on a monthly basis. The type of mining machine assigned to a section is designated within each block. The letter A represents the first longwall or continuous miner and B represents the second continuous miner. The time required to mine the proposed reserve ore is almost identical for both two or three continuous miners alternatives (approximately 14 years). Due to the precedence constraints among mining sections, increasing
the number of miners did not reduce the total mining time.

The program also generates a datafile containing the location of the various beltheads during the planning horizon. Other modules within ICAMPS use this information for budget preparation or electrical load balancing.

Table 1. Summary of machine utilization (2 continuous miners)

<table>
<thead>
<tr>
<th>Machine #</th>
<th>Total days Worked</th>
<th>Setup time</th>
<th>Idle Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (R/P)</td>
<td>2454</td>
<td>162</td>
<td>1328</td>
</tr>
<tr>
<td>2 (R/P)</td>
<td>1189</td>
<td>111</td>
<td>2356</td>
</tr>
<tr>
<td>1 (L/W)</td>
<td>3038</td>
<td>147</td>
<td>540</td>
</tr>
</tbody>
</table>
Table 2. Summary of machine utilization (3 continuous miners)

<table>
<thead>
<tr>
<th>Machine #</th>
<th>Total days Worked</th>
<th>Setup Time</th>
<th>Idle Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (R/P)</td>
<td>1787</td>
<td>143</td>
<td>1958</td>
</tr>
<tr>
<td>2 (R/P)</td>
<td>1240</td>
<td>74</td>
<td>2271</td>
</tr>
<tr>
<td>3 (R/P)</td>
<td>616</td>
<td>50</td>
<td>1168</td>
</tr>
<tr>
<td>1 (L/W)</td>
<td>3038</td>
<td>145</td>
<td>484</td>
</tr>
</tbody>
</table>

Table 3. Summary of simulated production (2 continuous miners)

<table>
<thead>
<tr>
<th>Period No.</th>
<th>Actual Production</th>
<th>Required Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>269620.000</td>
<td>200000.000</td>
</tr>
<tr>
<td>2</td>
<td>357906.000</td>
<td>200000.000</td>
</tr>
<tr>
<td>3</td>
<td>688106.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>4</td>
<td>871223.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>5</td>
<td>697908.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>6</td>
<td>1016539.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>7</td>
<td>804056.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>8</td>
<td>815131.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>9</td>
<td>915291.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>10</td>
<td>755199.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>11</td>
<td>781741.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>12</td>
<td>818134.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>13</td>
<td>649720.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>14</td>
<td>927810.000</td>
<td>500000.000</td>
</tr>
</tbody>
</table>
Table 4. Summary of simulated production (3 continuous miners)

<table>
<thead>
<tr>
<th>Period No.</th>
<th>Actual Production</th>
<th>Required Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>390802.000</td>
<td>200000.000</td>
</tr>
<tr>
<td>2</td>
<td>401780.000</td>
<td>200000.000</td>
</tr>
<tr>
<td>3</td>
<td>1027365.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>4</td>
<td>700446.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>5</td>
<td>806469.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>6</td>
<td>910142.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>7</td>
<td>750806.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>8</td>
<td>905128.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>9</td>
<td>837310.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>10</td>
<td>682315.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>11</td>
<td>896946.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>12</td>
<td>781977.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>13</td>
<td>693356.000</td>
<td>500000.000</td>
</tr>
<tr>
<td>14</td>
<td>927810.000</td>
<td>500000.000</td>
</tr>
</tbody>
</table>
Figure 16: Timing Map (2 continuous miners and 1 longwall shear)
Figure 17: Timing Map (3 continuous miners and 1 longwall shear)
6.2 CONCLUSIONS

The ability to generate long-term schedules has been demonstrated by this program. The computer generated schedule will be free of conflicts and provides the mining engineer with a tool to experiment with different scenarios. In the manual scheduling process conflicts in the machine assignments are difficult to detect and requires extensive computation. The time intensive task of generating a timing map is also simplified by the program. The time to generate the schedule, once all necessary input is available, is insignificant compared to the manual method. A mining engineer can test different machine combinations and production rates to achieve the best schedule.

