OPTIMIZATION TECHNIQUES FOR GRAIN DRYER DESIGN AND ANALYSIS

Thesis for the Degree of Ph. D.

MICHIGAN STATE UNIVERSITY

DAVID MICHAEL FARMER

1972

This is to certify that the

thesis entitled

OPTIMIZATION TECHNIQUES FOR GRAIN DRYER

DESIGN AND ANALYSIS

presented by

David Michael Farmer

has been accepted towards fulfillment
of the requirements for

PhD degree in Agricultural Engineering

F.W. Bakker-Arkema

Major professor

Date 5-18-72

O-7639

ABSTRACT

OPTIMIZATION TECHNIQUES FOR GRAIN DRYER DESIGN AND ANALYSIS

By

David Michael Farmer

This study uses techniques of mathematical simulation and optimization in development of user-oriented algorithms for analysis and optimal design of selected grain drying systems; digital computer programs and the necessary documentation for their use are included.

In Part I, a prototype program is developed for use by equipment manufacturers, extension personnel, and individual farm operators (via remote time-share computer terminals) in design and economic analysis of batch-in-bin type drying systems. A specialized optimization technique for minimizing operating cost, subject to constraints on available equipment and product quality, is presented; an empirical model, capable of rapid evaluation, for batch drying of corn is adapted for computational efficiency in the search techniques. Potential use of the package is illustrated by studies of the sensitivity of operating costs to economic and ambient conditions, to design parameters, and to variations in operating and marketing practice.

Selected results for the central-Michigan area are:

1. Fuel cost is a much larger component of total operating cost than electrical cost. Thus, dryer design changes intended to improve thermal

efficiency can substantially reduce operating cost. Alternate fuels should be carefully considered by operators, based on heating value and price.

2. Replacing heat from fossil fuels by resistance electric heat is not economical for batch dryers at the current energy price levels.

3. Under adverse ambient drying conditions a greater percentage rise in airflow costs than heating costs occurs.

4. If drying conditions are such that optimal airflows are low, operating costs fall with increasing depth until excessive condensation occurs; the operating cost reduction per unit depth, however, decreases with increasing depth and must be weighed against increased construction costs.

5. For an equivalent daily grain volume, drying a single batch on a 20-hour schedule is appreciably cheaper than drying two batches on 10-hour schedules.

Part II of the thesis study is concerned with development and digital computer implementation of two algorithms for optimal design of concurrent-flow dryer, counterflow cooler systems, with and without recycle of cooler exhaust air. The optimization technique of dynamic programming is employed to insure overall optimal choice of construction and operating parameters for each system; computer programming techniques designed to minimize time and memory requirements are introduced. The user is permitted maximum freedom in choice of constraints, component models, and performance criteria in adapting the algorithm to his needs. Examples of the use of both algorithms in the design of corn drying systems are given. Under a single set of ambient and economic conditions, operating expenses for the dryer-cooler system without air recycle were found to be slightly lower than those for the recycled air system.

Approved F. H. Bakker-Arkema
Major Professor 5-18-72

Approved B A Stout
Department Chairman

OPTIMIZATION TECHNIQUES FOR GRAIN DRYER DESIGN AND ANALYSIS

by

David Michael Farmer

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Agricultural Engineering

(1972)

This work is dedicated, in equal measure, to the two who have made it possible: to my wife Ginny for unfailing affection and helpfulness; and to Dr. Fred W. Bakker-Arkema, for inspiration, patience and friendship freely given.

ACKNOWLEDGMENTS

The author gratefully acknowledges the guidance and encouragement of Dr. Fred W. Bakker-Arkema who, by his example, has shown the path of excellence in personal as well as technical pursuits. The friendship and unselfish assistance of students--both past and present--in the Processing Group have been most appreciated; in the preparation of this work, special thanks are due to Lloyd E. Lerew, the "enigmatuer", and to Stephen F. DeBoer, Gonzalo Roa, and Mitchell G. Roth.

Appreciation is expressed to Dr. William G. Bickert, Dr. George Coulman, Professor Robert L. Maddex, and Dr. Walter M. Urbain, not only for serving on the guidance committee but also for their help during the course of study.

The partial financial support of The Andersons, Incorporated, Maumee, Ohio was much appreciated. A note of thanks is also due to those who assisted in keypunching, Ken Jones and G. R. Shidle, and to the typist, Ms. Georgianna Farmer.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

a = specific surface area of grain, ft^2/ft^3

a_1, a_2, a_3, a_4, a_5 = empirically determined constants, Equation(I.B.10)

b_1, b_2 = empirically determined functions

c_1, c_2 = empirical constants

c_a = specific heat of dry air, Btu/lb-deg F

c_p = specific heat of product, Btu/lb-deg F

c_w = specific heat of water, Btu/lb-deg F

C_a = airflow cost, cents per square foot of bed area

C_h = heating cost, cents per square foot of bed area

DF = dimensionless grain flow rate

DM = dimensionless moisture content

DX = dimensionless depth of bed

e = base of Napierian logrithms

E = average pounds of moisture removed per pound of dry air during a specified elapsed time, Equation(I.B.1)

F = total value of the objective function, cents per square foot of bed area

$\bar{g}$ = vector functional of systems equations

G_a = mass flow rate of air per unit area, $lb/hr/ft^2$

h = convective heat transfer coefficient, $Btu/hr/ft^2$

h_{fg} = unit heat of vaporization of moisture in grain, Btu/lb

H = fuel heat value, Btu/gallon

HP = horsepower

i = partial process cost incurred during stage n

I = optimal total process cost

I' = optimal process subtotal oost

j = exponent, dimensionless

k = exponent, dimensionless

K = time constant, hr^{-1}

m_1, m_2 = exponents, dimensionless

M = moisture content, decimal dry basis

$\bar{M}$ = average moisture content, decimal dry basis

M_e = equilibrium moisture content, decimal dry basis

M_o = initial moisture content, decimal dry basis

n = stage of the system

N = total number of process stages

N_t = time, dimensionless

N_x = depth, dimensionless

p = pressure drop per unit area of bed, inches of water

p_{atm} = atmospheric pressure, psia

p_s = saturation vapor pressure, psia

P_E = price of electricity, cents per kilowatt-hour

P_F = price of fuel, cents per gallon

Q = airflow rate, cfm of dry air per square foot of horizontal drying surface

RH = relative humidity of air, decimal

t = elapsed drying time, hours

t_m = elapsed drying time, minutes

t_H = time in hours required to reduce the dimensionless moisture ratio to 0.5

T_{amb} = ambient dry bulb temperature, deg F

T_G = dry bulb temperature of air at an infinite dryer length after a constant wet bulb process to an equilibrium relative humidity for grain at the initial moisture content, deg F

T_{inlet} = inlet air temperature, deg F

ΔT = temperature rise of incoming air due to its passage over the motor and fan, deg F

$\bar{u}$ = vector of control or decision variables

U = set of all feasible controls

V = air velocity, feet/minute

x = bed depth or distance in bed from air inlet, feet

$\bar{x}$ = vector of state variables

X = set of all feasible states

β = dimensionless constant

γ = dimensionless constant

η_f = fan efficiency, decimal

η_m = electric motor efficiency, decimal

η_{th} = thermal efficiency, decimal

θ = product temperature, deg F

ρ_a = density of dry air, lb/ft^3

ρ_g = dry grain bulk density, lb/ft^3

ρ_w = water vapor density, lb/ft^3

I. AN ECONOMIC STUDY OF BATCH-IN-BIN DRYER OPERATING COST

A. Introduction

On-farm drying is a fact of life for most corn farmers today. The advent of combine harvest of corn has increased the harvesting, drying and handling rates to a point where many country elevators no longer have the capacity to process the high moisture corn being delivered, thus forcing the grower to dry his own. In addition, on-farm drying and storage have given the farmer flexibility in selling his crop, enabling him to wait until the initial glut on the corn market has passed. Also, he may now avoid dockage charges for high moisture corn by drying his own. In assuming the responsibility for drying, however, the grower himself is liable for dryer operating costs and possible product degradation.

Farmers and elevator operators have usually taken care to avoid readily apparent deterioration due to rodents, insects or microorganisms during storage. Deterioration of processing quality due to overheating during drying has not been reflected in market prices as long as the average moisture content fell within an acceptable range. Lately, however, increasing complaints from cereal and snack food processors as well as from the wet milling (corn starch) industry indicate that a premium may soon be paid for properly dried grain.

Bussell(1969) attributes the wide diversity in corn quality to several sources in the grain harvesting, conditioning and marketing system. Improper dryer operation may occur on the farm or at the elevator because of ignorance of correct operating procedures or due to overloading a dryer too small to meet the flow rates in the harvest system. Dryer manufacturers, who have not evolved a uniform means of rating dryers, are held accountable for the latter charge. Farmers are blamed for not taking time to learn correct dryer operation or to teach it to their employees.

These criticisms from a manufacturer of drying equipment point out the need for a means of comparison of dryer capacities and for clearly defined dryer operating procedures. The recent development of good dryer computer models (Bakker-Arkema et al.,1971) help to define operating limits; nonetheless, an appreciation and study of dryer economics is needed by both the manufacturer for intelligent design and by the dryer operator for intelligent use.

To date, the studies of Morey et al.(1969) and Bloome et al.(1970) on the economic feasibility of layer drying cover the recently published work specifically concerned with the economics of bin dryer operation. These, however, are necessarily based on long-range weather conditions for a particular area and hence have limited applicability elsewhere. In this study, the intent is to develop an economic optimization model for a batch-in-bin dryer which will be of use to both dryer operators and manufacturers and which will be adaptable to the user's need, regardless of location. If such a program were inserted in a telephone-linked computer system (such as in Michigan State University's Future Plan Programs; Harsh et al.,1971) it would be within reach of any potential user at minimal cost. A series of similar programs, one for each dryer type, would allow rapid comparison of dryer performance and provide a rational means of rating dryer capacities.

The batch-in-bin dryer typically consists of a cylindrical metal bin up to 36 feet in diameter or larger, which is filled during drying to a depth of from one to three feet. A high volume fan equipped with a burner (usually LP gas) introduces heated air into a plenum chamber beneath the grain from whence it flows through a perforated metal floor and up through the bed of grain. The rate of drying is adjusted to the harvesting rate so that one or two batches are dried daily. Since the grain first begins to dry in the lower portion of the bin where the heated air enters, the upper portion of the grain is last to dry. In practice, drying is continued until an acceptable average moisture content is reached; the grain is then cooled and placed in storage or is shipped. Auger unloading of the dryer serves to mix the wet and dry grain after which equilibration takes place in storage.

It should be realized that choice of type and size of dryer is largely determined by the remaining components in the harvesting-handling-drying storage system. Thus, an economic optimization model of dryer operation should have the flexibility to be incorporated into large-scale systems models (Farmer,1971). Furthermore, operating cost, which this algorithm is designed to evaluate, is only one component of total dryer cost; Bloome(1970) also includes cost of storage structure, heater, fan, material handling components and their operating cost, depreciation, interest, repairs, taxes and insurance. The present model is set up on a per square foot basis so that it may be easily adjusted for evaluation of dryers having different diameters.

Some problems in dryer choice or operation, for which the model is applicable, are suggested:

1. Purchase of a new harvesting machine will increase the daily input to an existing batch-in-bin dryer.

a. What are the implications of this change in terms of dryer efficiency?

b. If the purchase of a new dryer is contemplated, what will be the expected operating cost?

2. Is there a cost per bushel improvement in drying two or more batches daily instead of a single batch?

3. What effect does the time of harvest have on drying costs, i.e., how do costs associated with higher harvest moisture contents early in the season balance against increased field losses and lower ambient temperatures expected later?

4. What effect will a change in the price of fuel or electricity have upon operating cost?

5. What are the operating cost implications of a change in heat source (e.g. LP gas vs. natural gas vs. electrical heat)?

6. How much can be saved in dryer operating costs by marketing at moisture contents above that needed for long-term storage?

The algorithm can also be used to provide insight into the following questions, which might occur in dryer design or sales and service:

1. Are automatic controls which adjust dryer operating conditions in accordance with changes in weather conditions worthwhile in terms of expected savings?

2. Is it feasible to tailor dryer design to location, based on expected weather conditions?

3. For a given location, what operating recommendations should be given to customers for most economical operation?

4. Is the investment in high quality motor and fan components worthwhile in terms of increased dryer efficiency?

5. What are the expected cost advantages of direct fired over indirect fired dryers?

6. How can construction costs (depth vs. diameter) be balanced against operating costs in order to size a line of dryers?

In the succeeding sections, the simulation models considered for batch-in-bin drying are first discussed. Secondly, the cost (objective) function is developed, followed by an exposition of the optimization technique employed. Next, constraints and sources of information to facilitate use of the algorithm are delineated along with a program listing and instructions for use. Finally, several of the questions posed above are investigated for conditions prevailing in the central Michigan area.

B. Batch-in-Bin Dryer Simulation Models

Use of the optimization option in the economic model requires that the simulation be performed a number of times in order to approach those operating conditions for which cost is minimized. In order that computing time and cost do not become prohibitive, a rapid process simulation model is needed to locate the vicinity of the optimum.

Two models from the literature were chosen as candidates for rapid simulation of the process. The first (Nelson,1960) utilizes a dimensional analysis approach to predict average moisture contents during drying of deep beds of various grains. The author reported a fit of within 4-5% on all experimental data. Corn drying trials were not included in the data; however, it was presumed that for a single type of grain the model could considerably improve its prediction accuracy over that for all grains. The form of the model is:

$$E = c_1\left[(M_o - M_e)(T_{inlet} - T_G)/T_{inlet}\right]^n \left[1. - e^{c_2 x/Qt_m}\right] \qquad \text{(I.B.1)}$$

The final average moisture content is computed by:

$$\bar{M} = M_o - E\rho_a/(Qxt_m\rho_g) \qquad \text{(I.B.2)}$$

In order to evaluate the effectiveness of the model, a comparison with deep bed corn drying experiments was needed. A thorough search disclosed little usable published deep bed corn data. In order to provide a balanced set of information against which the model could be measured, the experimental data of Kirk(1958), Hamdy et al.(1969), and Farmer(1969) were chosen, encompassing the inlet air temperature range of 100.-165. deg F at a variety of airflow rates, elapsed drying times, and ambient relative humidities. The predicted average moisture contents, using Nelson's constants, were calculated. In addition, a non-linear parameter estimation routine, GAUSHAUS (Meeter,1965) was used, retaining the form of the model, to estimate least-squares best-fit constants (c_1,c_2, and n) using the available data. A comparison of the results is shown in Table 1.

Between the inlet air temperatures of 100.-165. deg F, the predictions of the model using the "GAUSHAUS" constants fell within 2.% of each data point; however, there were no additional data available with which to make an independent comparison.

A second model, first proposed by Hukill(1953), was tried in an attempt to improve the accuracy of the predictions. In essence, the model in its original form used the following relations between dimensionless quantities:

Table 1: Comparison of experimental data with model simulations

Data Source and Test No.	M_o dec.,db	T_{inlet} deg F	RH dec	Bed Depth ft	Q cfm/ft^2	Drying Time hr	P_{atm} psia	g lb/ft^3	FINAL AVERAGE MOISTURE CONTENT: Data dec.,db	*	**	‡
(Farmer,1969)												
1	0.337	100.	0.50	2.0	32.	16.	14.34	34.1	0.229	0.170	0.243	0.207
2	0.336	100.	0.50	1.0	16.	16.	14.34	34.1	0.260	--	0.243	0.267
3	0.296	105.	0.20	0.833	32.	16.	14.34	34.1	0.117	--	0.118	0.118
4	0.319	105.	0.35	0.833	32.	16.	14.34	34.1	0.147	--	0.167	0.144
(Kirk,1958)												
1	0.432	165.	0.02	0.333	124.	1.3	14.6	34.1	0.163			
2	0.432	165.	0.02	0.667	72.	2.0	14.6	34.1	0.163			
3	0.432	165.	0.02	1.0	37.	5.1	14.6	34.1	0.163	(These data are above the maximum temperature limit of the model.)		
4	0.432	165.	0.02	1.333	53.	4.2	14.6	34.1	0.163			
5	0.432	165.	0.02	0.333	84.5	3.0	14.6	34.1	0.099			
6	0.432	165.	0.02	0.667	55.	5.0	14.6	34.1	0.081			
7	0.432	165.	0.02	1.0	42.5	6.0	14.6	34.1	0.112			
8	0.432	165.	0.02	1.333	35.5	8.5	14.6	34.1	0.130			
(Hamdy,1969)												
1	0.376	123.	0.26	1.5	59.28	1.	14.3	33.7	0.334	--	0.325	0.334
2	0.376	123.	0.26	1.5	59.28	2.	14.3	33.7	0.294	--	0.278	0.294
3	0.376	123.	0.26	1.5	59.28	3.	14.3	33.7	0.253	--	0.238	0.259
4	0.376	123.	0.26	1.5	59.28	4.	14.3	33.7	0.212	--	0.207	0.227
5	0.376	123.	0.26	1.5	59.28	5.	14.3	33.7	0.191	--	0.183	0.198
6	0.376	123.	0.26	1.5	59.28	6.	14.3	33.7	0.170	--	0.164	0.174
7	0.376	123.	0.26	1.5	59.28	7.	14.3	33.7	0.153	--	0.149	0.153
8	0.376	123.	0.26	1.5	59.28	8.	14.3	33.7	0.142	--	0.136	0.136

*** - Simulation results of the model of Nelson(1960), using recommended constants: c_1=0.308, c_2=10602., n=0.774**

****- Simulation results of the models of Nelson(1960), using GAUSHAUS constants: c_1=0.06425, c_2=17030., n=0.5272**

‡ - Simulation results of the QUICK model (Hukill,1953), using constants found by GAUSHAUS: a_1=-0.04564539, a_2=0.0055836759, a_3=0.33804141, a_4=0.94061342, a_5=6.3492118

$$N_m \equiv \frac{M-M_e}{M_o-M_e} = \frac{2^{N_x}}{2^{N_x} + 2^{N_t} - 1.} \qquad \text{(I.B.3)}$$

where:

$$N_x \equiv \frac{x\rho_g h_{fg}(M_o-M_e)}{6000.Q\rho_a c_a t_H(T_{inlet} - T_G)} \qquad \text{(I.B.4)}$$

and

$$N_t \equiv t/t_H \qquad \text{(I.B.5)}$$

The general shape of the moisture profile which develops during a deep bed drying process, given constant inlet conditions and a uniform initial moisture content (Figure 1), can be well approximated (aside from cases where large amounts of condensation occur) by a function of the form:

$$N_m = \beta^j/(\beta^j + \gamma^k - 1.) \qquad \text{(I.B.6)}$$

where:

$\beta, \gamma > 0$

$0 \leq j < \infty \quad ; \quad j \xrightarrow[x \to \infty]{} \infty \quad ; \quad j_{x=0} = 0$

$0 \leq k < \infty \quad ; \quad k_{t=0} = 0 \quad ; \quad k \xrightarrow[t \to \infty]{} \infty$

For fixed positive values of β and γ, the value of j determines the slope of the curve; the value of k regulates the shift of the curve along the distance axis.

In Hukill's model, $\beta = 2$, $\gamma = 2$ and $j = N_x$, $k = N_t$. Because a comparison of data with the prediction of the model showed considerable

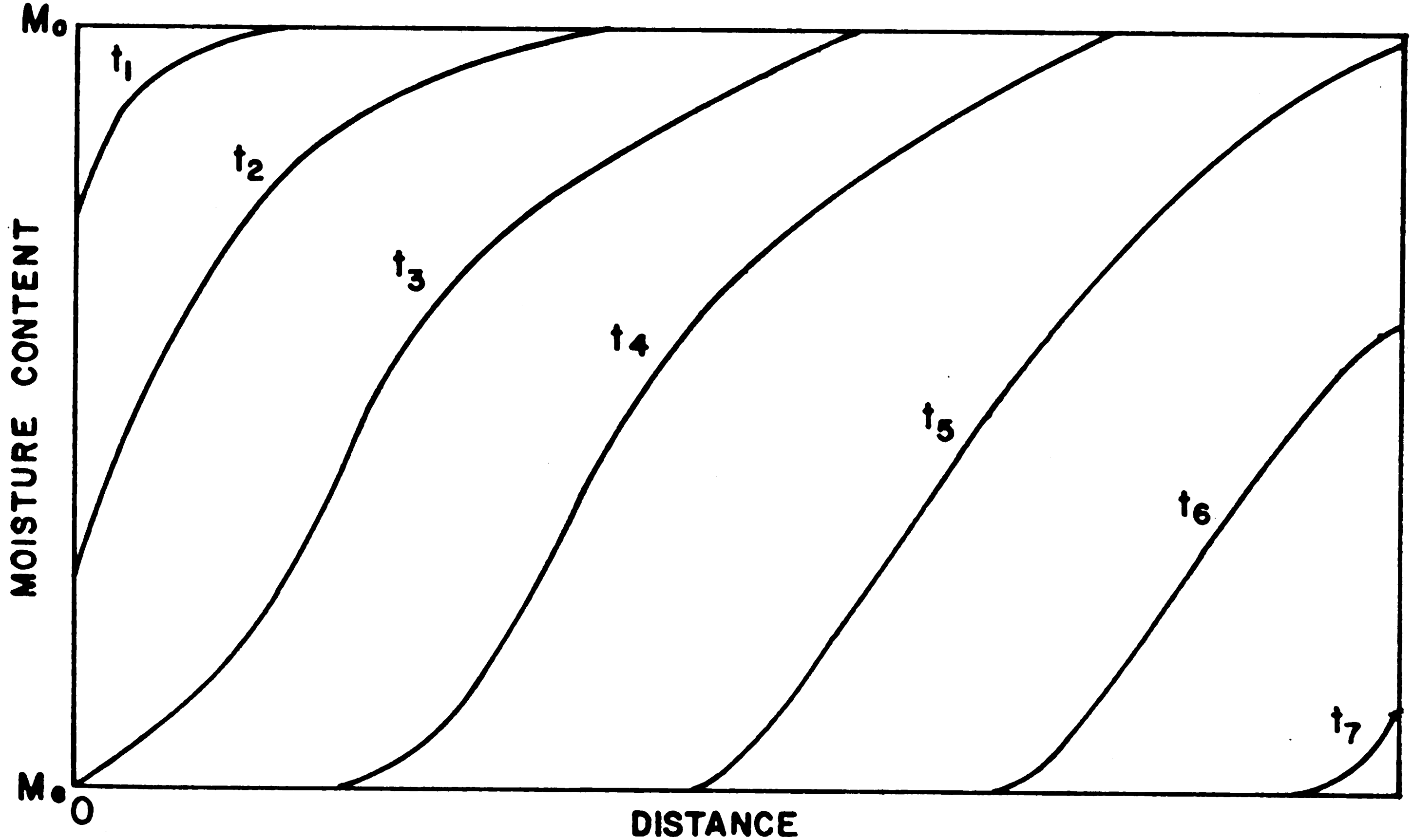

Figure 1: Variation in Moisture Content with Depth and Time

discrepancy, a later version of the model (Holtman et al.,1967; Barre et al.,1970) was tried in which $\beta = \gamma = e$; for this version j and k were redefined by:

$$j = \frac{x\rho_g h_{fg} K(M_o - M_e)}{Q\rho_a c_a (T_{inlet} - T_G)} \qquad \text{(I.B.7)}$$

$$k = Kt \qquad \text{(I.B.8)}$$

where K is a time constant

$$K = K(p_s^{m_1} V^{m_2}) \qquad \text{(I.B.9)}$$

and m_1, m_2 are each positive exponents less than one.

A good fit to a single drying test was obtained by Barre et al.(1970); however, they failed to report an explicit functional relationship for values of K. Since the available experimental data did not agree well using several trial functional forms for K, a strictly empirical modification was developed using the available data. The general functional form adopted for K was:

$$K = a_1 + a_2 p_s^{a_3} V^{a_4} \qquad \text{(I.B.10)}$$

where a_1, ..., a_4 are constants to be determined. Due to the appearance of the parameter K in the expressions for both j and k, a multiplicative constant a_5 for the expression for k was used to further differentiate the two dimensionless quantities; thus, in this study:

$$k = a_5 Kt \qquad \text{(I.B.11)}$$

Values of the constants a_1, a_2, ..., a_5 are given in Table 1.

In searching for constants which would provide the best fit of the model to the data, GAUSHAUS failed to work properly. An available pattern search optimization program, HCLMB (Rosenbrock et al.,1960), was used to determine the constants, again using a least-squares best-fit criterion. A comparison of the available data with the values generated by this model is shown in Table 1. Since this model showed improvement over the first considered and was essentially as fast for computer evaluation, it was chosen to represent the process for the subsequent sensitivity studies. This model will henceforth be referred to by its FORTRAN IV subroutine name, QUICK.

Because all available experimental data had been used to evaluate the empirical constants used in QUICK, none was on hand for an independent review of the model's authenticity. As a precaution against distortion of the model by the choice of data used in its construction, a secondary check was required; this check was also needed to prevent extrapolation of QUICK beyond the limits of its validity.

To provide this check on the optima predicted by QUICK, the Michigan State University deep bed drying simulation FXBD (Bakker-Arkema et al., 1971) was employed. This model, which is based on fundamental mass and energy balances, accounts for phenomena (e.g. condensation) which are beyond the scope of QUICK. In addition, it can be considered valid over the entire range of input parameters normally used in deep bed drying. FXBD, which utilizes an iterative method of solution, requires considerable time for a single simulation; thus it could not be readily used in the optimization process.

Using the experimental data of Table 1 covering inlet air temperatures from 100 - 165 deg F and initial moisture contents from 0.296 - 0.432 dry basis, simulations were made using the QUICK and FXBD models. From the summary of Table 2, the predicted dry basis final moisture contents of FXBD fall roughly in the range of 3% higher to ½% lower

Table 2: Comparison of QUICK and FXBD Models with Experimental Data*

Final Moisture Contents, decimal, dry basis

Data # - Source	Data	"QUICK"	"FXBD"
(Farmer,1969)			
#1	0.2280	0.2065	0.2172
#2	0.2625	0.2670	0.2187
#3	0.1150	0.1180	0.1300
#4	0.1472	0.1439	0.1410
(Hamdy,1969)			
#1	0.3335	0.3337	0.3445
#2	0.2935	0.2944	0.3084
#3	0.2530	0.2586	0.2749
#4	0.2120	0.2265	0.2476
#5	0.1910	0.1983	0.2257
#6	0.1700	0.1739	0.2074
#7	0.1533	0.1531	0.1916
#8	0.1422	0.1359	0.1779
Sum of squares fit	=	0.0855 =	1.1738

*Conditions for these tests are shown in Table 1.

than the data values, while the predictions of QUICK are within this range.

Optimal drying conditions predicted by QUICK in the subsequent sensitivity studies were used as inputs for FXBD. Using the stated tolerances as a standard, the model QUICK was considered valid only for those trials in which the difference of predicted final average moisture contents fell within this range; for trials in which the difference was greater, QUICK was presumed to be an inadequate model and the results were discarded.

C. Development of the Cost (Objective) Function

In line with the objective of a unitized economic model from which more complex models might be formed, the cost function was constructed on a per square foot basis.

In conventional deep-bed drying, two sources of energy are required: one (typically natural or LP gas) for heating the air, the second (typically electricity) for powering a fan to force the drying air through the bed. A formulation of the heating component of the cost function for fossil fuels is (Bloome,1970):

$$C_h = \frac{60.P_F Q(\rho_a c_a + \rho_w c_w)(T_{inlet} - T_{amb} - \Delta T)t}{H \eta_{th}} \qquad \text{(I.C.1)}$$

where ΔT is the temperature rise of the incoming air due to its passage over the motor and friction with the fan blades.

To derive an expression for ΔT, the factors influencing its magnitude must be known. Bloome(1970) states that this temperature rise is a function of fan efficiency and of the static pressure against which the fan operates. This reference cites the following field data: for fans providing an airflow rate of 12.8 cfm/ft^2 through a 16-foot depth of shelled corn, a temperature rise ΔT of approximately 2 deg F occurs. A 30% fan-motor efficiency factor is given.

The corresponding pressure drop, 1.585 inches of water, was calculated from the empirical relation of Thompson (1967):

$$p = x\left[\frac{Q}{58.}\right]^{1.528} \tag{I.C.2}$$

Assuming that temperature rise of the air is directly proportional to pressure drop and inversely proportional to fan motor efficiency, an expression for ΔT was developed:

$$\Delta T = 2.0 \left(\frac{p}{1.585} \right)\left(\frac{0.30}{\eta_m \eta_f} \right) \tag{I.C.3}$$

If electrical resistance heating is substituted for the conventional fossil fuel sources, the heating component of the objective function becomes:

$$C_h = \frac{60.Q(\rho_a c_a + \rho_w c_w)(T_{inlet} - T_{amb} - \Delta T)P_E t}{3413.} \tag{I.C.4}$$

The cost component for airflow arises from the electrical energy used by the fan motor. The trend toward higher horsepower electrical motors on farms, as well as the disadvantage of having an idle tractor, has almost obviated the use of the power take-off for this application. Theoretical horsepower required on a per square foot basis for the motor-fan combination is (Hall,1957):

$$HP = \frac{Qp}{6350.} \tag{I.C.5}$$

Using Equation (I.C.2) to calculate pressure drop, the cost of electricity for running the fan and motor is:

$$C_a = \frac{0.746\ (HP)P_E t}{\eta_f \eta_m} \tag{I.C.6}$$

Total operating cost per square foot is then given by the sum of the heating and airflow costs (Equations (I.C.1) or (I.C.4) and (I.C.6)). The objective function thus has the following form:

$$F = C_h + C_a \tag{I.C.7}$$

D. Optimization Technique

The following independent variables abstracted from the drying model and objective function are known to influence the fixed bed dryer economic performance: initial and final moisture contents; bed depth; elapsed drying time; ambient air conditions; fuel and electrical prices; thermal, fan and electrical-mechanical conversion efficiencies; inlet air temperature; and airflow rate. Assuming that a dryer has been purchased and harvesting policy decided upon, only inlet air temperature and airflow rate remain amenable to control. Viewed in this perspective, the problem reduces to a two-independent-parameter minimization of the cost function, subject to constraints on inlet air (or grain) temperature, airflow rate, final moisture content, and time.

The first optimization technique used on the problem was the pattern search technique HCLMB. This method, for several reasons, was unsatisfactory for the present problem. For two reasons, convergence to an optimum was very slow: (i) because the optimum was necessarily located on the final moisture content isostere, and (ii) because the line of constant final average moisture (isostere) and the iso-cost lines were often essentially parallel near the optimum. In addition, it was difficult to ascertain the relationship between the isosteres and isocost lines in the vicinity of the optimum. Also, the technique did not indicate whether convergence was to a local or global optimum, leaving open the possibility of a line of alternate optimal solutions or a better solution than that initially found. Because of these difficulties, it was advantageous to develop a specialized optimization technique which exploits the characteristics of the given problem. This algorithm will be explained after some preliminary remarks.

One special feature of this problem is the use of the final average moisture content isostere as a constraint. Unlike the constraints specified on airflow and inlet air temperature, the shape and location of this isostere are not known beforehand but must be evaluated during the course of solution.

Consideration of the drying process indicates that the isosteres when graphed on inlet air temperature-airflow rate coordinates, must exhibit a shape which is concave upward. That is, at high inlet air temperatures and low airflow rates additional heat has little effect on average moisture content; at high airflow rates and low temperatures, additional air does not appreciably lower the average moisture content. For the iso-cost lines, analysis of the objective function, Equation(I.C.7), shows that for high temperatures and low airflow rates, the near-linear fuel cost term predominates, while at low temperatures with high airflow rates, the non-linear electrical cost term takes precedence; therefore, the shape of the iso-cost lines is also concave upward.

Given these strictures, four different relationships between the final average moisture content isostere and isocost lines are possible (See Figure 2).

For Case I, the slope of the isocost line remains flatter than that of the isostere throughout the region. The unique least cost optimum occurs on either the maximum temperature or minimum airflow rate bound. Case II, the inverse situation occurs on either the minimum temperature or maximum airflow rate bound.

In Case III, the isocost lines cross the isostere at two locations within the region of interest but are flatter than the isostere at the left or top bound and steeper at the right or bottom bound. Since the isocost lines are essentially parallel, it is evident that the optimum will occur on the interior of the region and, depending on the slopes of the functions, may exhibit a line, rather than a point, solution.

Case IV is similar to Case III except that the isocost lines are flatter than the isostere on the bottom or right-hand side and steeper on the top or left hand side. In this situation two local optima occur. The minimum of the two (global optimum) depends on the particular configuration of the problem.

