A Microcomputer Decision Support System For Check Encoder Workforce Scheduling

Dr. Samuel G. Davis, Management Science, Penn State University, University Park

Dr. Edward T. Reutzel, Management Science, Penn State University, University Park

Abstract

As personal computers become more common in the workplace, the opportunity becomes available to place sophisticated decision support systems in the hands of managers. In this paper we describe such a system to be used by bank operations managers to facilitate the difficult task of scheduling employees responsible for processing checks after they have been deposited.

Introduction

Mathematical modelling of workforce staffing and scheduling problems constitutes the basis of many articles in management journals. The research papers generally include an exposition of the mathematical formulation, followed by a case study description where mathematical solutions identify operating schemes which are significantly superior to common past practices. Where the case study was based upon actual circumstances and the researchers were directly involved with the user organization, the calculated benefits are often, at least partially, realized by the user organization. These benefits, however, are frequently short-lived and not transferable to other comparable organizations. Without the direct involvement of competent researchers, the difficulties associated with modifying the mathematical model in response to changing operating characteristics may be overwhelming. In addition, the output containing the model's results may be unintelligible to those other than the program developers.

A microcomputer version of a decision support system, called ProofTime, has been constructed to assist bank managers in the staffing and scheduling of check encoder operators. As reflected in Figure 1 and summarized in an ensuing section entitled Decision Support System Description, there are three components to the scheduling system:

1. Interactive Data Entry (and model formulation),
2. Model Solution, and
3. Simulation Analysis (and report generation).

Figure 1
ProofTime
System Flow
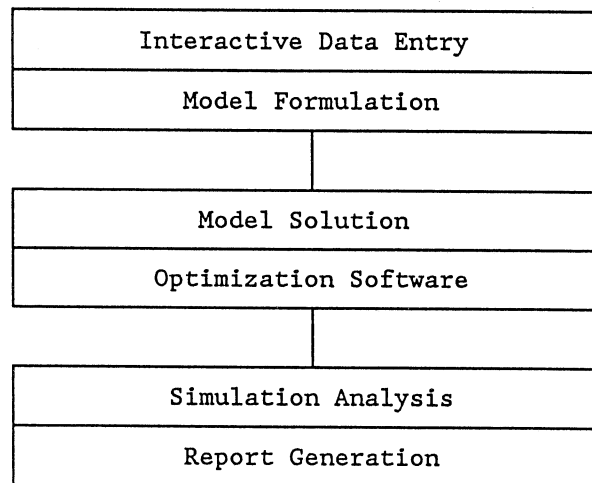
The mathematical model is formulated via a user-friendly interactive program that allows managers to respond to information requests without any knowledge of the mathematical relationships. The model is then, transparently

to the user, passed to a commercially available mathematical programming optimization package to obtain optimal mathematical solutions. These solutions, in turn, are automatically passed to a LOTUS 1-2-3 or Symphony worksheet for user analysis and examination via clear, understandable menus. The link to the familiar personal computer worksheet environment provides numerous benefits, including:

1. The mathematical results are translated into management reports, eliminating the need for mathematical knowledge to interpret and implement defined changes,

2. The user is afforded the opportunity to run "what-if" simulation tests of the proposed solution by varying the data parameters and immediately viewing the effects, and

3. The ProofTime worksheet can be linked to another user-defined worksheet or other system, for purposes of individual employee scheduling, attendance recording, productivity reporting, etc..

Problem Description

Encoder operator scheduling addresses the bank industry problem of determining the staffing and scheduling of individuals responsible for "writing" (encoding) the appropriate dollar amounts on checks in magnetic ink to enable computer capture and processing. The scheduling of check encoders is made complex due to the externally imposed time constraints on the process:

1. Checks are unavailable for encoding until they are deposited by customers, and

2. The completion of the processing task must consider the deadlines for presentation of checks drawn on other banks (transit checks) imposed by the Federal Reserve Bank system, local clearing houses, as well as other channels for check clearance.

