Bureau of Business Research
Graduate School of Business Administration
University of Michigan

October 1972

CASH MANAGEMENT MODEL FOR THE CITY OF ANN ARBOR, MICHIGAN

Working Paper No. 65

by

Joseph D. Vinso
Research Fellow

CONTENTS

TABLES

Preface

This paper is written as a user's guide to accompany a cash management model constructed for the City of Ann Arbor, Michigan. The model was designed to help the city improve its management of cash flows. Expenses occur at a fairly uniform rate; income is highly segmented yet, to some extent, predictable. The city is experiencing problems of overdrawn accounts and fund manipulation in months with low revenues. Therefore, the city enlisted the services of the Bureau of Business Research to provide guidance in better management of its cash flows. The model was originally intended for use in managing the General Fund, but it became apparent that it could be constructed for use in any fund and/or the city's total cash flow. As a matter of fact, the sample problem is applied to the city's cash flow from all sources. Thus the model discussed here is a fairly general one, with some specific uses indicated.

This guide is accompanied by a printout of the computer program and simplex tableaus, as well as a card deck, as supplementary materials; these items are referred to in this paper but are not included in it.

A note of thanks is extended to Pat Larkey, Program Manager of Community Planning and Management, who helped initiate and sponsor this project; Ken Sheehan, Assistant Administrator--Finance for the city who gave his support and provided input every step of the way; and Lorne Yedle, City Controller, who helped prevent this project from straying too far from reality.

I

Introduction

Project objectives

This report is intended to serve as a user's guide to a model for use in the cash management function of the City of Ann Arbor. The model was developed under the guidelines of a proposal prepared in March, 1972, and subsequently revised. The general project objective is to develop and implement analytical procedures for managing the city's cash flow. Expected benefits from the project include:

a) Greater return on investment through the more efficient use of revenue dollars which are collected but not needed immediately to meet expenditure requirements

b) Fewer problems with cash shortages during the fiscal year and avoidance of penalties incurred in liquidating investments before they mature

c) Pinpointing of areas where improved procedures for forecasting revenues and expenditures are needed

A specific objective of the cash management model is to maximize income through the financing function (primarily the investment portfolio) subject to the following constraints:

1. All governmental functions are funded when necessary.
2. Bank accounts are not overdrawn.
3. All legal requirements are satisfied.

The model has been constructed with these objectives in mind.

Current practice

At the present time excess cash is invested subjectively by the

Deputy Controller according to his estimates for future cash needs. This practice causes two types of problems. First, although the estimates are developed from experience, the process is complex and there may be times when all costs and cash needs are not recognized. Second, if the Deputy Controller leaves his job, the investment function is less than optimally performed until the new Deputy Controller obtains the experience needed for this complex task.

Two resources are utilized to provide for unexpected cash needs: (1) bank accounts are overdrawn and funds are shuffled from one account to another; (2) investments are sold early at a potential capital loss or money is left idle in anticipation of future needs. Either alternative entails real or opportunity costs. Furthermore, keeping track of over 50 funds and 100 bank accounts is a formidable task.

Thus it appears that a cash management model would be useful in aiding the investment officer (Deputy Controller) to fulfill his function more efficiently.

Outline of the report

Part II describes the model. The assumptions used in forming the objective of the program and various constraints on it are outlined. Sources of data are also discussed.

Part III describes in detail the derivation of the numbers used in the model as well as the equation systems. Terms are defined and the model is constructed.

Part IV presents a sample problem to facilitate understanding of the model. The problem presented is the situation faced by the city with some cash flow patterns assumed.

Finally, Part V indicates the role of the model in the investment process, suggestions for model revisions, and ways of updating the model. Then some suggestions on implementation are provided.

II

Model Description

The purpose of a cash management model is to suggest actions for the investment of cash. It must include alternatives, because if no alternatives exist there is no problem. Such a model essentially determines an investment schedule which will maximize the earnings of available cash. It must be emphasized that no model, no matter how sophisticated, can preclude managerial judgment. The model should only serve as a guide for action. Its validity will be determined by the validity of the data and assumptions used in its construction.

The cash management model designed for the City of Ann Arbor is based on a linear programming algorithm. Thus the linear programming format is used in the description of the model. There are basically two parts to the model: (1) the objective function, and (2) the constraints.

Objective function

The objective of the cash management program is to maximize income through investments. The realization of this objective is dependent on the types of securities held, their maturities, and the amount of each type held. Consequently, some assumptions on securities and time horizon are needed.

As a municipal government the City of Ann Arbor is limited by law as to what types of securities it may include in its portfolio. These include short-term issues of the United States Government (primarily Treasury bills) and time deposits in commercial banks in the form of Certificates of Deposit (CD) up to one year. It is questionable whether other government securities are eligible, but if they are, they too should be included.

The following assumptions pertain to the time horizons of the model. Since the city operates on a fiscal year budgetary system, the model should have at least a twelve-month horizon. A longer one is not necessary because a twelve-month horizon allows constant updating, i.e., planning would be for twelve months into the future. The model will allow for a shorter horizon, but initially the twelve-month period seems appropriate. Furthermore, a monthly model allows for sufficient detail without excessive calculation.

Several assumptions must be made about maturity of securities. Treasury bills are available on a 3-, 6-, 9-, or 12-month basis from the Federal Reserve. But using these intervals in the model assumes that Treasury fundings occur at the same time the cash to be invested becomes available, which is not necessarily a good assumption. Therefore it is assumed that Treasury bills would be bought from securities dealers (such as a bank) at a rate quoted in the financial section of the New York Times or the Wall Street Journal, which allows the purchase of a bill with a maturity for any time period, i.e., 1-, 2-, . . . 12-month bills.

Time deposits at commercial banks pose a special problem. The usual instrument is the CD. Unless the CD amounts to more than \$100, 000, it is not negotiable, which means that usually interest is lost if it is cashed in early.

It is assumed that for a given period a CD will be purchased to mature in that period and cannot be sold early. So a CD represents liquid funds only in the period in which it matures. Liquidity is handled internally in the model through the constraint functions discussed later. The return for a CD is just a multiple of its one-period return, e.g., a 60-day (Period 2) CD has a return of twice a 30-day (Period 1) CD. The CD provides a direct alternative to Treasury bills.