With the foregoing background the mechanics of the algorithm can be explained. In the first step the final moisture contents at each of the four corners of the region are determined. Since all isosteres exhibit the same shape, within any feasible region two and only two bounds will be crossed by the final average moisture content isostere (disregarding the unlikely degenerate case where it intersects a corner of the region). Examination of the four corners determines along which bounds to search for the crossing points. For this purpose the method of Dekker(1967) was chosen, due to its property of supralinear convergence

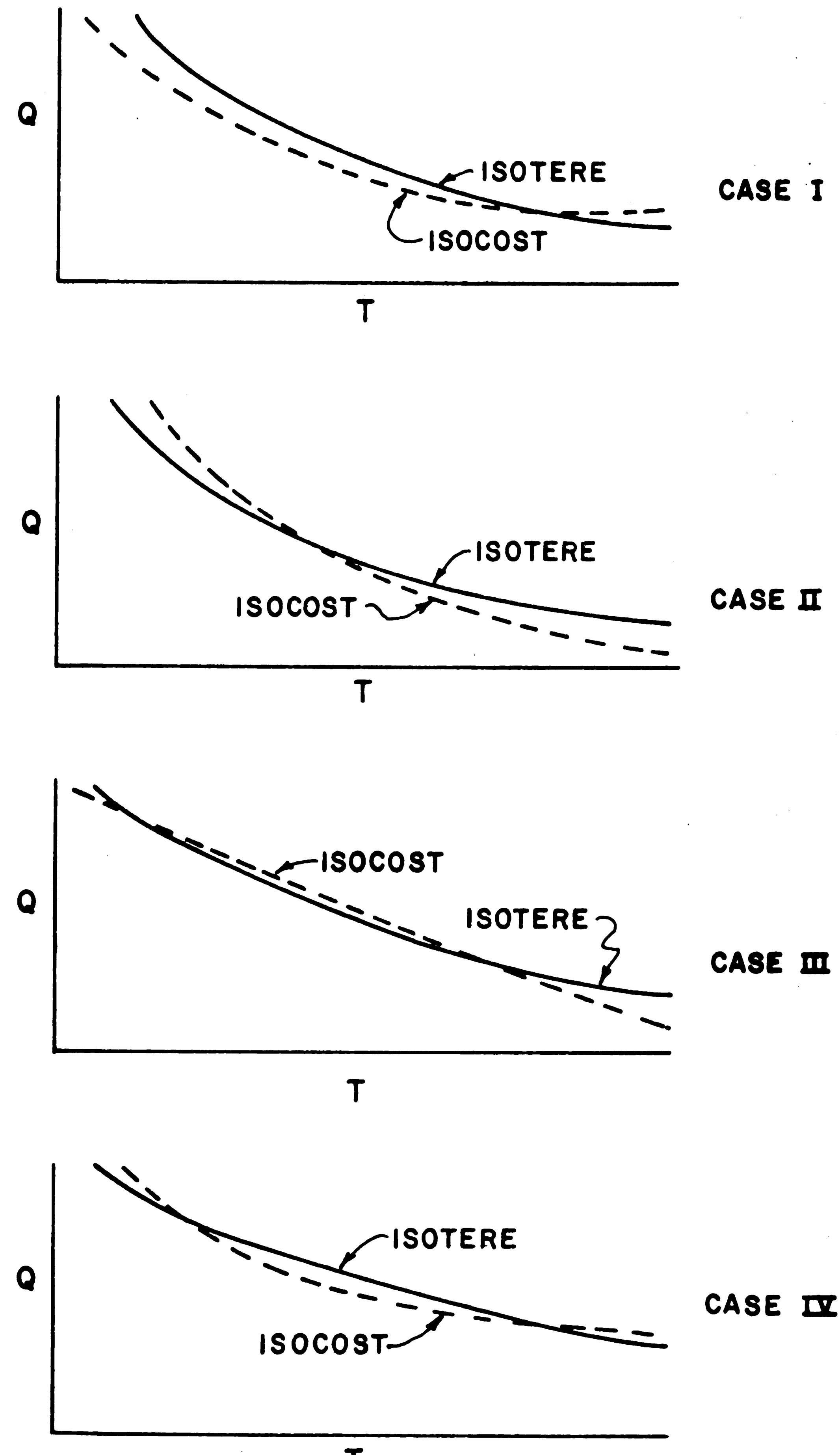

Figure 2: Possible Isotere-Isocost Configurations

to the zero of an unknown function. Thirdly, at both crossing points the slopes of both the isostere and isocost lines are sampled. By comparison of slopes, Cases I and II can be immediately recognized and the optimum identified. Case IV requires, in addition, a comparison of the two local optima for identification of the global optimum. In the remaining situation, Case III, the optimum occurs in the interior of the region; hence, further exploration is necessary. In this phase of the algorithm, the equation of the diagonal from upper right to lower left corner of the region is first determined. Again using the one-dimensional search, the crossing point of the desired isostere is located and the slopes of both the isostere and isocost line sampled. If the cost slope at that point is steeper than that of the isostere, the optimum must occur to the left and above the point sampled; hence a new and smaller rectangular region is defined containing the optimum. In like manner, if the cost slope at the point is flatter than that of the isostere, the optimum must occur to the right and below the point sampled; thus, a large portion of the region may be discarded in this step of the search. The preceding method is repeated on successively smaller regions until the optimum is approached arbitrarily closely. Since convergence takes place from both sides of the optimum a line of solutions may be obtained.

For design purposes, it may be desired to investigate cost per bushel as a function of depth or time. By imbedding the foregoing method into a minimization search with respect to either of these, its applicability can be extended. Alternatively, the method can be used selectively in order to survey a number of possible design combinations without a formal search.

E. Constraints and Parameter Values Needed for Input

In order to properly use the optimization procedure, constraints must be provided on the two independent variables, inlet air temperature and air flow rate. Because the model is primarily derived for inlet air temperatures less than 140 deg F, this is suggested as a maximum upper bound. Hall(1957) recommends air temperatures of less than 140 deg F for grain to be milled. The lower limit on air temperature is arbitrary; a lower bound of 70 deg F was chosen. An airflow range of 10-50 cfm/bushel is considered normal for this type of dryer; the airflow constraints may be chosen accordingly. In addition to deterioration in quality due to overheating of the bottom layers of grain, molding may occur in storage if the moisture content of the top layers of the drying bed is too great, even though the average moisture content after mixing is satisfactory. Maddex(1971) suggests a maximum moisture content of 16% wet basis in the upper layer of the drying bed if shelled corn is to be stored on a long-term basis; this boundary condition option can be inserted into the solution if desired.

In order to specify the cost function, several parameters related to the dryer components and operating conditions must be known. Of the three efficiencies inherent in the objective function, only fan efficiencies are normally specified in manufacturers literature; electric motor efficiency curves, however, can be developed by simple tests independent of the dryer configuration. The thermal efficiency determinations for a given dryer are a function of both the burner chosen and the air distribution system design. Thus this parameter must be measured by tests on the dryer itself; for simple calculations these tests can best be made without drying (simple heat transfer).

For general use of the algorithm, the following guidelines are suggested:

Thermal efficiency of heater:

direct fired, fall operation	70%	(Hall,1957)
direct fired, winter operation	50%	
indirect fired, fall operation	50%	
indirect fired, winter operation	30%	

Electrical-mechanical conversion efficiency of motor:

65-85% (Anon.,1958)

Fan efficiency:

40-70% (Perry et al.,1963)

Heating value:

Natural gas (methane)	22,190 Btu/lb
Propane	19,944 Btu/lb

(Hall,1957)

For the simulation models, as well as for the cost function, it is necessary to know the expected ambient air temperature and relative humidity (or equivalent property). The ASAE Yearbook(1971) features monthly weather maps for the continental United States, giving mean wet and dry bulb temperatures and standard deviations from the mean. The algorithm is written such that this information can be substituted directly if desired. Mean barometeric pressure for a particular location may be estimated from:

Altitude, feet	Mean barometric pressure, in. Hg
0	29.921
500	29.38
1000	28.86
5000	24.89

The remaining parameters necessary in the economic dryer model include: electrical price, fuel price, initial and final moisture contents, bed depth, and drying time. These inputs vary according to the prevailing economic climate and the purpose of the user.

F. Program Description and Use of the Algorithm

The computer implementation of the algorithm may be applied in several ways, subject to the discretion of the user. Available options, which are specified on the data input cards, include: 1) a single simulation by the QUICK model, with or without a cost evaluation, 2) cost optimization with either inlet temperature or airflow rate variable, and 3) cost optimization with both temperature and airflow rate variable.

To further facilitate use of the program, input data values can be specified in any common units (e.g. either wet or dry basis moisture contents), with conversion, if necessary, being done by the program. The flowchart, Figure 3, is written to allow the user to easily supply the necessary data cards.

The computer program, with accompanying comment cards, is listed in Appendix A. The psychrometric subroutines have been excerpted from Lerew (1971); the equilibrium moisture content subroutine is from Bakker-Arkema et al.(1971).

Several diagnostic messages have been included in the program, in the event of user difficulty. A brief explanation of each of these follows:

(1) Messages:

MOISTURE CONTENT IS NOT IN THE NORMAL RANGE

AIRFLOW RATE INPUT IS NOT IN THE NORMAL RANGE

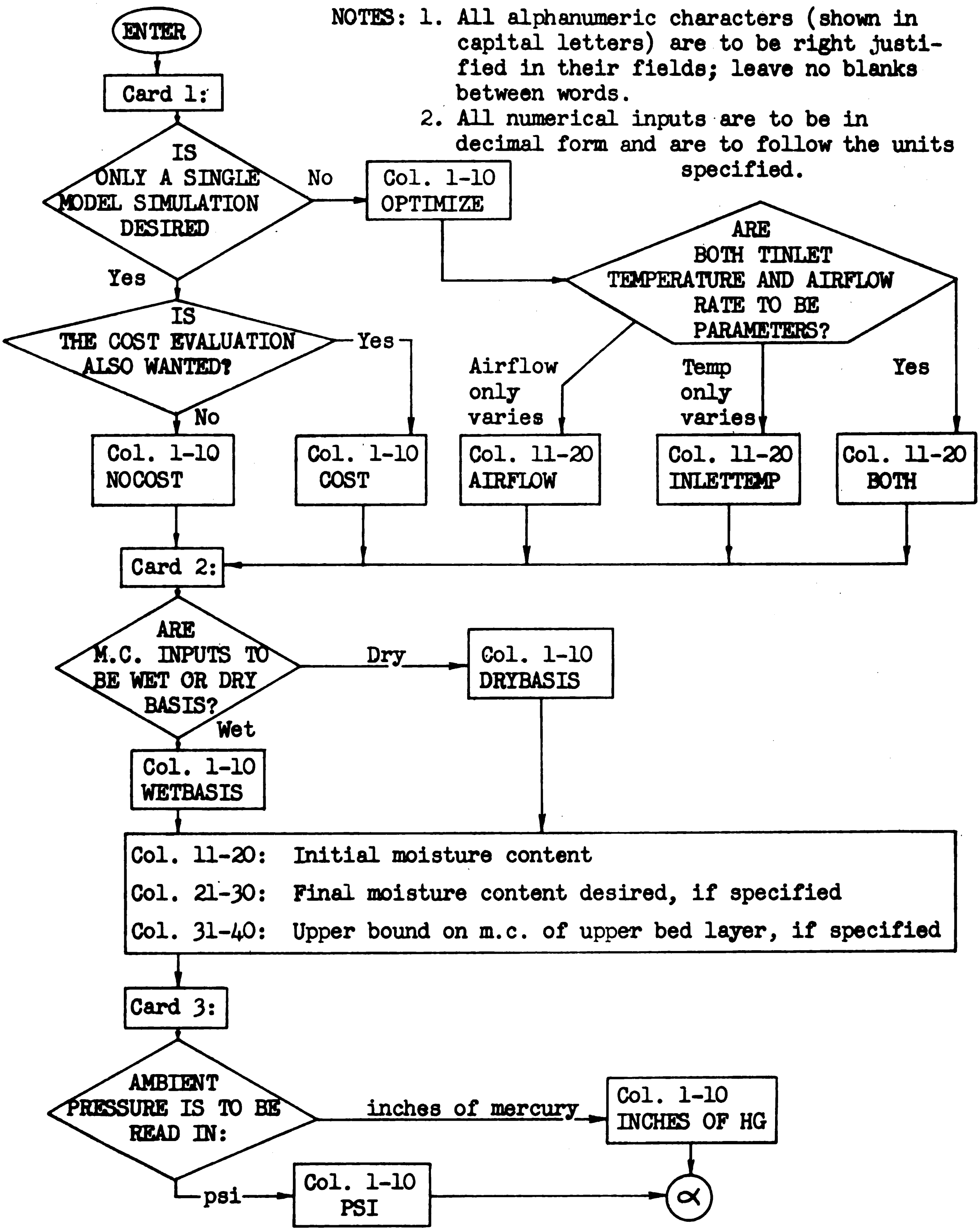

Figure 3: Flowchart of Necessary Data Cards

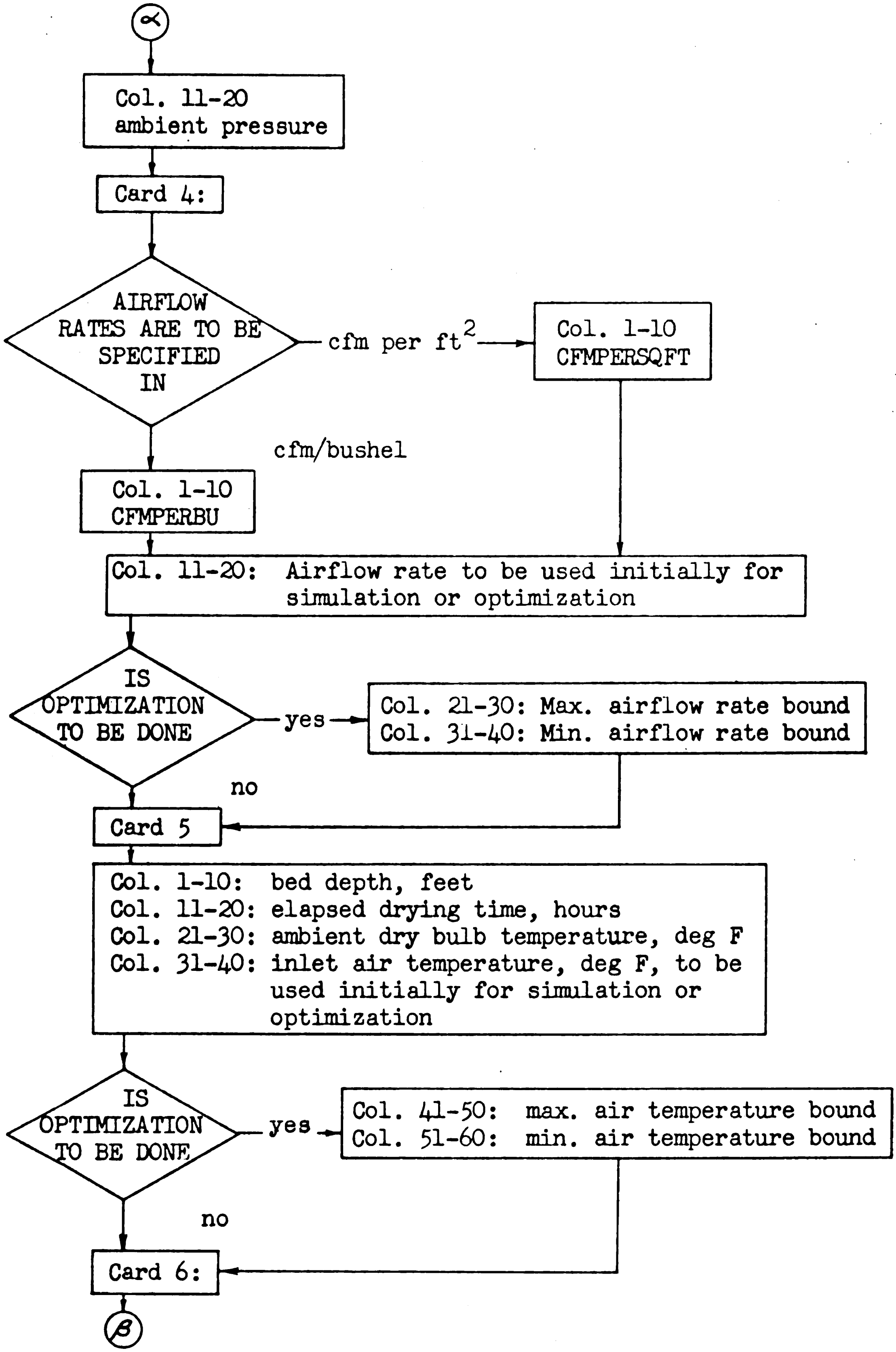

Figure 3 (cont'd)

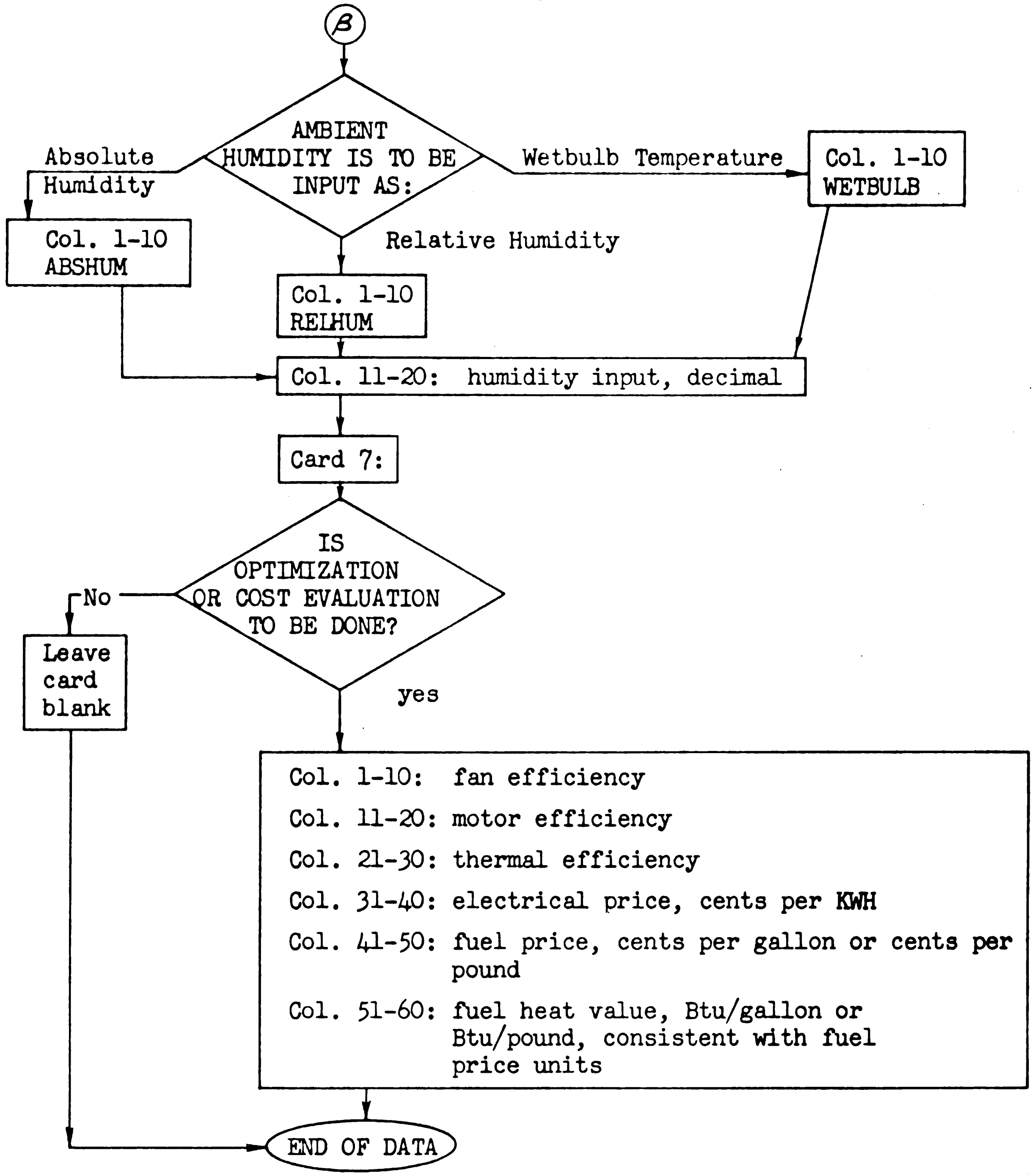

Figure 3(cont'd)

WETBULB TEMPERATURE INPUT IS NOT IN NORMAL RANGE

ABSOLUTE HUMIDITY IS NOT IN THE NORMAL RANGE

RELATIVE HUMIDITY INPUT IS INCORRECT

Explanation:

The reliable working range of the model has been exceeded by input data. Change the parameter specified.

(2) Messages:

AIRFLOW RATE BOUNDS ARE INCORRECT

INLET AIR TEMP BOUNDS ARE INCORRECT

Explanation:

The limits permissible for optimization, inlet air temperatures between 70 and 140 deg F and airflows between 10 and 50 cfm/ft^2, have been exceeded.

(3) Message:

THE MC OF THE TOP LAYER CANNOT REACH THE MAX PERMISSIBLE MC WITHIN THE STATED BOUNDS

Either the upper limit on airflow rate is too low to permit sufficient drying of the top layer or the bound on moisture content for the top layer is too stringent.

(4) Message:

IT IS IMPOSSIBLE TO REACH THE DESIRED MC WITHIN THE STATED BOUNDS

Explanation:

Even at the highest values of airflow rate and temperature, the specified average moisture cannot be attained.

(5) Message:

THE MODEL CANNOT HANDLE THIS CASE

Explanation:

The search for the correct vapor pressure at the temperature T_G was unsuccessful; therefore, the model QUICK cannot be used for the given input data set.

(6) Message:

THE FINAL MC ISOSTERE CROSSES OTHER THAN 2 BOUNDS - THE METHOD FAILS

Explanation:

This may be caused by an airflow rate bound which is too low. Reset the bound and rerun the trial.

G. Sensitivity Studies

To illustrate an application of the algorithm, a set of economic and climatic conditions typical of the Central Michigan area was chosen and the sensitivity of the optimal solution to expected variations in these parameters was explored. The following sections summarize and interpret the results obtained.

G.1. Standard conditions and results

For the month of October the Central Michigan area experiences, on the average, a dry bulb temperature of 49.5 deg F and wet bulb temperature of 45 deg F; mean barometric pressure is approximately 14.34 psi (ASAE Yearbook,1971). The prevailing 1971 LP gas and electrical prices are respectively, 16.5¢/gal and 2¢/KWH, when purchased in amounts ordinarily consumed by individual farms. Mid-range efficiencies were chosen for the fan (55%), electrical-mechanical conversion of the motor (75%), and heating system (70%). An operating policy of 10 hours (2 batches/day) at two foot depth was chosen arbitrarily, with a uniform harvest moisture content of 28.0% w.b. to be reduced to an average of 15.5% w.b. for marketing or storage.

Figure 4 shows the set of inlet air temperature-airflow rate

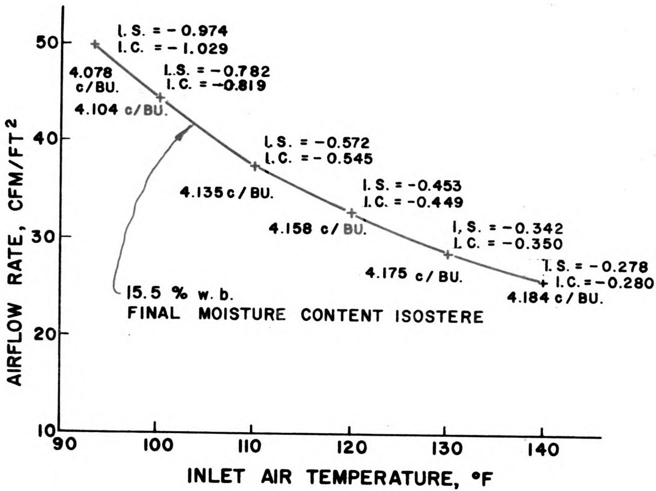

Figure 4: Drying Costs and Corresponding Isostere-Isocost Slopes for Standard October Tests

conditions for which the desired average final moisture content is achieved. The corresponding slope of some isocost lines and their values at selected intervals are also noted. With this choice of parameters, the optimal operating cost will occur on either the minimum temperature bound or the maximum airflow rate bound (Case II), since the slope of the isocost lines is steeper than that of the isostere anywhere within the feasible region. The optimal total cost/bu of 4.08¢ indicates a saving over the highest non-optimal cost within the feasible region, 4.18¢. However, relaxation of the maximum airflow rate bound would allow further improvement.

If the additional constraint of a maximum moisture content of 16.0% w.b. is applied, it is impossible for the top layer of the bed to reach this moisture and still satisfy the remaining conditions on the problem; within the bounds of the region shown (Figure 4), the top layer reaches, at best, 20.0% w.b. Either more time must be allotted to the process, the airflow rate bound increased, the depth decreased, or both in order to satisfy this top layer constraint.

G.2. Effect of the heating price:airflow price ratio

The value of an isocost line at a given point of intersection with an isostere will of course be altered by a change in the price of energy for heat or airflow, a change in thermal, electric motor or fan efficiencies, or any combination of these. The slope of the isocost lines at the intersection also varies with a change in the heating price: airflow price ratio (a change in efficiency is effectively a change in price). If the change in slope is sufficient to change the optimization problem from one case to another (Figure 2), the location of the minimum, as well as its value, may be drastically altered. In Table 3, the

Table 3: Results of Parameter Studies (unless otherwise noted, conditions are same as Standard Test, Trial A)

Trial and Comment	Initial Moisture W.B.	Bed Depth Ft.	Drying Time, Hrs	Fan Eff.	Motor Eff.	Thermal Eff.	Electric Price ¢/KWH	Fuel Price ¢/gal	Amb. Drybulb Temp Deg F	Amb. Wetbulb Temp	Final Moisture W.B.	Atm. Press. PSI	Temp Bounds Deg F	Airflow Bounds cfm/ft²	OPTIMAL COST ¢/ft²	OPTIMAL COST ¢/bu	Optimal Temp Deg F	Optimal Airflow cfm/ft²	Final M.C. in Top Layer, W.B.
A Standard Test (Oct.)	0.28	2.	10.	0.55	0.75	0.70	2.	16.5		45.	0.155	14.34	70-140	5-90	6.53	4.08	93.62	50.	0.198
B Min. Electric Eff.				0.40	0.65										6.67	4.17	93.62	50.	0.198
C Max. Electric Eff.				0.70	0.85										6.45	4.03	93.62	50.	0.198
D Indirect Fired						0.50									8.95	5.60	93.62	50.	0.198
E. Min. Electric Price							1.5								6.41	4.01	93.62	50.	0.198
F. Min. Fuel Price								14.							5.64	3.50	93.62	50.	0.198
G Max. Fuel Price								17.							6.71	4.19	93.62	50.	0.198
H Min. Total Price							1.5	14.							5.49	3.43	93.62	50.	0.198
I Electric Heating							2.0	-							14.27	8.92	93.62	50.	0.198
J Good Oct. Weather									53.	46.					5.89	3.68	92.92	50.	0.198
K Poor Oct. Weather									49.5	49.					6.85	4.28	140.	26.37	0.198
L Mean Nov. Weather									36.	33.					7.71	4.82	140.	25.23	0.198
M Sea Level Pressure												14.70			6.69	4.18	93.73	50.	0.198
N Variation in depth		1.													5.03	6.29	83.99	50.	0.175
O Variation in depth		1.5													5.77	4.81	83.74	50.	0.187
P Variation in depth		2.5													7.31	3.66	98.71	50.	0.209
Q Fan Cost Added		1.													5.72	7.15	140.	21.69	0.177
R Fan Cost Added		2.													7.05	4.41	140.	25.81	0.199
S Variation in Time			6.												6.39	4.00	140.	40.79	0.203
T Variation in time			8.												6.49	4.06	104.79	50.	0.200
U Variation in time			12.												6.49	4.05	85.75	50.	0.196
V Variation in time			16.												6.17	3.86	74.88	50.	0.195
W Variation in time			20.											10-50	5.90	3.69	70.	46.99	0.194
X Alternative Marketing Option	0.28										0.20				3.64	2.25	73.12	50.	0.237
Y Alternative Marketing Option	0.25										0.20			10-50	2.29	1.43	70.	37.23	0.220
Z Case III			20.	0.40	0.60	0.80	2.5	10.							4.12	2.57	92.84	28.55	0.198

sensitivity of the optimum to changes in several parameters is shown. The first set of results stems from changes in the heating price: airflow price ratio (Trials A through I):

1. For a Case I type isocost-isostere configuration the goal of economy of operation (associated with high inlet temperatures and low airflow rates) is opposed to the goal of high grain quality (provided by low inlet temperatures and high airflow rates); an overall economic model must simultaneously consider these contradictory objectives. In a Case II situation, the two goals are in harmony. No general statement can be made about configurations III and IV.

2. Fuel cost is by far the largest contributor to total operating cost in these trials; electrical cost plays only a minor role (a maximum of 7%). In terms of dryer design, this implies that little is to be gained by improvement of fan-motor efficiency. Emphasis should instead be placed on improving thermal efficiency. Similarly, any possible changes in the electrical price will not substantially affect the total operating cost of gas-fired dryers; however, consideration of the price structures and heat values for competing fuels may lead to a sizeable operating cost reduction.

3. Replacement of fossil fuel burning heaters by electrical resistance heaters is economically unsound under the prevailing electrical: fuel price ratio.

4. Indirect fired dryers operate at a substantially higher total cost per bushel than do direct fired dryers, due to the large contribution of fuel cost to total cost and the inefficiency of the heat exchanger.

G.3. Effect of variation in ambient conditions

Using values of wet and dry bulb temperature one standard deviation from their October mean values and holding all other parameters at their standard values, the variation in the optimal solution due solely to expected weather fluctuations was investigated (Table 3 - Trials J and K). The average ambient conditions for November were likewise used, with the remaining parameters held constant (Trial L). Finally the effect of atmospheric pressure on the optimal solution was explored (Trial M).

The effect of changes in ambient conditions is to shift the location and orientation of the average final moisture content isostere relative to the isocost lines. From consideration of Equations (I.C.1) and (I.C.6), increased humidity and decreased ambient air temperature lower the positions of the isocost lines on the inlet air temperature - airflow rate graph; this shift is uniform, due to the nearly constant specific volume of air at any ambient dry bulb temperature encountered during the drying season.

A comparison of the results of Trial A in Table 3, in which an ambient absolute humidity of 0.005 lb water vapor/lb dry air is used, with that of Trial K, using 0.0075 lb water vapor/lb dry air, shows an increase of 0.2¢/bushel at the higher humidity, using the same 49.5 deg F ambient dry bulb temperature. The slope of the latter isostere is steeper than that of the former at any comparable point. Clearly the comparatively adverse drying conditions of Trial K also shift the isostere upward.

The analogous situation, in which absolute humidity is held constant and ambient dry bulb temperature varied, is seen in Trials A and J. Maintaining an absolute humidity of 0.005 lb water/lb air and raising the dry bulb temperature from 49.5 deg F to 53.0 deg F lowers the optimal drying costs by 0.3¢/bushel, decreases the slope of the isostere, and shifts it downward.

Since the slope of the isostere becomes steeper due to adverse weather conditions, while the slope of the isocost lines remains the same, the <u>percentage</u> contribution of the airflow component to total cost increases. A season-long study of drying cost fluctuation, however, would account for the decline in the initial moisture content due to field drying, which would partially offset the increase in drying costs due to poor weather in the latter months.

An increase in atmospheric pressure to 14.70 psi, as shown by Trial M, results in increased drying costs over the Central Michigan average of 14.34 psi (Trial A). However, the magnitude of the increase, 0.1¢/bu, is relatively small compared to that due to expected monthly variation in wet and dry bulb temperatures.

<u>G.4. Effect of variation in grain depth</u>

The depth of grain in a deep bed dryer is a major consideration in design since increased batch volume can be achieved only by increased structural costs, either horizontal or vertical. Furthermore, harvesting machinery capacity and dryer operating schedule will influence the depth of grain to be dried in a given system. In order to analyze the effect of increasing depth on dryer operating performance, Trials A, N, O, and P (Table 3) utilized depth as a parameter, with all other conditions standard.

To properly perceive the problem, it is useful to visualize it in three dimensions (Figure 5). The three-dimensional representation of isocost and final average moisture isostere surfaces depicts the mode of intersection inferred from the simulations. Any cross-section of the inlet air temperature-airflow rate plane will exhibit one of the four cases shown in Figure 2.