The timing of the completion of check encoding is critical because transit checks which are not processed prior to clearing deadlines incur an opportunity cost. In banking language this opportunity cost is often called "float". Float represents the balance credited to a customer's account for which payment from the source bank

has not been received. This amount is therefore unavailable for investment opportunities. Thus, tradeoffs exist between processing efficiency and direct costs and the reduction in float which can be expected as more resources are committed to provide for greater throughput of work prior to clearing deadlines.

Traditionally, bank encoder departments use a mix of full- and part-time employees to respond to the sporadic nature of the hourly check volume distribution. Although the use of part-time employees enables the concentration of processing in the critical hours between the arrival of end of day branch volume and the onset of early evening clearing deadlines, the potential benefits must be compared with the additional equipment requirements, shift differentials, and productivity impact of part-time labor.

Optimum encoder schedules, given a fixed number of available encoding machines, are defined to be those which minimize the sum of encoding operator payroll costs and float costs. The decisions required include start times as well as shift lengths for all workers since:

1. Payroll costs are affected by the mix of full-time and part-time employees, requirements for shift differentials, and overtime, and

2. Float costs are affected by the timing of processing capacity in relation to the timing of check volume deliveries and clearing deadlines.

Appendix 1 contains a mathematical formulation of the problem of determining optimum encoder staffing. Approaches to solving this problem include dynamic programming as proposed by Davis and Reutzel (Spring 1981), heuristic methods described by Krajewski, et al (1980) and Mabert, et al in 1979, and integer programming (see Davis and Reutzel, February, 1981 and Mabert and McKenzie 1980). From a practical perspective, both dynamic programming and heuristic methods have severe limitations. The use of dynamic programming involves a heavy computational burden due to large storage requirements and high execution times. By definition, heuristic approaches are actually "rules of thumb" and provide approximate solutions to the problem. Only with integer programming are optimal solutions obtained directly in a reasonable amount of time.

Even where integer programming computer programs are available, however, the development of solutions for a given operation is a cumbersome process primarily due to the need to enter the appropriate data into the model equations. The complex structure of the model dictates the involvement of a knowledgeable individual or the development of a user-friendly yet sophisticated decision support system for model formulation.

Decision Support System Description

Figure 1 displayed a conceptual flowchart of an encoder staffing decision support system which assists in the definition of the optimal workforce schedule to match the check volume arrival pattern at minimum total cost. This system allows the user to obtain mathematically optimal solutions without any knowledge of the mathematics involved. The schedules obtained minimize the combination of payroll and float costs while recognizing:

1. Variable worker starting times,
2. Variable worker shift lengths,
3. Variable wage rates,
4. Variable productivity rates,
5. Lunch and break policies,
6. Machine availability limitations, and
7. Check clearing deadlines.

In addition, a simulation capability is provided to examine "what if" variation of any of the key operating parameters. This system has been developed to operate on IBM or compatible personal computers. The three building blocks in Figure 1 consist of:

1. A compiled BASIC program to guide the user through the data entry procedure as well as provide the ability to revise previous entries. When all data have been entered the appropriate integer programming formulation of the problem is generated.

2. The formulation is passed to a version of XA (a commercially available mathematical programming system marketed by Sunset Software Technology [7]).

3. The original data and the optimal mathematical solution are passed to a LOTUS 1-2-3 or Symphony worksheet template which contains provisions via macro menus to examine the implications of varying data parameters as well as generate managerial reports.

The specific features are best described through the use of an example case study, where system input and output can be observed.

Case Study Analysis

An operations management classroom case written by Mabert & Showalter (1984)[6] asks that students develop a daily encoding operator schedule for Centurion Bank. The following information is given:

1. A check volume arrival profile by hour from 12 noon to 8 P.M..

2. Checks not encoded by 7 P.M. will not be processed in time for the check clearing dispatch.

3. Two workforce categories; full-time and part-time workers. The two categories have differing wage rates and processing rates. In addition full-time operators work a total of 9 hours with a one hour lunch break and two 15 minute coffee breaks while part-time operators work 4 hours and receive one 15 minute coffee break. All shifts are scheduled to begin on the hour.

4. 37 encoding machines are available.

5. Transit checks comprise 25 percent of encoding volume, the average check size is $350, and the cost of capital is 5.4 percent.