Thus the objective function is as follows:

$$\text{Max}Z = \sum_{t=1}^{12} \sum_{i=1}^{2} C_{it} X_{it}$$

where C_{it} = return for security i for time t
X_{it} = amount held in each security

As a result the model attempts to search combinations of yield and maturity and to choose the set that results in the greatest amount of income.

Constraints

The maximization process has limits within which it can operate. The first type of constraint involves period cash flows. For each period the cash from revenue plus maturing investments must equal the expenses paid in cash, cash invested, plus change in cash balances. Since there are twelve periods (months), there are twelve cash flow constraints.

For example, in the first period various investments are made, including the possibility of Treasury bills and/or CD for Period 2. In Period 2 it is necessary to specify cash revenues, cash expenses, and the return from the Treasury bills and/or CD invested in the first period. Any residual would be reinvested in the period.

Another element in the cash flow constraint is the beginning and ending cash balances. These can be determined by the model or set as a policy decision. In Part IV the sample problem is outlined and the cash balances are set as a result of policy decisions. Cash balances can be determined by the model, but since they have a zero rate of return it is unlikely that they will appear in the solution unless some positive benefit can be assigned. Thus the constraints for the period cash flows are as follows:

$$\sum_{k=j+1}^{12} \sum_{j=1}^{11} \sum_{i=1}^{2} X_{ijk} - \sum_{k=j+1}^{12} \sum_{j=1}^{11} \sum_{i=1}^{2} A_{ijk} X_{ijk} + b_i - b_{i-1} = R_i - E_i = CF_i$$

where: X_{ijk} = amount invested in security i in period j to mature in period k

A_{ijk} = return on security i which was bought in period j and matures in period k

b_i = cash balance at the end of period i
b_{i-1} = cash balance at the beginning of period i
R_i = cash from revenue in period i
E_i = cash expenses in period i
CF_i = cash flow in period i

III

Model Construction

The primary problem in construction of the model concerns determination of the correct coefficients for the general equations outlined in Part II. The purpose here is to describe in detail the derivation of the numbers used in the model as well as the equation systems. First, terms are defined. Then the construction of constraints is reviewed, and finally construction of the objective function is given.

Definitions of variables

In the description of the model, variables were described by using subscripts such as X_{ijk} where k signified the maturity of the security, j signified the purchase period of the security, and i signified the type of security. Since the linear programming algorithm does not recognize subscripts, a method of describing variables compatible with the program was needed.

Table 1 shows the method used to adapt the model to the program. The amount of funds invested in Treasury bills in Period 1 maturing in Period 2 is identified as X_1. A CD bought in Period 2 for maturity in Period 5 is X_{34}. A 4-month CD bought in Period 5 for maturity in Period 9 is X_{83}. A similar method can be used for all variables up to X_{144}.

Certain variables, such as X_{21}, X_{22}, X_{41}, X_{58} etc. (indicated by b_0, b_1, b_2, b_3 etc.) represent the cash balance at the end of each

TABLE 1

Variable and Coefficient Specifications

Program Variable	Model Variable	D_i	$1 + D_i = d_i$
x_1	x_{211}	.00280	1.00280
x_2	x_{311}	.00614	1.00614
x_3	x_{411}	.00957	1.00957
x_4	x_{511}	.01320	1.01320
x_5	x_{611}	.01700	1.01700
x_6	x_{711}	.02120	1.02120
x_7	x_{811}	.02470	1.02470
x_8	x_{911}	.02940	1.02940
x_9	x_{1011}	.03340	1.03340
x_{10}	x_{1111}	.03730	1.03730
x_{11}	x_{212}	.00340	1.00340
x_{12}	x_{312}	.00680	1.00680
x_{13}	x_{412}	.01062	1.01062
x_{14}	x_{512}	.01402	1.01402
x_{15}	x_{612}	.01742	1.01742
x_{16}	x_{712}	.02124	1.02124
x_{17}	x_{812}	.02464	1.02464
x_{18}	x_{912}	.02804	1.02804
x_{19}	x_{1012}	.03186	1.03186
x_{20}	x_{1112}	.03526	1.03526
x_{21}	b_0	0.0	1.00
x_{22}	b_1	0.0	1.00
x_{23}	x_{321}	.00280	1.00280
x_{24}	x_{421}	.00614	1.00614
x_{25}	x_{521}	.00957	1.00957
x_{26}	x_{621}	.01320	1.01320
x_{27}	x_{721}	.01700	1.01700
x_{28}	x_{821}	.02120	1.02120
x_{29}	x_{921}	.02470	1.02470
x_{30}	x_{1021}	.02940	1.02940
x_{31}	x_{1121}	.03340	1.03340
x_{32}	x_{322}	.00340	1.00340
x_{33}	x_{422}	.00680	1.00680
x_{34}	x_{522}	.01062	1.01062
x_{35}	x_{622}	.01402	1.01402
x_{36}	x_{722}	.01742	1.01742
x_{37}	x_{822}	.02124	1.02124
x_{38}	x_{922}	.02464	1.02464

TABLE 1 (Cont.)

Variable and Coefficient Specifications

Program Variable	Model Variable	D_i	$1 + D_i = d_i$
x_{39}	x_{1022}	.02804	1.02804
x_{40}	x_{1122}	.03186	1.03186
x_{41}	b_2	0.0	1.00
x_{42}	x_{431}	.00280	1.00280
x_{43}	x_{531}	.00614	1.00614
x_{44}	x_{631}	.00957	1.00957
x_{45}	x_{731}	.01320	1.01320
x_{46}	x_{831}	.01700	1.01700
x_{47}	x_{931}	.02120	1.02120
x_{48}	x_{1031}	.02470	1.02470
x_{49}	x_{1131}	.02940	1.02940
x_{50}	x_{432}	.00340	1.00340
x_{51}	x_{532}	.00680	1.00680
x_{52}	x_{632}	.01062	1.01062
x_{53}	x_{732}	.01402	1.01402
x_{54}	x_{832}	.01742	1.01742
x_{55}	x_{932}	.02124	1.02124
x_{56}	x_{1032}	.02464	1.02464
x_{57}	x_{1132}	.02804	1.02804
x_{58}	b_3	.00340	1.00340
x_{59}	x_{541}	.00280	1.00280
x_{60}	x_{641}	.00614	1.00614
x_{61}	x_{741}	.00957	1.00957
x_{62}	x_{841}	.01320	1.01320
x_{63}	x_{941}	.01700	1.01700
x_{64}	x_{1041}	.02120	1.02120
x_{65}	x_{1141}	.02470	1.02470
x_{66}	x_{542}	.00340	1.00340
x_{67}	x_{642}	.00680	1.00680
x_{68}	x_{742}	.01062	1.01062
x_{69}	x_{842}	.01402	1.01402
x_{70}	x_{942}	.01742	1.01742
x_{71}	x_{1042}	.02124	1.02124
x_{72}	x_{1142}	.02464	1.02464
x_{73}	b_4	0.0	1.00
x_{74}	x_{651}	.00280	1.00280
x_{75}	x_{751}	.00614	1.00614
x_{76}	x_{851}	.00957	1.00957

TABLE 1 (Cont.)