Although ICAMPS is essential for generating the input that is needed for scheduling, the ICAMPS output can also serve several other planning purposes. The program has been designed to be executed by any experienced mining engineer and the terminology is easily understood by the user.

The scheduling program is user-friendly and all the scheduling parameters can be changed without affecting the other data. The indexed file structure makes the program execute faster while operating in the interactive mode.
The schedule generated by the program may not be feasible based on other constraints such as ventilation or coal quality and thus may require user intervention to make the schedule practical.

6.3 RECOMMENDATION FOR FUTURE ACTION:

1) The multitude of constraints and requirements for mine production scheduling seems to make the use of an expert system inevitable. Working independently, several mining engineers will invariably come up with a different schedule for the same mine. This result means that there is no standard practice in real-time scheduling of underground mines. The knowledge of several experienced engineers could be consolidated to create a knowledge base for mine scheduling. The various scheduling decisions would become rules and, with some user interaction, an optimum schedule might be generated.

2) Provide options for the user to change the final schedule generated by the program and re-run the timing map generation option. This capability will enable the engineer to incorporate some constraints which cannot be specified within the program.

3) In addition to computing production and quality data, a cost analysis of the proposed schedule could be performed. This feature would require additional
user input to convert the schedule into time based requirements for auxiliary equipment and supplies such as shuttle cars, roof bolters, rails, belts and roof bolts. The equipment and supply requirements data would make the budgetary and procurement functions more effective.

4) With some modification to the program, constraints on quality, power and ventilation requirements could be incorporated into the scheduling process.
BIBLIOGRAPHY


University Park, PA, 1966, pp DD1-DD27.


ings, 14 th APCOM, The Pennsylvania State University, October, 1976.


APPENDIX A
(USER'S MANUAL)

The user of this program must have a working knowledge of the VAX/VMS operating system and ICAMPS. Experience with Tektronix terminals will be helpful in understanding the graphics associated with the program.

Prior to executing this program, the user must use ICAMPS to input the mine layout and design the longwall and room-and-pillar sections. The necessary data files are given in Appendix B.

To execute the program, the user must default to the directory containing the data files and type "RUN SCHEDULE". The introductory screen will appear as shown below.
The name of the mine along with the version number will then be requested. Once a valid mine name has been entered the main menu will appear as shown below.

**Main Menu For Scheduling**

[0] Return to Previous Menu
[1] Define Mineable Areas
[2] Create Network
[3] Edit Previous Input
[4] Run Quality/Shifts
[5] Schedule Areas
[6] Plot Timing Map
[7] Summary of Schedule
[8] Reset Screen Boundary
[9] DCL Command Mode

Enter your option •

Options 1 through 5 are essential for generating the schedule and they must be executed in the sequence in which they appear in the menu. Options 6 through 9 are utility options which can be used after or before a schedule is generated.

**Option 1** Define Mineable Areas

This option allows you to define the sections that are to be scheduled and/or to input the mine calendar data. If you have already executed the section definition option in ICAMPS, you can omit this option. Further details about the various sub-options can be obtained from the ICAMPS User's Manual. If you choose this option, the following menu appears.
INFORMATION FOR MINE PRODUCTION MENU

[0] RETURN TO PREVIOUS MENU
[1] MINE CALENDAR OF WORK DAYS
[2] AVERAGE PRODUCTION DATA FOR EACH AREA
[3] DIVIDE/INTEGRATE AREAS FOR TIMING PURPOSES
[4] DISPLAY AREAS ALREADY IDENTIFIED
[5] DISPLAY AREAS DESIGNED

ENTER YOUR OPTION

OPTION 2 CREATE NETWORK

If you choose this option, the following menu will appear.