In the discussion to follow, the ProofTime system is described and demonstrated using the Centurion Bank data. The data entry procedure, tabular and graphical reports reflecting the optimum solution, and opportunities for performing "what if" simulation analyses are presented. Finally, a comparison to the solution contained in the Mabert and Showalter (1984) Teaching Note is offered.

System Input

Through a series of five input screens, the user answers simple questions at the keyboard to initiate the process. (All user responses are printed in **boldface**.) Input screen 1 (Figure 2)

contains the basic parameter data described above for Centurion Bank. Input screen 2 (Figure 3) shows the check volume arrival pattern. The user is prompted for input for each interval between the first and last volume availability indicated on screen 1. Since the response to item 3 on screen 1 indicated that there were two worker categories, screen 3 (Figure 4) prompts for the items required to fully describe the first worker category and screen 4 (Figure 5) prompts for the possible shift starting times and minimum and maximum number to assign each shift. (Similar screens would appear for worker category 2.) Unless overridden through the keyboard, the minimum assigned will be zero and the maximum will be the number of machines available. Finally, on screen 5 (Figure 6) the clearing deadlines are identified.

Figure 2

SCREEN 1

```
Encoder Scheduling Data Capture
```
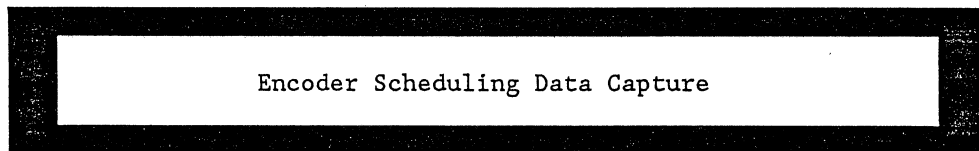
```
        ProofTime - Encoder Scheduling System

                 Type '?' for Help

     1. Timing of First Volume Availability   1200
     2. Timing of Last Volume Availability    2000
     3. Number of Worker Categories              2
     4. Number of Machines                      37
     5. Number of Processing Cutoffs             1
     6. Average Check Size ($)                 350
     7. Cost of Capital                       .054

     All ok? Y or N [.]
```
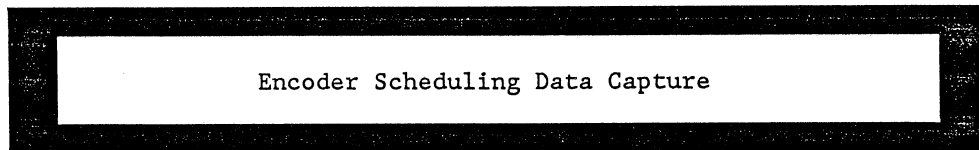
Figure 3

SCREEN 2

```
Encoder Scheduling Data Capture
```

```
          Volume Available by Hour (Items)

     Volume Available by 1200   32430
     Volume Available by 1300    8530
     Volume Available by 1400    5340
     Volume Available by 1500   28530
     Volume Available by 1600   31630
     Volume Available by 1700   47780
     Volume Available by 1800    6925
     Volume Available by 1900   10230
     Volume Available by 2000    2420

     All ok? Y or N [.]
```
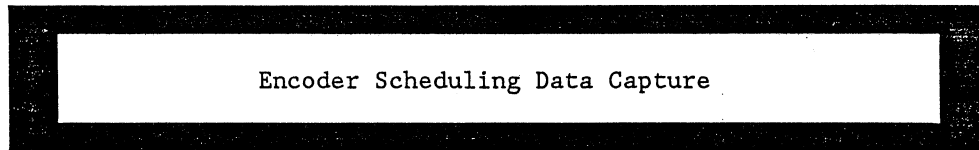
Figure 4

SCREEN 3

Encoder Scheduling Data Capture

Encoding Operator Category No. 1
Type '?' for Help

1.	Description	Full-Time
2.	Hourly Processing Rate	900
3.	Hourly Payroll Cost	3.56
4.	Shift Length	9
5.	No. of Potential Start Times	5
6.	No. of Breaks (incl. Lunch)	3
7.	1st Break Hour	2
8.	Break 1 (minutes)	15
9.	2nd Break Hour	5
10.	Break 2 (minutes)	60
11.	3rd Break Hour	7
12.	Break 3 (minutes)	15

All ok? Y or N [.]