Variable and Coefficient Specifications

Program Variable	Model Variable	D_i	$1 + D_i = d_i$
x_{77}	x_{951}	.01320	1.01320
x_{78}	x_{1051}	.01700	1.01700
x_{79}	x_{1151}	.02120	1.02120
x_{80}	x_{652}	.00340	1.00340
x_{81}	x_{752}	.00680	1.00680
x_{82}	x_{852}	.01062	1.01062
x_{83}	x_{952}	.01402	1.01402
x_{84}	x_{1052}	.01742	1.01742
x_{85}	x_{1152}	.03186	1.03186
x_{86}	b_5	0.0	1.00
x_{87}	x_{761}	.00280	1.00280
x_{88}	x_{861}	.00614	1.00614
x_{89}	x_{961}	.00957	1.00957
x_{90}	x_{1061}	.01320	1.01320
x_{91}	x_{1161}	.01700	1.01700
x_{92}	x_{762}	.00340	1.00340
x_{93}	x_{862}	.00680	1.00680
x_{94}	x_{962}	.01062	1.01062
x_{95}	x_{1062}	.01402	1.01402
x_{96}	x_{1162}	.01742	1.01742
x_{97}	b_6	.00340	1.00340
x_{98}	x_{871}	.00280	1.00280
x_{99}	x_{971}	.00614	1.00614
x_{100}	x_{1071}	.00957	1.00957
x_{101}	x_{1171}	.01320	1.01320
x_{102}	x_{872}	.00340	1.00340
x_{103}	x_{972}	.00680	1.00680
x_{104}	x_{1072}	.01062	1.01062
x_{105}	x_{1172}	.01402	1.01402
x_{106}	b_7	0.0	1.00
x_{107}	x_{981}	.00280	1.00280
x_{108}	x_{1081}	.00614	1.00614
x_{109}	x_{1181}	.00957	1.00957
x_{110}	x_{982}	.00340	1.00340
x_{111}	x_{1082}	.00680	1.00680
x_{112}	x_{1182}	.01062	1.01062
x_{113}	b_8	0.0	1.00
x_{114}	x_{1091}	.00280	1.00280

TABLE 1 (Cont.)

Variable and Coefficient Specifications

Program Variable	Model Variable	D_i	$1 + D_i = d_i$
x_{115}	x_{1191}	.00614	1.00614
x_{116}	x_{1092}	.00340	1.00340
x_{117}	x_{1192}	.00680	1.00680
x_{118}	b_9	.00340	1.00340
x_{119}	x_{11101}	.00280	1.00280
x_{120}	x_{11102}	.00340	1.00340
x_{121}	b_{10}	0.0	1.00
x_{122}	b_{11}	0.0	1.00
x_{123}	x_{1211}	.04170	1.04170
x_{124}	x_{1212}	.04240	1.04240
x_{125}	x_{1221}	.03730	1.03730
x_{126}	x_{1222}	.03526	1.03526
x_{127}	x_{1231}	.03340	1.03340
x_{128}	x_{1232}	.03186	1.03186
x_{129}	x_{1241}	.02940	1.02940
x_{130}	x_{1242}	.02804	1.02804
x_{131}	x_{1251}	.02470	1.02470
x_{132}	x_{1252}	.02464	1.02464
x_{133}	x_{1261}	.02120	1.02120
x_{134}	x_{1262}	.02124	1.02124
x_{135}	x_{1271}	.01700	1.01700
x_{136}	x_{1272}	.01742	1.01742
x_{137}	x_{1281}	.01320	1.01320
x_{138}	x_{1282}	.01402	1.01402
x_{139}	x_{1291}	.00957	1.00957
x_{140}	x_{1292}	.01062	1.01062
x_{141}	x_{12101}	.00614	1.00614
x_{142}	x_{12102}	.00680	1.00680
x_{143}	x_{12111}	.00280	1.00280
x_{144}	x_{12112}	.00340	1.00340
x_{145}	b_{12}	0.0	1.00

period and simultaneously the cash balance at the beginning of the next period. For example, X_{22} (b_1) is the cash balance at the end of Period 1. It is also the cash balance at the beginning of Period 2. The variable X_{41} (b_2) is the cash balance at the end of Period 2, and variables X_{22} less X_{41} represent the change in cash balance over the period.

The next step is to provide coefficients of the variables so that the optimum choices can be made. Before specifying these coefficients, construction of the constraints should be examined.

Construction of constraints

The first constraints to be reviewed are the period constraints. For all periods the overall equation is that cash in equals cash out. Thus in the first period revenues less expenses equal the cash invested in securities plus the change in cash balances. That is,

$$X_1 + X_2 + \ldots + X_{20} + X_{123} + X_{124} - X_{21} + X_{22} = \text{Revenue} - \text{Expenses}$$

It is assumed that there are no securities owned at the beginning of the period (X_{21} is all in cash). So the excess of revenues over expenses is to be split among several competing investments. The proportion which goes into each is determined by the other twelve periods.

The second period has a similar construction--with an important exception. Now the excess of revenues over expenses is not the only potential source of income. If securities were bought in Period 1 to mature in Period 2, this income must be considered. The constraint for Period 2 then is:

$$X_{23} + X_{24} + \ldots + X_{40} + X_{125} + X_{126} - X_{22} + X_{41} - d_1X_1 - d_{11}X_{11}$$
$$= \text{Revenue - Expenses}$$

Here, d_1 and d_{11} are the increase in funds over the period. The increase d_i is determined for each security as follows: a 30-day Treasury bill is quoted at 3.56 percent interest on a yearly basis. If bought and held to maturity, the bill would be worth 0.280 percent on a monthly basis. Thus, if $100 were invested in Treasury bills in Period 1 to mature in Period 2, it would be worth $100.28 at the end of Period 2, or 1.0028 x 100. Thus, d_1 is 1.0028. Similarly, the coefficient d_{11} is calculated by taking the annual rate of interest on a CD and converting it to a monthly basis. Table 1 gives the d_i for all securities, and Part IV discusses more precisely the numbers used in the sample program.