NETWORK DETERMINATION STEPS

[0] RETURN TO PREVIOUS MENU
[1] RUN PASS ONE
[2] RUN PASS TWO
[3] RUN PASS THREE
[4] DRAW AREAS ON SCREEN

ENTER YOUR OPTION, PLEASE

The first three choices must be executed sequentially in order to generate the network.
OPTION 2.1                   RUN PASS ONE

This option will read the room-and-pillar and long-wall definition files which were generated by ICAMPS and create an indexed file of area coordinates named with an extension "AC". Each record in this file contains the sequentially assigned area identifying numbers and its coordinates. Based on the newly created areas, the adjacent areas will be determined for all areas. The adjacent areas are stored in the "ADJ" file and the adjacency relationships are stored in the "SID" file. Each record in the 'ADJ' file contains the area identification number, the number of adjacent areas and the corresponding identification numbers. The relation between each pair of adjacent areas is stored in the 'SID' file.

OPTION 2.2                   RUN PASS TWO

Before executing pass two you will be required to input the offset distance to split the areas based on precedence relationships. The areas are split and a new area coordinates file is created. The adjacent areas are determined again for the newly created areas. The potential belt head coordinates are also calculated at this stage.

OPTION 2.3                   RUN PASS THREE

Pass three will generate the network of sections starting from the first area (inputted by user). The net-
work is stored as i-j (from-to sequences) in the "NET" file. The followers for each area in the network are then determined and stored in the "FOL" data file. The predecessors for the longwall areas are calculated and stored in the "PRE" data file.

OPTION 2.4 DRAW AREAS ON SCREEN

This option allows you to either draw the sections on the screen or on the plotter.

OPTION 3 EDIT PREVIOUS INPUT

This option displays the previously identified areas which can be mined in opposite directions and allows the user to modify this list.

OPTION 4 RUN QUALITY/SHIFTS

This option will bring up the following menu:

CALCULATION INFORMATION MENU

[0] RETURN TO PREVIOUS MENU
[1] CALCULATE QUALITY
[2] CALCULATE NUMBER OF SHIFTS
[3] INPUT PRODUCTION DATA
[4] INPUT MACHINE DATA

ENTER YOUR OPTION, PLEASE
Options 4.1 and 4.2 are based on the logic used in ICAMPS and the user must refer to the User Manual for ICAMPS for details.

**OPTION 4.1 CALCULATE QUALITY**

This option calculates the quality of coal in each area using the ".GRD" file and the fast quality determination method.

**OPTION 4.2 CALCULATE NUMBER OF SHIFTS**

This option calculates the number of shifts that will be needed to mine each area and the tonnages that can be obtained. The tonnage data are stored in the ".TON" file.

**OPTION 4.3 INPUT PRODUCTION DATA**

This option should be used to input the production requirements from the mine. The following input table appears and allows you to modify the input data. The data will be stored in the ".PRO" data file.

<table>
<thead>
<tr>
<th>DEFAULT VALUES</th>
<th>VARIABLES</th>
<th>USER INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.00</td>
<td>TOLERANCE PERCENTAGE</td>
<td>10.00</td>
</tr>
<tr>
<td>0.00</td>
<td>TOLERANCE PERIOD(MONTHS)</td>
<td>12.00</td>
</tr>
</tbody>
</table>

DO YOU WANT TO MAKE ANY CHANGES (YES=1 NO=0)
OPTION 4.4

INPUT MACHINE DATA

The machine specifications for the continuous miners and the longwall shears can be entered in this option. If the machine specification data file has already been created then the following menu will appear:

[0] RETURN TO PREVIOUS MENU
[1] RE-ENTER MACHINE DATA
[2] CONTINUE USING OLD DATA

ENTER YOUR OPTION
If you choose to re-enter the data, the Production Information Menu will appear. The production of the machines can be specified as Tons/Shift or Feet of Advance/Shift.

**PRODUCTION INFORMATION MENU**

[0] RETURN TO PREVIOUS MENU  
[1] PRODUCTION GIVEN IN TONS/SHIFT  
[2] PRODUCTION GIVEN IN FEET OF ADVANCE/SHIFT

ENTER YOUR OPTION, PLEASE

The next table will require you to enter the number of continuous miners and longwall shears to be used in the schedule.