Figure 5

SCREEN 4

Encoder Scheduling Data Capture
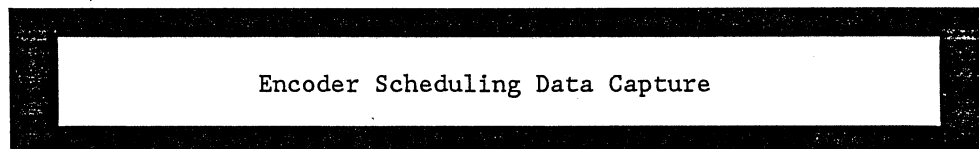
Encoding Operator Category 1 Full-Time

$3.56   900. items/hr

5 Shift Starting Times

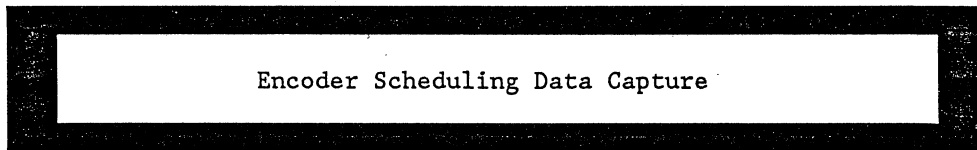| Start Time 1 | 1100 | Minimum Assigned: | 0 | Maximum Assigned: | 37 |
Start Time 2	1200	Minimum Assigned:	0	Maximum Assigned:	37
Start Time 3	1300	Minimum Assigned:	0	Maximum Assigned:	37
Start Time 4	1400	Minimum Assigned:	0	Maximum Assigned:	37
Start Time 5	1500	Minimum Assigned:	0	Maximum Assigned:	37

All ok? Y or N [.]

System Output

The model generated by the front-end capture program is passed to and solved by an integer programming optimization package (XA) developed by Sunset Software Technology and specially adapted to the encoder scheduling problem. However, all mathematical output is suppressed and ProofTime automatically transfers control to a LOTUS worksheet for analysis and reporting via

98

Figure 6

SCREEN 5

```
┌──────────────────────────────────────────────────────┐
│                                                      │
│            Encoder Scheduling Data Capture           │
│                                                      │
└──────────────────────────────────────────────────────┘
```

Processing Cutoff Data

Processing Cutoff 1 1900 % of Total Item Volume for this Cutoff:25

    All ok? Y or N [.]

macro menus. The ProofTime worksheet main menu is shown in Figure 7.

Figure 7
ProofTime Main Menu

COMMAND	COMMAND DESCRIPTION
Import	Import the Data for Encoder Cost and Throughput Analysis
Encoder	Review and Analyze the Encoder Schedule and Costs
Throughput	Review the Throughput Analysis Report
Save	Save the Current Worksheet
Retrieve	Retrieve Another Worksheet
Quit	Exit the ProofTime System

The **Import** command imports the optimal mathematical solution and the parameter data and **Encoder** shows the Encoder Operator Schedule and Cost Report (Figure 8) reflecting the staffing pattern and associated costs. Displayed with this report is the Encoder submenu which allows the user to generate printed output, perform "what if" analyses, and view the costs in graphical form.

Selecting **Throughput** displays the Throughput Analysis Report (Figure 9) reflecting when the volume is available for processing, the capacity for processing in each time interval as a result of the staffing plan, and the items processed and backlogged at the end of each time interval. This report is also accompanied by a submenu allowing for printed output, "what if" analysis, and a graphical display of the operating plans. Figure 10 contains a graph of the volume processed and backlogged in each time interval for Centurion Bank.

From a decision support perspective, it is important to note that the Encoder and Throughput Reports are linked together. Therefore the user may immediately examine not only the effect on costs of changes to wage rates or the staffing complement but also the effect on volume throughput. In addition, it is also possible to assess the impact on throughput of variations in the check volume availability profile.