Similar constraints are constructed for the remaining periods. That is, income from investments in previous periods plus change in cash balances less amount invested in securities equals revenue minus expenses. The computer printout of the program is available.

The second type of constraint included in the model is specification of cash balances. The beginning cash balance, X_{21}, is specified, as is X_{145}, the ending cash balance. Each period's cash balance, b_1 to b_{11}, is also specified. It should be noted that it is possible to specify a level for all of the cash balances, as was done in the sample program.

The liquidity for each period provides another set of constraints. Since it was felt that the best way to take care of potential emergencies was to have a specified cash balance and a required investment in 30-day

Treasury bills, a positive balance was required. Thus the variables which specified investing in Treasury bills maturing in the next period (such as X_{23}, X_{42}, X_{59}, etc. to X_{143}) were given positive balances. The coefficient was the 30-day rate on a Treasury bill. Again, the discussion of the sample program provides detail on the specification.

Objective function

Finally, the objective function is needed. The objective of the cash management model is to maximize income from investments; therefore, the coefficients of all decision variables must express that objective. Using the previous example of the constraint coefficient determination of X_1, the objective function is now determined. Since a 30-day investment in a Treasury bill will yield 0.280 percent if the annual rate is 3.56 percent, then that amount is the return on that investment Thus, the objective function is

$$\text{Max}Z = D_1 X_1 + D_2 X_2 + D_3 X_3 + \ldots + D_{145} X_{145}$$

where $D_i = d_i - 1$. In this equation d_i is the d_i previously determined in Table 1, and D_i for all variables is shown in Table 1.

The model construction is now complete. It consists of a linear programming model using the Simplex algorithm. Exhibit 1 in the Appendix describes how to run the Simplex algorithm on the University of Michigan MTS system. In the discussion of data construction, option number one was used for minimum problems.

For a detailed description of similar model construction, see Orgler.[1/] Exhibit 2 in the Appendix gives the standard linear programming description of the program.

Part IV presents a sample problem using the model and data from fiscal 1972. It also discusses the method used to update the program.

1/ Yair E. Orgler, Cash Management (Belmont, Calif.: Wadsworth Publishing Co., 1970).

IV

Sample Problem

A sample problem has been devised to facilitate understanding of the model. This problem is essentially the city's problem, and most of the coefficients would be unchanged when running the city's data.

Cash flow assumptions

Previous sections have discussed the form of the model and the derivation of the coefficients for the decision variables (the securities to be invested). The rest of the needed input data are specified, as well as the form of each constraint. As outlined, the sum of income and investments plus change in cash balances are the revenues less expenses for the period. The actual values for the cash inflow and outflow of the periods are now being gathered. In the meantime, the following scheme was used to determine net requirements.

According to the Deputy Controller, Earl Hoenes, the city's revenue accrues as follows:

1. Approximately 60 percent of the total revenue comes in July and August in the form of property tax collections--most of it in August. Delinquent tax collections are spread evenly through the year. Another 20 percent of total revenue comes in January, also from property tax collections.

2. State-collected taxes are received in April, July, October, and January.

3. Fines and other revenue come in evenly throughout the year.

Next, Mr. Hoenes mentioned that expenses tend to be fairly uniform throughout the year.

So yearly revenue of $12, 000, 000, as reported in the financial report of the city, was used to determine the period revenues and expenses, as shown in Table 2.

All that remains is specification of cash balances and liquid funds. The beginning cash balance was set at $50, 000, as reported in the city's financial statements. The balance at the end was also set at $50, 000. Every other period balance was set at $20, 000, so a positive balance of liquid funds was available.

The second set of constraints included the requirement that at least $30, 000 be invested in 30-day Treasury bills in every period. Thus liquid funds are kept in cash and maturing Treasury bills.

Review of computer program

A printout of the computer deck is available at the Bureau of Business Research and should be followed as the entire program is reviewed. Cards 2-6 are the control cards described in the program description. Card 5 changes the input format since the input data has up to five decimal places. Cards 6-269 represent the period constraints. Period 1 is covered by cards 6-27; Period 2 by cards 28-49; Period 3 by cards 50-71, etc. In Exhibit 2 of the Appendix periods are represented by constraint 1, constraint 2, etc. up to constraint 12.

Next are the cash balance constraints. Cards 270-91 cover X_{21} (b_2); cards 292-313 are for X_{22} (b_1); cards 314-35 are for X_{41} (b_2),

TABLE 2

Revenue and Expense Assumptions

Period	Month	Revenue	Expenses
1	July	2,000,000	1,000,000
2	August	4,500,000	1,000,000
3	September	300,000	1,000,000
4	October	900,000	1,000,000
5	November	200,000	1,000,000
6	December	100,000	1,000,000
7	January	2,000,000	1,000,000
8	February	500,000	1,000,000
9	March	400,000	1,000,000
10	April	600,000	1,000,000
11	May	400,000	1,000,000
12	June	600,000	1,000,000

etc. up to and including cards 534-55, which are for X_{145} (b_{12}). The coefficients of the variables must be minus one (-1) since these constraints are greater than specifications. Likewise, the "b" vector must be minus (in card 555 it is -50000) to insure the proper sign.

Following the cash balance constraints are the liquidity constraints, i.e., specification of minimum 30-day Treasury bill investments. Cards 556-77 cover Period 1 (X_1); cards 578-99 represent Period 2 (X_{23}), etc. through cards 776-97, which are for Period 12 (X_{143}). Again it should be noted that both the coefficients and the "b" vectors are minus, since these constraints specify "greater than" requirements.

Finally, the objective function is covered by cards 798-819. The objective function comes last and does not depend on the number of constraints.

Thus each of the cards has its role and must be kept in numerical sequence, especially the blank ones. Any card or part of a card left blank means that the coefficient for the variable associated with that card is zero (0). So it is obvious that changing numerical sequence can lead to infeasible solutions.