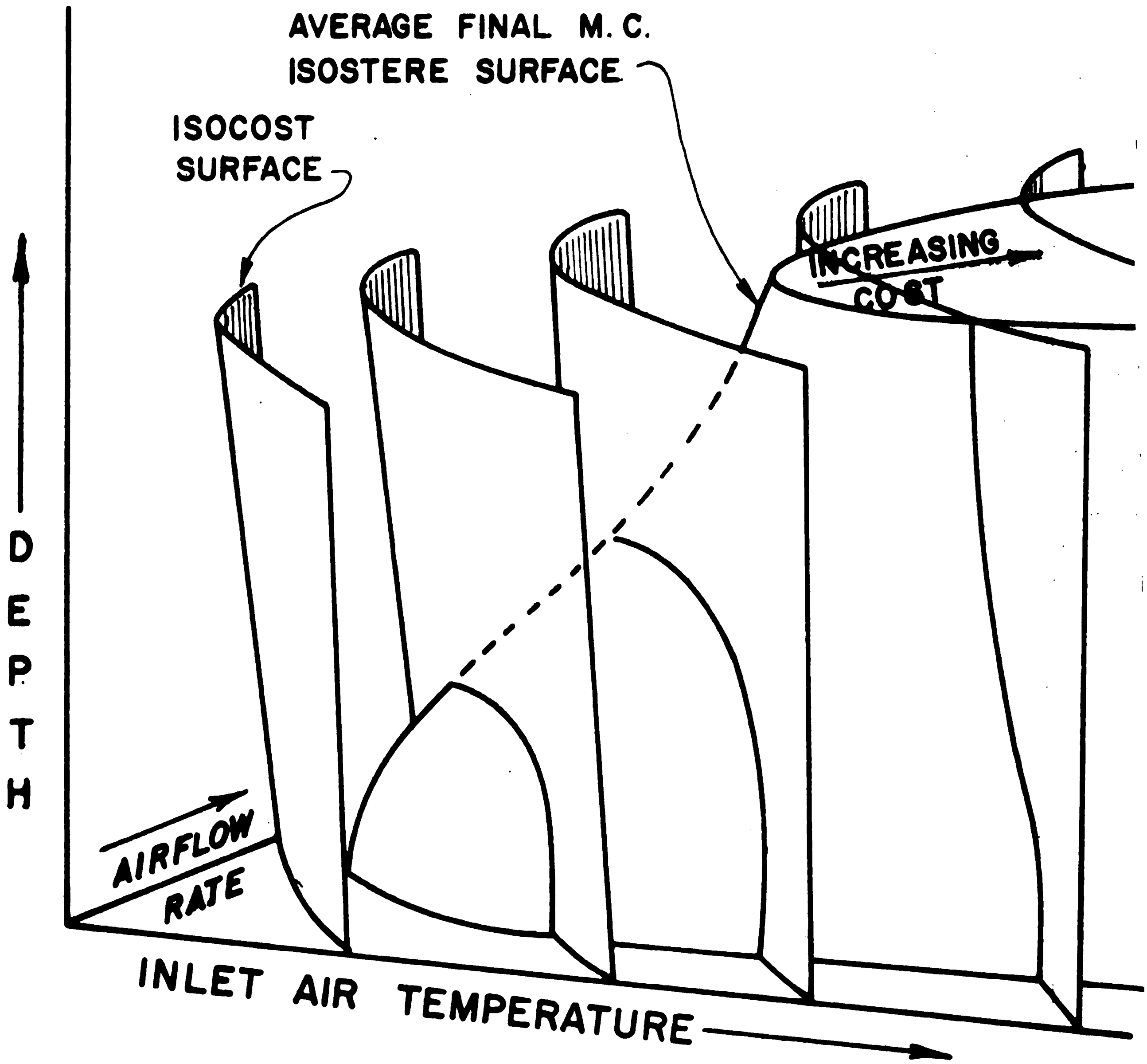

Figure 5: 3-Dimensional Schematic View of the Economic Behavior of the Model with Increasing Bed Depth

The simulation results show a pattern of decreasing cost per bushel with increased depth, but with a decreasing rate of change, i.e. the incremental gain in dryer efficiency becomes smaller as depth increases. In these trials, the model was found to be inadequate at depths greater than 2½ feet, presumably because excessive condensation in the upper bed occurred at the low airflow rates specified by the optimal solution. At the 2½ foot depth, the rate of improvement in per bushel cost was very small; since condensation is more pronounced at greater depths, per bushel operating cost can be expected to begin increasing as depth is increased much beyond 2½ foot level.

In order to use the algorithm for optimal design it is necessary to balance construction costs against performance for a dryer of specified volume. The following example, illustrating two hypothetical dryers of equal volume but different geometries, demonstrates the relative magnitude of capital and operating costs for each as well as the net cost differential between the two.

Assume a corn harvesting system with a 9000 bushel per year throughput and a specified 10 hour per day drying schedule. The depth of Dryer I is arbitrarily chosen to be one foot and its radius 15 feet. Dryer II is filled to a depth of two feet and has a corresponding radius of 10.6 feet, (Trials Q and R, respectively, in Table 3). Assuming that burner capital costs will be approximately equal for the two solutions, the total cost components which vary between the two dryers are: fixed cost of fan and motor, operating cost, horizontal structural costs, and vertical structural costs.

Using the fixed cost coefficients for layer dryers given by Morey et al.(1969), and the operating cost evaluation of the QUICK model, a comparison of the two dryers was made. Due to the similarity in construction of layer and batch-in-bin dryers, adoption of the layer dryer fixed cost coefficients for this example is a reasonable approximation.

Because specification of dryer dimensions automatically fixes the construction cost components, these were not imbedded in the operating cost function. Fan and motor cost, however, is a function of operating conditions and hence was incorporated in the operating cost optimization. For this example, the standard October operating conditions were taken as average for the entire harvesting season. A comparison of component and total costs is shown in Table 4.

It is evident in the example that horizontal construction costs and operating costs are the major contributors to total annual dryer cost. As shown previously, the per bushel drying costs are decreased by increasing the bed depth (at least within a range of depths). The decreased horizontal surface area gives an economic advantage to the deeper dryer. However, as depth increases, the moisture content of the top layer becomes too high for safe long-term storage.

In an actual fixed-volume design problem, depth could be expressed as a parameter and the algorithm restructured to search for minimum total annual cost, with time-variant constraints and climatic and economic conditions specified by the designer. In this manner, dryer design could be tailored to specifications of the locale for which it was intended.

G.5. Effect of variation in required drying time

One means of increasing system capacity is to dry two batches daily, rather than one. It is also conceivable that a three batch schedule could be employed, although the final moisture gradient for a bed of the same depth would not favor long-term storage; materials handling considerations also favor a fewer number of batches per day.

Table 4: Comparison of Annual Dryer Costs for Two 9000 bu/yr Systems

Dryer I: Depth = 1 foot; radius = 15 feet; standard October drying conditions.

Construction costs, horizontal	=	46¢/ft^2/yr x 707 ft^2	=	\$325.
Construction costs, vertical	=	10¢/ft^2/yr x 90 ft^2	=	\$ 9.
Fan-motor cost	=	\$10/HP/yr x 1.3HP	=	\$ 13.
Optimal operating cost	=	7.1483¢/bu x 9000 bu/yr	=	\$643.
(Trial Q)		Total	=	\$990.

Dryer II: Depth = 2 feet; radius = 10.6 feet; standard October drying conditions.

Construction costs, horizontal	=	46¢/ft^2/yr x 352 ft^2	=	\$162.
Construction costs, vertical	=	10¢/ft^2/yr x 130 ft^2	=	\$ 13.
Fan-motor cost	=	\$10/HP/yr x 2 HP	=	\$ 20.
Optimal operating cost	=	4.4077¢/bu x 9000 bu/yr	=	\$397.
(Trial R)		Total	=	\$592.

To examine the effect on operating cost of the drying schedule alone, Trials P and W of Table 3 were compared. Using standard October conditions, drying a two foot bed for twenty hours incurred a cost of 5.90¢/ft^2. An equivalent daily volume can be dried in two one-foot deep batches each using a 10 hour drying time; the operating cost for this process is 2(5.03) = 10.06¢/ft^2. Thus a saving of 4.16¢/ft^2 or 2.60¢/bu is realized by the twenty hour schedule. If this figure were the average over the entire drying season for a 10,000 bu/yr operation, the yearly savings would be \$260. A 5-year cost amortization for the dryer would then allow \$1300 breakeven cost to be spent for the additional height needed to dry by the twenty hour single batch schedule.

For a single drying depth it is also of interest to examine the change in operating cost with increased required drying times. The Trials A (10 hours), and S (6 hours) of Table 3 correspond to doubling and tripling the system capacity, respectively, over the capacity for the 20 hour process Trial W. Three intermediate required drying times (Trials T, U, V) were also simulated for additional cost information.

Again, a 3-dimensional schematic view (Figure 6) is useful in interpreting the simulation results. Because operating cost increases linearly with time (for fixed inlet air temperature, airflow rate, depth, ambient conditions, and price structure), the isocost surfaces change with time as shown. According to the model, drying costs vary from a local minimum of 6.39¢/ft^2 in a 6 hour test, to a maximum of 6.53¢/ft^2 in a 10 hour test, and again to a local minimum of 5.90¢/ft^2 for a 20 hour test. For this example, the alternation implies that at least two lines of intersection exist between some of the isocost surfaces and the constant final average moisture content surface. It is also possible, with a

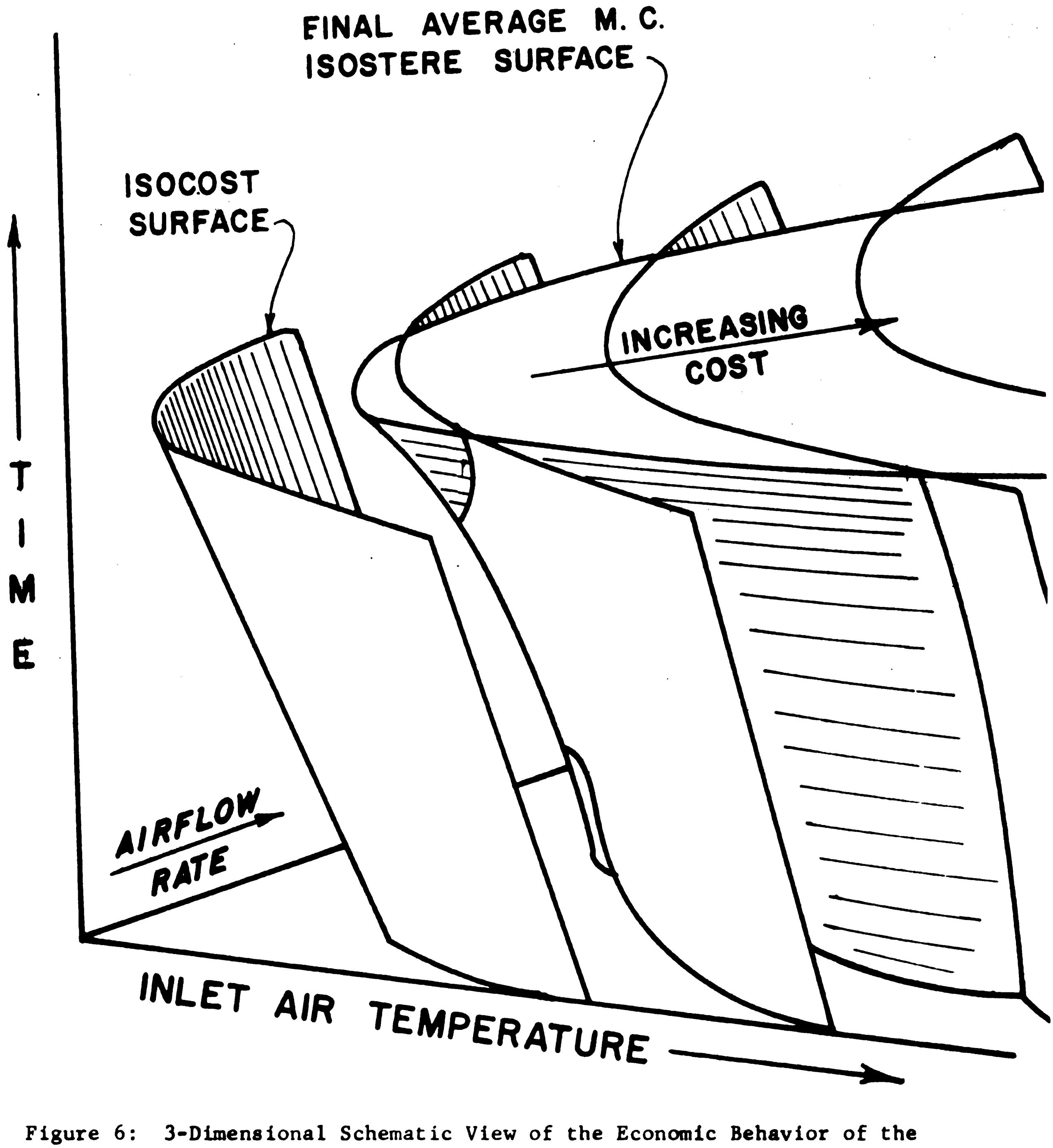

Figure 6: 3-Dimensional Schematic View of the Economic Behavior of the Model with Increased Elapsed Drying Time

flatter slope of the isocost surfaces relative to that of the isostere surface for the minimum total cost to occur at the minimum time bound; for a steeper slope, the minimum total cost will pass to the upper time bound.

Due to the small change in operating cost with time, the duration of drying for a single batch is probably of minor importance. Consequently the dryer operating schedule can be principally determined by the considerations of increased system throughput with a multiple-batch schedule or increased grain quality possible with a single batch per day.

G.6. Effect of variation in marketing options

A suggested use of the algorithm is to assist in investigation of the profitability of various marketing alternatives open to the grower. Several of the common choices are shown in Figure 7 (Scott,1969). To incorporate this model into a overall economic picture, it would be necessary to also include aeration, storage, and handling costs, a penalty (dockage) function for failure to reach the specified storage moisture, plus seasonal variation in rate of harvest, harvest moisture, and market prices. Nonetheless, insight into the relative profitability of marketing options can be gained by experimentation with the present algorithm.

Trials A, X and Y (Table 3) illustrate three of the options suggested by Scott, using the standard set of parameters. For comparison, drying to 15.5 and 20.5% w.b. (Scott, 1969), a net loss in operating cost of 7.17¢/bu is sustained on each bushel of corn sold at 20.0% w.b. The third option, Trial Y, allows the corn to field dry to 25.0% w.b. before harvest, after which it is artificially dried to 20.0% w.b. for marketing.

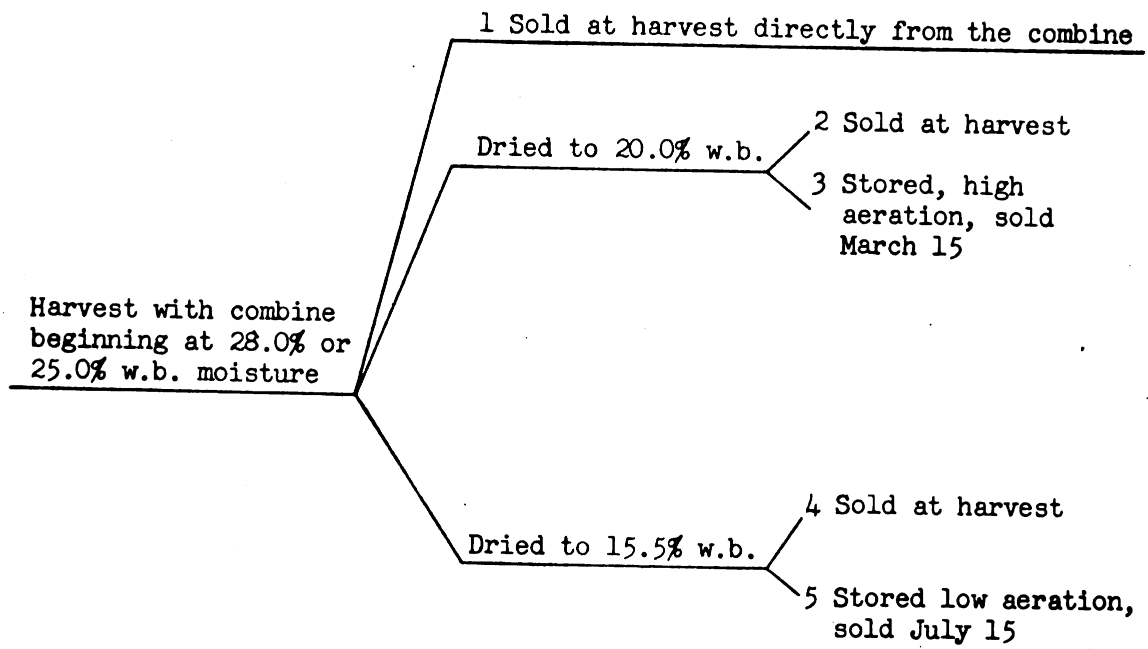

Figure 7: Common Marketing Options for Shelled Corn

Although this alternative cuts the operating cost to 1.43¢/bu the large dockage penalty remains and the net difference per bushel is 6.35¢ over the cost of the first option.

The foregoing example demonstrates a simple marketing analysis which can be easily made using the algorithm. Because the optimum can be located by the algorithm very quickly, the model can also be combined with weather and crop data in large-scale models to explore seasonal harvesting and marketing strategies.

H. Observations of the Performance of the Algorithm

Using the criterion for acceptable model performance (Section I.B.), admissible results were obtained with all values of input parameters shown in Table p30 3. However, in Trials W and Y it was necessary to raise the lower airflow bound to 10 cfm/bu. This procedure did not affect the final result but was necessary in order for the algorithm to begin the solution properly; such a failure is indicated by a program diagnostic. In separate tests, a model deficiency was also detected at low inlet air temperatures and airflow rates of 70 cfm/bu. The weakness of the model in this region will ordinarily b~~y~~e avoided by use of the suggested upper airflow rate bound of 50 cfm/bu.

The optimization procedure, using the data presented, encountered Case I, II, and IV type solutions (Figure 2) during the Trials A-Y. As mentioned previously, the location of the optimum depends on the relative slopes of the final moisture content isostere and the isocost lines. If their point of tangency lies outside the feasible region, as in these trials, the optimal solution will occur on a bound; with this condition, a small change in the constraints, final moisture content isostere, or isocost lines can shift the optimum from one boundary of the feasible region to another. Case III, that of an interior optimum, is shown by Trial Z, using a similar data set. It is therefore necessary to retain the generality of the algorithm, since any one of the four cases may occur in a particular solution.

An indication of the speed of the algorithm is given by its performance in the search of Trial Z, Table 3. For this search, requiring 10 reductions of the initial feasible region, execution time was 3.7 seconds using a CDC 6500 computer.

I. Proposal for Dryer Testing

As shown in prior sections, the model and optimization techniques can be used to investigate the adaptability of the dryer to individual operations or to study the effects of one or more variables on dryer performance. These applications, however, require access to a computer and manipulation of input parameters; at the present stage of farming sophistication, this may be too difficult or inconvenient for many grain growers. Despite these drawbacks, this package and similar programs for other dryers can still be useful tools for extension workers by enabling them to quickly generate data which can be transformed into useful publications.

In this light, there are two unfulfilled needs to which this type of model may be applied: (i) a mapping of the expected relative performance of each dryer type with respect to climatic variation, and (ii) development of meaningful uniform rating procedures for performance comparison of competing dryers.

For both applications, at least two indices of performance are important for comparison: capacity and efficiency. Because of differences in weather conditions, energy prices, and operating parameters, it has heretofore been difficult to compare the results of tests of similar dryers; in the following discussion a standardization of test conditions is proposed which, through computer simulation and limited testing, will permit comparative studies to be made.

A means of localizing dryer tests with respect to climate is suggested by the air conditioning design procedure of the American Society of Heating, Refrigeration, and Air Conditioning Engineers. For this system, long-term nationwide weather records for the season of interest have been analyzed to determine the wet and dry bulb temperature below which 95% of the total hours of the season may be expected to fall; these temperatures and their corresponding locations are listed in the ASHRAE Guide and Data Book. Adaptation of this procedure for dryer analysis would require calculation of the wet and dry bulb temperatures <u>above</u> which a percentage of the total hours of the season can be expected to fall. Moreover, the length and occurrence of the harvesting season could be varied geographically. Analysis of weather bureau data by computer would reduce the amount of labor required to produce these tables. Atmospheric pressure, which also affects the drying process, is mainly a function of altitude. Thus, location-specific values could be included in the table with the expected temperature data.

In order to fairly compare dryer performances, a wide initial to final moisture content span should be used. A standard initial moisture content of 28% wet basis (near the maximum harvest moisture) and a mean final moisture content of 14.5% wet basis (required for long-term storage) are suggested. In addition, for product quality consideration the permissible final moisture content range could be specified as well as an upper temperature bound.

For comparison of manufactured dryers, fan, motor and thermal efficiencies can be determined by simple tests; these could be performed by an independent organization, similar to the Nebraska Tractor Tests. Average efficiency figures can be assumed for testing the variation in dryer efficiency with location.

With these conventions, comparison of batch-in-bin dryer performance (assuming model adequacy and uniformity of dryer air distribution) could be done as follows:

1. For each location (two or three per state would probably be sufficient) relatively severe weather conditions would be determined by the procedure described above.

2. The standard moisture content range and published efficiencies would be used in the simulation.

3. Bed depth, airflow rate, and inlet air temperature would be specified by the manufacturer.

4. Simulation would be performed on a per square foot basis for determination of the time required to dry to the specified final average moisture content.

5. Output information, for comparison with similar dryers, would include: bed depth, time required, and air and heat energy requirements. Using these data and prevailing energy prices, total dryer capacities and operating costs could be easily calculated.

For a study of the variation in efficiency of batch-in-bin dryers with location, the following modifications could be made:

1. Average efficiency values could be used.

2. A four-dimensional search (with appropriate bounds) in bed depth, time, airflow rate, and inlet air temperature could be performed, with the objective to be the minimization of the energy input:dryer capacity ratio. For this search, a cost ratio for high grade (electrical) to low grade (heat) energy must be assumed. Again, a one-square-foot bed cross-section would be assumed.

J. Suggestions for Further Study

Suggestions for further investigation in the field of deep fixed bed drying economics center principally on improvement of the algorithm and applications:

1. Because of the small data sample from which the model was constructed, its accuracy and range of applicability are not firmly established. Thus, there is a need for additional well instrumented drying data over the entire range of model parameters. In addition to serving as a check on the present form of the model QUICK, these data can be used to improve the values of constants used in the model.

Other models which are developed should be considered as replacements for QUICK, provided that their computational times and accuracies are better than or comparable to that of QUICK. For example, an extension of the dimensionless parameter model of Bakker-Arkema et al.(1971) accounts for condensation in its prediction of final average moisture content and requires less time for evaluation than QUICK.

2. The user should be able to improve the speed of the optimization routine, OPTWHIZ, without significant loss of accuracy by relaxing the termination requirements for the one-dimensional searches and by reducing the interval to be searched.

3. If optimization with respect to depth or time is desired for purposes of design, an extension of the OPTWHIZ technique to three dimensions might be made if the geometry of the isocost and isostere surfaces were known for all possible cases.

4. For purposes of control, it would be valuable to know precisely under what conditions each of the cases of Figures 2, 6 and 7 occurs. For example, if the optimal conditions are consistently at the upper temperature bound for constant values of depth and time, only the air-flow rate need be adjusted to maintain optimality.

5. Equation I.C.3, which gives the contribution of the fan and motor to inlet air temperature rise, is an approximation based on available information. Since at low inlet air temperatures and high airflow rates this factor becomes important, a study should be made to improve or confirm this expression.

6. Similar operating cost models should be developed for other common grain dryers. (If the QUICK model is shown to be valid for temperatures higher than 140 deg F, the algorithm is directly applicable to the column batch drying system.) Incorporation of these into their respective systems models would facilitate comparison of competing systems for a particular enterprise, or enable the grower to determine the effect of a proposed change in a given system.

A complementary study of capital costs for each type of dryer is needed to extend the capabilities of the operating cost models. These costs, however, unlike operating costs, vary somewhat from manufacturer to manufacturer and tend to rise faster with economic inflation than do the utility prices. It would therefore be necessary to update the fixed cost data frequently to maintain current capital cost models.

7. The model might be useful in the development of a set of operating guidelines (e.g. nomographs) for a particular locale to be used by operators to minimize drying cost while maintaining grain quality.

8. This and similar models could aid in development of a rational dryer performance rating scheme, as suggested in Section I. A testing program generally accepted by manufacturers, elevators, growers, and universities could be set up to determine equipment efficiency factors, a concept similar to that of the Nebraska Tractor Tests. With the availability of telephone-linked computer systems, evaluation of competing dryers for a particular locale could thus easily and impartially be made.

II. OPTIMIZATION OF CONCURRENT DRYER-COUNTERCURRENT COOLER COMBINATIONS USING DYNAMIC PROGRAMMING

A. Introduction

Due to the large grain volumes handled daily by commercial elevators, a high degree of automation is possible; this permits reduction in labor costs but requires generally higher capital investments. Thus, in a highly competitive market the costs required to process the incoming grain for storage or shipment can spell the difference between success and failure; small differences in operating cost per bushel or dryer capacity rather than capital cost can be the deciding factor in choosing between two competitive designs.

The continuous flow convection-type dryers used in elevator operations can be classified into three categories based on the relative direction of air and product flow: (i) cross-flow, in which the air moves perpendicular to the flow of the grain; (ii) concurrent flow, where the air and product travel in the same direction; and (iii) counterflow, in which the air flow is opposite to that of the grain. In addition to these basic classes, commercial dryers are built which use combinations of these in sequential sections.

Thompson(1967) has commented on the performance characteristics of the three basic dryer types. Cross-flow type dryers, when built in a single stage, overdry the grain on the side where the inlet air enters and underdry on the air exhaust side. Counterflow dryers, in which the hottest air meets the driest grain, have the highest moisture removal

rate per foot of depth of the three types but in doing so subject the grain to considerable stress. Concurrent flow dryers typically remove the majority of the moisture within a relatively short depth and then provide a tempering period through the rest of the bed.

Improper drying of the product can cause two major types of damage: reduced milling quality--caused primarily by overexposure to high temperatures--and stress cracking, with consequent kernel breakage, caused by rapid moisture removal. Thus, choice of dryer design can greatly influence product quality as well as system capacity, and the two objectives are rarely compatible. Some combination of existing dryer types in a single machine will ultimately be the compromise required to meet the rising raw product standards of the cereal products manufacturers.

A design currently of interest employs a concurrent flow dryer coupled with a counterflow cooler for the grain (Anderson,1971; Kline et al.,1971). The concurrent dryer, which exposes the grain to the least stress, is used to remove the majority of the moisture. The counterflow design, used as a cooler, allows the grain to gradually cool, thereby slowly reducing the temperature and moisture gradients within the kernel; this cooler also accomplishes a slight amount of drying. Some sacrifice in capacity, for a given sized machine, may be expected from the use of the concurrent dryer instead of the more efficient counterflow design. It is claimed, however, that higher dryer temperatures with consequent higher moisture removal efficiencies can be achieved with a machine of this design combination. In addition, there exists the possibility of recycling to the dryer all or a portion of the air exhausted from the cooler, utilizing the heat energy which has been added to the air during the cooling process. Despite the promise of high product quality with this machine configuration, Thompson(1967) found that "counterflow cooling immediately following (concurrent) drying was not an adequate substitute for delayed cooling in reducing the susceptibility of the corn to break or form stress cracks."

Clearly, the optimal design problem is not a straightforward one. Variation in economic and ambient air conditions, product quality considerations, temperature, airflow, and depth parameters must all be considered in making design recommendations. Due to the extreme variability of weather and product moisture over the short harvest season, as well as the high labor costs and large number of variables involved in the construction of prototype machines, another design technique was required--the use of simulation and optimization.

Earlier researchers in the field of optimal design of grain drying equipment have been hampered by the lack of adequate models to describe the drying process. Although Hukill(1954) pointed out the inaccuracy of his model of the batch drying process, Ahn et al.(1964) and Schroeder et al.(1965) adapted it for modeling the crossflow dryer. Fortunately, there now exist several more reliable models from which the researcher can choose; these are summarized in Bakker-Arkema et al.(1971).

Within the published drying literature, only two attempts at optimization of multistage dryers have been reported. Both Ahn et al.(1964) and Schroeder et al.(1965) used the dynamic programming technique to determine optimal airflow allocation in multistage crossflow dryers. Because different performance criteria were used, the two studies were not comparable.

In related, single-stage optimization work on grain dryer design, Thompson(1967) employed gradient and one-dimensional search techniques for determination of depth, grain flow and airflow rates in all three basic continuous dryer types. Thygeson et al.(1970) used the differential algorithm technique of Wilde and Beightler(1967) to predict optimal bed depth and gas flow rate in a dryer during the constant rate drying period. The deficiencies of both techniques for the multistage problem are discussed in the following section.

In designing an optimization procedure, one of the initial steps must be the formulation of a performance criterion or objective function for the system to be investigated. The criteria used in previous studies of grain drying optimization fit three separate classifications: minimization of cost (capital and/or operating), maximization of throughput, and maximization of quality. Usually these are stated in some combination; the most comprehensive objective function (Thompson,1967) had the following form:

$$\begin{aligned} E = {} & c_1 \left(\frac{\text{drying speed}}{6\%/\text{hr}} \right)^4 + c_2 \int (\text{drying temperature})\,dt \\ & + c_3(\text{heat energy supplied}) + c_4(\text{fan energy supplied}) \\ & + c_5(\text{size of dryer needed}) \qquad \text{(II.A.1)} \end{aligned}$$

In this thesis study the goal is to develop user-oriented techniques for optimization of two concurrent flow dryer-counterflow cooler combinations and to demonstrate their use. Choice of simulation models, constraints, and objective function will be left, insofar as possible, to the user. Adequate instructions for use of the associated computer programs will be given and possible modifications discussed.

B. Dynamic Programming

Dynamic programming (d.p.) is an optimization technique developed especially for problems characterized by multistage decision processes. These problems arise, e.g., in a sequential manufacturing process when decisions made during each production step affect the properties and accumulated cost of the product at each subsequent step in the process. Other problems of this type originate in such diverse fields as economics, physics, and transportation scheduling.

Unlike linear programming, no standard algorithm exists which can be applied to the problem at hand; instead, dynamic programming is essentially a philosophy which allows the user to structure problems of considerable complexity into a form amenable to solution. That is, d.p. is used for determination of all process inputs, outputs, and controls necessary to achieve the objective in the most desirable way (least cost, maximum profit, etc.) for a given process.

To be structured into a form for optimization by means of d.p., a problem must have the following properties (Hadley,1964):

(i) it must be possible to visualize the problem as a sequential decision problem, where each step in the process requires one or more decisions.

(ii) the system must have a measure of performance which is expressible in the same units at each stage of the process.

(iii) the parameters measuring the performance of the system and its response to controls which are applied must remain the same at each stage in the process.

Some alternative approaches to the optimization of multistage decision processes are discussed by Rosenbrock and Storey(1966), including hill-climbing on parameters, the gradient method in function space, and Pontryagin's method. Of the available methods only Pontryagin's method and d.p. will always yield the "best" overall solution; the other techniques may converge to a "good", but suboptimal solution. Further difficulties with the other algorithms arise in the awkward manner in which constraints are handled and in finding an initial feasible solution from which to begin computations.

Pontryagin's method has not found wide acceptance in process control problems due to difficulties in formulation of the solution and excessive numerical computations necessary for solution. Although d.p. also suffers from these two disadvantages, it yields a solution which is more useful in the present context. A discussion of the "pros and cons" of d.p. will follow a discussion of the d.p. formulation.

In order to explain the dynamic programming principle, some definitions and an example are useful; the following notation and description is due to Larson(1968):

Let $\bar{x}$ denote the state vector, the components of which describe the state of the system at each stage in the process.

Let $\bar{u}$ denote the control or decision vector, whose components are those variables which may be manipulated to influence the values of the state variables.

The stage of the system will be denoted by the scalar index n, which varies monotonically through the N process stages.

The system equations describe how the state variables at stage n are transformed into a new vector of state variables at stage n+1 by application of a control vector $\bar{u}$ at stage n. The general relation is:

$$\bar{x}(n+1) = \bar{g}(\bar{x}(n), \bar{u}(n), n) \qquad \text{(II.B.1)}$$

The performance criterion (objective function) evaluates the performance of the system when a given control sequence $\bar{u}(n)$, $n=1,\ldots,N$ is applied. If the criterion is a cost function, the sequence of controls which minimizes it are sought; a reward function is associated with a maximizing sequence of controls. For the problems considered herein, a cost function criterion is used, hence all further reference will be to minimization of cost. The corresponding objective function takes on the general form:

$$I = \sum_{n=0}^{N} i[\,\bar{x}(n),\bar{u}(n),n\,] \tag{II.B.2}$$

where I is the total cost for the process, which is to be minimized, and i is the partial cost incurred at stage n.

The purpose of constraints on the problem is to limit the values of the state and control variables at a given stage. These controls may be expressed as: $\bar{x} \in X(n)$ and $\bar{u} \in U(\bar{x},n)$.