Solution Comparison

The ProofTime system was applied to the Centurion Bank data, yielding the optimum staffing plan shown in Figure 8. Since the clearing deadline was 7 P.M. (1900 in military time), the float cost incurred is a result of the 9,590 items available but unprocessed at 7 P.M. Figure 11 contains a comparison of this solution to the "best known solution" given in the Teaching Note to [6]. The textbook solution required one additional part-time operator yet still resulted in a higher number of

99

Figure 8

ProofTime

Encoder Operator Schedule and Costs for
Monday, September 22, 1986

Worker Category	Hourly Wage	Number of Workers	Cost
Full-Time	$3.56		*
1100 - 2000		8	$256
1200 - 2100		7	$224
1300 - 2200		1	$32
1400 - 2300		0	$0
1500 - 2400		0	$0
Part-Time	$3.50		*
1100 - 1500		0	$0
1200 - 1600		0	$0
1300 - 1700		6	$84
1400 - 1800		1	$14
1500 - 1900		21	$294
1600 - 2000		0	$0
1700 - 2100		0	$0
1800 - 2200		0	$0
1900 - 2300		0	$0
2000 - 2400		0	$0

Payroll Cost	$904
Float Cost	$126
Total Cost	$1,030

Figure 9

ProofTime

Throughput Analysis for Monday, September 22, 1986

Time of Day	Items Available	Capacity Available	Items Processed	Backlog	% Capacity Utilized
1100 - 1200	0	7,200	0	0	0%
1200 - 1300	32,430	11,700	11,700	20,730	100%
1300 - 1400	8,530	17,025	17,025	12,235	100%
1400 - 1500	5,340	18,025	17,575	0	98%
1500 - 1600	28,530	26,625	26,625	1,905	100%
1600 - 1700	31,630	24,025	24,025	9,510	100%
1700 - 1800	47,780	27,100	27,100	30,190	100%
1800 - 1900	6,925	27,525	27,525	9,590	100%
1900 - 2000	10,230	14,175	14,175	5,645	100%
2000 - 2100	2,420	7,200	7,200	865	100%
2100 - 2200	0	900	865	0	96%
2200 - 2300	0	0	0	0	100%
2300 - 2400	0	0	0	0	100%
Total	173,815		173,815	0	