An obvious question at this point concerns changes in data input. Part V reviews data input changes, the mechanics of which are discussed here. The easiest way to change the input data is to use option number three in the program discussion (see Exhibit 1, Appendix) so that the original card deck is not disturbed. The new values may be stored in a file and reprinted. Of course another way is to change the cards. For example, if it is desired to end the 12-month period with an amount of \$100000 instead of \$50000, one would pull card 555 and punch up a new card with -100000 in columns 1-10. Any new change can be introduced in a similar way.

Explanation of output results

In conjunction with the discussion of output, reference should be made to the output printout.

Output is divided into four parts: the initial tableau or matrix, the iteration log, the final tableau, and the summary. The summary will be the primary tool for use by the investment officer.

The initial tableau is a printout of input information. The "basis" consists of artificial variables needed to begin the solution and is followed by the "b" vector and then the variables. Here A(1) is the same as X_1. Above each variable is the coefficient of that variable as entered in the objective function. The matrix is the enumeration of the coefficients for each variable in each constraint. It is easy to check the coefficients, since each constraint is numbered. The "-z" row gives the current value of the solution and the "opportunity costs" of each variable, which in the initial tableau are the coefficients of the objective function. The purpose of this tableau is to ensure that all of the elements of the matrix are correct so that a feasible solution results.

Next is the iteration log. The incoming and outgoing variables are listed for each iteration, as well as the current value of the objective function. The log also shows the rate at which the optimal solution is approached. In this problem fifty iterations were needed to reach an optimal solution.

The final tableau gives the identity of the variables in the optimal solution as well as the value that the variables take. Thus, it is a listing of those securities which should be held and the level that they should hold. Although it is interesting, the other material in the final tableau is not important. This tableau should not be used for decision making since the information is much more clearly described in the summary.

Finally, the summary of results presents the values of all the variables in numerical order. If a variable is in the optimal set, it is listed as a basic variable, and the amount of that variable is listed as the activity level. If it is not in the optimal set, it is listed as a nonbasic variable and its opportunity cost or "shadow price" is also given. For example, X_1 is in the optimal set and has a value of \$30, 000; X_2 is not in the optimal set, so it is listed as a nonbasic variable and the opportunity cost is 0. 0077.

What is important to the investment officer is the listing of the basic variables and their values. Table 3 gives the list of basic variables in the optimal set.

As was expected, most of the cash balance and liquidity constraints are just met. The question to be asked of the output of this model is what type of investments should be procured in the coming month. To answer that question the basic variables relating to Period 1 are considered. In Period 1 the beginning cash balance is \$50, 000 and the ending

TABLE 3

Optimal Set of Variables

Basic Variable	Model Variable	Activity Level
x_1	x_{211}	30,000
x_8	x_{911}	478,963
x_{11}	x_{212}	422,170
x_{13}	x_{412}	98,866
x_{21}	b_0	50,000
x_{22}	b_1	20,000
x_{23}	x_{321}	30,000
x_{32}	x_{322}	1,584,986
x_{34}	x_{522}	2,338,703
x_{41}	b_2	20,000
x_{42}	x_{431}	30,000
x_{52}	x_{632}	890,459
x_{58}	b_3	20,000
x_{59}	x_{541}	30,000
x_{73}	b_4	20,000
x_{74}	x_{651}	30,000
x_{85}	x_{1152}	1,563,625
x_{86}	b_5	20,000
x_{87}	x_{761}	30,000
x_{97}	b_6	20,000
x_{98}	x_{871}	30,000
x_{102}	x_{872}	604,370
x_{104}	x_{1072}	395,713
x_{106}	b_7	20,000
x_{107}	x_{981}	30,000
x_{110}	x_{982}	106,509
x_{113}	b_8	20,000
x_{114}	x_{1091}	30,000
x_{118}	b_9	20,000
x_{119}	x_{11101}	30,000
x_{121}	b_{10}	20,000
x_{122}	b_{11}	20,000
x_{143}	x_{12111}	30,000
x_{144}	x_{12112}	1,013,526
x_{145}	b_{12}	667,056

balance is \$20,000, thus releasing \$30,000 for investment. Revenues exceed expenditures, leaving \$1,030,000 to be invested. This amount should be invested as follows:

Treasury bills:	
30-day (X_1)	30,000
8-month (X_8)	478,963
Certificates of Deposit:	
30-day (X_{11})	422,170
3-month (X_{13})	98,866
	\$1,030,000

The model is saying that excess cash in the next month should be invested in the portfolio listed. Deviations in cash flows would necessitate changes in the portfolio. If substantial changes in cash flow occur, the model should be rerun.

The rest of the output is not of concern since it describes actions to be taken in future months. But it can be seen that the twelve-month horizon is considered and will affect the actions of the current period. Part V discusses updating of the model in detail; a brief discussion follows here.

As the model is used each month, a set of actions will be described for the coming month. These actions may be different from those described for the same time period several months previously because of new information. It should be noted that when the model is run in the next month, investments purchased in the preceding month will be kept and their value at maturity is added to the cash balance of the period. For example, when running the model at the beginning of the next month, the 8-month Treasury bill bought in Period 1 now matures in Period 7. This means that b_7 (X_{106}) is \$478, 963, plus \$20, 000, or \$498, 963; changes are also made for other securities. The 30-day securities are added to b_0 (X_{21}), the cash balance at the beginning of Period 1. That is, the model does not provide for selling securities. If securities are to be sold before maturity, this action must be handled by adjusting cash balances in the period. However, one of the assumptions included in the construction of the model was that securities were not to be sold before maturity so as to maximize income.

Finally, the optimal value should be mentioned. What this value means is that if the cash flows occur exactly as specified, if investments are made as specified, and if interest rates do not change, then the city could earn \$117, 056 on its investments. However, these conditions are unlikely. Each month will bring new and better forecasts of cash flow and changes in interest rates. As a matter of fact, at the present time the interest rate on short-term Treasury bills is beginning to rise. For these reasons the pattern of investments will change over time. Thus one would not expect to find that \$117, 056 will be earned over the fiscal year. However, since the optimal value considerably exceeds that which was actually earned in fiscal 1971, one would expect that a longer-range planning horizon would result in more efficient operation <u>and</u> the generation of more income. The optimal value will change from month to month as flows change, but the path of its changes can indicate what effect current or contemplated actions may have on investment income as well as investment decisions. Change in the optimal value will also indicate the effect of interest rate changes on the investment portfolio and investment income.