The optimization problem can then be summarized as follows:

Given: 1) a system described by a set of system equations

$$\bar{x}(n+1) = \bar{g}(\bar{x}(n),\bar{u}(n),n)$$

2) the constraints

$$\bar{x} \in X(n)$$

$$\bar{u} \in U(\bar{x},n)$$

3) an initial state $\bar{x}(0)$

Find: the control sequence $\bar{u}(n)$, $n=0,\ldots,N$ which minimizes:

$$I = \sum_{n=0}^{N} i[\,\bar{x}(n),\bar{u}(n),n\,] \tag{II.B.3}$$

An illustration of a problem having this structure is provided by

an exhaustive review of a crossflow grain dryer design problem solved by Ahn et al.(1964) using d.p. In this type of dryer, as shown in Figure 8, a long column of grain, which moves steadily by gravity flow through the dryer, is exposed to an airflow stream passing perpendicular to the grain. The hot entering air, which picks up moisture during its exposure to the grain, is then exhausted. Wet grain entering the top of the column dries to an acceptable average moisture content before its release at the bottom of the column. Prior to this study, the usual practice in crossflow dryer design had been to use an approximately evenly spaced airflow pattern along the length of the column. These authors wished to investigate the possibility of improving dryer performance by partitioning the airflow into three equally spaced sections along the column with each section receiving a different portion of the total air - flow. Inlet and outlet moisture contents, entering air temperature and humidity, grain mass flow rate, and column width were taken to be fixed.

In terms of dynamic programming, the crossflow drying process can be drawn schematically as shown in Figure 8, where the three equally spaced sections of the dryer are taken to be the three stages of the process. In this case the state vector has a single component, the moisture content of the grain, which completely describes the variable of interest in the system. A constraint on the grain moisture content (m.c.) is given by the final m.c.; no outlet m.c. above this point is acceptable. Likewise, only the initial m.c. specified is used as an input to the process. The type of control available to the process is the same at each stage, namely the amount of air allotted to the grain passing through the section. The control variables are subject to the constraint that the total airflow be less than or equal to a value fixed

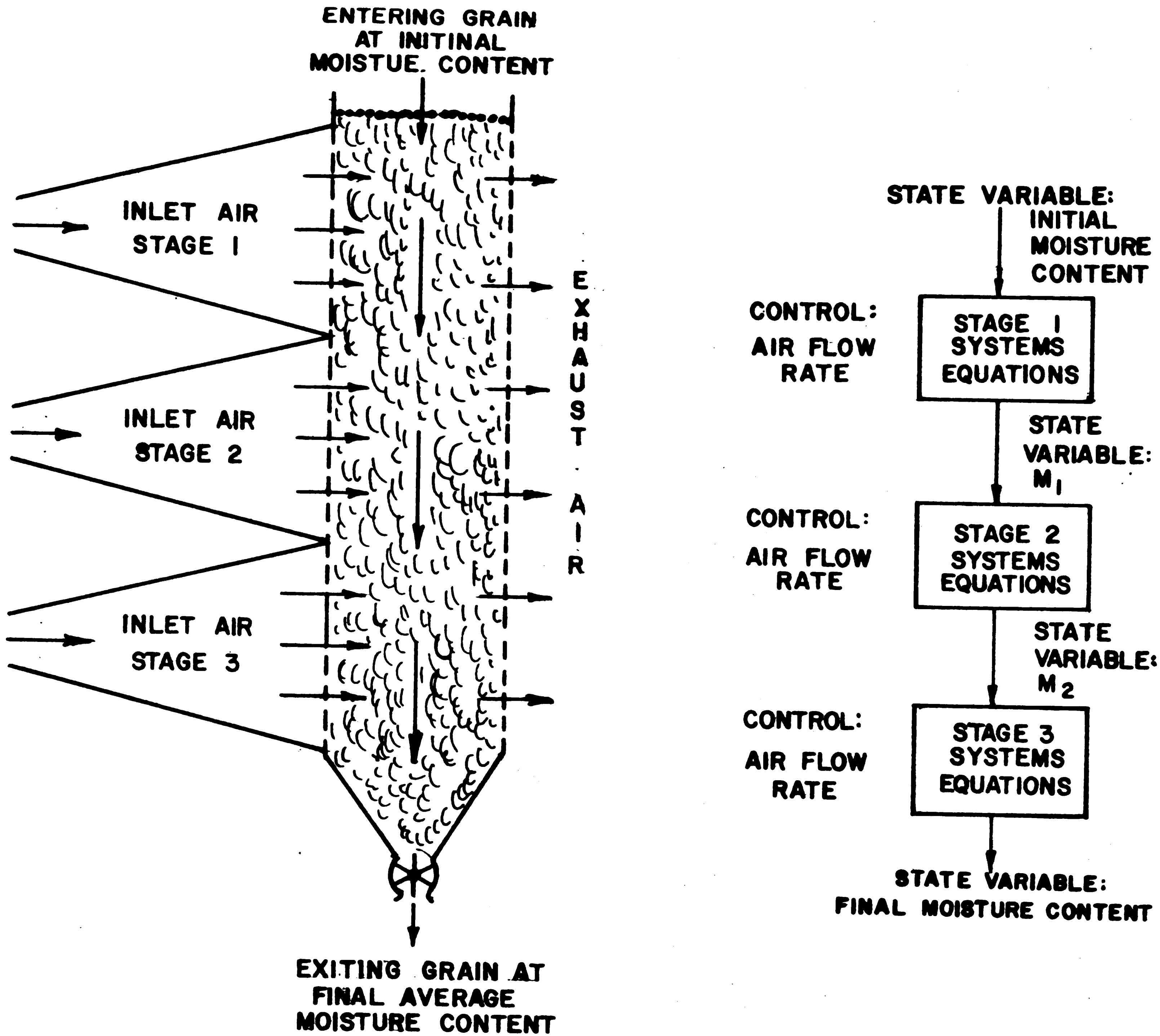

Figure 8: Schematic Representation of Crossflow Dryer and Dynamic Programming Model

by the available fans.

The system equations are provided by a crossflow dryer adaptation of a batch dryer model (Hukill,1953; also Section I.B. of this study) which gives the average m.c. as a function of dryer inputs and grain properties. The same system equations thus may be used for all three stages. Finally, the objective function to be minimized is given by the ratio of the mass flow rate of drying air to the mass moisture removal rate, or simply the sum of the airflow rates of the individual sections, since moisture removal rate is fixed by the specified inlet and outlet m.c. and grain flow rate.

To implement the solution to this problem, it is necessary to introduce the Principle of Optimality (Bellman,1965) and the basic iterative relation associated with this problem. The Principle of Optimality can be stated as follows: "an optimal policy (i.e. sequence of decisions) has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with respect to the state resulting from the first decision". From this observation- which can be rigorously proved mathematically- a technique for solution of multistage decision problems was developed. Smith et al.(1970) have interpreted the Principle of Optimality by stating "every component in a serial structure influences every downstream component; only the last component is independent. The last component can be suboptimized independently for each possible state of the input it receives. Once this is done, the last two components are grouped together and suboptimized independently for each possible state of input. This process continues until the entire structure is included in the optimization."

If the solution process begins at the final stage of the process and proceeds iteratively backward to the initial stage, the method is called "backward d.p."; using the earlier notation, the recurrance relation can then be stated mathematically as:

$$I'(\bar{x},n) = \min_{\bar{u} \in \mathbf{U}} \left\{ i(\bar{x},\bar{u},n) + I'[\,\bar{g}(\bar{x},\bar{u},n),n+1\,] \right\} \qquad \text{(II.B.4)}$$

where I' denotes an optimal sum of the partial costs incurred in stages up to and including the current stage n; i identifies the cost of a trial solution during the current stage n. Thus the optimal solution including the present stage is found by minimizing the sum of the costs incurred during the present stage and the optimal cost associated with the resulting state in the stage which computationally preceded it.

Implementation of this procedure on a digital computer requires that the optimal costs as well as the associated optimal controls be stored at discrete node points in the state space. These results are required for interpolation among nodes adjacent to the state point in stage n+1 resulting from discrete applied controls in stage n.

Returning to the example of the crossflow dryer, the solution begins at the third stage, which brings the grain to the desired final moisture. The state space at the interface between the second and third stages is partitioned into discrete, evenly spaced nodes representing moisture contents in the range from which the final moisture content can feasibly be reached during the third stage. The possible values of airflow to be alloted to the third stage are also discretized; this range is bounded above by the constraint on total airflow and below by zero, in which case no drying would occur in the third stage.

Starting at each of the nodes designated at the interface of the second and third stages, all possible discrete airflows are tried in turn. Of those which result in the desired final moisture content at the dryer outlet, the optimum control is that which achieves this goal at the minimum cost (i.e. uses the least air to accomplish it). The minimum cost and associated optimal control are then stored in the computer memory at a location designated by the node at the second-third stage interface from which the solution proceeded. For those nodes from which it is impossible to reach the desired final moisture content using available airflows, an absurd value indicating their infeasibility is stored in memory.

The solution now proceeds to the second stage of the dryer. A new grid of possible moisture contents is set up at the interface of the first and second stages. As before, each of the possible airflow controls is applied at each of the m.c. nodal points of the grid and the resulting m.c. (obtained from the model) at the interface of the second and third stages is noted. In general, the resulting m.c. will not fall on a node and a value of the cost and associated airflow will have to be obtained by interpolation between the two adjoining nodes at the interface of the second and third stages. By summing the cost at the interpolated point and the cost of the applied control of the current stage, a total cost for the process through the current stage is obtained. Repeating this procedure for each of the possible controls and comparing all sums, the optimal total cost for each m.c. node at the first-second stage interface can be found. These values are stored in the computer memory along with their associated node point. In order that the maximum airflow constraint is not violated, it is also necessary to store the total accumulated airflow from previously computed stages.

At the first or beginning stage of the dryer, the initial m.c. is specified. Thus there is no need to set up a grid of moisture contents from which to work. Starting at the initial moisture content, all feasible controls are again applied. For each control which results in a feasible state, the cost of the control is assessed and added to an interpolated value from the first-second stage interface. When all resulting sums are compared, the minimum sum is the optimal total process cost and the associated airflow control is the optimal control for the first stage. To determine the optimal controls for the remaining stages, the first control is again applied to the model and interpolation is performed among the nodes at the second-third stage interface to recover the optimal airflow control for the second stage. Optimal airflow for the third stage is then simply obtained by subtraction of the two optimal controls from the accumulated total. The first, second, and third dryer stages were found to have optimal airflow rates of 6300, 5700, and 5400 lb/hr, respectively.

In addition to serving as an introduction to grain dryer design by dynamic programming, (c.f. Schroeder et al.,1965) the preceding example illustrates several of the advantages and disadvantages of the method:

(i) the algorithm for each process must be separately designed. Although the formulation in the dryer example was straightforward, many more complex problems require a combination of "experience, intuition, and luck" (Smith et al.,1971). Familiarity with the system to be optimized allows the designer to choose relatively narrow ranges for the state variables--grids which are fine enough to preserve accuracy but coarse enough to speed computation--and to use all known constraints to limit the number of computations necessary.

(ii) a large number of evaluations of the systems equations is necessary even for a well designed problem. Therefore, it is necessary that the models are capable of rapid solution by the computer; in particular, this rules out long iterative type solutions.

(iii) it is necessary to store a large number of results generated at each stage in order to be able to interpolate in the state space of previous stages and to recover the optimal policy at the end of the problem. This has been the primary limitation in the use of the d.p. technique, since present computers are limited in the amount of fast access memory available. As the number of state variables increases, the storage location requirement increases as the product of the number of levels of all state variables. Most authors state a limit of three as the absolute limit on the number of state variables that can be handled by present computers.

(iv) in the solution of a dynamic programming problem, not just one, but a family of solutions is generated. Thus at the end of the example problem, optimal solutions are available for a range of moisture contents in either a one or two stage dryer as well as the single solution for the three stage dryer. Moreover, because the grids of optimal costs and controls for the second and third stages are unaffected by the inlet conditions specified in the first stage, they can be used to evaluate optimal cost and policy for any initial moisture content within a wide range and to serve as a base for the evaluation of dryers having four or more stages.

(v) once the problem has been solved with a relatively coarse grid, additional refinement is possible by narrowing the ranges of states and controls to be considered and by employing a finer grid mesh.

Several variations on the basic d.p. technique have been explored in the literature. Among these is "forward d.p." in which it is desired to find the optimal policy for a problem which starts at a known initial state but for which a variety of desirable final states is possible (Larson,1968). Optimization of stochastic processes has received attention (Bellman,1965). Manufacturing processes having loops and branches have been treated (Mitten et al.,1963) as well as countercurrent processes (Dranoff et al.,1961).

The development of dynamic programming is still in a state of flux. Several recent developments in diverse fields have shown promise in extending the size and variety of problems which can be handled by this technique. For example, special methods which minimize size requirements for problems having continuous rather than discrete stages have been developed (Larson,1968; Wong,1967). A new type of magnetic memory using orthoferrite crystals promises an inexpensive, fast-access, high-capacity memory for the current computers. Finally, a new generation of computers (Illiac IV of the University of Illinois and Burroughs Corporation) is being developed expressly to handle the type of computations found in dynamic programming. Although dynamic programming may become more useful as a design tool, its use will continue to be limited by the skill in formulation of the user.

C. Optimization of Concurrent Dryer-Counterflow Cooler Configuration Without Air Recycle

C.1. Dryer-cooler description

The concurrent flow dryer-counterflow cooler configuration to be analyzed in this section is shown schematically in Figure 9. In this design, inlet air to the dryer is first drawn over the motor or engine used to power the fan, thus serving to slightly increase the air temperature as well as to cool the power source. The air then passes into a combustion chamber where it is heated by combustion of a fossil fuel, typically LP or natural gas. Next the hot air is forced through the bed in the same direction as the grain, which moves downward by gravity flow. As the air traverses the bed it cools by evaporation and accumulates moisture until it is saturated or is exhausted from the system. The hot grain passes from the dryer into the counterflow cooler where it is progressively cooled by an opposing airstream being drawn by suction through the moving bed. By means of this gradual cooling process, steep internal temperature gradients in the grain are avoided and stress cracking is minimized. The air used to cool the grain may perform a certain amount of drying, increasing in humidity during the process. At the exit of the cooler, the cooling air temperature approaches that of the incoming grain; thus the cooling air could be a potential warm air source for the dryer. In this design configuration, however, it is exhausted from the system by the suction fan.

To optimize the design of this dryer by use of mathematical techniques, it is first necessary to define the objective of the optimization and its mode of measurement and secondly to identify the state and control variables, their constraints, the stages of the system, and

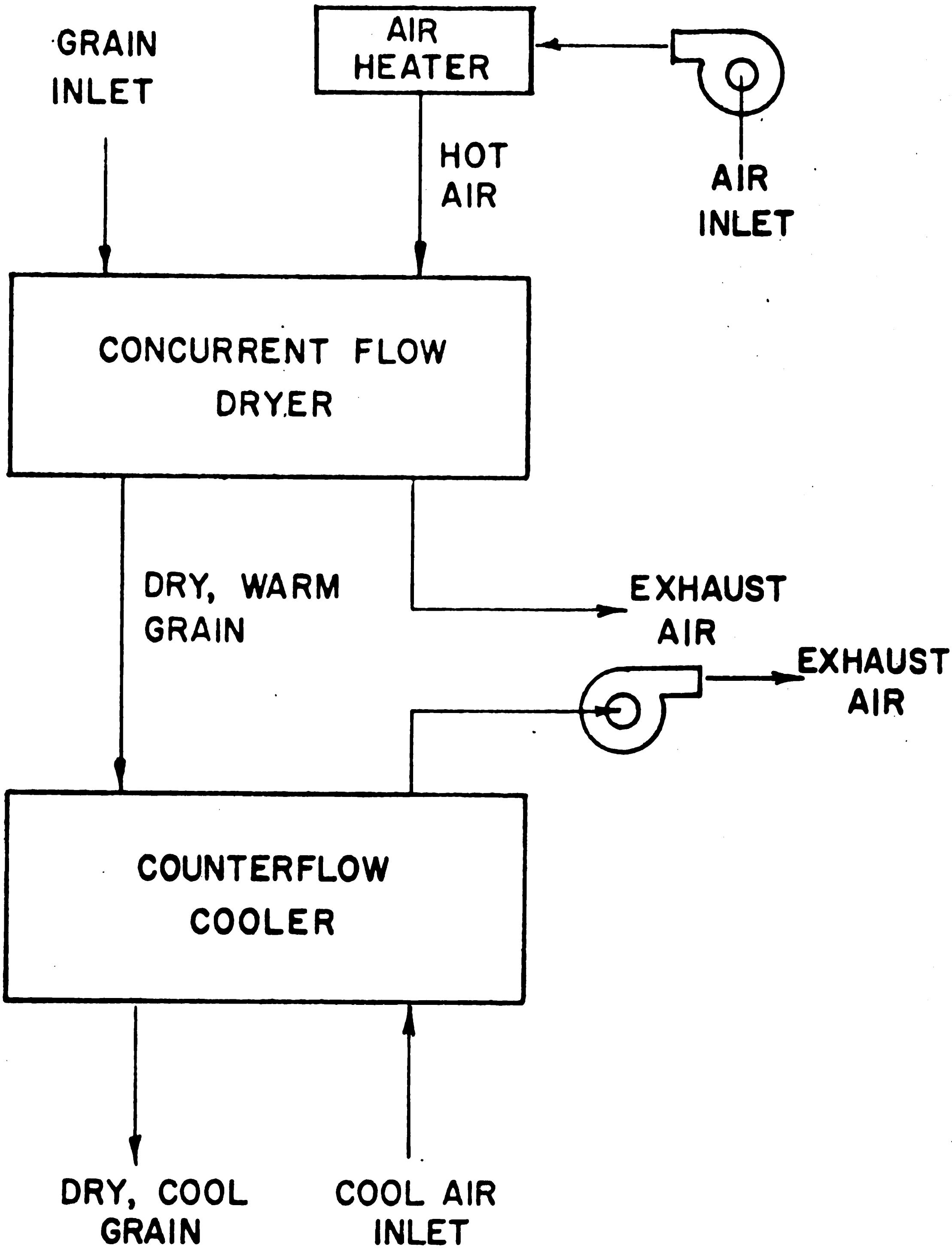

Figure 9: Schematic Representation of Combination Dryer-Cooler Without Air Recycle

the system equations necessary to model the components.

For the concurrent-countercurrent dryer system, it was decided to minimize the operating cost for a given grain throughput rather than to maximize throughput under cost constraints. Thus the optimum conditions for airflow, depth, et cetera were sought. Capital costs were not included in the analysis since, on a large volume operation, small differences in construction cost are overshadowed by operating cost. Also, unlike Equation (II.A.1), no penalty terms were included in the objective function, although a constraint on maximum dryer air temperature was imposed. Anderson(1971) stated that at present no premium is paid by the cereal industry for quality dried grain. Hence, the objective function used reflects the actual processing cost; additional quality terms can be easily added to the objective function should the marketing situation change.

Operating costs arise from three major sources in the system. In the dryer, energy is required to heat the incoming air. The efficiency of this process depends on burner and dryer design, the fuel used, and on prevailing ambient conditions. The second energy requirement of the dryer is for driving the fan; its magnitude depends principally upon the depth of the bed and the airflow rate required, as well as on the efficiency of the fan and driver designs. As noted before, a portion of the "wasted" energy input-dissipated as frictional heat-is recovered by using it to preheat the drying air. The operating cost of the cooler fan, which exhausts the air by suction, is a function of the same variables. However, its inefficiency does not permit a savings on the heating component of dryer cost unless the inlet air to the dryer is routed over the cooler fan and motor. In the configuration chosen in the

formulation, utilization of the "wasted" energy of the dryer fan, but not that of the cooler fan, is assumed. For the continuous flow operation a time base of one hour was chosen. By specifying the fixed grain flow in bushels per hour, and measuring operating costs in cents per hour the common performance ratio of cents per bushel could be calculated. The cost functions, which are the same as in Part I of the Thesis, are given in Part I, Section C. A second common measure of performance, Btu per pound of water evaporated, can be calculated after the optimization is complete.

Since the product (grain) is the only flow component which passes through both the dryer and cooler, the process divides naturally into two major stages. The additional components, fans and heater, can be lumped with whichever major stage each is associated because they do not operate directly on the product.

In many respects this problem is an analog of the cross-flow dryer design example presented. Like that case, the air is not reused from stage to stage; hence its properties need not be designated as state variables. Again, because the primary function of the process is drying, product moisture content is the natural choice as one state variable. Unlike the example, however, the temperature of the product in this problem is also required for assessment of the performance of the two stages. This requires the addition of product temperature as a second state variable.

Controls on the dryer include its depth, the temperature of the entering air, and the airflow rate. Cooler controls are provided by depth and airflow rate.

It is possible to impose many constraints, both artificial and natural, on the problem to limit the number of possible solutions under consideration. In this context, experience with the system components is invaluable in choosing maximum airflow and temperature constraints, placing upper and lower bounds on bed depth, and establishing feasible ranges and increment sizes for the two state variables at both stages. The major preliminary task remaining before the final d.p. formulation is the choice or development of suitable systems equations, or models, which adequately describe the performance of the two stages of the process under all potential conditions of operation. In the following section, the development of these models is considered.

C.2. Development of models for cooler and dryer

For the two process stages of the d.p. formulation two separate models are required, one for the dryer, the other for the cooler. From each model, the resultant values of the state variables (product moisture and temperature) are needed as a function of all possible controls over the anticipated range of the model. In addition, the formulation also required prediction of the air temperature and humidity in order to satisfy physical constraints on saturation of the air.

The presently available models for these system components are of an iterative nature (Bakker-Arkema et al.,1971) requiring time-consuming computations for a single simulation. These fundamental models, however, were used in generating data from which strictly empirical models could be constructed. Two of these fundamental models, CONCUR and COUNT, were chosen to provide information for the empirical dryer and cooler models, respectively. Fortran IV coding and documentation for these models is found in Bakker-Arkema et al.(1972). Theoretical considerations are discussed in Bakker-Arkema et al.(1971).

Since product flow rate was not intended as a parameter in this

study, most of the models were constructed based on a single flow rate, 9 bu/hr/ft^2. Since several different techniques were used in the construction of the models, these will be discussed under separate headings.

Concurrent dryer outlet moisture content model.-- DeBoer(1971) constructed an empirical model for the moisture content of corn at the dryer exit as a function of inlet air and grain parameters, air flow rate, and dryer length. A plot of the dimensionless depth $DX = \frac{hax}{G_a c_a}$ versus the dimensionless moisture ratio $DM \equiv \frac{M - M_e}{M_o - M_e}$ for various values of dimensionless grain flow rate $DF \equiv \frac{ha}{G_p c_p}$ indicated that all resulting curves can be approximated by the functional form $DX = b_1(DM - 1)^{b_2}$, where b_1 and b_2 are each functions of DF. GAUSHAUS (Meeter,1965), a non-linear least squares curve fitting subroutine, was used to approximate the constants. The equation was then re-written with DM as the independent variable. In a similar fashion multiplicative modifying functions for b_2 were successively incorporated into the model accounting for the parameters: airflow rate, inlet air temperature, initial moisture content and inlet grain temperature. Air humidity was found not to be a significant factor in the model except as implicit in the equilibrium moisture content.

Within the range:

inlet air temperature = 200.-500. deg F

initial m.c. = 0.20-0.40 d.b.

airflow rate = 60.-240. cfm/ft^2

grain flow rate = 3.-9. bu/hr/ft^2

inlet product temperature = 50.-100. deg F

the model was found to have an average error of approximately 0.5% d.b.

with a maximum error of 1.6% d.b.

Concurrent dryer outlet air and grain temperatures and air humidity models.-- Using the midpoint value of each parameter as a base, the grain temperature profile through the bed was generated, using the CONCUR model. This profile was approximated, using GAUSHAUS, by a function of the form: $\theta_{avr} = c_1(\mathrm{Depth})^{c_2}$; this will hereinafter be designated the "standard profile".

Next, the control variable airflow rate was varied from its midpoint value over the intended range of the model being constructed, holding all other parameters constant at their midpoint values. From the profiles generated by this procedure, the deviation from the standard profile caused by the change in airflow rate was plotted as a function of the value of airflow rate, for a particular depth in the bed. The plot at each depth was found to be essentially linear. Thus the product temperature at a particular bed depth could be predicted by adding or subtracting the deviation from the value of the standard profile at that depth. In order to bring the parameter bed depth into the model, it was then necessary to express the slopes of the lines as a function of depth. GAUSHAUS was again used to provide the empirical constants to the functional form: $\mathrm{Slope} = c_1(\mathrm{Depth})^{c_2}$.

The same procedure was used for the parameters inlet product temperature and inlet air temperature, giving three correction factors for the standard profile, each based on the value of the parameter and depth within the bed.

Variations in the inlet moisture content and absolute humidity parameters also resulted in near-linear plots of deviation from the standard profile versus parameter value. These corrections, however, were nearly independent of depth and hence required no depth factors in their expressions.

Finally, it was determined that the sum of the correction factors for all parameters, when added to the value of product temperature taken from the standard profile, provided a good approximation to the results of the CONCUR model, using the same input data. Within the same ranges as the preceding empirical moisture content model, this model produced results generally within 5.0 deg F of the data of the CONCUR model, at bed depths of one foot or greater.

The same technique was used by Roth(1971) to produce empirical models for the outlet air temperature and humidity of the concurrent dryer. The results are: outlet humidity, maximum deviation .0168 lb water/lb dry air, maximum percent error 40.%, mean percent error 18.%; outlet air temperature, maximum deviation 52. deg F, maximum percent error 35.%, mean percent error 12.%. The four empirical models for the concurrent dryer--outlet air temperature, humidity, grain temperature, and moisture content--are programmed in Appendix B, SUBROUTINE DRYSIM.

Model of outlet air and product conditions for the countercurrent cooler.--

Roth(1971) found that essentially the same procedure could be used in formulating the empirical models for the countercurrent cooler, except that a single additive depth correction factor could be used. That is, instead of adding the depth dependent correction factors to the stored profiles of the dependent variables, a linear depth correction factor plus the sum of the other correction factors and a constant yielded a good approximation to the data from COUNT.

Significant nonlinearity of the correction factor for airflow rate in the humidity and air temperature models required approximation of the factor by a polynomial in the former case and by an exponential in the latter. Only inlet air temperature and airflow rate were found to significantly affect the outlet air temperature; inlet moisture content and bed depth were the sole major determinants of outlet product moisture content.

In testing the model over the ranges:

inlet air temperature, 35-60 deg F

inlet air humidity, .002-.005 lb water/lb dry air

inlet product temperature, 110-160 deg F

inlet moisture content, 0.19-0.24 decimal, dry basis

airflow rate, 80.-175. cfm

depth, 0.3-2.0 ft; ($\leq$ 1.0 ft for outlet product temperature)

Results were: outlet humidity, maximum deviation .00113 lb water/ lb dry air, mean percent error 12%; outlet air temperature, maximum deviation 3.9 deg F, mean percent error 1.0%; outlet product temperature, maximum deviation 6.1 deg F, mean percent error 4.5%; outlet moisture content, maximum deviation .0135 decimal dry basis, mean percent error 2.0%. The four empirical models for the cooler outlet air temperature, humidity, grain temperature, and moisture content are found in Appendix B, SUBROUTINE COOLSIM.

C.3. Dynamic programming formulation and computer programming

With the preliminary information necessary for the solution now available, the mechanics of the dynamic programming formulation will be considered. Because the application of the dynamic programming principle to a practical problem is inseparable from the problems of computer program coding and the characteristics of the machine being used, the following discussion will detail the rationale for the program flow pattern; the associated programs COOLONE, DRYONE, and RECONE with their attendant subprograms are listed in Appendix B.

There are two variations of the dynamic programming principle which can be applied to a process of this nature. In the first type, forward dynamic programming, the solution starts at a single state specified at the beginning of the process and proceeds iteratively to the end of the process, where the optimal solutions for a number of final states can be explored without reworking the entire problem. For the other formulation, backward dynamic programming, the solution begins at a single terminal state and works iteratively backward to the beginning of the process; in this case, only the last step must be repeated in order to investigate alternate initial states.

For the present problem, the backward dynamic programming formulation was chosen as the most useful. Corn received at an elevator in a single day can vary widely in moisture content; the final moisture content necessary to prevent deterioration in storage provides a relatively invariant terminal state toward which the system is directed. For a single solution, no variation in ambient or economic conditions is permitted; a parameter study of these factors using the present formulation would require multiple trials.

In coding a d.p. computer program, there are two basic machine limitations which must be considered: computer fast access memory and available computer time. Of these, the first is the more serious problem; a long program with many checks on constraints may be more efficient computationally but may require too much memory to accomodate the large arrays which are needed for interpolation. On the other hand, there are economic penalties for computationally inefficient programs, which may be the overriding consideration. In practice, a balance must be struck.

The overall strategy employed for the solution was to write a separate program for each stage in the process rather than to solve the entire problem in a single program. This allowed the portion of the memory allotted to program instruction storage to be kept to a minimum. In addition, visual scanning of the intermediate output at each stage permitted significant reduction of the size of the interpolating arrays for the subsequent stage, thus requiring only a moderate amount of memory for the solution of large scale problems.

The general computational scheme for the cooler stage will now be described; coding for the program is given in Appendix B, PROGRAM COOLONE.

COOLONE first reads the user supplied information, including ambient and economic conditions (fixed during the solution), the ranges and increment sizes for the state variables, product moisture content and temperature at the dryer-cooler interface, the range and increment size to be used for the application of the control variables airflow and bed depth, and the maximum permissible values of outlet product moisture content and temperature. In addition, a minimum permissible value for moisture content is read; the use of this value will be discussed later.

The second phase of the program is a preliminary screening of the grid sizes of both the state and control variables, in order to limit the number of model evaluations necessary. This screening is accomplished by using the constraints on maximum allowable outlet product temperature and moisture content.

In the screening procedure, all parameters, except the one being checked, are held constant at that end of their respective ranges which maximizes the probability of a feasible trial with respect to a given constraint. The range of the remaining parameter is then searched until the point of failure (if any) is found. Several such simple checks using different constraints can reduce the problem size considerably at the start. The initial screening can be done several times before the iterations are begun in order to insure a near-optimal starting grid size and a minimum number of controls to be applied.

The third phase of the program is designed to evaluate the cooler simulation model for the remaining node points of the inlet product temperature-moisture content grid, using the remaining controls. The order of these computations is arranged such that the desired outlet conditions are most apt to be achieved at the beginning of this computational sequence; termination of the program occurs when no further feasible solutions starting from the remaining nodes are possible. Thus, the order of the DO loops, outside to inside, is: (i) inlet product moisture (starting at its lowest value), (ii) inlet product temperature (starting at its lowest value), (iii) airflow rate (starting at its upper bound), and finally (iv) depth (starting at its lower bound).

The constraints on outlet product properties and outlet air properties can be classified into two groups if all other parameters except depth are temporarily held constant (as they are while the inner DO loop is incrementing). A failure to meet the constraints on maximum allowable outlet product temperature and moisture content indicates that the bed is too shallow; excess bed length is indicated by the conditions of air saturation, air-product equilibrium, water absorption by the product, or by excessive drying (indicated by the user-supplied moisture constraint read in at the beginning of the program). These checks are inserted within the inner DO loop. A "bed too shallow" check indicates that feasible solutions are still possible within the depth loop but that no cost evaluation for the present iteration is necessary. These procedures reduce considerably the number of process and economic model evaluations required.

For each node of the inlet product temperature-moisture content grid, the least cost feasible solution, if any, is determined by comparison with the costs incurred in feasible trials using alternate controls; the indices of the grid, the minimal cost, and the corresponding depth and airflow are then written sequentially on a low-speed storage device. No arrays are used in this program and storage requirements are thus kept to a minimum.

The second program in the computer package for this concurrent dryer-countercurrent cooler configuration, DRYONE, was designed to: (i) evaluate the airflow and heating costs of the concurrent dryer using all feasible controls, (ii) interpolate in the grid of partial costs stored at the dryer-cooler interface, and (iii) perform a comparison of the total costs to find the minimum total operating cost for the system. Having found the optimum, DRYONE then interpolates in the grid of stored cooler controls at the dryer-cooler interface to recover the optimal depth and airflow for the cooler.

The general computational strategy is similar to that of COOLONE, with several additional features. Recalling the values of cooler cost and corresponding depth and airflow stored sequentially on the low speed memory device, DRYONE uses the associated indices of product temperature and moisture content to index the three two-dimensional arrays needed for interpolation. Because many of the values of the array would normally be infeasible if the original indices were retained, DRYONE uses information supplied by the user (from a scan of the COOLONE output) to reduce the size of the arrays to a minimum. If memory capacity is still a problem, the program can be rewritten so that the control arrays need not be formed at this time, but may have the needed values retrieved for interpolation after the minimum cost is determined.

User-supplied values for the economic and ambient conditions are read, in addition to the bounds and increment sizes in the state and control variables. Two new constraints for the dryer, maximum inlet air temperature and maximum allowable pressure drop, may be specified by the user to conform to the requirements of currently available equipment.

The iterative scheme used is to initialize the control DO loops such that maximum drying occurs, with the order of the loops (from outer to inner) being: airflow rate, air temperature, and bed depth. All available constraints are used to minimize the number of model and cost function evaluations necessary. Next, all feasible control possibilities are evaluated and the sums of all dryer and interpolated cooler costs compared in turn to identify the minimum sum. The associated controls for the cooler are then found by interpolation in the stored grids. Finally a third program, RECONE, uses all optimal controls and input data to identify all intermediate states for the process.