100
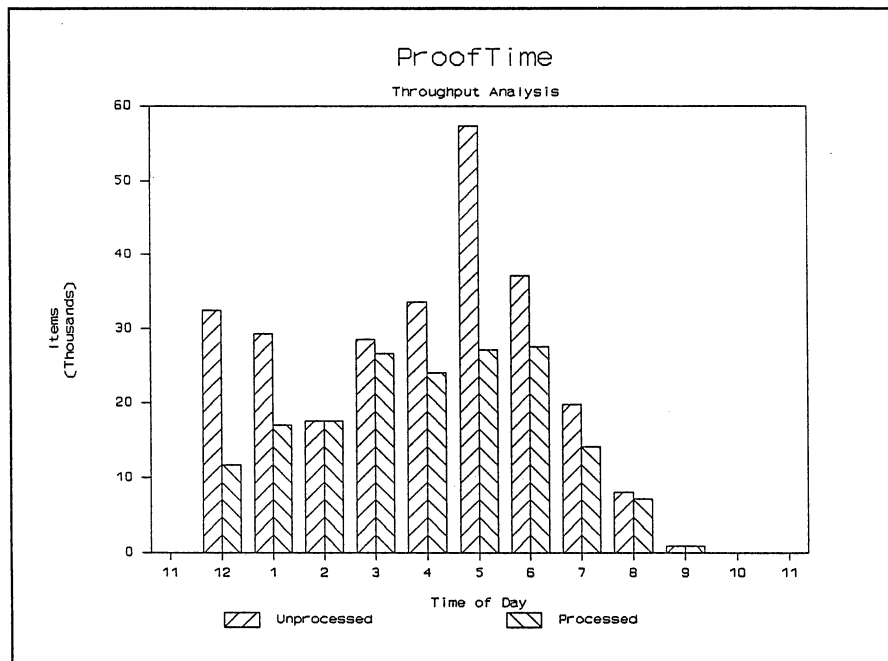
Figure 10

Figure 11

SOLUTION COMPARISON

Worker Category	Hourly Wage	ProofTime Solution Number of Workers	Cost	Textbook Solution Number of Workers	Cost
Full-Time	$3.56		*		
1100 - 2000		8	$256	7	$224
1200 - 2100		7	$224	9	$288
1300 - 2200		1	$32	0	$0
1400 - 2300		0	$0	0	$0
1500 - 2400		0	$0	0	$0
Part-Time	$3.50		*		
1100 - 1500		0	$0	0	$0
1200 - 1600		0	$0	0	$0
1300 - 1700		6	$84	7	$98
1400 - 1800		1	$14	0	$0
1500 - 1900		21	$294	21	$294
1600 - 2000		0	$0	0	$0
1700 - 2100		0	$0	0	$0
1800 - 2200		0	$0	0	$0
1900 - 2300		0	$0	0	$0
2000 - 2400		0	$0	1	$14

Payroll Cost $904 $918

Unprocessed Items @ 7 PM   9,590   10,340

Corresponding Float Cost $126 $135

Total Cost $1,030 $1,053

101

unprocessed items(10,340) at the 7 P.M. clearing deadline.

Summary

As previously indicated, the development of solutions for math programming problems can be a cumbersome and tedious process due to the need to enter the appropriate data into the model equations. The structure of the encoder scheduling problem requires that time "periods" be defined for analysis. Usually the time periods are either hour or one-half hour intervals and two constraints must be created for each time period in the horizon: (1) An inventory balance relationship (Constraint set (2) in the Appendix.); (2) A machine availability restriction (Constraint set (3) in the Appendix).

The structural complexity in generating this formulation is characterized by the a_{ijt} elements in constraint set (2) and the e_{ijt} elements in constraint set (3). Non-zero a_{ijt} and e_{ijt} entries reflect which worker categories and shift numbers are active in each time period. Naturally, $e_{ijt} = 1$ if worker category i, shift number j is active in time period t. The a_{ijt} elements indicate the processing rate per period for worker category i on shift number j in time period t. This processing rate, however, must be adjusted for those periods where the given worker category-shift number is assigned to coffee and lunch breaks.

Thus, although the problem formulation requires only an objective function and two constraints per period, the complex structure dictates the involvement of a knowledgeable individual or the development of a user-friendly yet sophisticated matrix generator for model formulation. Furthermore, revising the formulation to account for reparameterization or redefinition of the options to be investigated requires not only the revision of the coefficient values but may change the structure and number of equations as well. Obviously, care must be taken that clerical errors do not invalidate the results obtained. For data validation purposes the need exists for the active involvement of a math programmer who is familiar with the formulation.

In this paper, a decision support system has been presented which mitigates the concerns mentioned above and goes significantly beyond previous research published on the encoder scheduling problem. The tedious task of problem formulation and revision has been overcome by providing an interactive facility to capture the operating system parameters and automatically generate the mathematical model. Bank managers can make use of the mathematical sophistication and directly obtain optimal results without any knowledge of the mathematics. Moreover, the mathematical results are presented in banking language in the environment of the familiar LOTUS worksheet. In the ideal, the mathematical programming solution provides the starting point for managerial evaluation. Even where the assumptions required by an optimization technique are violated, the results can be expected to at least place the user in the "neighborhood of optimality" prior to embarking upon "what if" comparisons.

A final note for optimism is the fact that the system has been developed to run on a micro-computer. Since the personal computer has become second nature to many bank operations officers, start up time is minimized. As demonstrated by the solution to the Centurion Bank case, realistic-sized problems are well within the capacity of the micro-computer (the elapsed time required for a branch and bound solution to the problem was a mere 15 minutes on an IBM XT with an 8086 processor).

REFERENCES

1 Davis, S.G. and Reutzel E.T. Joint Determination of Machine Requirements and Shift Scheduling in Banking Operations. *Interfaces*, 11, 1(February, 1981), 41-42.

2 Davis, S.G. and Reutzel, E.T. A Dynamic Programming Approach to Work Force Scheduling with Time-Dependent Performance Measures. *Journal of Operations Management*, 1, 3(Spring, 1981), 165-171.

3 Krajewski, L.J., Ritzman, L.P. and McKenzie P. Shift Scheduling in Bank Operations: A Case Application. *Interfaces*, 10, 2(April 1980), 1-8.