**NUMBER OF MINING UNITS (0 TO EXIT)**

<table>
<thead>
<tr>
<th>DEFAULT VALUES</th>
<th>VARIABLES</th>
<th>USER INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>R/P MINERS =)</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>LW MINERS =)</td>
<td></td>
</tr>
</tbody>
</table>

FOR DEFAULT VALUES PRESS RETURN
After you verify your input, you must input the specification for each of the machines as shown in the following example:

**INFORMATION FOR P/Z UNIT # 1**

<table>
<thead>
<tr>
<th>DEFAULT VALUES</th>
<th>VARIABLES</th>
<th>USER INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>MINER UNIT ID (max 4 Nos.)</td>
<td>1.00</td>
</tr>
<tr>
<td>0.00</td>
<td>TONS/SHIFT</td>
<td>500.00</td>
</tr>
<tr>
<td>8.00</td>
<td>MAX. CUT HEIGHT (ft)</td>
<td>7.00</td>
</tr>
<tr>
<td>0.00</td>
<td>MIN. CUT HEIGHT (ft)</td>
<td>5.00</td>
</tr>
<tr>
<td>0.00</td>
<td>STARTING DAY</td>
<td>1.00</td>
</tr>
<tr>
<td>0.00</td>
<td>STARTING MONTH</td>
<td>1.00</td>
</tr>
<tr>
<td>0.00</td>
<td>STARTING YEAR</td>
<td>1987.00</td>
</tr>
<tr>
<td>0.00</td>
<td>SETUP TIME(DAYS)</td>
<td>5.00</td>
</tr>
<tr>
<td>0.00</td>
<td>TRAMMING SPEED(FT./DAY)</td>
<td>1000.00</td>
</tr>
</tbody>
</table>

**OPTION 5**

**SCHEDULE AREAS**

After you have created all the required data files, this option is ready to be executed. If the schedule has already been generated, the following menu will appear:

**SCHEDULING HAS BEEN COMPLETED! **

[0] RETURN TO PREVIOUS MENU
[1] RE-RUN SCHEDULE

ENTER YOUR OPTION
If you decide to re-run the schedule, the following menu will appear:

**DO YOU WISH TO INCLUDE RETREAT CALCULATIONS?**

[0] CONTINUE WITHOUT RETREAT CALCULATIONS
[1] INPUT RETREAT REQUIREMENTS

**ENTER YOUR OPTION**

If you want to include retreat calculations in the schedule, the following series of tables will appear. You will have to specify the mining rate in the retreat sections and the precedence area for each retreat mining section.

**MINING RATE IN RETREAT SECTIONS (0--TO QUIT)**

<table>
<thead>
<tr>
<th>DEFAULT VALUES</th>
<th>VARIABLES</th>
<th>USER INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>MINING RATE =⇒</td>
<td></td>
</tr>
</tbody>
</table>

FOR DEFAULT VALUES PRESS RETURN
### Precendence-Areas for Retreat-Mining Section 1 (0 to exit)

<table>
<thead>
<tr>
<th>Default Values</th>
<th>Variables</th>
<th>User Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>Area (1) to be completed</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>Area (2) to be completed</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>Area (3) to be completed</td>
<td></td>
</tr>
</tbody>
</table>

For default values press return.

Next you enter the start date as shown in the following example.

### Start date of Schedule (0 to return)

<table>
<thead>
<tr>
<th>Default Values</th>
<th>Variables</th>
<th>User Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>Starting day</td>
<td>1.00</td>
</tr>
<tr>
<td>0.00</td>
<td>Starting month</td>
<td>1.00</td>
</tr>
<tr>
<td>0.00</td>
<td>Starting year</td>
<td>1987.00</td>
</tr>
</tbody>
</table>
In the next table you identify areas which were previously mined.

<table>
<thead>
<tr>
<th>AREAS ALREADY MINED</th>
<th>FOR DEFAULT VALUES PRESS RETURN</th>
<th>TO CONTINUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT VALUES</td>
<td>VARIABLES</td>
<td>USER INPUT</td>
</tr>
<tr>
<td>0.00</td>
<td>AREA NUMBER =)</td>
<td></td>
</tr>
</tbody>
</table>

After you verify your data, the outline of the sections and the location of the belt heads are drawn on the graphics screen as the schedule is generated. The belt-head locations are stored in the file ".BELT". The schedule is written on to the ".SCH" data file.