C.4. Procedure for use of the algorithm and an example

The input information necessary for the use of the non-air-recycle process programs, COOLONE, DRYONE, and RECONE is summarized in this section. Options available to the user are explained for each program. Finally, an example of the use of the algorithm is given.

For program COOLONE, the first of the three programs to be used, the following information is to be supplied on data cards, using FORTRAN IV F10.0 and I10 fields for real and integer values, respectively.

CARD 1

a. Maximum moisture content allowable at the entrance to the cooler, decimal, dry basis (real).

b. Minimum moisture content allowable at the entrance to the cooler, decimal, dry basis (real).

c. Number of levels of moisture content to be included in the grid between the maximum and minimum values (integer).

CARD 2

a. Maximum product temperature allowable at the entrance to the cooler, deg F (real).

b. Minimum product temperature allowable at the entrance to the cooler, deg F (real).

c. Number of levels of product temperature to be included in the grid between the maximum and minimum values (integer).

CARD 3

a. Maximum airflow rate to be considered as a cooler control variable, lb dry air/hr/ft^2 (real).

b. Minimum airflow rate to be considered as a cooler control variable, lb dry air/hr/ft^2 (real).

c. Number of discrete levels of airflow rate between the maximum and minimum values (integer).

CARD 4

a. Maximum cooler depth to be considered as a control variable, feet (real).

b. Minimum cooler depth to be considered as a control variable, feet (real).

c. Number of discrete levels of cooler depth between the maximum and minimum values (integer).

CARD 5

a. Inlet air temperature to the cooler, deg F (real).

b. Inlet air humidity to the cooler, lb vapor/lb dry air (real).

c. Atmospheric pressure, psi (real).

CARD 6

a. Maximum allowable cooler outlet product temperature, deg F (real).

b. Maximum allowable cooler outlet product moisture content, decimal, dry basis (real).

c. Minimum allowable cooler outlet product moisture content, decimal, dry basis (real).

CARD 7

a. Price of fan energy (for electrical energy, as programmed) required, ¢/KWH (real).

b. Time base over which the optimization is taken, hr (real).

c. Fan efficiency, decimal (real).

d. Efficiency of fan power source, decimal (real).

In addition, the user has the option of using the cooler model supplied, SUBROUTINE COOLSIM, and the electrical-energy-based objective function, SUBROUTINE CSTCOOL, or substituting his own subroutines. In this as well as the following programs, constraints read in as data values may be deactivated by setting them to absurdly high or low values and allowing the method to find an optimum unconstrained in that variable.

The output format for the program COOLONE, which is written on a storage device as well as printed, gives the following information: grid index for moisture content, grid index for product temperature, current optimal cost, airflow rate, and depth for the node.

In using COOLONE, some experience is required to be able to choose an optimal sized grid. Recalling that the primary limitation in the following program, DRYONE, is the size of the arrays which can be accommodated by the fast access memory, the user is urged to carefully consider the range and increment size in the product temperature and moisture content dimensions. Because only the optimal controls of depth and airflow rate will be stored, computer time for the COOLONE program is the only restriction on the range and increment size of the two controls. In initial trials using COOLONE, the use of a STOP card after the initial screening process in the program is recommended; this allows the program to use the constraints specified to initially adjust the variable ranges to feasible size.

Before running the second program, DRYONE, the user should scan the indices of the product moisture and temperature dimensions to note the largest and smallest feasible values of each found by the COOLONE simulation; this information is used to reduce the size of the arrays needed in DRYONE. For large scale problems the computer can be programmed to do the scan automatically. In choosing ranges and increment sizes for the control variables, the program is limited only by computer time, not storage space.

A summary of the information needed for the program DRYONE follows; values are read in FORTRAN IV F10.0 and I10 fields for real and integer values, respectively.

CARD 1

a. Minimum value of product moisture content to be considered in the interpolating grid (found by scanning the COOLONE output), decimal, dry basis (real).

b. Increment size in product moisture dimension (from COOLONE), decimal, dry basis (real).

c. Lowest value of moisture content dimension used in COOLONE program (real).

d. Number of feasible levels remaining in the moisture content dimension (from the scan of the COOLONE output) (integer).

CARD 2

a. Minimum value of product temperature to be considered in the interpolating grid (found by scanning the COOLONE output), deg F (real).

b. Increment size in product temperature dimension, deg F (real).

c. Lowest value of product temperature dimension used in COOLONE program (real).

d. Number of feasible levels remaining in the moisture content dimension (from the scan of COOLONE output), integer.

CARD 3

a. Lowest value of airflow rate to be used as a control for the dryer, lb dry air/hr/ft^2 (real).

b. Increment size of the airflow rate, lb dry air/hr/ft^2 (real).

c. Number of levels of airflow rate to be used as controls (integer).

CARD 4 (These inputs do not include the temperature rise due to the fan.)

a. Lowest value of inlet air temperature to be used as a control for the dryer, deg F (real).

b. Increment size in inlet air temperature, deg F (real).

c. Number of levels of inlet air temperature to be used as controls (integer).

CARD 5

a. Lowest value of dryer bed depth to be used as a control for the dryer, feet (real).

b. Increment size in bed depth, feet (real).

c. Number of levels of bed depth to be used as controls (integer).

CARD 6

a. Inlet air temperature to heater, deg F (real).

b. Inlet air humidity to heater, lb vapor/lb dry air (real).

c. Atmospheric pressure, psia(real).

d. Inlet grain moisture content to dryer, decimal, dry basis (real).

e. Inlet product temperature to dryer, deg F (real).

CARD 7

a. Time base used for optimization, hr (real).

b. Price of energy (electrical energy) for fan, ¢/KWH (real).

c. Fan efficiency, decimal (real).

d. Efficiency of fan power source, decimal (real).

e. Heat value of fuel used, Btu/gal (real).

f. Price of fuel used, ¢/gal (real).

g. Thermal efficiency of heater, decimal (real).

CARD 8

a. Maximum temperature constraint for dryer, deg F (real).

b. Maximum pressure drop constraint for dryer, inches of water (real).

Output from DRYONE includes the optimal total process cost, optimal dryer depth, airflow rate, and inlet air temperature (without heat added by the fan). In addition, the grain moisture content and temperature at the dryer-cooler interface are specified. Interpolation performed by DRYONE also identifies the optimal depth and airflow rate controls for the cooler.

The remaining program, RECONE, uses the results obtained in DRYONE along with the component models to identify the remaining air and grain conditions and to summarize the steady-state operating conditions for the process.

The required input information is specified in F10.0 fields as follows:

CARD 1 (From DRYONE output)

a. Optimal dryer airflow rate, lb dry air/hr/ft^2.

b. Optimal dryer inlet air temperature (without heat added by the fan), deg F.

c. Inlet air humidity to dryer, lb water vapor per lb dry air.

d. Optimal dryer depth, ft.

e. Moisture content at dryer-cooler interface, decimal, dry basis.

f. Product temperature at dryer-cooler interface, deg F.

CARD 2

a. Inlet air temperature to heater, deg F.

b. Atmospheric pressure, psia.

c. Fan efficiency, decimal.

d. Motor efficiency, decimal.

e. Inlet grain moisture content, decimal, dry basis.

f. Inlet grain temperature, deg F.

CARD 3 (From COOLONE input and DRYONE output)

a. Inlet air temperature, deg F.

b. Inlet air humidity, lb water vapor/lb dry air.

c. Optimal cooler depth, ft.

d. Optimal cooler airflow rate, lb dry air/hr/ft^2.

An example of the use of the program is next discussed; the conditions for the example were arbitrarily chosen.

Input information for the cooler program:

Air input conditions -- temperature, 52 deg F; absolute humidity, .003 lb vapor/lb dry air; atmospheric pressure, 14.34 psia.

Desired outlet conditions -- grain temperature, 80 deg F; product moisture content, 0.19, dry basis.

State and control variable ranges and increment sizes (after initial screening):

Moisture content: 0.203-0.195 dry basis, increment, 0.001

Product inlet temperature: 100.-140. deg F; increment, 5.0

Airflow rate: 100.-400. lb dry air/ft^2/hr; increment 10.

Depth of bed: 1.-2. ft; increment, 0.1

In the execution of this program, all remaining nodes were found to be feasible; hence, in the setup procedure for running DRYONE, no adjustments in the grid size were necessary. The range and increment size for the dryer controls were chosen to be:

Airflow rate: 400.-850. lb dry air/ft^2/hr; increment, 50.

Inlet air temperature: 200.-500. deg F; increment, 50.

Bed depth: 1.-3. ft; increment, 0.1

Inlet grain conditions were 0.25 moisture content, dry basis and 50. deg F.

Economic factors for both machine components were: electrical price 2¢/KWH; LP gas price, 16.5¢/gal; fuel heat value, 91547. BTU/gal; fan efficiency, 0.55; motor efficiency, 0.75; thermal efficiency, 0.70. An optimization time base of one hour was used for each model; grain flow rate was specified as 9.0 bu/hr/ft^2. The maximum total air temperature bound was 500. deg F; the maximum allowable pressure drop constraint was deactivated. The objective function used reflected only operating cost per square foot of dryer cross-section; no capital costs or product quality penalties were included.

The optimal conditions found were:

Cost: 15.7¢/ft^2/hr (or 1.74¢/bu)

Dryer depth: 2.6 feet

Dryer airflow rate: 600. lb dry air/hr/ft^2 (or 230.cfm/ft^2)

Inlet fan dryer temperature: without fan component, 450. deg F;
with fan component, 469. deg F

Cooler depth: 2.0 feet

Cooler airflow rate: 163. lb dry air/hr/ft^2 (or 35.cfm/ft^2).

Using RECONE, the resulting intermediate steady state air and grain conditions were identified. Outlet conditions for the air from the cooler were: 131. deg F and 0.0143 lb vapor/lb dry air; for the dryer air: 128. deg F and 0.0155 lb vapor/lb dry air. Outlet grain conditions for the cooler were: 79. deg F and 0.185 dry basis; for the dryer, 132. deg F and 0.202, dry basis. The overdrying from the desired final moisture content is a result of the specified product temperature constraining the solution.

From these results, mass and energy balances were performed as a check on the accuracy of the models used. For the cooler the moisture content model indicated that the amount of water lost by the grain was 7.36 lb/hr/ft^2 whereas the less accurate humidity model predicted a gain of only 1.84 lb/hr/ft^2 by the air. Using the former more reliable figure and performing an energy balance on the cooler, an average figure of 1100 Btu/lb water was obtained for latent heat of vaporization. This value compares well with experimental data (Hall,1957).

Similar computations for the dryer indicated a moisture loss of

20.8 $lb/hr/ft^2$ from the grain versus a 7.5 $lb/hr/ft^2$ gain by the air. Again using the product moisture loss as a basis, the average latent heat of vaporization was computed to be 1660 Btu/lb water. This value is higher than would be expected, indicating poor predictions by one or more of the dryer models. The foregoing balances are not efficiency computations for the system. Their calculation serves only as a check on the predictions of the component models used and hence provides no measure of either component or system performance. If, in the judgement of the user, the results of these checks are unacceptable, the component models must be improved before further use with the optimization technique.

A summary of measured process parameters for a dryer-cooler combination of this type are next given (Anderson,1971). Because these data are incomplete, no direct comparison can be made; however, they may provide a useful benchmark in future process optimization work.

Cooler: depth, approximately 3 feet

average outlet grain temperature, 72.4 deg F

average outlet grain moisture content, 16.5% wet basis

Dryer: average inlet air temperature, 530 deg F

average outlet air temperature, 130 deg F

depth, approximately 5.5 ft

average inlet grain flow rate, 725.4 $lb/min/100ft^2$

average inlet moisture content, 21.9% wet basis

airflow rate, 5710 $ft^3/hr/100ft^2$

Ambient conditions:

average dry bulb temperature, 52.3 deg F

average relative humidity, 63.7%

D. Optimization of Configuration with Air Recycle

D.1. Dryer-cooler description

The second dryer-cooler configuration for which an optimization technique was developed employed recycle of the warm air leaving the cooler (Figure 10). This air, which has increased in humidity and temperature during its passage through the cooler, is mixed with fresh air, further heated, and forced through the dryer. Since the temperature of the air at the cooler exit approaches that of the grain from the dryer, there seems to be the potential for minimizing the heat which must be added by the burner. This potential, however, will depend strongly on the amount of moisture added to the air in the cooler and on the ratio of cooler air to fresh air which enters the dryer. The fan, which draws air through the cooler by suction also adds frictional heat to the air prior to forcing it through the dryer.

Clearly, this is a much more difficult optimization problem than that for the process without recycle because of the added control variables and the fact that the air properties must be accounted for in the recycle process. In the development of the optimization procedure which follows, the grain flow rate, as well as economic and ambient conditions, will again be considered fixed. Thus, if the same models, constraints, and objective function are used as in the previous configuration, the optimal results of the two solutions can be directly compared.

D.2. Dynamic programming formulation and computer programming

The following development is in many respects an extension of the non-recycle technique. Hence, to avoid repetition only the crucial differences in the approach will be developed. Frequent reference to the exposition of the former technique will be made.

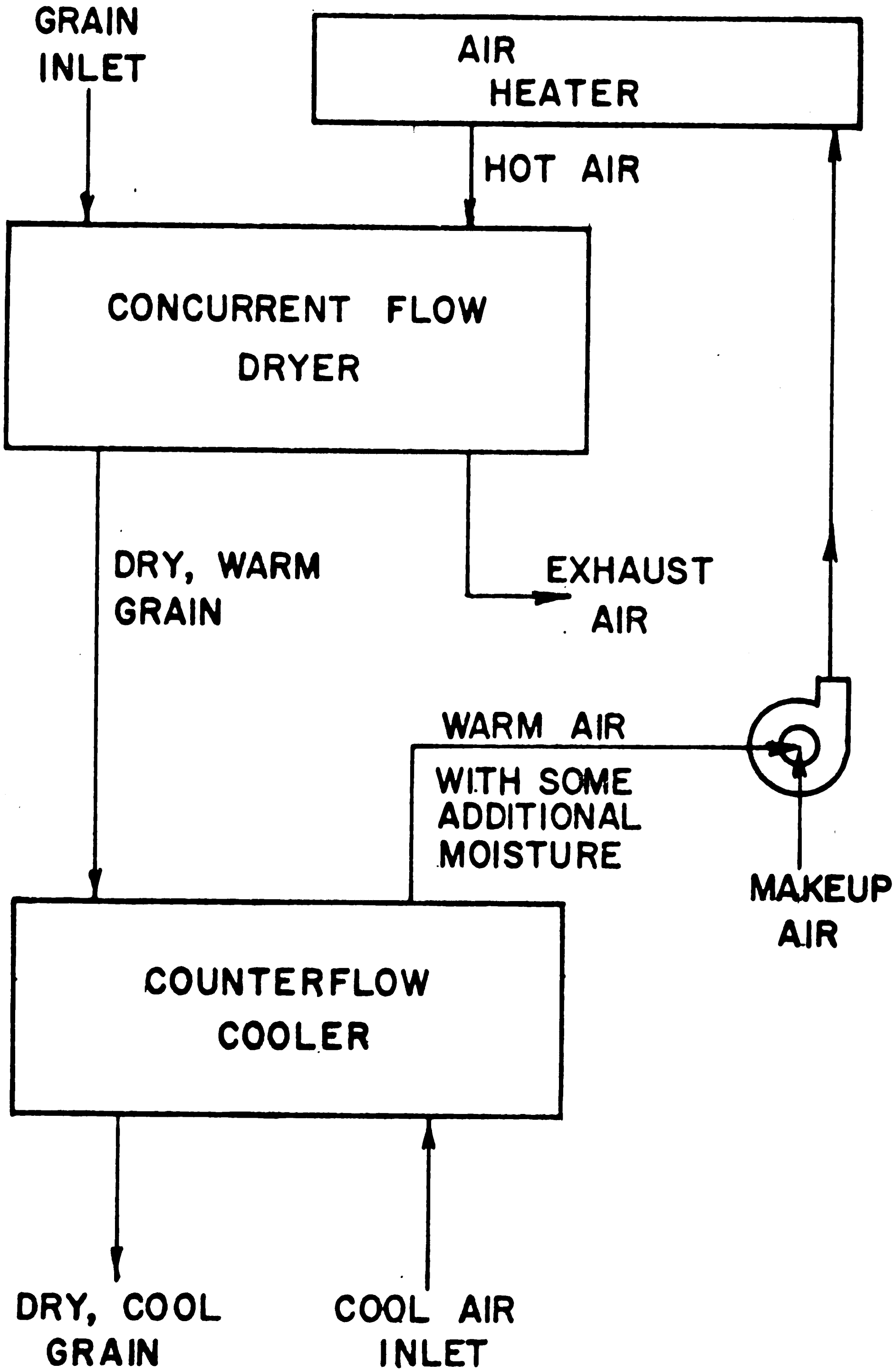

Figure 10: Schematic Representation of Combination Dryer-Cooler With Air Recycle

An overview of the system in terms of the dynamic programming formulation reveals some significant differences from the preceding problem. Unlike the process without air recycle, which could be called a straight-through process (the air was discarded at each stage), the process with recycle exhibits two streams--product and air--flowing in opposite directions between stages. In addition, there is a sidestream of fresh air entering the system prior to the heater. Such a system is termed a countercurrent flow process.

The dynamic programming ramifications for this type of system have been treated extensively by Dranoff et al.(1961). Basically, the authors show that just as in the straight-through process the backward dynamic programming recursion formula (Equation II.B.1) applies if each stage can be shown to be independent of those preceding it in the product stream. For the recycled air grain drying process, this condition is met by dividing the system into three d.p. stages: (i) the dryer, without fan or heater; (ii) the fan, heater, and mixing chamber for the air; and (iii) the cooler. In the cooler, the last d.p. stage in the system, the process of cooling from any given inlet moisture content and product temperature will be the same, regardless of the use that is made of the cooler exit air. The second stage (heater) model requires a knowledge of the downstream air properties as well as of the properties of the inlet air to achieve a specified set of outlet air conditions, but needs no information about the dryer upstream. The first d.p. stage is the dryer, which uses the air from the heater, but has no upstream counterpart.

In determining which properties of the air and product to employ as state variables, the product temperature and moisture content logically can be used to uniquely identify the state of the grain. It is also necessary to identify the state of the air between stages. This is accomplished by the use of three additional state variables: air humidity, temperature, and flow rate. The airflow rate, which ordinarily would be designated a control variable, must become a state variable because of the addition of outside air in the middle stage of the process.

The resulting five-dimensional state space poses a critical test for the computer implementation since a very large number of storage locations are required even to accomodate a rough grid mesh. Dranoff et al.,(1961), for example, consider the problem to be computationally feasible only in the dimensionality is two or less. However, by taking advantage of the properties and configuration of the process, the technique presented can give a good approximation to the optimal costs and controls for the system.

One of the difficulties which initially arose in the implementation of the problem was due to the inability of the computer, the CDC 6500, to work with arrays having a dimensionality greater than three. To circumvent this problem, a subprogram (SUBROUTINE IENCODE, Appendix C) was programmed to combine three of the indices into a single index, so that the values could be stored in three dimensions.

The interpolation subroutine (SUBROUTINE INTERP, Appendix C) was programmed to do linear interpolation in multiple dimensions. Other auxilliary subprograms which are required are: (i) DRYHEAT, associated with the interpolation subroutine INTERP; (ii) the cooler and heater models, COOLSIM and DRYSIM; (iii) the cooler and heater operating cost evaluations, CSTCOOL and DRYCOST; (iv) ZEROIN, a search routine for the zeroes, or roots, of an unknown function (Dekker, 1967), and associated functions DEEPMNT, DEEPMNM, and HOWDEEP; and (v) assorted psychrometric functions of the SYCHART package (Lerew,1971).

To reduce the memory requirements for the program instructions, the routines for the three stages were coded separately as programs DRYER, HEATER, and COOLER. Simulation and cost evaluation for the HEATER program were done internally; the program required no additional subprograms.

The overall strategy for the computer implementation was to avoid forming the cost interpolation array until the final dryer stage, thus enabling the intermediate results to be stored in low-access-speed storage devices. This was accomplished at the sacrifice of some computational efficiency and extra low-speed storage requirements. In the interim, by employment of scanning techniques such as described earlier, the size of the array created in the last stage could be substantially reduced to manageable size.

In the first program of the sequence, COOLER, a grid of values of each of the five dimensions at the cooler-heater interface is formed, although the values are stored sequentially on the low-access-speed storage device rather than forming an array. To accomplish this, an initial feasibility screening of the specified input ranges takes place, as in the previous algorithm, and user adjustments are made. Next, for each pair of discrete (indexed) product temperatures and moisture content values, a series of discrete decreasing airflow rates is applied. For each airflow rate, the ZEROIN search routine is used to fill the other two dimensions by searching for the bed depths where discrete values of air humidity occur and noting the corresponding temperature of the air at the cooler exit. The indices of the moisture content, product temperature, airflow rate, and humidity are stored as they are generated, along with associated nondiscrete values of air temperature and cost. Various feasibility checks are made during the computations to minimize the number of model and cost evaluations necessary.

In the second program, HEATER, the objective is to apply all available controls to each of the states represented by the nodal points at the heater-cooler interface and to produce a discrete (indexed) five-dimensional state space at the dryer-heater interface for interpolation in cost and controls.

Because the grain is unaffected by the heater in this stage, only the process costs and airflow conditions are altered. HEATER reads and processes as a block all those state nodal values at the heater-cooler interface which have repeated indices for product moisture content and temperature; this is an efficient process since they had been generated as a block in the COOLER program. Seven one-dimensional arrays are needed for temporary storage and processing of a block; the minimum dimensions needed for each is equal to the largest number of values to be read in a block.

For each state in the block, all controls on heat and airflow added are applied in turn and the cumulative cost computed. With the controls available, it is only possible to get two of the three state variables affected by this stage, air temperature and humidity, into the discrete, evenly spaced values necessary for simple interpolation in the DRYER stage. Therefore, an interpolation between the airflow rate values resulting from successive controls is performed at this point, and the resulting five state variable index values with associated costs and heater controls are written on a second low-speed storage device.

Since no attempt is made at this point to determine which of the values corresponding to a given set of state indices is the optimal cumulative cost, this procedure may produce many redundant data sets. However, since the values are written in a block corresponding to the indices of moisture content and product temperature, they are stored somewhat compactly on the memory device.

The third program, DRYER, using information supplied by the user regarding the ranges of indices generated by the HEATER program, initially reads the indices and cost values from the low-speed memory device and adjusts the indices to be used in the cost array to minimum size. By means of IENCODE, the indices in the airflow, air temperature, and air humidity dimensions are combined into a single index. This index, with the remaining moisture content and product temperature indices, is used to identify the location in the three dimensional array to be filled with a cost value. By comparison of all costs having the same set of indices, each position in the array is filled with the optimal cumulative cost at the dryer-heater interface; gaps in the array are filled by absurdly high cost values to prevent their use in the interpolation procedure.

Using the input air values indicated by the indices of the airflow rate, air temperature, and humidity dimensions, and applying discrete values of the only remaining control, dryer length, the operating costs due to airflow in this stage are assessed and added to the interpolated cost obtained from the dryer-heater-interface cost array. Appropriate feasibility checks and the program flow pattern are designated to limit the number of evaluations necessary. The minimum of all costs found by this procedure is the optimal total operating cost for the process.

The remaining task is to recover all optimal controls for the process. This is accomplished by an auxilliary program called RECOVRY, which uses information supplied by the user from the preceding programs to find optimal bed depths for both cooler and dryer, optimal airflows through both components, and the optimal amount of heat added by the heater stage. RECOVRY also uses the simulation models to summarize the conditions of operation between each of the stages and at both ends of the process.

By the same interpolation procedure used to find optimal cost, the optimal amount of added air and heat in the heater stage are recovered. Simple heat and mass balances, plus model evaluations recover all intermediate air and grain conditions and remaining controls except cooler depth. This is found by using the one-dimensional search technique, ZEROIN, to locate the depth at which the known cooler outlet humidity occurs.

To facilitate use of the algorithm, a detailed description of program inputs and intermediate steps follows. Information is to be supplied on data cards using FORTRAN IV F10.0 and I10 fields for real and integer values, respectively.

In preparation for using COOLER, the first of the four programs, the desired outlet product moisture content and temperature, airflow and depth ranges, and humidity increment must be chosen. Ambient and economic conditions to be fixed for the duration of the solution are needed. The following data cards are to be supplied by the user:

CARD 1

a. Maximum moisture content allowable at the entrance to the cooler, decimal, dry basis (real).

b. Minimum moisture content allowable at the entrance to the cooler, decimal, dry basis (real).

c. Number of levels of moisture content to be included in the grid between the maximum and minimum values (integer).

CARD 2

a. Maximum product temperature allowable at the entrance to the cooler, deg F (real).

b. Minimum product temperature allowable at the entrance to the cooler, deg F (real).

c. Number of levels of product temperature to be included in the grid between the maximum and minimum values (integer).

CARD 3

a. Maximum airflow rate through the cooler, lb dry air/hr/ft^2 (real).

b. Minimum airflow rate through the cooler, lb dry air/hr/ft^2 (real).

c. Number of levels of airflow rate to be included in the grid between the maximum and minimum values (integer).

CARD 4

a. Maximum cooler depth to be considered as a control variable, feet (real).

b. Minimum cooler depth to be considered as a control variable, feet (real).

CARD 5

a. Size of humidity increment desired for setting up the humidity dimension at the cooler-heater interface, lb water/lb dry air (real).

CARD 6

a. Inlet air temperature to the cooler, deg F (real).

b. Inlet air humidity to the cooler, lb water/lb dry air (real).

c. Atmospheric pressure, psia (real).

CARD 7

a. Constraint on maximum cooler outlet product temperature, deg F (real).

b. Constraint on maximum cooler outlet product moisture content, decimal, dry basis (real).

c. Constraint on minimum cooler outlet product moisture content, decimal, dry basis (real).

CARD 8

a. Price for fan energy (electrical energy, as programmed) required, ¢/KWH (real).

b. Time base over which optimization is taken, hr (real).

c. Fan efficiency, decimal (real).

d. Efficiency of fan power source, decimal (real).

The cooler model COOLSIM, and the electrical-energy-based objective function, CSTCOOL, may be replaced by equivalent subprograms supplied by the user, if desired. In the COOLER, HEATER, and DRYER programs, constraints read in as data values may be deactivated by setting them to absurdly low or high values.

The output format for COOLER, which is written onto a low access speed storage device as well as printed, provides the following information for each of the grid points at the cooler-heater interface: product moisture content index, product temperature index, airflow rate index, absolute humidity index, corresponding outlet air temperature and corresponding cost.

As explained previously, the initial part of COOLER is designed to use feasibility checks for limiting the ranges of state and control variables to be considered. Therefore, in initial trials, the use of a STOP card immediately after the screening section is recommended.

The second program, HEATER, reads the information generated by COOLER, then performs heat and mass balances calculating the amount of added fresh air and heat necessary to achieve discrete values of outlet conditions; the following information is to be read from data cards:

CARD 1

a. Minimum airflow rate at the cooler-heater interface as read from COOLER data, lb dry air/hr/ft^2 (real).

b. Airflow rate increment, as read from COOLER data, lb dry air/hr/ft^2 (real).

c. Minimum air humidity at the cooler-heater interface, as read from COOLER data, lb water/lb dry air (real).

d. Air humidity increment, as read from COOLER data, lb water/lb dry air (real).

e. Minimum moisture content bound at heater-cooler interface, as read from COOLER data, decimal, d.b. (real).

f. Maximum moisture content bound at heater-cooler interface, as read from COOLER data, decimal, d.b. (real).

g. Minimum product temperature bound at heater-cooler interface, as read from COOLER data, deg F (real).

h. Maximum product temperature bound at heater-cooler interface, as read from COOLER data, deg F (real).

CARD 2

a. Lower airflow rate bound at the heater-dryer interface, lb dry air/hr/ft^2 (real).

b. Upper airflow rate bound at the heater-dryer interface, lb dry air/hr/ft^2 (real).

c. Maximum permissible ratio of cooler exhaust air to total air, decimal (real).

d. Lower air temperature bound at the heater-dryer interface, deg F (real).

e. Upper air temperature bound at the heater-dryer interface, deg F (real).

f. Lower air humidity bound at the heater-dryer interface, lb water/lb dry air (real).

g. Upper air humidity bound at the heater-dryer interface, lb water/lb dry air (real).

CARD 3

a. Number of levels of moisture content at heater-cooler interface, from COOLER data (integer).

b. Number of levels of product temperature at heater-cooler interface, from COOLER data (integer).

c. Number of levels of airflow rate at dryer-heater interface (integer).

d. Number of levels of air temperature at dryer-heater interface (integer).

e. Number of levels of air humidity at dryer-heater interface (integer).

CARD 4

a. Inlet air temperature to heater, deg F (real).

b. Inlet air humidity to heater, lb water/lb dry air (real).

c. Atmospheric pressure, psi (real).

CARD 5

a. Heat value of fuel used, Btu/gal (real).

b. Price of fuel used, ¢/gal (real).

c. Heater efficiency, decimal (real).

d. Time base for optimization, hours (real).

The size of each of the seven one-dimensional arrays in HEATER is found by counting the maximum number of data sets to be read from the COOLER output which have the same indices of product moisture content and temperature.

The output from HEATER, which is both printed and written on a low-access-speed storage device, consists of indices for the moisture content, product temperature, air temperature, air humidity, and airflow rate dimensions, plus values of cumulative cost, added heat, and added air from the HEATER model. To reduce the size of the cost array, to be formed in the DRYER program, the range of indices for each state variable is found from a scan of the HEATER data and is input on the data cards for DRYER; the size of the 3-dimensional cost array is given by the maximum number of remaining feasible levels for (i) moisture content and (ii) product temperature and (iii) by the product of the remaining feasible levels of the other three state variables.

CARD 1

a. Number of feasible levels remaining in moisture content dimension (integer).

b. Number of feasible levels remaining in product temperature dimension (integer).

c. Number of feasible levels remaining in airflow rate dimension (integer).

d. Number of feasible levels remaining in air temperature dimension (integer).

e. Number of feasible levels remaining in humidity dimension (integer).

CARD 2

a. Lower feasible bound of moisture content at the heater-dryer interface, decimal, d.b. (real).

b. Moisture content increment at heater-dryer interface, decimal, d.b. (real).

CARD 3

a. Lower feasible bound of product temperature at the heater-dryer interface, deg F (real).

b. Product temperature increment at heater-dryer interface, deg F (real).

CARD 4

a. Lower feasible bound of airflow rate at the heater-dryer interface, lb air/hr/ft^2 (real).

b. Increment of airflow rate at the heater-dryer interface, lb air/hr/ft^2 (real).

CARD 5

a. Lower feasible bound of air temperature at the heater-dryer interface, deg F (real).

b. Increment of air temperature at the heater-dryer interface, deg F (real).

CARD 6

a. Lower feasible bound of humidity at the heater-dryer interface, lb water/lb air (real).

b. Increment of humidity at the heater-dryer interface, lb water/lb air (real).

CARD 7

a. Lower moisture content bound before adjustment, decimal, d.b. (real).

b. Lower product temperature bound before adjustment, deg F (real).

c. Lower airflow rate bound before adjustment, lb dry air/hr/ft^2 (real).

d. Lower air temperature bound before adjustment, deg F (real).

e. Lower humidity bound before adjustment, lb water/lb air (real).

CARD 8

a. Lower bound of dryer depth to be considered as a control, feet (real).

b. Upper bound of dryer depth to be considered as a control, feet (real).

c. Number of levels of dryer depth between upper and lower bounds to be considered (integer).

CARD 9

a. Moisture content of grain entering dryer, decimal, d.b. (real).

b. Temperature of grain entering dryer, deg F (real).

c. Time base used for optimization, hours (real).

d. Price of energy (electrical energy) for fan, ¢/KWH (real).

e. Fan efficiency, decimal (real).

f. Efficiency of fan power source, decimal (real).

g. Atmospheric pressure, psia (real).

CARD 10

a. Maximum inlet air temperature to the dryer, deg F (real).

b. Maximum pressure drop through dryer, inches of water (real).

The optimal total operating cost is given by DRYER, in addition to the optimal dryer depth and the resulting grain temperature and moisture content at the dryer exit .

The last program, RECOVRY, uses this, plus additional information from COOLER and HEATER to recover by interpolation all intermediate states and optimal controls. The following data cards are required for the use of RECOVRY; all input values are real and are read in F10.0 fields.

CARD 1 (from DRYER output)

a. Optimal airflow rate, lb dry air/hr/ft^2.

b. Optimal inlet air temperature, deg F.

c. Optimal inlet absolute humidity, lb water vapor/lb dry air.

d. Optimal dryer depth, feet.

e. Resultant dryer outlet moisture content, decimal, dry basis.

f. Resultant dryer outlet product temperature, deg F.