4 Mabert, V.A., Fairhurst, R. and Kilpatrick, M.A. Chemical Bank's Encoder Daily Shift Scheduling System. *Journal of Bank Research*, 10, 3(Autumn, 1979), 173-180.

5 Mabert, V.A. and McKenzie, J.P. Improving Bank Operations: A Case Study at BancOhio/Ohio National Bank. *Omega-The International Journal of Management Science*, 8, 3(1980), 345-354.

6 Mabert, V.A. and Showalter, M.J. *Cases in Operations Management*, Plano, Texas, Business Publications Inc., 1984. Centurion National Bank (A) -Worker Scheduling for Back Office Operations, 243-250.

7 *Professional Linear Programming System*, Sunset Software Technology, 1613 Chelsea Road, Suite 153, San Marino, CA 91108.

APPENDIX 1 - ENCODER SCHEDULING PROBLEM FORMULATION

The model presented below reflects a typical day in the operation of a check encoder department in a commercial bank. The index sets with variable/parameter definitions and the model equations with explanations are given below.

Index Sets:

i indexes the worker category
j indexes the shift number, and
t indexes the time period

Variables:

x_{ij} = number of workers in category i assigned to shift j
y_t = number of unprocessed items at the end of time period t

Parameters:

a_{ijt} = processing rate (items/period) for worker i on shift j in period t
b_t = number of items arriving in period t
c_{ij} = cost for worker i on shift j
d_t = opportunity cost of an unprocessed item at time t
e_{ijt} = 0-1 element indicating the availability of worker i on shift j in period t
m_t = number of machines available in period t
T = number of time periods in the work horizon

The objective is to minimize total daily cost (function (1)), subject to constraints on maintaining a balance of inventory (constraint set (2)), machine availability restrictions (constraint set (3)), and guaranteeing that all work is completed (constraint (4)). The model is:

$$\text{MINIMIZE} \qquad \sum_i \sum_j c_{ij} x_{ij} + \sum_t d_t y_t \qquad\qquad (1)$$

The x_{ij} variables are introduced to represent each possible worker category (i) and shift number (j) where the worker category reflects a certain wage, productivity rate, and shift length and the shift number indicates the shift length for the worker category. The coefficients c_{ij} reflect the cost of a worker for a typical individual assigned to that shift.

The y_t variables refer to the number of checks not encoded at the end of each time period t. (Usually the time periods in the analysis are hour or one-half hour intervals.) These variables must appear in the objective function for each time period that represents a cutoff to make a clearing deadline, since there is a cost associated with those checks not encoded at that time. The coefficients of y, are calculated as follows:

d_t = (average check $ size) * (cost of capital) * (% of checks available for clearance at this deadline)

$$\text{SUBJECT TO: } \sum_i \sum_j a_{ijt} x_{ij} + (y_t - y_{t-1}) \qquad \geq \quad b_t \quad t = 1,...,T \qquad (2)$$

In constraint set (2), the right hand side values (b_t), represent the number of checks available for encoding in time period t. Thus, the number of checks received prior to the beginning of the time horizon and prepared for encoder processing would be entered as b_1 with subsequent values of b corresponding to the number of checks that have arrived for encoding during that period. The a_{ijt} elements would normally indicate the processing rate per period for the given worker category. However, the processing rate per period must be adjusted for periods where the given worker

category, shift number is assigned to coffee and lunch breaks.

$$\sum_i \sum_j e_{ijt} x_{ij} \qquad \leq \qquad m_t \qquad t = 1,...,T \qquad (3)$$

In constraint set (3), the number of machines available for assignment to operators in that time period are represented by the right hand side values (m_t). Non-zero e_{ijt} entries reflect which worker categories and shift numbers are active in each time period. Naturally, $e_{ijt} = 1$ if worker category i, shift number j is active in time period t.

$$y_T \qquad = \qquad 0 \qquad (4)$$

Constraint (4) ensures that all checks have been encoded by the end of the planning horizon.

$$x_{ij} \qquad \geq \qquad 0 \text{ and integer}$$

$$y_t \qquad \geq \qquad 0$$

Obviously the variables representing people must take on whole numbered values to have a practical meaning.