V

Interpretation and Implementation

The preceding has been a technical discussion of the structure of the model produced. Now the role of the model in the investment process, updating of the model, and suggestions for model revisions and capabilities will be reviewed.

Role of the model in investment

As mentioned in the introduction, the model does not serve as a substitute for decision making, but rather as an aid to the process. For example, for a given month the model will indicate the amount of investment needed and the type of instrument best suited to meet that need which will result in maximizing income. This information allows the investment officer to plan for the full year--not just the next several months. When funds become available, the information generated by the model will indicate the type and maturity of the security in which the money should be placed. If an unexpected event occurs which changes cash flows, the model can be updated to see if the change in cash flows is sufficient to change investment patterns. Thus, the model can be used to lengthen the time horizon of the investment officer and to take longer-term concerns into consideration.

The model can be used to direct attention to changes in yield patterns of securities. For example, with the current yield structure short-term CDs would be preferred over Treasury bills, but change in yield structure could change this situation. Thus the model can change the pattern of security investments needed to maximize income.

Finally, the model can be used to determine investment patterns for specific funds. As constructed, the "b" vector is the net revenue less expenses for the city. It is a simple matter to exchange the estimates of a fund cash flow for the city cash flow. After providing the cash balances required, the optimum investment program for a fund can be determined.

Updating the model

There are two types of considerations involved in updating the model. First are the month-to-month changes. That is, cash flow in Period 4, when the model is run on July 1, will be the cash flow for Period 3, when the model is run on August 1. Thus, each month the "b" vectors have to be changed. In addition, each of the yield coefficients must be checked. It is suggested that these coefficients be allowed to change by at least 1 percent of yield on an annual basis before changing the model. For example, interest on Treasury bills goes from 4 to 5 percent before the model is changed. Therefore, each month the model should be checked to see that all of the explicit and implicit assumptions still hold, and if they do not, then those that do not should be changed.

Finally, there is the updating of strategy, that is, revision of portfolio strategy if the model run on August 1 shows a different optimal set from that run on July 1. Again, what the optimal set indicates is the current optimal set for each period. Either adjustments can be made or the optimal set gives an indication of where new money available for investment should be placed. Long-term CDs cannot be changed without penalty, but Treasury bills can. Thus a modification or change in the makeup of the portfolio can be attempted if desired. It should be noted that major changes in the portfolio will occur only with major changes in the yield structure. Since there is a cash buffer built into the model, changes in cash flow should have less effect on the portfolio.

Implementation suggestions

As presently constructed, the coefficients and cash constraints are set. Changes in these would be policy changes. The first step in implementation of the model would be to take the actual cash flow data from last year, less income from investments, and estimate the level of income expected from investments. Then the "b" vector should be changed to the estimates of the current year and rerun. Finally, an updated forecast should be run each month to determine whether any changes in variables will change investment patterns. It might be added that constantly extending the time horizon by one month will help make the budget preparation process much easier and more efficient since time and effort have already been expended in forecasting.

After proficiency has been achieved in the use and interpretation of the model, many changes are possible. Time periods can be changed, other securities can be considered, and other assumptions can be incorporated in the model. It is strongly suggested that these changes not be considered before adequate facility has been achieved with the present model.

APPENDIX

Exhibit 1

SIMPLEX ALGORITHM FOR LINEAR PROGRAMMING PROBLEMS*

A. DESCRIPTION OF PROGRAM

This program (designated by the acronym SIMPLEX) uses the standard simplex algorithm for solving linear programming problems. The program accommodates problems in which the constraints are inequalities or equalities as well as problems in which variables are either restricted or unrestricted in sign.

The program inserts slack and/or artificial variables as needed via control card instructions which are described in Section C. Artificial variables are handled by the Charnes M-Method.

SIMPLEX2 is written in Fortran IV, Level H for batch implementation on The University of Michigan MTS/360 computer system. The program is designed to handle small linear programming problems where the total size of the initial tableau does not exceed 10,000 elements.

B. GENERAL FORM OF THE INPUT STREAM

The sign-on, command, control, and data cards for a problem run in SIMPLEX will take the general form outlined below:

```
$SIGNON xxxxx
yyyyy
$RUN BSAD:SIMPLEX2      3=-A
      Control Card  }
      Data Cards    }  One set for each problem.
$ENDFILE
$SIGNOFF
```

where xxxx is the users' computing center ID number and yyyyy is the users' password. (Note: To maintain password security, it is a good policy to set the keypunch "PRINT" toggle switch to OFF when punching the password card.)

C. CONTROL CARD FORMAT

The control card must be punched according to the following format, where all fields should be right-justified:

---

*Reprinted with permission of the authors from "Optimization Programs at the University of Michigan," by William K. Hall, Archer McWhorter, and W. Allen Spivey (Ann Arbor: Bureau of Business Research, University of Michigan, 1972), pp. 2-8.

Field	Control Card (Columns)	
1	1-4	m + 1 = the number of inequalities (and/or equalities) plus the objective function ($m \leq 140$). The nonnegativity conditions $x_i \geq 0$ are handled by the program and they need not be entered either on control or data cards.
2	5-8	n + 1 = the number of unknowns plus the "b" vector. $n \leq 400$ (not precise) and $m \times (n + m) \leq 10,000$.
3	9-10	If left blank, the algorithm maximizes the objective function. A one (1) in this field will cause minimization of the objective function.
4	11-12	Leave blank.
5	13-14	If left blank, the initial (expanded) tableau and the final tableau will be printed out. A one (1) in this field will cause all iterations to be printed out. If ties are found among vectors to leave the basis, the row numbers including the one to break the tie are printed out. Ties are broken by a random device designed to give any row an equally likely chance of being chosen.
6	15-16	If all the constraints are in fact equalities, then a one (1) in this field establishes this fact. Otherwise, this field should be left blank.
7	17-18	Input format control; if left blank, a standard format is used. If a one (1) is punched, a new format is on the following card. If a two (2) is punched, the format from the preceding problem is used. See section F on data format options for additional details.
8	19-20	Input option. A one (1) in this field causes data to be read one equation per card with the objective function last. A two (2) in this field causes data to be read one value per card. A three (3) in this field causes data to be read continuously. A four (4) in this field allows updating of the previous problem. See section E on data input options for additional details.
9-36	21-80	(30 2-column fields) All constraints are to be read in on data cards from less than or equal inequalities unless some or all (i. e., a 1 in column 16) are

specified to be equalities. Hence an inequality with a $\geqq$ sign should be multiplied through by -1. If each constraint is "$\leqq$", then nothing is punched in columns 21-80. If each constraint is "=" nothing is punched in columns 21-80 since columns 15 and 16 accommodate this situation. If just a few (up to 30) rows are equalities, each such row should be specified by starting in field 9 (columns 21 and 22) and using the consecutive 2-column fields for placing the row numbers that are equalities (right justify all integers). The program is designed to utilize information from columns 21-80 and introduces appropriate artificial variables without further action on the part of the user. The format is (2I4, 36I2) for the control card.