Next, note the indices in the HEATER grid corresponding to the optimal values of airflow rate, inlet air temperature, and inlet absolute humidity from the optimal DRYER results. For the resultant DRYER outlet moisture content and outlet product temperature values, determine the next lowest discretized values of both variables and their corresponding indices in the HEATER grid. From the HEATER output, read the least cost "added airflow" and "added heat" controls corresponding to the five indices: (i) moisture content, (ii) product temperature, (iii) airflow rate, (iv) air temperature, (v) absolute humidity.

CARD 2 (from HEATER output)

a. Discretized moisture content corresponding to the index found in the above procedure, decimal, dry basis.

b. Discretized product temperature corresponding to the index found in the above procedure, deg F.

c. Least cost "added airflow" control corresponding to the five indices found in the above procedure, lb dry air/hr/ft^2.

d. Least cost "added heat" control corresponding to the five indices found in the above procedure, Btu/hr/ft^2.

Now raise the moisture content index by one, holding the other four constant, and read the least cost "added airflow" and "added heat" controls in the HEATER output corresponding to the new set of indices.

CARD 3 (from HEATER output)

a. Increment size in moisture content dimension, decimal, dry basis.

b. Least cost "added airflow" control corresponding to the five indices found by the preceding method, lb dry air/hr/ft^2.

c. Least cost "added heat" control corresponding to the five indices found by the preceding method, Btu/hr/ft^2.

Next, return to the set of five indices used for Card 2 (i.e. return the moisture content index to its original value). Then, raise the value of the product temperature index by one, holding the other four constant, and read the least cost "added airflow" and "added heat" controls in the HEATER output corresponding to the new set of indices.

CARD 4 (from HEATER output)

a. Increment size in product temperature dimension, deg F.

b. Least cost "added airflow" control corresponding to the five indices found by the preceding method, lb dry air/hr/ft^2.

c. Least cost "added heat" control corresponding to the five indices found by the preceding method, Btu/hr/ft^2.

The final three data cards are taken from the input parameters supplied for the three prior programs.

CARD 5 (from DRYER input)

a. Inlet moisture content to dryer, decimal, dry basis.

b. Inlet product temperature to dryer, deg F.

CARD 6 (from HEATER input)

a. Temperature of inlet air to heater, deg F.

b. Absolute humidity of inlet air to heater, lb water vapor/lb dry air.

c. Fan efficiency, decimal.

d. Motor efficiency, decimal.

CARD 7 (from COOLER input)

a. Temperature of inlet air to cooler, deg F.

b. Absolute humidity of inlet air to cooler, lb water vapor/lb dry air.

c. Maximum allowable cooler depth, feet.

d. Minimum allowable cooler depth, feet.

e. Atmospheric pressure, psia.

D.3. Example and discussion of results

A second example problem was formulated, using ambient and economic conditions (summarized on pp. 84-85) identical to those of the non-air-recycle case, but employing the dryer configuration with air recycle.

In the COOLER program, after initial adjustments, the resulting state variable ranges and increment sizes were:

product moisture content: 0.195-0.203 d.b.; increment, 0.002

product temperature: 130.-160. deg F; increment, 5. deg F

airflow rate: 200.-500. lb dry air/hr/ft^2; increment, 50. lb dry air/hr/ft^2

humidity increment: 0.0001 lb vapor/lb dry air

The indices of these four state variables, the corresponding outlet temperatures, and the associated costs were stored sequentially on the low-access-speed storage device.

In the HEATER program, blocks of these cost data were processed according to the indices of the first two state variables, product moisture content and temperature. The size of the seven one-dimensional arrays needed for block processing was chosen by scanning the COOLER output data and locating that pair of product temperature and moisture content indices which were most often repeated. The corresponding (decimal) number of storage locations needed for each array was 49 for a total of 343. HEATER then generated the indices for the five state variables, the cumulative cost, and associated heater controls, which were stored sequentially on a low-access-speed storage device. A constraint was applied which limited the maximum recycled air:total air ratio to 0.5. Hence the number of feasible combinations was substantially reduced.

Prior to the running of the third program, DRYER, a scan of the stored indices allowed adjustment of the ranges of the five state variables to:

product moisture content: 0.195–0.203 d.b.; increment 0.002

product temperature: 130.-160. deg F; increment 5.0 deg F

airflow rate: 650.–800. lb dry air/hr/ft^2; increment 50.

air temperature: 200.-500. deg F; increment 50.

air humidity: .0055-.0069 lb vapor/lb dry air; increment .0002.

The resulting three-dimensional cost array size, using this relatively coarse grid, was 5x6x224 or 6720 decimal storage locations.

Total execution time required for the three programs using a CDC 6500 computer was approximately 95 seconds. The optimal total process cost was 15.8¢/ft^2/hr or 1.76¢/bu.

The final program, RECOVRY, identified all optimal controls for the process and produced values of the state variables at the inlet, outlet, and intermediate stages of the process. These are

dryer exit air temperature: 133.7 deg F

dryer exit air humidity: 0.0141 lb vapor/lb dry air

dryer inlet product temperature: 50. deg F

dryer inlet grain moisture content: 0.25 dry basis

optimal dryer length: 4.5 feet

optimal airflow rate through dryer: 650. lb dry air/hr/ft^2 (231.cfm/ft^2

optimal inlet air temperature to dryer: 379.8 deg F.

dryer exit moisture content: 0.202 dry basis

dryer exit product temperature: 143.2 deg F.

heat added by burner: 42031. Btu/hr/ft^2

optimal fresh air added to system: 388.9 lb dry air/hr/ft^2 (84. cfm/ft^2)

cooler exit air temperature: 142.1 deg F

optimal cooler depth: 1.47 feet

optimal cooler airflow rate: 261.1 lb dry air/hr/ft^2 (56.4 cfm/ft^2)

The outlet product conditions, as calculated by working backward using the optimal controls, are: 83.8 deg F and 18.9% d.b. The good agreement with the specified outlet conditions of 80.0 deg F and 19.0% d.b. indicates that even with the rough grid employed, a near-optimal solution has been found and that little accuracy has been lost in the linear interpolations performed. Because no optimal controls or resulting states lie on the bounds prescribed for the problem, a good choice of ranges is indicated.

Mass and energy balances were performed on the dryer and cooler components of the system; since the heater model was itself a mass and energy balance, no information on adequacy of its model could be gained by this procedure.

For the cooler, using the output data given, a mass balance indicated a loss of 5.5 lb water/hr/ft^2 from the corn versus a gain of 6.8 lb vapor/hr/ft^2 by the air. Based on the change in moisture content of the grain (the more accurate model), a latent heat of vaporization of 1205. Btu/lb of water was calculated. Because the range of the cooler model had been extrapolated beyond its proven depth limit for the purposes of this example, better results can be expected using shallower beds or more comprehensive empirical equations. Improvement in the cooler model, which comes computationally first in the algorithm, will reduce the error carried on in subsequent calculations.

Similar calculations performed on the dryer output data indicated a moisture input to the dryer of 112.6 lb moisture/ft^2/hr versus an output of 96.6 lb moisture/ft^2/hr. This descrepancy was due primarily to the model for outlet air humidity; in the solution process this model was used only as a feasibility check and introduced no inherent error into the subsequent calculations if the optimal predicted dryer conditions were indeed feasible.

Presuming that the model for outlet air temperature is reasonable (using the outlet product temperature as a basis) and adding all additional moisture to the air which is needed to balance the equation, the resulting outlet air humidity of 0.0386 lb water/lb dry air is far below the saturation value of 0.1200 lb water/lb dry air at the predicted outlet air temperature, 133.7 deg F. Based on the more reasonable moisture content difference, a latent heat of vaporization of 1041. Btu/lb water is calculated for the dryer.

Comparing the values obtained for latent heat with those given by Hall(1957), the calculated values are reasonable; the drier grain passing through the cooler requires additional heat, on the average, to evaporate a pound of water than does that passing through the dryer.

The efficiency calculated for the system, based only on heat energy reaching the dryer inlet, is 1940. Btu per pound of water evaporated. The energy expended for airflow only is 0.063 HP-hours per pound of water evaporated. If thermal, motor and fan efficiencies are included, the figures become 2775. Btu and 0.153 HP-hours per pound of water evaporated.

D.4. Comparison of three example programs

The optimal controls and resulting states for the two previous examples are, as might be expected, quite different; results of these tests are summarized in Table 5. The slight superiority in per bushel operating cost for the dryer without recycled air is most likely attributable to the favorable drying weather conditions chosen for the example. In the recycling case, the amount of moisture picked up by the air in its passage through the cooler makes it unprofitable to use for recycle, despite the heat it has gained. Commercial dryers of the recycle design are rated for operating cost at 0 deg F (Anon.,1971), which suggests that under cold temperature conditions these dryers may have the advantage of greater economy. It may be possible to develop dryer designs which can be easily switched from one type to the other, depending on weather conditions.

Table 5: Summary of optimal conditions for three drying systems

	Recycle System	Non-recycle System	Dryer Alone
COOLER			
Inlet air temp, deg F	52.	52.	-
Inlet air humidity, lb water vapor/lb dry air	0.003	0.003	-
Inlet grain temperature, deg F	143.2	131.8	-
Inlet grain moisture content, decimal, dry basis	0.202	0.202	-
Optimal depth, feet	1.47	2.00	-
Optimal airflow rate, $lb/hr/ft^2$	261.1	163.	-
Outlet air temperature, deg F	142.1	131.	-
Outlet air humidity, lb water vapor/lb dry air	0.0117	0.0143	-
Outlet grain temperature, deg F	83.8	79.	-
Outlet grain moisture content, decimal, dry basis	0.189	0.185	-
HEATER			
Inlet air temperature, deg F	142.1 and 52.	-	-
Inlet air humidity, lb water vapor/lb dry air	0.0117 and 0.003	-	-
Optimal airflow rate, $lb/hr/ft^2$	650.	-	-
Optimal heat input, $Btu/hr/ft^2$	42031.	-	-
Outlet air temperature, deg F	379.8	-	-
Outlet air humidity, lb water vapor/lb dry air	0.0065	-	-
DRYER			
Inlet air temperature, deg F	379.8	469.	491.
Inlet air humidity, lb water vapor/lb dry air	0.0065	0.003	0.003
Inlet grain temperature, deg F	50.	50.	50.
Inlet grain moisture content, decimal, dry basis	0.25	0.25	0.25
Optimal depth, feet	4.5	2.6	5.0
Optimal airflow rate, $lb/hr/ft^2$	650.	600.	500.
Outlet air temperature, deg F	133.7	128.	142.
Outlet air humidity, lb water vapor/lb dry air	0.0141	0.0155	0.0401
Outlet grain temperature, deg F	143.2	132.	119.9
Outlet grain moisture content, decimal, dry basis	0.202	0.202	0.190
TOTAL COST, ¢/bu	1.76	1.74	1.57

Because the optimal operating conditions for the non-recycle system lie on several bounds imposed by the model, it is possible that additional savings in operating cost may be gained by extension of the model limits.

As expected, the amount of labor, computer memory, and time required for running a program of the non-recycle design are minimal compared to those for the more complex system. A single solution, using a fine mesh in state and control variables, may be sufficient to achieve acceptable results for the former algorithm, while the latter may require several applications to narrow the limits of uncertainty.

A third program, utilizing the concurrent dryer only, was run in order to compare operating costs with those of the cooler-dryer combinations. For this case, the outlet grain moisture content desired was again 0.19 dry basis, while the outlet grain temperature was unconstrained. A comparison of the optimal results is shown in Table
Because no cooling was done, the cost per bushel is lower than for the dryer-cooler combinations. The outlet grain temperature, however, is 119.9 deg F, which is unsuitable for storage without further cooling.

E. Summary and Conclusions

The primary goal of this study, two user-oriented algorithms for design and analysis of competing grain dryer designs, has been achieved. Instructions for the use and modification of the two programs have been included. Both algorithms have been tested for performance, flexibility, ease of use, and computational feasibility on an existing computer.

The empirical models which were necessary for the testing of the two algorithms have shown deficiencies when subjected to heat and mass balance checks; for successful design, some of these will require improvement and/or extension of the ranges over which they apply; alternate approaches to modeling the dryer and cooler components may be required.

In the two comparable examples of algorithm use presented, the dryer not using the air recycle was shown to have slightly superior operating cost performance under the constraints imposed and economic and ambient conditions assumed. Optimal operating conditions for the two dryer systems were quite different.

F. Suggestions for Further Study

The following suggestions for use and improvement of the algorithm are made:

1. The primary requirement for further use of the algorithms is the development of more accurate models for the system components. Although the present models are sufficient to provide much information about the nature of the optimal solution, they are inadequate for the purpose of making precise design recommendations. For the cooler model, in particular, an extension to greater depths is suggested.

2. User experience with the algorithms is required in order to reduce the computational requirements for the solutions; parameter studies of the effect of economic and ambient conditions will permit a priori narrowing of the state and control ranges to be considered.

3. The algorithms, as presently programmed, are primarily written for comprehensibility rather than computational efficiency. By reconsideration of the order in which evaluations are processed, by increased efficiency in programming, and by devising new checks on the limits of feasibility, computational speed and program memory requirements can be further optimized.

4. Adaptation of these algorithms to other possible process variations should be considered. For instance, the recycling solution technique can be easily changed to investigate the possibility of recycling only a fraction of the air from the cooler. Another design worthy of interest is the insertion of a heat exchanger into the system to utilize heat of the exhausted air from the dryer.

5. Two of the expressions used in the program require further attention. The available formula for pressure drop through the dryer and cooler is based on depth and an airflow rate measured in cfm. Because the air volume is temperature dependent, this requires a correction in the solution procedure, in addition to the computational inefficiency of conversion to lb dry air/hr, the basic units of the airflow rate terms in the algorithms. An improvement would be an expression for pressure drop based on mass flow rate and bed depth.

The second expression to be analyzed for reliability in the algorithms is the term expressing the amount of heat added to the air by the fan and motor inefficiency. In the absence of a better equation, this expression, developed from low-temperature low-airflow data in Part I of the study, has been extrapolated far beyond its intended limits. Because this inefficiency can make an appreciable contribution to the heat added to the air under conditions of high airflow rates and deep beds, some experimental checks on the adequacy of this estimate are needed.

6. Modifications to the constraints and objective function may be made, if desired. For example, fixed costs may be included in the objective function if they vary smoothly with respect to the control variables; this requirement on smoothness prevents errors in interpolation. In addition, compatible time scales must be chosen for fixed and

operating cost in order to keep both equally represented in the objective function.

7. In the event that premium prices are paid for quality grain, attention should be given to the choice of an appropriate penalty term for the objective function which will accurately reflect the economic cost of dryer operation on grain quality.

REFERENCES

Ahn, Y. K., H. C. Chen, L. T. Fan and C. G. Wan (1964). Optimum design of a moving bed grain dryer by dynamic programming. The Canadian Journal of Chemical Engineering 42:117-120.

American Society of Agricultural Engineers (1971). Agricultural Engineers Yearbook. American Society of Agricultural Engineers, St. Joseph, Michigan, 572 pp.

Anderson, R. (1971). President of The Andersons Incorporated, Maumee, Ohio. Personal communication.

Anonymous (1958). Electric motors for farm use. Vocational Agricultural Service, University of Illinois, Urbana, Illinois.

Anonymous (1971). Carter-Day's HC dryer--An exclusive new concept in grain drying. Carter-Day Incorporated, Minneapolis, Minnesota.

Bakker-Arkema, F. W. and D. B. Brooker, Eds. (1971). Proceedings of the Institute for Simulation of Cooling and Drying Beds of Agricultural Products. Department of Agricultural Engineering, Michigan State University, East Lansing, Michigan.

Bakker-Arkema, F. W. and L. E. Lerew (1972 in press). Simulation of shelled corn drying. Research Report - Michigan Agricultural Experiment Station, Michigan State University, East Lansing, Michigan.

Barre, H. J., G. R. Baughman, and M. Y. Hamdy (1970). Application of the logarithmic model to deep bed drying of grain. Paper 70-327, American Society of Agricultural Engineers, St. Joseph, Michigan.

Bellman, R. and R. Kalaba (1965). Dynamic Programming and Modern Control Theory. Academic Press, New York. 112 pp.

Bloome, Peter Dale (1970). Simulation of low temperature drying of shelled corn leading to optimization. Thesis for the degree of Ph.D., University of Illinois, Champaign. (Unpublished).

Bussell, J. D. (1969). Problems of grain conditioning as viewed by the manufacturer. pp. 8-12, in Corn Conditioning Conference Proceedings. University of Illinois, Champaign, Illinois.

DeBoer, S. F. (1971). Graduate Research Associate, Department of

Agricultural Engineering, Michigan State University, East Lansing, Michigan. Personal communication.

Dekker, T. J. (1967). Finding a zero by means of successive linear interpolation. pp. 37-48 in Conference of Constructive Aspects of the Fundamental Theorem of Algebra, Zurich, Switzerland.

Dranoff, J. S., L. G. Mitten, W. F. Stevens and L. A. Wanninger Jr. (1961). Application of dynamic programming to countercurrent flow processes. Operations Research 9:388-401.

Farmer, D. M. (1969). Department of Agricultural Engineering, Michigan State University East Lansing, Michigan. Unpublished deep bed drying tests.

Farmer, D. M. (1971). Materials handling system design for shelled corn. Unpublished research report. Department of Agricultural Economics, Michigan State University, East Lansing, Michigan.

Hadley, G. (1964). Nonlinear and Dynamic Programming. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts. 484 pp.

Hall, Carl W. (1957). Drying Farm Crops. Edwards Bros., Inc., Ann Arbor. 336 pp.

Hamdy, M. Y. and H. J. Barre (1969). Theoretical analysis and simulation of deep bed drying of grain. Paper 69-328, American Society of Agricultural Engineers, St. Joseph, Michigan.

Hamdy, M. Y. and H. J. Barre (1970). Analysis and hybrid simulation of deep-bed drying of grain. Transactions of the American Society of Agricultural Engineers:13,6, 752-757.

Harsh, S. B. and J. R. Black (1970). On the use of time-share computers in extension education--the Michigan experience. Western Agricultural Economics Association Proceedings.

Hukill, William V. (1954). Drying of grain. Chapter 9 in J. A. Anderson and A. W. Alcock, Storage of Cereal Grains and their Products. American Association of Cereal Chemists, St. Paul, Minnesota.

Kirk, D. E. (1959). Column thickness for shelled corn dryers. Transactions of the American Society of Agricultural Engineers:2,1, 42-3.

Kline, C. M. and M. Best. (1971). Grain Drying Process and Apparatus. U. S. Patent Number 3,373,503 Re. 27,030, January 12.

Larson, R. E. (1968). State Increment Dynamic Programming. American Elsevier Publishing Company, Inc., New York. 256 pp.

Lerew, L. E. (1971). Guide to the use of SYCHART, a Fortran psychrometric package. Unpublished report. Department of Agricultural Engineering, Michigan State University, East Lansing, Michigan.

Maddex, R. L. (1971). Raising corn for profit. Michigan Field Crops short course, Lessons 5 and 6. Drying-Handling-Storing Shelled Corn. Agricultural Extension Service, Department of Agricultural Engineering, Michigan State University, East Lansing.

Meeter, D. A. (1965). Non-Linear Least Squares (GAUSHAUS). Michigan State University Computer Laboratory Program Documentation No. 87.

Mitten, L. G. and G. L. Nemhauser (1963). Multistage optimization. Chemical Engineering Progress:59,1,52-60.

Morey, R. V. and R. M. Peart (1969). Optimization of a natural air corn drying system. Paper 69-834, American Society of Agricultural Engineers, St. Joseph, Michigan.

Nelson, G. L. (1960). A new analysis of batch grain dryer performance. Transactions of the American Society of Agricultural Engineers: 3,2,81-88.

Perry, John H. (1963). Perry's Chemical Engineers' Handbook, 4th Edition. McGraw-Hill Book Company, New York.

Rosenbrock, H. H. (1960). An automatic method for finding the greatest or least value of a function. Computer Journal:3,175-184. Algorithm programmed by D. M. Koenig, Department of Chemical Engineering, Ohio State University, Columbus, Ohio.

Rosenbrock, H. H. and C. Storey (1966). Computational Techniques for Chemical Engineers.Pergamon Press, Oxford, Great Britain. 328 pp.

Roth, M. G. (1971). Research Associate, Department of Agricultural Engineering, Michigan State University, East Lansing, Michigan. Personal communication.

Schroeder, M. E. and R. M. Peart (1965). Dynamic programming method of air allocation in a grain dryer. Paper 65-923, American Society of Agricultural Engineers, St. Joseph, Michigan.

Scott, J. T.,Jr. (1969). The economics of corn conditioning and storage alternatives for farmers. Agricultural Economics Research Report 98. Department of Agricultural Economics, Agricultural Experiment Station, University of Illinois, Champaign, 22 pp.

Smith, C. L., R. W. Pike, and P. W. Murill (1970). Formulation and Optimization of Mathematical Models. International Textbook Company, Scranton, Pennsylvania, 581 pp.

Thompson, Thomas L. (1967). Predicted performances and optimal designs of convection grain dryers. Thesis for the degree of Ph.D, Purdue University, Lafayette, Indiana. (Unpublished).

Threlkeld, J. L. (1962). Thermal Environmental Engineering. Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 514 pp.

Thygeson, J. R.,Jr. and E. D. Grossman (1971). Optimization of a continuous through-circulation dryer. AIChE Journal:16,5,749-754.

Wilde, D. J. and C. S. Beightler (1967). Foundations of Optimization. Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 480 pp.

Wong, Peter J. (1967). Dynamic programming using shift vectors. Thesis for the degree of Ph.D, Stanford University, Stanford, California. (Unpublished).

APPENDICES

APPENDICES

(Note: The subprograms required for each main program are given after the program name; the subprograms are listed alphabetically in Appendix C.)

APPENDIX A

```
      PROGRAM CONTROL(INPUT,OUTPUT)
C  PROGRAM CONTROL READS DATA, CONVERTS UNITS,AND CALLS ROUTINES
C  DETERMINED BY OPTIONS CHOSEN BY THE USER.
      COMMON/PRESS/PATM/FCON/FANETA,EMETA,THERETA,ELPRICE,FUPRICE,TAMB,F
     1ACTOR,ELCOST,FULCOST/ICHECK/IFLAG/DT/DEPTH,TIME/FINAL/XMFINAL/STUF
     2F/DIAGSLP,QMIN,TMIN,QMAX,TMAX/BOUND/XMBND,BNDM/AMB/PSWB,HPFG,ATWB,
     3XMZERO/OTHER/DFLX,PV/QT/QUE,TEE
      EXTERNAL COVFR,SIDE
      DATA IBASIS,IQ,IHUMID,JHUMID,JCOST,JOPT,JBOTH,JWHICH,IPRESS,IFUEL/
     110H  WETBASIS,10HCFMPERSQFT, 10H   WETBULB,10H    ABSHUM,10H    NO
     1COST,10H  OPTIMIZE,10H      BOTH,10H INLETTEMP,10H       PSI,10H
     3  GALLON/
      READ 10020,ICHECK6,ICHECK7
      PRINT 20001,ICHECK6,ICHECK7
C     THE FOLLOWING SECTION SETS UP FOR QUICK
      READ 10001,ICHECK1,XMZERO,XMFINAL,XMBND
      PRINT 20002,XMZERO,XMFINAL,XMBND,ICHECK1
      IF(ICHECK1.EQ.IBASIS)10,13
 10   XMZERO=XMZERO/(1.-XMZERO) $ XMFINAL=XMFINAL/(1.-XMFINAL)$XMBND=XMB
     2ND/(1.-XMBND)
 13   IF(XMZERO.LE.0.4.AND.XMZERO.GE.0.0)GO TO 20 $ PRINT 10002
      STOP
  20  READ 10001,ICHECKA,PATM
      PRINT 20003,PATM,ICHECKA
      IF(ICHECKA.EQ.IPRESS)GO TO 35$PATM=0.491*PATM
  35  READ 10001,ICHECK2,Q,QMAX,QMIN
      PRINT 20004,Q,QMAX,QMIN,ICHECK2
      IF(ICHECK2.EQ.IQ)60,50
50    Q=Q*DEPTH/1.25$QMAX=QMAX*DEPTH/1.25$QMIN=QMIN*DEPTH/1.25
 60   IF(Q.GE.5..AND.Q.LE.50. )GO TO 61 $ PRINT 10003
      STOP
 61   IF(ICHECK6.EQ.JOPT.AND.(ICHECK7.NE.JBOTH.AND.ICHECK7.NE.JWHICH))62
     1,64
62    IF(QMAX.GT.50..OR.QMIN.LT.5..OR.QMAX.LT.QMIN)63,64
63    PRINT 10008
      STOP
64    READ 10010,DEPTH,TIME,TAMBDB,TINLET,TMAX,TMIN$ATAMB=TAMBDB+459.69
      PRINT 20005,DEPTH,TIME,TAMBDB
      PRINT 20055,TINLET,TMAX,TMIN
      TAMB=TAMBDB
      IF(ICHECK6.EQ.JOPT.AND.(ICHECK7.EQ.JBOTH.OR.ICHECK7.EQ.JWHICH))
     1645,65
645   IF(TMAX.GT.140..OR.TMIN.LT.70..OR.TMAX.LT.TMIN)646,65
646   PRINT 10007
      STOP
  65  READ 10001,ICHECK3,HUMID
      PRINT 20006,HUMID,ICHECK3
      IF(ICHECK3.EQ.IHUMID)70,90
70    IF(HUMID.GT.32..AND.HUMID.LT.100.)GO TO 78
75    PRINT 10004
      STOP
78    AWBAMB=HUMID+459.69$PV=PVDBWB(ATAMB,AWBAMB)$HSP=HAPV(PV)$GO TO 120
90    IF(ICHECK3.EQ.JHUMID)100,110
100   IF(HUMID.GT.0.001.AND.HUMID.LT.0.1)103,105
103   PV=PVHA(HUMID)$HSP=HUMID$GO TO 120
105   PRINT 10005
      STOP
110   IF(HUMID.GT.0.0.AND.HUMID.LT.1.0)115,113
113   PRINT 10006
      STOP
```

```
115   HSP=HADBRH(ATAMB,HUMID)$PV=PVHA(HSP)
120   FACTOR=(0.242+0.45*HSP)/VSDBHA(ATAMB,HSP)$DELX=DEPTH/20;
C THE FOLLOWING SECTION SETS UP FOR THE COST FUNCTION,IF USED;
130   READ 10022,FANETA,EMETA,THERETA,ELPRICE,FUPRICE,HTVAL
      PRINT 20007,FANETA,EMETA,THERETA
      PRINT 20075,ELPRICE,FUPRICE,HTVAL
      IF(ICHECK6.EQ.JCOST)GOTO140$FUPRICE=FUPRICE/HTVAL$IF(ICHECK6.EQ.JO
     1PT)GOTO150$CALL COSTFUN(COST,TINLET,Q)$COSTPBU=COST/DEPTH*1.25
      PRINT 10023,ELCOST,FULCOST,COST,COSTPBU
140   CALL QUICK(AVEMCDB,TINLET,Q)$AVEMCWB=AVEMCDB/(1.+AVEMCDB)$PRINT
     110024,TINLET,Q,AVEMCDB,AVEMCWB
      STOP
150   IF(ICHECK7.EQ.JBOTH)GO TO 180$IF(ICHECK7.EQ.JWHICH)160,170
C     AIRFLOW IS FIXED,ONLY INLET TEMP CAN VARY
160   GUESS1=TMAX$GUESS2=TMIN$QUE=Q$CALL ZEROINA(GUESS1,GUESS2,0.000001,
     1COVER)$TEE=(GUESS1+GUESS2)/2.
165   CALL COSTFUN(COST,TEE,QUE)$COSTPBU=COST/DEPTH*1.25
      AVEMCWB=XMFINAL/(1.+XMFINAL)$PRINT 10023,ELCOST,FULCOST,COST,COSTP
     1BU $ PRINT 10022,TEE,QUE,XMFINAL,AVEMCWB $ STOP
C     INLET TEMP IS FIXED,ONLY AIRFLOW CAN VARY
170   TEE=TINLET $ GUESS1=QMAX$GUESS2=QMIN$CALL ZEROINA(GUESS1,GUESS2,0.
     1000001,SIDE)$QUE=(GUESS1+GUESS2)/2.$GO TO 165
180   CALL OPTWHIZ
10001 FORMAT(A10,7F10.0)
10002 FORMAT(3X,*MOISTURE CONTENT IS NOT IN THE NORMAL RANGE*)
10003 FORMAT(3X,*AIRFLOW RATE INPUT IS NOT IN THE NORMAL RANGE*)
10004 FORMAT(3X,*WETBULB TEMPERATURE INPUT IS NOT IN NORMAL RANGE*)
10005 FORMAT(3X,*ABSOLUTE HUMIDITY IS NOT IN NORMAL RANGE*)
10006 FORMAT(3X,*RELATIVE HUMIDITY INPUT IS INCORRECT*)
10007 FORMAT(3X,*INLET AIR TEMP BOUNDS ARE INCORRECT*)
10008 FORMAT(3X,*AIRFLOW RATE BOUNDS ARE INCORRECT*)
10010 FORMAT(8F10.0)
10020 FORMAT(2A10)
10022 FORMAT(8F10.0)
10023 FORMAT(3X,*ELECT.COMPONENT =*,F7.2,*FUEL COMPONENT =*,F7.2,*TOTAL
     1COST,CTS/FT2 =*,F7.2,* CTS/BU COST =*,F7.2)
10024 FORMAT(3X,*INLET AIR TEMP,DEG F =*,F10.7,*AIR FLOW RATE,CFM/FT2 =
     1*,F10.7,*FINAL AVE MC,DB= *,F10.3,*FINAL AVE MC,WB = *,F10.7)
20001 FORMAT(1H0,3X,*THE OPTION CHOSEN IS*,2A10)
20002 FORMAT(1H0,3X,*INIT MC = *,F7.4,3X,*DESIRED FINAL MC = *,F7.4,3X,
     1*TOP LAYER MOISTURE BOUND = *,F7.4,3X,A10)
20003 FORMAT(1H0,3X,*ATM PRESSURE = *,F7.4,3X,A10)
20004 FORMAT(1H0,3X,*AIRFLOW INITIAL GUESS = *,F7.4,3X,*UPPER BOUND = *,
     1F7.4,3X,*LOWER BOUND = *F7.4,3X,A10)
20005 FORMAT(1H0,3X,*BED DEPTH,FT = *,F7.4,3X,*DRYING TIME,HR = *,F7.4,3
     1X,*AMBIENT TEMP,F = *,F7.4)
20006 FORMAT(1H0,3X,*HUMIDITY = *,F7.4,A10)
20007 FORMAT(1H0,3X,*EFFICIENCY OF FAN = *,F7.4,3X,*OF MOTOR = *,F7.4,3X
     1,*OF HEATER = *,F7.4)
20055 FORMAT(1H0,3X,*INLET TEMP INITIAL GUESS,F = *,F7.4,3X,*UPPER BOUND
     1 = *,F7.4,3X,*LOWER BOUND = *,F7.4)
20075 FORMAT(1H0,3X,*ELECTRIC PRICE,CTS/KWH = *,F7.4,3X,*FUEL PRICE,CTS
     1/GAL OR /LB = *,F7.4,3X,*FUEL HEAT VALUE,BTU/GAL OR /LB=*,F7.4,//)
      END
```