OPTION 6  PLOT TIMING MAP

After the schedule is generated, this option will draw the timing map for the period you previously specified. The output can be obtained on the graphics screen or the plotter. A sample timing map is shown in Figure 17 in Chapter 6.

OPTION 7  SUMMARY OF SCHEDULE

The machine utilization and deviations from the desired production levels summary data is provided by this option. Tables 1 through 4 of Chapter 6 are examples
of the summary data.

**OPTION 8**  
RESET SCREEN BOUNDARY

The screen boundary can be changed by defining a new working area in the following Table.

<table>
<thead>
<tr>
<th>DEFAULT VALUES</th>
<th>VARIABLES</th>
<th>USER INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>LOWER LEFT X</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>LOWER LEFT Y</td>
<td></td>
</tr>
<tr>
<td>1000.00</td>
<td>UPPER RIGHT X</td>
<td></td>
</tr>
<tr>
<td>10000.00</td>
<td>UPPER RIGHT Y</td>
<td></td>
</tr>
</tbody>
</table>

**OPTION 9**  
DCL COMMAND MODE

This option allows you to enter the DCL mode and to execute any DCL command from within the program.
APPENDIX B

The following tables are lists of the various data files and their extensions. The extensions are concatenated with the mine name to obtain the actual file name.

DATA FILES REQUIRED BY THE SCHEDULING PROGRAM

<table>
<thead>
<tr>
<th>EXTENSION</th>
<th>EXPLANATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot; &quot;</td>
<td>Mine layout data file with version number</td>
</tr>
<tr>
<td>PD</td>
<td>Development Picture Directory</td>
</tr>
<tr>
<td>BD</td>
<td>Bleeder Picture Directory</td>
</tr>
<tr>
<td>LD</td>
<td>Longwall Picture Directory</td>
</tr>
<tr>
<td>QY</td>
<td>Quality data for all defined sections</td>
</tr>
<tr>
<td>SE</td>
<td>Section definition data file</td>
</tr>
<tr>
<td>SP</td>
<td>Split areas data file</td>
</tr>
<tr>
<td>CR</td>
<td>Machine specification data file</td>
</tr>
<tr>
<td>.CLD</td>
<td>Calendar data file</td>
</tr>
<tr>
<td>Files.Def</td>
<td>File containing the names of the grid and Calendar data files</td>
</tr>
<tr>
<td>Boundary</td>
<td>File with the boundary or working area definitions</td>
</tr>
</tbody>
</table>
DATA FILES GENERATED BY THE SCHEDULING PROGRAM

<table>
<thead>
<tr>
<th>EXTENSION</th>
<th>EXPLANATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>Area coordinates file</td>
</tr>
<tr>
<td>.ADJ</td>
<td>Adjacent areas file</td>
</tr>
<tr>
<td>.BELT</td>
<td>File containing belt head locations generated during scheduling</td>
</tr>
<tr>
<td>.FOL</td>
<td>File containing the followers for each area</td>
</tr>
<tr>
<td>.GRD</td>
<td>Grid file containing the quality data at the grid points</td>
</tr>
<tr>
<td>.MACH</td>
<td>Individual machine utilization file</td>
</tr>
<tr>
<td>.MAP</td>
<td>File containing the month/year in which each segment will be mined</td>
</tr>
<tr>
<td>.NET</td>
<td>Network stored as i-j labels</td>
</tr>
<tr>
<td>.PRE</td>
<td>File containing predecessors for the longwall areas</td>
</tr>
<tr>
<td>.PRO</td>
<td>File containing production requirements inputted by the user</td>
</tr>
<tr>
<td>.SCH</td>
<td>Scheduling data which is stored as a sequential data file</td>
</tr>
<tr>
<td>.SID</td>
<td>File containing adjacency relationships between areas</td>
</tr>
<tr>
<td>.SUM</td>
<td>Production summary file</td>
</tr>
</tbody>
</table>
.TIM  Timing information file.

.TON  File containing tonnage data.