D. DATA CARD STRUCTURE AND SAMPLE PROBLEM

Following the control card the initial tableau is put in as follows:

$\left|\begin{array}{l} Ax \leqq b \\ cx \end{array}\right|$. The a_{ij} are floating point numbers and each must have a decimal point. All constraints must be either less than or equal to (or equal to as indicated above). Therefore, any b_i can be negative in the entering data. An exception is that the b_i in an equality must be nonnegative.

Example

The following example will be used to illustrate many of the options which the program allows.

(1) maximize $f = x_1 + 3x_2 + 5x_3 + 4x_4 + 9x_5$

(2) subject to $3x_1 + 2x_2 + 4x_3 + x_4 + 5x_5 = 15$

$$x_1 + 2x_2 + x_3 + 5x_4 + 5x_5 = 13$$

$$2x_3 + 6x_4 + 3x_5 \geqq 6$$

$$x_j \geqq 0 \quad (j = 1, \ldots, 5).$$

Punching the Control Card

(a) $m + 1 = 4$, so that the number 4 is punched in column 4.

(b) $n + 1 = 6$, so that the number 6 is punched in column 8.

(c) We are maximizing so columns 9-10 are left blank.

(d) Assume that each iteration is to be printed out; this means a 1 is punched in column 14.

(e) All the constraints are not equalities, so columns 15-16 are left blank.

(f) The possibilities for input and format control are discussed in Sections E & F below.

(g) We have two equality constraints, so beginning with columns 21-22 we punch a 1 in column 22, then punch a 2 in column 24. The third inequality would be multiplied through by -1 for entering on data cards; since the inequality is then "≤", no control card punch is then needed. Note: Suppose in a large problem the 1st, 2nd, 3rd, 4th, 17th, 25th constraints were equalities. Then we would have:

Col.	21 22	23 24	25 26	27 28	29 30	31 32	33 . . . 80
	1	2	3	4	1 7	2 5	← blank →

The program handles the introduction of artificial variables and their subsequent elimination, so the user need do nothing more with respect to equality constraints (except as noted above).

E. DATA INPUT OPTIONS

There are four input options available. Option number one reads one row (equation) at a time. Each row (equation) begins on a new card. Assuming the standard format for option one, the example above would be punched as follows:

Field Columns	1 1-10	2 11-20	3 21-30	4 31-40	5 41-50	6 51-60	7 61-70	8 71-80
Card 1	3.	2.	4.	1.	5.	15.		
Card 2	1.	2.	1.	5.	5.	13.		
Card 3	0.		-2.	-6.	-3.	-6.		
Card 4	1.	3.	5.	4.	9.	0.		

Several things are worth noting. Once a row (equation) is finished, the subsequent row begins on a new card. A number can be placed anywhere in the ten column field, but a decimal point must be punched. The third equation was given as "greater than or equal". It was multiplied by minus one so that it is "less than or equal". Fields 1 and 2 in card three represent zero coefficients. Zeroes are not to be punched, since a blank field is read as zero. The zero in field 6 of card four is the constant term in the objective function. Like other zeroes, it need not be punched, but room must be left for it. Note also that the objective function comes last.

Option number two reads one tableau value per card. Since it initializes the tableau to zero, only non-zero values need to be read in. Assuming the standard format for option two, the example above would be punched as follows:

Field Columns	1 1-5	2 6-10	3 11-20
Card 1	1	1	3.
Card 2	1	2	2.
Card 3	2	1	1.
Card 4	3	4	-6.
Card 5	3	0	-6.
Card 6	3	3	-2.
Card 7	3	5	-3.
Card 8	0	1	1.
Card 9	0	2	3.
Card 10	0	3	5.
Card 11	0	4	4.
Card 12	0	5	9.
Card 13	1	3	4.
Card 14	1	4	1.
Card 15	1	5	5.
Card 16	1	0	15.
Card 17	2	2	2.
Card 18	2	3	1.
Card 19	2	4	5.
Card 20	2	5	5.
Card 21	2	0	13.
Card 22	-1	-1	

The important features to be noted are: the row (equation) number in columns 1-5 and the column (or variable) number in columns 6-10 must be right-justified. Values in columns 11-20 need not be right-justified but must have a punched decimal point. The order of input data cards in this option is unimportant. A row number of zero indicates that the variable is in the objective function. A column number of zero indicates that the associated value is in the "b" vector. If a constant in the objective function is to be entered, Field 1 should contain a zero, and Field 2 should contain the number "n+1" (the same value that was entered in Field 2 of the control card). A card with both row and column numbers negative must be punched to terminate the reading of values for the tableau.

Option number three resembles option one. The data appears below for our sample problem under option three with its standard format:

Field Columns	1 1-10	2 11-20	3 21-30	4 31-40	5 41-50	6 51-60	7 61-70	8 71-80
Card 1	3.	2.	4.	1.	5.	15.	1.	2.
Card 2	1.	5.	5.	13.			-2.	-6.
Card 3	-3.	-6.	1.	3.	5.	4.	9.	

The format is the same as option one. The decimal points are required as in option one. Zeroes need not be punched as in option one. The only difference is that you do not begin a new card for each row. This causes a loss of some ease in reading, but obtains some economies in number of cards used.