APPENDIX B

```
      PROGRAM COOLONE(INPUT,OUTPUT,CHAN,TAPE11=CHAN)
C PROGRAM COOLONE IS THE INITIAL PROGRAM TO BE RUN IN THE 3-PROGRAM SEQUENCE FOR
C OPTIMAL DESIGN OF A NON-AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM, ASSOCIATED SUBPROGRAMS REQUIRED-COOLSIM,CSTCOOL,EMC,PSDB,PVHA,RHPSPV,
C VSDBHA
      COMMON/FIXED/TAMB,HAMB,ATAMB,SPVLCON/INCOOL/XMOI,XTH,XGA,XDP/OUTCO
     1OL/HOUT,TOUT,THOUT,XMOUT/PRICE/ELPRICE,TIME,FANETA,EMETA/PRESS/PAT
     1M
      PRINT 55555
C READ MAX AND MIN ALLOWABLE MCS AT THE ENTRANCE TO THE COOLER AND THE NUMBER OF
C LEVELS USED, CALCULATE INCREMENT SIZE.
      READ 10000,XMMXIN,XMMNIN,LEVELM
      DELM=(XMMXIN-XMMNIN)/FLOAT(LEVELM-1)
      PRINT 40000,XMMXIN,XMMNIN,LEVELM,DELM
C READ MAX AND MIN ALLOWABLE PRODUCT TEMPERATURES AT THE ENTRANCE TO THE COOLER
C AND THE NUMBER OF LEVELS USED, CALCULATE INCREMENT SIZE.
      READ 10000,THMXIN,THMNIN,LEVELTH
      DELTH=(THMXIN-THMNIN)/FLOAT(LEVELTH-1)
      PRINT 40001 ,THMXIN,THMNIN,LEVELTH,DELTH
C READ MAX AND MIN ALLOWABLE AIRFLOW RATES THRU THE COOLER AND THE NUMBER OF
C LEVELS USED, CALCULATE INCREMENT SIZE.
      READ 10000,GMXOUT,GMNOUT,LEVELG
      DELG=(GMXOUT-GMNOUT)/FLOAT(LEVELG-1)
      PRINT 40003,GMXOUT,GMNOUT,LEVELG,DELG
C READ MAX AND MIN ALLOWABLE BED DEPTHS OF THE COOLER AND THE NUMBER OF LEVELS
C USED, CALCULATE INCREMENT SIZE.
      READ 10000,DPTHMAX,DPTHMIN,LEVELD
      DELD=(DPTHMAX-DPTHMIN)/FLOAT(LEVELD-1)
      PRINT 78787,DPTHMAX,DPTHMIN,LEVELD,DELD
      PRINT 40005
C READ COOLING AIR INLET TEMPERATURE AND HUMIDITY PLUS ATMOSPHERIC PRESSURE.
      READ20000,TAMB,HAMB,PATM
C READ MAX ALLOWABLE OUTLET PRODUCT TEMPERATURE AND MAX AND MIN ALLOWABLE OUTLET
C MCS,
      READ 20000,THOUTMX,XMOUTMX,XMOUTMN
      PRINT 70000,TAMB,HAMB,PATM
      PRINT 70001,THOUTMX,XMOUTMX,XMOUTMN
      ATAMB=TAMB+459.69
      SPVLCON=VSDBHA(ATAMB,HAMB)/60.
      RHIN=RHDBHA(ATAMB,HAMB)$EQM=EMC(RHIN,TAMB)
C READ ELECTRIC PRICE, TIME OVER WHICH OPTIMIZATION IS TO BE PERFORMED, FAN AND
C MOTOR EFFICIENCIES,
      READ 20000,ELPRICE,TIME,FANETA,EMETA
      PRINT 40006,ELPRICE,TIME,FANETA,EMETA
C BEGIN SEARCH FOR UPPER BOUND IN MC GRID DIMENSION.  FIRST CHECK FEASIBILITY OF
C MIN POSSIBLE OUTLET MC AGAINST CONSTRAINT,
      PRINT 40007
      XMOI=XMMNIN$XTH=THMXIN$XGA=GMXOUT$XDP=DPTHMAX
      CALL COOLSIM(3)
      PRINT 90005,XMOUT
      IF(XMOUT.GT.XMOUTMX)STOP
C CHECK FEASIBILITY OF MIN POSSIBLE OUTLET PRODUCT TEMPERATURE AGAINST CON-
C STRAINT,
      XTH=THMNIN
      CALL COOLSIM(1)
      PRINT 40008,THOUT
      IF(THOUT.GT.THOUTMX)STOP
C USING BISECTION, FIND UPPER MOISTURE CONTENT BOUND;
      MHIGH=LEVELM$MLOW=1
      MUPTRY=LEVELM
```

```
10      XMOI=XMMNIN+FLOAT(MUPTRY-1)*DELM$XTH=THMXIN
        CALL COOLSIM(3)
        IF(XMOUT.GT.XMOUTMX)35,15
15      XTH=THMNIN
        CALL COOLSIM(1)
        IF(THOUT.GT.THOUTMX)35,20
20      MLOW=MUPTRY
        IF((MHIGH-MLOW).LE.1)25,40
25      MUP=MLOW
        IF(MUP.EQ.1)30,45
30      PRINT 90000
        STOP
35      MHIGH=MUPTRY
40      MUPTRY=(MHIGH-MLOW)/2+MLOW
        GOTO10
45      PRINT 40009,MUP
C BEGIN SEARCH BY BISECTION FOR LOWER BOUND IN AIRFLOW RATE GRID DIMENSION.
C USING MAX OUTLET PRODUCT TEMPERATURE CONSTRAINT.
46      PRINT 40010
        KGHIGH=LEVELG$KGLOW=1$LOWGTRY=1
        XGA=GMXOUT$CALL COOLSIM(1)
        PRINT 40008,THOUT
        IF(THOUT.GT.THOUTMX)STOP
50      XGA=GMNOUT+FLOAT(LOWGTRY-1)*DELG$CALL COOLSIM(1)
        IF(THOUT.GT.THOUTMX)55,60
55      KGLOW=LOWGTRY
        IF ((KGHIGH-KGLOW).EQ.1)62,57
57      LOWGTRY=(KGHIGH-KGLOW)/2+KGLOW
        GOTO 50
60      KGHIGH=LOWGTRY
        IF((KGHIGH-KGLOW).GT.1)GOTO57
62      LOWG=KGHIGH
        IF(LOWG.EQ.LEVELG)65,70
65      PRINT 90001
        LOWG=LEVELG
70      PRINT 40011,LEVELG,LOWG
C BEGIN SEARCH BY BISECTION FOR UPPER BOUND IN PRODUCT TEMP GRID DIMENSION.
C USING MAX OUTLET PRODUCT TEMPERATURE CONSTRAINT.
        PRINT 40012
        XMOI=XMMNIN$XDP=DPTHMAX$XGA=GMXOUT
        XTH=THMNIN$CALL COOLSIM(1)
        PRINT 40008,THOUT
        IF(THOUT.GT.THOUTMX)STOP
        KTHHIGH=LEVELTH$KTHLOW=1$IUPTHTR=LEVELTH
 75     XTH=THMNIN+FLOAT(IUPTHTR-1)*DELTH$CALL COOLSIM(1)
        IF(THOUT.GT.THOUTMX)95,80
80      KTHLOW=IUPTHTR
        IF((KTHHIGH-KTHLOW).LE.1)85,100
85      IUPTH=KTHLOW
        IF(IUPTH.EQ.1)90,105
90      PRINT 90002
        STOP
95      KTHHIGH=IUPTHTR
100     IUPTHTR=(KTHHIGH-KTHLOW)/2+KTHLOW
        GOTO 75
105     PRINT 40009,IUPTH
C BEGIN SEARCH BY BISECTION FOR THE LOWER BOUND OF BED DEPTH.
        PRINT 40013
        XMOI=XMMNIN$XGA=GMXOUT$XTH=THMNIN
        KDHIGH=LEVELD$KDLOW=1$LOWDTRY=1
```

```
      XDP=DPTHMAX$CALL COOLSIM(1)
      PRINT 40008,THOUT
      IF(THOUT.GT.THOUTMX)STOP
110   XDP=DPTHMIN+FLOAT(LOWDTRY-1)*DELD$CALL COOLSIM(1)
      IF(THOUT.GT.THOUTMX)115,120
115   KDLOW=LOWDTRY
      IF((KDHIGH-KDLOW).EQ.1)122,117
117   LOWDTRY=(KDHIGH-KDLOW)/2+KDLOW
      GOTO110
120   KDHIGH=LOWDTRY
      IF((KDHIGH-KDLOW).GT.1)GOTO117
122   LOWD=KDHIGH
      IF(LOWD.EQ.LEVELD)125,130
125   PRINT 90099$STOP
130   PRINT 40011,LEVELD,LOWD
C FIND OPTIMAL FEASIBLE DEPTH AND AIRFLOW RATE CONTROLS, FILLING THE PRODUCT MC-
C TEMPERATURE GRID WITH CORRESPONDING COSTS.
      PRINT 40016
      DO 230 III=1,MUP
      XMOI=FLOAT(III-1)*DELM+XMMNIN
      DO 220 JJJ=1,IUPTH
      XTH=FLOAT(JJJ-1)*DELTH+THMNIN$BESTCST=10.E25$BESTGA=0.0$BESTDP=0.0
      DO 210 KKK=LOWG,LEVELG
      KKK1=LOWG-KKK+LEVELG
      XGA=FLOAT(KKK1-1)*DELG+GMNOUT
      DO 200 LLL=LOWD,LEVELD
      LLL1=LOWD-LLL+LEVELD
      XDP=FLOAT(LLL1-1)*DELD+DPTHMIN
      CALL COOLSIM(4)
      IF(THOUT.GT.THOUTMX.OR.XMOUT.GT.XMOUTMX)GOTO210
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO200
      IF(((XMOUT-EQM)/(XMOI-EQM)).GT.1.)GOTO200
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XMOUT)GOTO200
      CALL CSTCOOL(COST)
      IF(COST.LT.BESTCST)190,200
190   BESTCST=COST$BESTGA=XGA$BESTDP=XDP
200   CONTINUE
210   CONTINUE
      IF(BESTCST.LT.10.E23)211,220
211   WRITE(11,33333)III,JJJ,BESTCST,BESTGA,BESTDP
      PRINT 33333,III,JJJ,BESTCST,BESTGA,BESTDP
  220 CONTINUE
  230 CONTINUE
      ENDFILE 11
10000 FORMAT(2F10.0,I10)
20000 FORMAT(8F10.0)
33333 FORMAT(2I5,3E22.15)
40000 FORMAT(1X*MAX MC=*F5.3*  MIN MC=*F5.3*  NUMBER OF LEVELS=*I3*  MC
     1INCREMENT=*F6.4)
40001 FORMAT(1H0*PROD TEMP-MAX=*F5.0*  MIN=*F5.0*  NUMBER OF LEVELS=*I3*
     1  INCREMENT SIZE=*F5.1)
40003 FORMAT(1H0*AIRFLOW RATE-MAX=*F5.0*  MIN=*F5.0*  NUMBER OF LEVELS=*
     1I3*  INCREMENT SIZE=*F5.1,///)
40005 FORMAT(1X*ADDITIONAL INFORMATION REQD FOR SOLUTION*///)
40006 FORMAT(1H0*ELECTRICAL PRICE=*F4.2*  TIME SCALE=*F4.2*  FAN EFF=*F4
     1.3*  MOTOR EFF=*F4.3,///)
40007 FORMAT(1X*SEARCH FOR FEASIBLE UPPER BOUND IN MC DIMENSION*)
```

```
40008 FORMAT(1H0*FOR THESE INPUTS, THE MINIMUM OUTLET PRODUCT TEMPERATUR
     1E IS *F7.2)
40009 FORMAT(1H0*ADJUSTED UPPER NODE IS *I5*,LOWER NODE IS 1*///)
40010 FORMAT(1X*SEARCH FOR FEASIBLE LOWER BOUND IN AIRFLOW DIMENSION*)
40011 FORMAT(1H0*UPPER NODE IS*I5*,ADJUSTED LOWER NODE IS*I5,///)
40012 FORMAT(1X*SEARCH FOR FEASIBLE UPPER BOUND IN THETA DIMENSION*)
40013 FORMAT(1X*SEARCH FOR FEASIBLE LOWER BOUND OF DEPTH*)
40016 FORMAT(1X*BEGIN ITERATION, PRINTING INDICES OF MC AND PRODUCT TEMP
     1 AND ASSOCIATED COST,AIRFLOW,AND DEPTH*)
55555 FORMAT(1H1*SET UP GRID SYSTEM AT THE COOLER-DRYER INTERFACE*///)
70000 FORMAT(1X*INLET CONDITIONS-TEMP=*F7.3*  ABS HUM=*F8.5*  ATM PRESS=
     1*F5.2)
70001 FORMAT(1H0*DESIRED OUTLET CONDITIONS*MAX GRAIN TEMP=*F7.3*  MAX MC=
     1=*F7.4*  MIN MC=*F7.4)
78787 FORMAT(1H0*DEPTH-MAX=*F4.2*  MIN=*F4.2*  NUMBER OF LEVELS=*I3*  IN
     1CREMENT SIZE=*F5.3,///)
90000 FORMAT(3X,*MOISTURE DIMENSION CONTAINS ONLY ONE FEASIBLE LEVEL*)
90001 FORMAT(3X*AIRFLOW DIMENSION CONTAINS ONLY ONE FEASIBLE LEVEL*//)
90002 FORMAT(3X,*THETA DIMENSION CONTIANS ONLY ONE FEASIBLE LEVEL*)
90005 FORMAT(1H0*FOR THESE INPUTS, THE MINIMUM OUTLET MC IS *F5.4)
90099 FORMAT(3X*DEPTH CONTROL CONTAINS ONLY ONE FEASIBLE LEVEL*)
      END
```

```
      PROGRAM DRYONE(INPUT,OUTPUT,TAPE37)
C PROGRAM DRYONE IS THE 2ND PROGRAM TO BE RUN IN THE 3-PROGRAM SEQUENCE FOR OPT-
C IMAL DESIGN OF A NON-AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM.  ASSOCIATED SUBPROGRAMS REQUIRED--COOLERC,COOLERD,COOLERG,COSTONE,
C DRYSIM,EMC,INTERP,PSDB,PVHA,PHPSPV,VSDBHA,
      DIMENSION INDEX(2),ICECREM(2)
      COMMON/CON/CON1,CON2/VALUES/XMCIN,THIN,GP,SA,CA,CP,CV/DOLLAR/TIME,
     1ELPRICE,FUELPHI,FUELHT,EMETA,FANETA,THERETA/RECALL/OPTCOST(  ,  ),
     1GCOOL(  ,  ),DCOOL(  ,  ),XNEW(2),DXINV(2),XMLOW,DELM,THLOW,DELTH/
     2PRESS/PATM/IN/SUMTEMP,HAMB,GIN,DEPTH/OUT/XM,THETA,TOUT,HOUT/SIMCOS
     3T/G,TAMB,TIN
      EXTERNAL COOLERC,COOLERD,COOLERG
      DATA PATM,RHOP,BPH,SA,CA,CP,CV/14.34,38.71,9.,239.,.242,.268,.45/
      NDIM=2
      BESTCST=10.E22
      GP=BPH*1.244*RHOP
      CON1=SA/(GP*CP)$CON2=SA/CA
C READ MIN FEASIBLE MC VALUE FROM COOLONE OUTPUT, INCREMENT SIZE, MIN VALVE OF
C MC USED IN COOLONE PROGRAM, AND NUMBER OF FEASIBLE LEVELS REMAINING IN COOLONE
C OUTPUT.
      READ 10001,XMLOW,DELM,XMMNIN,LEVELM
      PRINT 20000,XMLOW,DELM,XMMNIN,LEVELM
      DXINV(1)=1./DELM
      XMHIGH=XMLOW+FLOAT(LEVELM-1)*DELM
      MINDXM=IFIX((XMLOW+DELM/2.-XMMNIN)/DELM)
C READ MIN FEASIBLE PRODUCT TEMP VALUE FROM COOLONE OUTPUT, INCREMENT SIZE, MIN
C VALUE OF PRODUCT TEMP USED IN COOLONE PROGRAM, AND NUMBER OF FEASIBLE LEVELS
C REMAINING IN COOLONE OUTPUT.
      READ 10001,THLOW,DELTH,THMNIN,LEVELTH
      PRINT 20001,THLOW,DELTH,THMNIN,LEVELTH
      DXINV(2)=1./DELTH
      THHIGH=THLOW+FLOAT(LEVELTH-1)*DELTH
      MINDXTH=IFIX((THLOW+DELTH/2.-THMNIN)/DELTH)
```

```
C READ LOWEST AIRFLOW RATE,INCREMENT SIZE,AND NUMBER OF LEVELS TO BE USED:
      READ 10003,GLOW,DELG,LEVELG
      PRINT 20002,GLOW,DELG,LEVELG
C READ LOWEST INLET AIR TEMP, INCREMENT SIZE, AND NUMBER OF LEVELS TO BE USED:
      READ 10003,TLOW,DELT,LEVELT
      PRINT 20003,TLOW,DELT,LEVELT
C READ SHORTEST BED DEPTH,INCREMENT SIZE, AND NUMBER OF LEVELS TO BE USED:
      READ 10003,DLOW,DELD,LEVELD
      PRINT 20004,DLOW,DELD,LEVELD
C READ INLET AIR TEMPERATURE AND HUMIDITY TO HEATER PLUS ATMOSPHERIC PRESSURE
C AND INLET GRAIN MC AND TEMPERATURE TO THE DRYER;
      READ 10002,TAMB,HAMB,PATM,XMCIN,THIN
      PRINT 20005,TAMB,HAMB,PATM,XMCIN,THIN
C READ TIME BASE FOR OPTIMIZATION, ELECTRIC PRICE, FAN EFFICIENCY,MOTOR EFFIC-
C IENCY, FUEL HEAT VALUE, FUEL PRICE, AND THERMAL EFFICIENCY.
      READ 10002,TIME,ELPRICE,FANETA,EMETA,FUELHT,FUELPRI,THERETA
      PRINT 20006,TIME,ELPRICE,FANETA,EMETA
      PRINT 20007,FUELHT,FUELPRI,THERETA
C READ MAXIMUM AIR TEMP AND PRESSURE DROP CONSTRAINTS;
      READ 10002,TMAX,PDROPMX
      PRINT 20008,TMAX,PDROPMX
      ATAMB=TAMB+459.69
      SPVLCON=VSDBHA(ATAMB,HAMB)/60.
      RHIN=RHDBHA(ATAMB,HAMB)
      EQM=EMC(RHIN,TAMB)
C INITIALIZE THE COST INTERPOLATION GRID TO ABSURD VALUES.
      DO 9 I=1,LEVELM
      DO 7 J=1,LEVELTH
      OPTCOST(I,J)=10.E25
7     CONTINUE
9     CONTINUE
C READ IN VALUES OF COST AND CORRESPONDING AIRFLOW AND COOLER DEPTHS.  BY
C COMPARISON, FILL THE GRID WITH OPTIMAL VALUES.
11    READ (37,30000)IM,ITH,EXPENSE,GACOOL,DPCOOL
      IF(EOF(37))20,15
15    IM=IM-MINDXM
      ITH=ITH-MINDXTH
      OPTCOST(IM,ITH)=EXPENSE
      GCOOL(IM,ITH)=GACOOL
      DCOOL(IM,ITH)=DPCOOL
      GOTO11
C APPLY IN TURN ALL DRYER AIRFLOW RATE ,AIR TEMPERATURE, AND BED DEPTH CONTROLS
C TO EACH NODAL POINT OF THE GRID.
20    DO 180 IG=1,LEVELG
      IGG=LEVELG-IG+1
      GIN=GLOW+FLOAT(IGG-1)*DELG
      DO 170 IT=1,LEVELT
      ITT=LEVELT-IT+1
      TIN=TLOW+FLOAT(ITT-1)*DELT
      DO 160 ID=1,LEVELD
      IDD=LEVELD-ID+1
      DEPTH=DLOW+FLOAT(IDD-1)*DELD
C CHECK MAXIMUM AIR TEMPERATURE AND PRESSURE DROP CONSTRAINTS;
      Q=GIN*SPVLCON
      PDROP=DEPTH*(Q/58.)**1.528
      IF(PDROP.GT.PDROPMX)GOTO160
      DELTADD=.37854869*PDROP/FANETA/EMETA
      SUMTEMP=TIN+DELTADD
      IF(SUMTEMP.GT.TMAX)GOTO160
C CALL DRYER SIMULATION TO PREDICT OUTLET GRAIN MC AND TEMPERATURE AND CHECK IF
```

```
C RESULT FALLS WITHIN GRID.
      CALL DRYSIM(1)
      IF(XM.GT.XMHIGH)GOTO170
      IF(XM.LT.XMLOW)GOTO160
      IF(THETA.LT.THLOW)GOTO160
      IF(THETA.GT.THHIGH)GOTO170
C APPLY SATURATION, ABSORPTION , AND EQUILIBRIUM CHECKS TO OUTLET AIR AND
C HUMIDITY PREDICTIONS OF DRYER MODEL.
      CALL DRYSIM(2)
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO160
      IF(((XM-EQM)/(XMCIN-EQM)).GT.1.)GOTO160
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XM)GOTO160
C SET UP INTERPOLATION PROCEDURE AND INTERPOLATE.
      XNEW(1)=XM$XNEW(2)=THETA
      NSTORE=0
      DO 85 I=1,NDIM
85    ICECREM(I)=0
      INDEX(1)=IFIX((XM-XMLOW)/DELM)+1
      INDEX(2)=IFIX((THETA-THLOW)/DELTH)+1
      XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      IF(ABS(XVAL-XM).LT.10.E-8)90,100
90    NSTORE=NSTORE+1$ICECREM(NSTORE)=1
100   XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      IF(ABS(XVAL-THETA).LT.10.E-8)110,120
110   NSTORE=NSTORE+1$ICECREM(NSTORE)=2
120   IF(NSTORE.EQ.NDIM)130,140
130   PARCOST=OPTCOST(INDEX(1),INDEX(2))
      GOTO 145
140   CALL INTERP(PARCOST,NDIM,INDEX,ICECREM,NSTORE,COOLERC)
145   IF(PARCOST.LT.1.E-8.OR.PARCOST.GT.10.E22)GOTO160
C EVALUATE COST OF DRYING AND ADD TO INTERPOLATED COST.  COMPARE THE RESULT TO
C THE CURRENT OPTIMAL COST.
      CALL COSTONE(COST,PDROP)
      TRYCOST=PARCOST+COST
      IF(TRYCOST.LT.BESTCST )150,160
C IF CURRENT TOTAL COST IS OPTIMAL, REPLACE THE PREVIOUS BEST COST AND INTER-
C POLATE TO FIND OPTIMAL COOLER DEPTH AND AIRFLOW RATE.  PRINT ALL OPTIMAL
C RESULTS.
150   BESTCST=TRYCOST$BESTDPH=DEPTH$BESTG=GIN$BESTT=TIN$BESTM=XM$BESTTH=
     1THETA
      PRINT 20013,BESTCST,BESTM,BESTTH
      PRINT 20014,BESTDPH,BESTG,BESTT
      CALL INTERP(OPTDCL ,NDIM,INDEX,ICECREM,NSTORE,COOLERD)
      CALL INTERP(OPTGCL ,NDIM,INDEX,ICECREM,NSTORE,COOLERG)
      PRINT 20015,OPTDCL,OPTGCL
160   CONTINUE
170   CONTINUE
180   CONTINUE
10001 FORMAT(3F10.0,I10)
10002 FORMAT(8F10.0)
10003 FORMAT(2F10.0,I10)
20000 FORMAT(1H1*  MIN FEASIBLE MC=*F5.3*  MC INCREMENT=*F5.3*MIN MC IN
     1COOLONE GRID=*F5.3*  REMAINING FEASIBLE LEVELS=*I4,//)
20001 FORMAT(1H0*  MIN FEASIBLE PRODUCT TEMP=*F6.2*  PRODUCT TEMP INCREM
     1ENT=*F6.2*  MIN PRODUCT TEMP IN COOLONE GRID=*F6.2*  REMAINING FEA
     2SIBLE LVL=*I3,//)
20002 FORMAT(1H0*LOWER AIRFLOW RATE BOUND=*F6.2*  INCREMENT=*F6.2*  NUMB
```

```
     1ER OF LEVELS=*I4,//)
20003 FORMAT(1H0*LOWER AIR TEMP BOUND=*F6.2*  INCREMENT=*F6.2*  NUMBER O
     1F LEVELS=*I4,//)
20004 FORMAT(1H0*LOWER BED DEPTH BOUND=*F4.2*  INCREMENT=*F4.2*  NUMBER
     1OF LEVELS=*I4,//)
20005 FORMAT(1H0*HEATER INLET AIR TEMP=*F5.2*  INLET HUMIDITY=*F6.4*  AT
     1M PRESSURE=*F6.2*  GRAIN INLET MC=*F5.3*  INLET GRAIN TEMP=*F6.2,/
     1/)
20006 FORMAT(1H0*TIME BASE FOR OPTIMIZATION=*F4.2*  ELECTRIC PRICE=*F4.2
     1*  FAN EFFICIENCY=*F5.3*  MOTOR EFFICIENCY=*F5.3,//)
20007 FORMAT(1H0*FUEL HEAT=*F7.0*  FUEL PRICE=*F5.2*  THERMAL EFFICIENCY
     1=*F5.3,//)
20008 FORMAT(1H0*MAX AIR TEMP CONSTRAINT=*F7.2*  MAX PRESSURE DROP CONST
     1RAINT=*F6.2,//)
20013 FORMAT(1H0*CURRENT OPTIMAL COST=*F10.5*  OUTLET MC=*F5.4*  OUTLET
     1 GRAIN TEMP=*F6.2)
20014 FORMAT(1H0*CURRENT OPTIMAL DRYER DEPTH=*F4.2*  AIRFLOW RATE=*F6.2*
     1  AIR TEMP=*F6.2)
20015 FORMAT(1H0*OPTIMAL COOLER DEPTH=*F6.3*  AIRFLOW RATE=*F6.2,///)
30000 FORMAT(2I5,3F22.15)
      END
```

```
      PROGRAM RECONE(INPUT,OUTPUT)
C PROGRAM RECONE IS THE FINAL PROGRAM TO BE RUN IN THE 3-PROGRAM SEQUENCE FOR
C OPTIMAL DESIGN OF A NON-AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM.  ASSOCIATED SUBPROGRAMS REQUIRED-COOLSIM,DRYSIM,HOWDEEP,VSDBHA.
      COMMON/PRESS/PATM/IN/THEAT,BESTH,Q,BESTDPH/OUT/Z,YZ,TOUTDRY,HOUTDR
     1Y/INCOOL/BESTM,BESTTH,GINHEAT,XDP/OUTCOOL/HOUT,TOUT,THOUT,XMOUT/FIX
     2XED/TAMB,HAMB,ATAMB,SPVLCON/AITCH/HINCOOL
      COMMON/VALUES/XMCIN,THIN,GP,SA,CA,CP,CV
      EXTERNAL HOWDEEP
      DATA EPS/.01/
C READ, FROM THE DRYONE OUTPUT, OPTIMAL DRYER AIRFLOW RATE, INLET AIR TEMP,
C INLET AIR HUMIDITY, DEPTH, PLUS RESULTING OUTLET MC AND PRODUCT TEMP.
      READ 10000,BESTG,BESTT,BESTH,BESTDPH,BESTM,BESTTH
C READ INLET AIR TEMP TO HEATER, ATMOSPHERIC PRESSURE, FAN AND MOTOR EFFIC-
C IENCIES, AND DRYER INLET MC AND PRODUCT TEMP.
      READ 10000,TINHEAT,PATM,FANETA,EMETA,XMCIN,THIN
C READ INLET AIR TEMP AND HUMIDITY TO COOLER,OPTIMAL COOLER DEPTH AND COOLER
C AIRFLOW RATE.
      READ 10000,TAMB,HAMB,XDP,GINHEAT
C CALCULATE ACTUAL AIR TEMP ENTERING DRYER.
      ATAMB=TAMB+459.69
      Q=BESTG*VSDBHA(ATAMB,HAMB)/60.
      PDROP=BESTDPH*(Q/58.)**1.528
      THEAT=BESTT+.37854889*PDROP/FANETA/EMETA
C SIMULATE DRYER TO PREDICT OUTLET AIR TEMP AND HUMIDITY.
      CALL DRYSIM(2)
      PRINT 20000,TOUTDRY,HOUTDRY,XMCIN,THIN,BESTDPH
      SPVLCON=VSDBHA(ATAMB,HAMB)/60.
      DEPTH=XDP$CALL COOLSIM(4)
C SIMULATE COOLER TO PREDICT COOLER OUTLET AIR TMEP AND HUMIDITY, PRODUCT TEMP
C AND MC.
      CALL COOLSIM(4)
      PRINT 50000,TOUT,HOUT,GINHEAT
      PRINT 55000,BESTM,BESTTH
```

```
      PRINT 60000,TAMB,HAMB,DEPTH,XMOUT,THOUT
10000 FORMAT(8F10.0)
20000 FORMAT(1H1*DRYER SPECS-OUTLET AIR TEMP=*F6.2*  HUMIDITY=*F6.4*  IN
     1LET GRAIN MC=*F5.3*  PROD TEMP=*F5.1*  DRYER DEPTH=*F4.2,//)
 5000 FORMAT(1X*COOLER EXIT AIR CONDITIONS-AIR TEMP=*F6.2*  HUMIDITY=*F6
     1.4*  AIRFLOW RATE=*F6.2//)
55000 FORMAT(1X*DRYER EXIT GRAIN CONDITIONS-MC=*F4.3*  PROD TEMP=*F6.2,/
     1/)
60000 FORMAT(1X*COOLER SPECS-INLET AIR TEMP=*F6.2*  HUMIDITY=*F6.5*  DEP
     1TH=*F6.2*  OUTLET MC=*F6.4*  OUTLET PROD TEMP=*F6.2)
      END
```

```
      PROGRAM COOLER(INPUT,OUTPUT,GIN,TAPE11=GIN)
C PROGRAM COOLER IS THE INITIAL PROGRAM TO BE RUN IN THE 4-PROGRAM SEQUENCE FOR
C OPTIMAL DESIGN OF AN AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM.  ASSOCIATED SUBPROGRAMS REQUIRED--COOLSIM,CSTCOOL,DEEPMNM,DEEPMNT,EMC,
C HOWDEEP,VSDBHA,RHPSPV,ZEROIN.
      COMMON/FIXED/TAMB,HAMB,ATAMB,SPVLCON/INCOOL/XMOI,XTH,XGA,XDP/OUTCO
     1OL/HOUT,TOUT,THOUT,XMOUT/PRICE/ELPRICE,TIME,FANETA,EMETA/AITCH/HNO
     2W/PRESS/PATM/IPIAL/THOUTMX,XMOUTMX
      EXTERNAL HOWDEEP,DEEPMNT,DEEPMNM
      DATA EPS/.01/
      PRINT 40004
C READ MAXIMUM AND MINIMUM ALLOWABLE MCS AT THE ENTRANCE TO THE COOLER AND THE
C NUMBER OF LEVELS USED.  CALCULATE INCREMENT SIZE.
      READ 10000,XMMXIN,XMMNIN,LEVELM
      DELM=(XMMXIN-XMMNIN)/FLOAT(LEVELM-1)
      PRINT 40000,XMMXIN,XMMNIN,LEVELM,DELM
C READ MAXIMUM AND MINIMUM ALLOWABLE PRODUCT TEMPERATURES AT THE ENTRANCE TO THE
C COOLER AND THE NUMBER OF LEVELS USED.  CALCULATE INCREMENT SIZE.
      READ 10000,THMXIN,THMNIN,LEVELTH
      DELTH=(THMXIN-THMNIN)/FLOAT(LEVELTH-1)
      PRINT 40001 ,THMXIN,THMNIN,LEVELTH,DELTH
C READ MAXIMUM AND MINIMUM ALLOWABLE AIRFLOW RATES THRU THE COOLER AND THE
C NUMBER OF LEVELS USED.  CALCULATE INCREMENT SIZE.
      READ 10000,GMXOUT,GMNOUT,LEVELG
      DELG=(GMXOUT-GMNOUT)/FLOAT(LEVELG-1)
      PRINT 40003,GMXOUT,GMNOUT,LEVELG,DELG
      PRINT 40005
C READ MAXIMUM AND MINIMUM COOLER DEPTHS USED. THE INCREMENT SIZE IN HUMIDITY,
C COOLING AIR INLET TEMPERATURE AND HUMIDITY, ATMOSPHERIC PRESSURE, MAXIMUM
C ALLOWABLE OUTLET PRODUCT TEMPERATURE, AND MAXIMUM AND MINIMUM ALLOWABLE
C OUTLET MOISTURE CONTENTS.
      READ 20000,DPTHMAX,DPTHMIN$READ 20000,DELH
      READ20000,TAMB,HAMB,PATM
      READ 20000,THOUTMX,XMOUTMX,XMOUTMN
      PRINT 70000,TAMB,HAMB,PATM
      PRINT 70001,THOUTMX,XMOUTMX,XMOUTMN
      PRINT 70002,DPTHMAX,DPTHMIN,DELH
      ATAMB=TAMB+459.69
      SPVLCON=VSDBHA(ATAMB,HAMB)/60.
      RHIN=RHDBHA(ATAMB,HAMB)$EQM=EMC(RHIN,TAMB)
C READ ELECTRIC PRICE, TIME OVER WHICH OPTIMIZATION IS TO BE PERFORMED, FAN AND
C MOTOR EFFICIENCIES.
      READ 20000,ELPRICE,TIME,FANETA,EMETA