Option number four allows one to modify the input from that of the immediately preceding problem. Suppose that in addition to our sample problem we wanted to run a problem identical to it except that the objective function is:

$$f = 3x_1 + 3x_2 - x_3 + 4x_4 + 9x_5$$

and the new equation three is:

$$x_1 + 6x_4 + 3x_5 \geq 7.$$

Under the standard input format for option four, a control card would be punched indicating that option 4 is to be used and the data would be punched on data cards as follows:

Field Columns	1 1-5	2 6-10	3 11-20
Card 1	0	1	3.
Card 2	0	3	-1.
Card 3	3	1	-1.
Card 4	3	3	0
Card 5	3	0	-7.
Card 6	-1	-1	

Note that the format to be used under option four is the same as in option two, including the requirement for a trailer card containing -1 in Fields 1 and 2. The original problem, however, could have been inputed option one, two, or three. If option four is used, only the values being changed need to be inputed. A constant in the objective function should be entered the same way as in option two.

F. DATA FORMAT OPTIONS

As indicated in the section on data input options, each option has its own standard format. For options one and three, the standard format is (8F10.4). For options two and four, the standard format is (2I5,F10.4). These standard formats are obtained by leaving columns 17-18 blank. If you desire to use a format of your own choosing, column eighteen should be punched with a one (1) and the format you desire should be punched on the succeeding card (beginning with its left parenthesis in column one). As an example, consider our familiar sample problem with the revised format (12F6.0):

Field	1	2	3	4	5	6	7	8	9	10	
Columns	1-4	5-8	9-10	11-12	13-14	15-16	17-18	19-20	21-22	23-24	25-80
Control Card	4	6			1		1	3	1	2	blank

Format Card (12F6. 0) in columns one through eight

Field	1	2	3	4	5	6	7	8	9	10
Columns	1-6	7-12	13-18	19-24	25-30	31-36	37-42	43-48	49-54	55-60
Card 1	3.	2.	4.	1.	5.	15.	1.	2.	1.	5.
Card 2			-2.	-6.	-3.	-6.	1.	3.	5.	4.

Field	11	12
Columns	61-66	67-72
Card 1	5.	13.
Card 2	9.	

This change of format would allow the data to be compressed onto two cards. There are occasions on which you might wish to repeatedly use a non-standard format. This can be done by reentering the non-standard format with each problem (as described above). An easier way has been built into the program. After explicitly supplying the non-standard format on the first of this series of problems, by punching a two (2) in column 18 of succeeding control cards the original non-standard format will be retained.

Exhibit 2

MODEL DESCRIPTION

Objective Function $\quad \text{Max } Z = \sum_{i=1}^{145} C_i X_i$

Constraint	Period Constraints
1	$\sum_{i=1}^{20} X_i + X_{123} + X_{124} - X_{21} + X_{22} = R_1 - E_1$
2	$\sum_{i=23}^{40} X_i + X_{125} + X_{126} - X_{22} + X_{41} - d_1X_1 - d_{11}X_{11} = R_2 - E_2$
3	$\sum_{i=42}^{57} X_i + X_{127} + X_{128} - X_{41} + X_{58} - d_2X_2 - d_{12}X_{12} - d_{23}X_{23} - d_{32}X_{32} = R_3 - E_3$
4	$\sum_{i=59}^{72} X_i + X_{129} + X_{130} - X_{58} + X_{73} - \sum_{j=w} d_jX_j = R_4 - E_4$ where w = 3, 13, 24, 33, 42, 50
5	$\sum_{i=74}^{85} X_i - \sum_{j=w} d_jX_j - X_{73} + X_{86} + X_{131} + X_{132} = R_5 - E_5$ where w = 4, 14, 25, 34, 43, 51, 59, 66
6	$\sum_{i=87}^{96} X_i - \sum_{j=w} d_jX_j + X_{133} + X_{134} - X_{86} + X_{97} = R_6 - E_6$ where w = 5, 15, 26, 35, 44, 52, 60, 67, 74, 80
7	$\sum_{i=98}^{105} X_i - \sum_{j=w} d_jX_j + X_{135} + X_{136} - X_{97} + X_{106} = R_7 - E_7$ where w = 6, 16, 27, 36, 45, 53, 61, 68, 75, 81, 87, 92
8	$\sum_{i=107}^{112} X_i - \sum_{j=w} d_jX_j + X_{137} + X_{138} - X_{106} + X_{113} = R_8 - E_8$ where w = 7, 17, 28, 37, 46, 54, 62, 69, 76, 82, 88, 93, 98, 102
9	$\sum_{i=114}^{117} X_i - \sum_{j=w} d_jX_j + X_{139} + X_{140} - X_{113} + X_{118} = R_9 - E_9$ where w=8, 18, 29, 38, 47, 55, 63, 70, 77, 83, 89, 94, 99, 103, 107, 110
10	$X_{119} + X_{120} - \sum_{j=w} d_jX_j + X_{141} + X_{142} - X_{118} + X_{121} = R_{10} - E_{10}$ where w = 9, 19, 30, 39, 48, 56, 64, 71, 78, 84, 90, 95, 100, 104, 108, 111, 114, 116
11	$X_{143} + X_{144} - \sum_{j=w} d_jX_j - X_{121} + X_{122} = R_{11} - E_{11}$ where w = 10, 20, 31, 40, 49, 57, 65, 72, 79, 85, 91, 96, 101, 105, 109, 112, 115, 117, 119, 120

Constraint	Period Constraints
12	$-\sum_{j=123}^{144} d_j X_j - X_{122} + X_{145} = R_{12} - E_{12}$

Constraint	Cash Balance Constraints
13	$X_{21} \geqq \$50000$
14	$X_{22} \geqq 20000$
15	$X_{41} \geqq 20000$
16	$X_{58} \geqq 20000$
17	$X_{73} \geqq 20000$
18	$X_{86} \geqq 20000$
19	$X_{97} \geqq 20000$
20	$X_{106} \geqq 20000$
21	$X_{113} \geqq 20000$
22	$X_{118} \geqq 20000$
23	$X_{121} \geqq 20000$
24	$X_{122} \geqq 20000$
25	$X_{145} \geqq 50000$

Constraint	Liquidity Constraints (Treasury Bills)
26	$X_{1} \geqq \$30000$
27	$X_{23} \geqq 30000$
28	$X_{42} \geqq 30000$
29	$X_{59} \geqq 30000$
30	$X_{74} \geqq 30000$
31	$X_{87} \geqq 30000$
32	$X_{98} \geqq 30000$

Constraint	Liquidity Constraints (Treasury Bills)
33	$X_{107} \geqq 30000$
34	$X_{114} \geqq 30000$
35	$X_{119} \geqq 30000$
36	$X_{143} \geqq 30000$
	All X_i's $\geqq 0$