      PRINT 40006,ELPRICE,TIME,FANETA,EMETA
C BEGIN SEARCH FOR UPPER BOUND IN MC GRID DIMENSION.  FIRST CHECK FEASIBILITY OF
C MIN POSSIBLE OUTLET MC AGAINST CONSTRAINT.
      PRINT 40007
      XMOI=XMMNIN$XTH=THMXIN$XGA=GMXOUT$XDP=DPTHMAX
      CALL COOLSIM(3)
      PRINT 90005,XMOUT
      IF(XMOUT.GT.XMOUTMX)GOTO250
C CHECK FEASIBILITY OF MIN POSSIBLE OUTLET PRODUCT TEMPERATURE AGAINST CON-
C STRAINT.
      XTH=THMNIN
      CALL COOLSIM(1)
      PRINT 40008,THOUT
      IF(THOUT.GT.THOUTMX)GOTO250
C USING BISECTION, FIND UPPER MOISTURE CONTENT BOUND.
      MHIGH=LEVELM$MLOW=1
      MUPTRY=LEVELM
```

```
10     XMOI=XMHNIN+FLOAT(MUPTRY-1)*DELM$XTH=THMXIN
       CALL COOLSIM(3)
       IF(XMOUT.GT.XMOUTMX)55,15
15     XTH=THMNIN
       CALL COOLSIM(1)
       IF(THOUT.GT.THOUTMX)55,20
20     MLOW=MUPTRY
       IF((MHIGH-MLOW).LE.1)25,40
25     MUP=MLOW
       IF(MUP.EQ.1)30,45
30     PRINT 90000
       STOP
35     MHIGH=MUPTRY
40     MUPTRY=(MHIGH-MLOW)/2+MLOW
       GOTO10
45     PRINT 40009,MUP
C BEGIN SEARCH BY BISECTION FOR LOWER BOUND IN AIRFLOW RATE GRID DIMENSION,
C USING MAX OUTLET PRODUCT TEMPERATURE CONSTRAINT,
46     PRINT 40010
       KCHIGH=LEVELG$KGLOW=1$LOWGTRY=1
       XGA=GMXOUT
       CALL COOLSIM (1)
       PRINT 40008,THOUT
       IF(THOUT.GT.THOUTMX)STOP
50     XGA=GMNOUT+FLOAT(LOWGTRY-1)*DELG
       CALL COOLSIM(1)
       IF(THOUT.GT.THOUTMX)55,60
55     KGLOW=LOWGTRY
       IF ((KGHIGH-KGLOW).EQ.1)62,57
57     LOWGTRY=(KGHIGH-KGLOW)/2+KGLOW
       GOTO 50
60     KGHIGH=LOWGTRY
       IF((KGHIGH-KGLOW).GT.1)GOTO57
62     LOWG=KGHIGH
       IF(LOWG.EQ.LEVELG)65,70
65     PRINT 90001$STOP
70     PRINT 40011,LEVELG,LOWG
C BEGIN SEARCH BY BISECTION FOR UPPER BOUND IN PRODUCT TEMP GRID DIMENSION,
C USING MAX OUTLET PRODUCT TEMPERATURE CONSTRAINT,
       PRINT 40012
       XMOI=XMMNIN$XDP=DPTHMAX$XGA=GMXOUT
       XTH=THMNIN
       CALL COOLSIM(1)
       PRINT 40008,THOUT
       IF(THOUT.GT.THOUTMX)71,72
71     PRINT 40020$STOP
72     KTHHIGH=LEVELTH$KTHLOW=1$IUPTHTR=LEVELTH
75     XTH=THMNIN+FLOAT(IUPTHTR-1)*DELTH
       CALL COOLSIM(1)
       IF(THOUT.GT.THOUTMX)95,80
80     KTHLOW=IUPTHTR
       IF((KTHHIGH-KTHLOW).LE.1)85,100
85     IUPTH=KTHLOW
       IF(IUPTH.EQ.1)90,105
90     PRINT 90002
       STOP
95     KTHHIGH=IUPTHTR
100    IUPTHTR=(KTHHIGH-KTHLOW)/2+KTHLOW
       GOTO 75
105    PRINT 40009,IUPTH
```

```
C BEGIN SEARCH FOR LOWER BOUND OF BED DEPTH.  FIRST CHECK FEASIBILITY OF MIN
C POSSIBLE OUTLET PRODUCT TEMPERATURE AGAINST CONSTRAINT.
      PRINT 40013
      XMOI=XMMNIN$XGA=GMXOUT$XTH=THMNIN
      XDP=DPTHMAX
      CALL COOLSIM(1)
      PRINT 40006,THOUT
      IF(THOUT.GT.THOUTMX)106,107
106   PRINT 40020
      STOP
C CHECK FEASIBILITY OF MIN POSSIBLE OUTLET MC AGAINST CONSTRAINT.
107   CALL COOLSIM(3)
      PRINT 90005,XMOUT
      IF(XMOUT.GT.XMOUTMX)108,109
108   PRINT 40020$STOP
109   KOUNT=0
      XDP=DPTHMIN
      CALL COOLSIM(1)
      IF(THOUT.GT.THOUTMX)115,120
C USING 1-D SEARCH, FIND DEPTH AT WHICH PRODUCT TEMPERATURE CONSTRAINT IS
C SATISFIED.
115   GUESS1=DPTHMIN$GUESS2=DPTHMAX
      CALL ZEROIN(GUESS1,GUESS2,EPS,DEEPMNT)
      DPTHMIN=(GUESS1+GUESS2)/2.
      KOUNT=1
C USING 1-D SEARCH, FIND DEPTH AT WHICH MC CONSTRAINT IS SATISFIED.
120   XDP=DPTHMIN$CALL COOLSIM(3)
      IF(XMOUT.LT.XMOUTMX.AND.KOUNT.EQ.0)150,130
130   GUESS1=DPTHMIN$GUESS2=DPTHMAX
      CALL ZEROIN(GUESS1,GUESS2,EPS,DEEPMNM)
      IF(KOUNT.EQ.0)135,140
135   DPTHMIN=(GUESS1+GUESS2)/2.
      GOTO150
C COMPARE DEPTHS AND CHOOSE THE MAXIMUM BECAUSE IT SATISFIES BOTH CONSTRAINTS.
140   DPTHMIN=AMAX1(DPTHMIN,((GUESS1+GUESS2)/2.))
150   PRINT 10014,DPTHMIN
      PRINT 40014
C DEFINE THE LIMITS BETWEEN WHICH HUMIDITY CAN VARY. THIS IS HELPFUL IN CHOOSING
C THE INCREMENT SIZE IN HUMIDITY.
      XMOI=XMMNIN$XTH=THMNIN$XGA=GMXOUT$XDP=DPTHMIN$CALL COOLSIM(2)
      HLOW=HOUT
      XMOI=XMMNIN+FLOAT(MUP-1)*DELM
      XTH=THMNIN+FLOAT(IUPTH-1)*DELTH
      XGA=FLOAT(LOWG-1)*DELG+GMNOUT
      XDP=DPTHMAX$CALL COOLSIM(2)
      HHI=HOUT
      PRINT 40015,HLOW,HHI
C THE USE OF A STOP CARD AT THIS POINT IS RECOMMENDED FOR INITIAL GRID SIZING.
C THE DO LOOP ITERATIONS TO FILL THE GRID WITH COSTS AND CORRESPONDING CONTROLS
C BEGIN.
      PRINT 40016
      DO 240 III=1,MUP
      PRODM=FLOAT(III-1)*DELM+XMMNIN
      DO 230 JJJ=1,IUPTH
      THETA=FLOAT(JJJ-1)*DELTH+THMNIN
      DO 220 KKK=LOWG,LEVELG
      KKK1=LOWG-KKK+LEVELG
      GA=FLOAT(KKK1-1)*DELG+GMNOUT
      LSEARCH=0$ISEARCH=0
      XMOI=PRODM$XTH=THETA$XGA=GA$XDP=DPTHMIN$CALL COOLSIM(4)
```

```
      IHSTART=IFIX((HOUT-HAMB)/DELH)+2
C BEGIN CHECKS ON MODEL PREDICTIONS FOR HUMIDITY AT MINIMUM BED DEPTH.
C CHECK SATURATION OF AIR.
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO230
C CHECK MOISTURE RATIO FOR ABSORPTION.
      IF(((XMOUT-EQM)/(PRODM-EQM)).GT.1.)GOTO230
C CHECK FOR EQUILIBRUIM.
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XMOUT)GOTO 230
C CK OUTLET PRODUCT TEMPERATURE AND MC CONSTRAINTS.
      IF(XMOUT.LT.XMOUTMN)GOTO230
      IF(THOUT.GT.THOUTMX.OR.XMOUT.GT.XMOUTMX)LSEARCH=1
C BEGIN CKS ON MODEL PREDICTIONS FOR HUMIDITY AT MAXIMUM BED DEPTH.
      XDP=DPTHMAX$CALL COOLSIM(4)
      IHTOP=IFIX((HOUT-HAMB)/DELH)+1
      IF(IHSTART.EQ.(IHTOP+1))151,153
C IF VARIATION OF HUMIDITY BETWEEN MAX AND MIN BED DEPTHS IS LT DELH, SKIP TO
C NEXT ITERATION OF INNER DO LOOP.
151   PRINT 99900
      GOTO220
C PERFORM MODEL FEASIBILITY CHECKS.
153   IF(THOUT.GT.THOUTMX.OR.XMOUT.GT.XMOUTMX)GOTO 230
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO155
      IF(((XMOUT-EQM)/(PRODM-EQM)).GT.1.)GOTO155
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XMOUT)155,154
154   IF(XMOUT.LT.XMOUTMN)155,160
155   ISEARCH=1
C IF NECESSARY BEGIN SEARCH FOR LOWER FEASIBLE BOUND ON HUMIDITY DURING THIS DO
C LOOP ITERATION.
160   IF(LSEARCH.NE.1)GOTO181
165   LTRY=(IHTOP-IHSTART)/2+IHSTART
167   HNOW=FLOAT(LTRY-1)*DELH+HAMB
      GUESS1=DPTHMIN$GUESS2=DPTHMAX
      CALL ZEROIN(GUESS1,GUESS2,EPS,HOWDEEP)
      XDP=(GUESS1+GUESS2)/2.$CALL COOLSIM(4)
      IF(THOUT.GT.THOUTMX.OR.XMOUT.GT. XMOUTMX)170,175
170   IHSTART=LTRY
      IF((IHTOP-IHSTART).LE.1)230,165
175   LTRY=(LTRY-IHSTART)/2+IHSTART
      IF(LTRY.EQ.IHSTART)180,167
180   IHSTART=LTRY+1
181   IF(ISEARCH.EQ.1)185,205
C IF NECESSARY, BEGIN SEARCH FOR UPPER FEASIBLE BOUND ON HUMIDITY DURING THIS DO
C LOOP ITERATION.
185   ITRY=(IHTOP-IHSTART)/2+IHSTART
187   HNOW=FLOAT(ITRY-1)*DELH+HAMB
      GUESS1=DPTHMIN$GUESS2=DPTHMAX
      CALL ZEROIN(GUESS1,GUESS2,EPS,HOWDEEP)
      XDP=(GUESS1+GUESS2)/2.$CALL COOLSIM(4)
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO 190
      IF(((XMOUT-EQM)/(PRODM-EQM)).GT.1.)GOTO190
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XMOUT)190,188
```

```
188    IF(XMOUT.LT.XMOUTMN)190,195
190    IHTOP=ITRY
       IF((IHTOP-IHSTART).LT.1)230,185
195    ITRY=(IHTOP-ITRY)/2+ITRY
       IF(ITRY.EQ.(IHTOP-1))200,187
200    IHTOP=ITRY
205    DEEP1=DPTHMIN
C FIND THE DEPTH AT WHICH EACH INTERMEDIATE HUMIDITY OCCURS AND EVALUATE
C CORRESPONDING AIR TEMPERATURE AND PROCESS COST.
       DO 210 IH=IHSTART,IHTOP
       HNOW=FLOAT(IH-1)*DELH+HAMB
       GUESS1=DEEP1$GUESS2=DPTHMAX
       CALL ZEROIN(GUESS1,GUESS2,EPS,HOWDEEP)
       XDP=(GUESS1+GUESS2)/2.$CALL COOLSIM(4)
       Q=GA*SPVLCON
       PDROP=XDP*(Q/5P.)**1.528
       TOUT=TOUT+.37854889*PDROP/FANETA/EMETA
       CALL CSTCOOL(COST)
       WRITE(11,33333)III,JJJ,KKK1,IH,TOUT,XDP,COST
       PRINT 33333,III,JJJ,KKK1,IH,TOUT,XDP,COST
       DEEP1=XDP
210    CONTINUE
  220  CONTINUE
  230  CONTINUE
  240  CONTINUE
250    CONTINUE
       ENDFILE 11
10000 FORMAT(2F10.0,I10)
10014 FORMAT(1H0*MINIMUM FEASIBLE BED DEPTH IS *F5.3,///)
20000 FORMAT(8F10.0)
33333 FORMAT(4I5,3F22.15)
40000 FORMAT(1X*MAX MC=*F5.3*  MIN MC=*F5.3*  NUMBER OF LEVELS=*I3*  MC
     1INCREMENT=*F6.4)
40001 FORMAT(1H0*PROD TEMP-MAX=*F5.0*  MIN=*F5.0*  NUMBER OF LEVELS=*I3*
     1  INCREMENT SIZE=*F5.1)
40003 FORMAT(1H0*AIRFLOW RATE-MAX=*F5.0*  MIN=*F5.0*  NUMBER OF LEVELS=*
     1I3*  INCREMENT SIZE=*F5.1,///)
40004 FORMAT(1H1*SET UP GRID SYSTEM AT THE COOLER-HEATER INTERFACE*///)
40005 FORMAT(1X*ADDITIONAL INFORMATION REQD FOR SOLUTION*///)
40006 FORMAT(1H0*ELECTRICAL PRICE=*F4.2*  TIME SCALE=*F4.2*  FAN EFF=*F4
     1.3*  MOTOR EFF=*F4.3,///)
40007 FORMAT(1X*SEARCH FOR FEASIBLE UPPER BOUND IN MC DIMENSION*)
40008 FORMAT(1H0*FOR THESE INPUTS, THE MINIMUM OUTLET PRODUCT TEMPERATUR
     1E IS *F7.2)
40009 FORMAT(1H0*ADJUSTED UPPER NODE IS *I5*,LOWER NODE IS 1*///)
40010 FORMAT(1X*SEARCH FOR FEASIBLE LOWER BOUND IN AIRFLOW DIMENSION*)
40011 FORMAT(1H0*UPPER NODE IS*I5*,ADJUSTED LOWER NODE IS*I5,///)
40012 FORMAT(1X*SEARCH FOR FEASIBLE UPPER BOUND IN THETA DIMENSION*)
40013 FORMAT(1X*SEARCH FOR FEASIBLE LOWER BOUND OF DEPTH*)
40014 FORMAT(1X*TO CHOOSE DELH, CHECK THE LIMITS OF OUTLET HUMIDITY*)
40015 FORMAT(1H0*LOWER BOUND=*F7.5*  UPPER BOUND=*F7.5,///)
40016 FORMAT(1X*BEGIN ITERATIONS,PRINTING INDICES OF MC, PROD TEMP,AIRFL
     1OW,ABS HUM, AND ASSOCIATED AIR TEMP, DEPTH, COST*)
40020 FORMAT(1H0*STOPPED-MAX DEPTH TOO SHORT TO REACH DESIRED OUTLET CON
     1DITIONS*)
70000 FORMAT(1X*INLET CONDITIONS-TEMP=*F7.3*  ABS HUM=*F8.5*  ATM PRESS=
     1*F5.2)
70001 FORMAT(1H0*DESIRED OUTLET CONDITIONS-MAX GRAIN TEMP=*F7.3*  MAX MC=
     1=*F7.4*  MIN MC=*F7.4)
70002 FORMAT(1H0*MAX DEPTH=*F4.2*  MIN DEPTH=*F4.2*INCREMENT SIZE IN ABS
```

```
     1 HUM=*F8.5)
90000 FORMAT(3X,*MOISTURE DIMENSION CONTAINS ONLY ONE FEASIBLE LEVEL*)
90001 FORMAT(3X,*AIRFLOW DIMENSION CONTAINS ONLY ONE FEASIBLE LEVEL*)
90002 FORMAT(3X,*THETA DIMENSION CONTIANS ONLY ONE FEASIBLE LEVEL*)
90005 FORMAT(1H0*FOR THESE INPUTS, THE MINIMUM OUTLET MC IS *F8.4)
99900 FORMAT(3X*DIFFERENCE IN HUMIDITY THRU BED LT DELH*)
      END
```

```
      PROGRAM HEATER(INPUT,OUTPUT,TAPE7,DOUT,TAPE11=DOUT)
C PROGRAM HEATER IS THE 2ND PROGRAM IN THE 4-PROGRAM SEQUENCE FOR OPTIMAL DESIGN
C OF AN AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER SYSTEM, ASSOCIATED
C SUBPROGRAMS REQUIRED-NONE.
C MINIMUM DIMENSIONS FOR THE ARRAYS ARE FOUND BY EXAMINATION OF COOLER OUTPUT,
C EQUAL TO THE MAXIMUM NUMBER OF STATE NODAL VALUES HAVING REPEATED INDICES FOR
C PRODUCT MC AND TEMPERATURE.
      DIMENSION IGSCRH(  ),IHSCRH(  ),TSCRACH(  ),CSTSCRH(  )
      DIMENSION GOUT(  ),GADDED(  ),DELHEAT(  )
      COMMON/PRESS/PATM
      DATA CA,CV/.242,.45/
C READ INFORMATION FROM COOLER OUTPUT-MIN AIRFLOW RATE AND INCREMENT, MIN HUMID-
C ITY AND INCREMENT, MIN AND MAX MC, AND MIN AND MAX PRODUCT TEMPERATURES,
C READ INFORMATION NEEDED TO SETUP GRID SYSTEM AT HEATER-DRYER INTERFACE AND
C LIMITING RECYCLED AIR-FRESH AIR RATIO.
      PRINT 50002
      READ 20000,GLOWIN,DELGIN,HLOWIN,DELHIN,XMLOW,XMHIGH,THLOW,THHIGH
      PRINT 50000
      PRINT 30000,GLOWIN,DELGIN,HLOWIN,DELHIN
      READ 20000,XLOWG,XHIG,RATIO1,XLOWT,XHIT,XLOWH,XHIH
      PRINT 30002,XMLOW,XMHIGH,THLOW,THHIGH,XLOWG,XHIG,XLOWT,XHIT,XLOWH,
     1XHIH
      PRINT 30005,RATIO1
C READ THE NUMBER OF LEVELS OF MC, PRODUCT TEMPERATURE, AIRFLOW RATE, AIR TEMP-
C ERATURE, AND HUMIDITY TO BE USED IN THE GRID, THEN CALCULATE INCREMENT SIZES
C IN EACH DIMENSION.
      READ 20001,LEVM,LEVTH,LEVG,LEVT,LEVH
      DELG=(XHIG-XLOWG)/FLOAT(LEVG-1)
      DELT=(XHIT-XLOWT)/FLOAT(LEVT-1)
      DELH=(XHIH-XLOWH)/FLOAT(LEVH-1)
      DELM=(XMHIGH-XMLOW)/FLOAT(LEVM-1)
      DELTH=(THHIGH-THLOW)/FLOAT(LEVTH-1)
      PRINT 30003,LEVM,LEVTH,LEVG,LEVT,LEVH,DELM,DELTH,DELG,DELT,DELH
C READ INLET AIR CONDITIONS TO THE HEATER-TEMPERATURE, HUMIDITY, AND ATMOSPHERIC
C PRESSURE.
      PRINT 50001
      READ 20000,TAMB,HAMB,PATM
      PRINT 30001,TAMB,HAMB,PATM
C READ ECONOMIC FACTORS-FUEL HEAT VALUE, FUEL PRICE,HEATER EFFICIENCY, AND THE
C TIME BASE USED FOR OPTIMIZATION.
      READ 20000,FUELHT,FUELPRI,THERETA,TIME
      PRINT 30004,FUELHT,FUELPRI,THERETA,TIME
      PRINT 50003
C READ THE FIRST PAIR OF MC AND PRODUCT TEMPERATURE INDICES AND BEGIN READING
C ALL GRID VALUES HAVING THESE TWO INDICES.
      IEND=0
      READ(7,11111)IM,ITH
      ITHOLD=ITH$IMOLD=IM
```

```
      BACKSPACE 7
89    ISCRACH=0
90    READ(7,10000)IM,ITH,IG,IH,T,COSTCL
      IF(EOF(7))109,94
94    IF(ITH.NE.ITHOLD)95,100
95    BACKSPACE 7
      ITHOLD=ITH$IMOLD=IM
      GOTO 110
C STORE AIRFLOW AND HUMIDITY INDICES PLUS CORRESPONDING AIR TEMPERATURE AND COST
C VALUES FOR ALL DATA GROUPS HAVING THE SAME MC AND PRODUCT TEMPERATURE INDICES.
100   ISCRACH=ISCRACH+1
      IGSCRH(ISCRACH)=IG
      IHSCRH(ISCRACH)=IH
      TSCRACH(ISCRACH)=T
      CSTSCRH(ISCRACH)=COSTCL
      GOTO 90
C APPLY ALL FEASIBLE CONTROLS (HEAT AND AIRFLOW ADDED) TO INLET AIR CONDITIONS.
109   IEND=1$IM=IMOLD$ITH=ITHOLD
110   DO 140 JJJ=1,LEVT
      TOUT=XLOWT+FLOAT(JJJ-1)*DELT
      DO 130 KKK=1,LEVH
      HOUT=XLOWH+FLOAT(KKK-1)*DELH
      IF(ISCRACH.EQ.1)GOTO 130
      DO 120 LLL=1,ISCRACH
      IF(CSTSCRH(LLL).GE.10.E22)115,116
115   GOUT(LLL)=-1.$GOTO 120
116   GIN=GLOWIN+FLOAT(IGSCRH(LLL)-1)*DELGIN
      HIN=HLOWIN+FLOAT(IHSCRH(LLL)-1)*DELHIN
      TIN=TSCRACH(LLL)
      IF(ABS(HOUT-HAMB).LT.1.E-12)117,118
117   GOUT(LLL)=1000000000.
      GOTO119
118   GOUT(LLL)=GIN*(HIN-HAMB)/(HOUT-HAMB)
119   GADDED(LLL)=GOUT(LLL)-GIN
      IF((GIN/GOUT(LLL)).LT.RATIO1)GOTO1195
      DELHEAT(LLL)=1.
      GADDED(LLL)=-1.
      GOTO120
1195  DELHEAT(LLL)=TOUT*GOUT(LLL)*(CA+HOUT*CV)-TIN*GIN*(CA+HIN*CV)-TAMB*
     1GADDED(LLL)*(CA+HAMB*CV)
120   CONTINUE
C HOLDING THE AIR TEMPERATURE AND HUMIDITY AND PRODUCT TEMPERATURE AND MC DIM-
C ENSIONS CONSTANT, INTERPOLATE IN THE AIRFLOW DIMENSION TO OBTAIN EVENLY SPACED
C VALUES OF CONTROLS AND COST.
      DO 125 MMM=2,ISCRACH
      GONE=GOUT(MMM-1)
      IF(GONE.LT.XLOWG.OR.GONE.GT.XHIG)GOTO 125
      IF(GADDED(MMM-1).LT.0.0.OR.DELHEAT(MMM-1).LT.0.0)GOTO125
      COSTONE=DELHEAT(MMM-1)/FUELHT/THERETA*FUELPRI*TIME+CSTSCRH(MMM-1)
      LWHERE1=IFIX((GONE-XLOWG)/DELG)+1
      GTRY=XLOWG+FLOAT(LWHERE1)*DELG
      IF(ABS(GONE-GTRY).LT.1.E-10)LWHERE1=LWHERE1+1
      GTWO=GOUT(MMM)
      IF(GTWO.LT.XLOWG.OR.GTWO.GT.XHIG)GOTO 125
      IF(GADDED(MMM  ).LT.0.0.OR.DELHEAT(MMM  ).LT.0.0)GOTO125
      COSTTWO=DELHEAT(MMM)/FUELHT/THERETA*FUELPRI*TIME+CSTSCRH(MMM)
      LWHERE2=IFIX((GTWO-XLOWG)/DELG)+1
      GTRY=XLOWG+FLOAT(LWHERE2)*DELG
      IF(ABS(GTWO-GTRY).LT.1.E-10)LWHERE2=LWHERE2+1
      IF(LWHERE1.EQ.LWHERE2)GOTO 125
```

```
      IF(GTWO.LT.GONE)GOTO 123
      NUMB=1
122   LW1PN=LWHERE1+NUMB
      G=XLOWG+FLOAT(LW1PN-1)*DELG
      RATIO=(G-GONE)/(GTWO-GONE)
      BESTCST=(COSTTWO-COSTONE)*RATIO+COSTONE
      BESTGAD=(GADDED(MMM)-GADDED(MMM-1))*RATIO+GADDED(MMM-1)
      BESTDHT=(DELHEAT(MMM)-DELHEAT(MMM-1))*RATIO+DELHEAT(MMM-1)
      WRITE(11,40000)IM,ITH,LW1PN,JJJ,KKK,BESTCST,BESTGAD,BESTDHT
      PRINT 77777    ,IM,ITH,LW1PN,JJJ,KKK,BESTCST,BESTGAD,BESTDHT
      IF(LW1PN.EQ.LWHERE2)GOTO 125
      NUMB=NUMB+1
      GOTO 122
123   NUMC=1
124   LW2PN=LWHERE2+NUMC
      G=XLOWG+FLOAT(LW2PN-1)*DELG
      RATIO=(G-GONE)/(GTWO-GONE)
      BESTCST=(COSTONE-COSTTWO)*RATIO+COSTTWO
      BESTGAD=(GADDED(MMM-1)-GADDED(MMM))*RATIO+GADDED(MMM)
      BESTDHT=(DELHEAT(MMM-1)-DELHEAT(MMM))*RATIO+DELHEAT(MMM)
      WRITE(11,40000)IM,ITH,LW2PN,JJJ,KKK,BESTCST,BESTGAD,BESTDHT
      PRINT 77777    ,IM,ITH,LW2PN,JJJ,KKK,BESTCST,BESTGAD,BESTDHT
      IF(LW2PN.EQ.LWHERE1)GOTO 125
      NUMC=NUMC+1
      GOTO 124
125   CONTINUE
130   CONTINUE
140   CONTINUE
150   CONTINUE
      IF(IEND.EQ.0)GOTO 89
      ENDFILE 11
10000 FORMAT(4I5,E22.15,22X,E22.15)
11111 FORMAT(2I5)
20000 FORMAT(8F10.0)
20001 FORMAT(8I10)
30000 FORMAT(1X*AIRFLOW-LOWER BOUND=*F7.2*  INCREMENT SIZE=*F7.3*  HUMID
     1ITY-LOWER BOUND=*F7.5*  INCREMENT SIZE=*F7.5,///)
30001 FORMAT(1X*AIR TEMPERATURE=*F7.2*  ABS HUMIDITY=*F7.5*  ATM PRESSUR
     1E=*F7.2,///)
30002 FORMAT(1X*MC-LOWER BND=*F5.4*  UPPER BND=*F5.4,//* PROD TEMP-LOWER
     1 BND=*F7.2*  UPPER BND=*F7.2,//* AIRFLOW-LOWER BND=*F7.2*  UPPER B
     2ND=*F7.2,//* AIR TEMP-LOWER BND=*F7.2*  UPPER BND=*F7.2,//* ABS HU
     3MIDITY-LOWER BND=*F7.5*  UPPER BND=*F7.5,//)
30003 FORMAT(1X*VARIABLE LEVELS-MC=*I4*  PROD TEMP=*I4*  AIRFLOW=*I4*  A
     1IR TEMP=*I4*  ABS HUM=*I4,//* INCREMENT SIZE-MC=*F5.4*  PROD TEMP=
     2*F5.1*  AIRFLOW=*F5.1*  AIR TEMP=*F5.1*  ABS HUM=*F7.5,///)
30004 FORMAT(1X*ECONOMIC FACTORS-FUEL HEAT VALUE=*F9.0*  FUEL PRICE=*F5.
     12*  THERMAL EFFICIENCY=*F4.3*  TIME SCALE=*F4.1,///)
30005 FORMAT(1X*MAX ALLOWABLE COOLER AIR/TOTAL AIR RATIO=*F5.3,///)
40000 FORMAT(5I5,3E22.15)
50000 FORMAT(1H1*READ INFORMATION FROM COOLER SOLUTION*//)
50001 FORMAT(1X*READ CONDITIONS OF ADDED AIR*//)
50002 FORMAT(1X*SET UP GRID SYSTEM AT THE HEATER-DRYER INTERFACE*///)
50003 FORMAT(1X*PRINT INDICES OF FEASIBLE MC,PROD TEMP,AIRFLOW,AIRTEMP,A
     1BS HUM, AND ASSOCIATED ACCUMULATED COST,ADDED AIR,ADDED HEAT*//)
77777 FORMAT(3X,5I7,3E22.15)
      END
```

```
      PROGRAM DRYER(INPUT,OUTPUT,TAPE37)
C PROGRAM DRYER IS THE 3RD PROGRAM TO BE RUN IN THE 4-PROGRAM SEQUENCE FOR
C OPTIMAL DESIGN OF AN AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM.  ASSOCIATED SUBPROGRAMS REQUIRED--DRYCOST,DRYHEAT,DRYSIM,EMC,IENCODE,
C INTERP,PSDB,PVHA,RHPSPV,VSDBHA.
      DIMENSION INDEX(3),ICECREM(3),IND(3),MAX(3)
      COMMON/CON/CON1,CON2/VALUES/XMCIN,THIN,GP,SA,CA,CP,CV/DOLLAR/TIME,
     1ELPRICE,EMETA,FANETA/RECALL/OPTCOST(  ,  ,  ),XNEW(2),DXINV(2),XML
     2OW,DELM,THLOW,DELTH/PRESS/PATM/IN/SUMTEMP,HIN,GIN,DEPTH/OUT/XM,THE
     3TA,TOUT,HOUT
      EXTERNAL DRYHEAT
      DATA PATM,RHOP,BPH,SA,CA,CP/14.34,38.71,9.,.239,.242,.268/
      BESTCST=10.E22
      GP=BPH*1.244*RHOP
      CON1=SA/(GP*CP)$CON2=SA/CA
C READ NUMBER OF FEASIBLE LEVELS REMAINING IN MC, PRODUCT TEMP, AIRFLOW RATE,
C AIR TEMPERATURE, AND HUMIDITY DIMENSIONS.
      READ 10000,LEVELM,LEVELTH,LEVELG,LEVELT,LEVELH
      PRINT 20001,LEVELM,LEVELTH,LEVELG,LEVELT,LEVELH
      MAX(1)=LEVELH$MAX(2)=LEVELT$NDIM=3
C READ LOWER FEASIBLE MC BOUND AND INCREMENT SIZE AT HEATER-DRYER INTERFACE.
      READ 10002,XMLOW,DELM
      DXINV(1)=1./DELM
      XMHIGH=XMLOW+FLOAT(LEVELM-1)*DELM
      PRINT 20002,XMLOW,XMHIGH,DELM
C READ LOWER FEASIBLE PRODUCT TEMPERATURE BOUND AND INCREMENT SIZE AT HEATER-
C DRYER INTERFACE
      READ 10002,THLOW,DELTH
      DXINV(2)=1./DELTH
      THHIGH=THLOW+FLOAT(LEVELTH-1)*DELTH
      PRINT 20003,THLOW,THHIGH,DELTH
C READ LOWER FEASIBLE AIRFLOW RATE BOUND AND INCREMENT SIZE AT HEATER-DRYER
C INTERFACE.
      READ 10002,GLOW,DELG
      PRINT 20004,GLOW,DELG
C READ LOWER FEASIBLE AIR TEMPERATURE BOUND AND INCREMENT SIZE AT HEATER-DRYER
C INTERFACE.
      READ 10002,TLOW,DELT
      PRINT 20005,TLOW,DELT
C READ LOWER FEASIBLE HUMIDITY BOUND AND INCREMENT SIZE AT HEATER-DRYER INTER-
C FACE.
      READ 10002,HLOW,DELH
      PRINT 20006,HLOW,DELH
C READ THE LOWER BOUNDS OF EACH DIMENSION AS THEY ARE BEFORE ADJUSTMENT--MC,
C PRODUCT TEMP,AIRFLOW RATE,AIR TEMP,AND HUMIDITY;
      READ 10002,XMMNIN,THMNIN,XLOWG,XLOWT,XLOWH
      PRINT 20007,XMMNIN,THMNIN,XLOWG,XLOWT,XLOWH
      MINDXM=IFIX((XMLOW+DELM/2.-XMMNIN)/DELM)
      MINDXTH=IFIX((THLOW+DELTH/2.-THMNIN)/DELTH)
      MINDXG=IFIX((GLOW+DELG/2.-XLOWG)/DELG)
      MINDXT=IFIX((TLOW+DELT/2.-XLOWT)/DELT)
      MINDXH=IFIX((HLOW+DELH/2.-XLOWH)/DELH)
C READ BOUND OF DRYER DEPTH AND INCREMENT SIZE.
      READ 99999,DLOW, DHIGH,LEVELD
      PRINT 20008,DLOW,DHIGH,LEVELD
      DELD=(DHIGH-DLOW)/FLOAT(LEVELD-1)
C READ DRYER INLET GRAIN MC AND TEMP PLUS ECONOMIC FACTORS--TIME ON WHICH OPT-
C IMIZATION IS BASED,ELECTRIC PRICE,MOTOR EFFICIENCY,FAN EFFICIENCY, AND AT-
C MOSPHERIC PRESSURE.
      READ 10002,XMCIN,THIN,TIME,ELPRICE,EMETA,FANETA,PATM
```

```
      PRINT 20009,YMCIN,THIN,PATM
      PRINT 20010,TIME,ELPRICE,EMETA,FANETA
C READ CONSTRAINTS ON MAXIMUM AIR TEMPERATURE INTO DRYER AND MAXIMUM PRESSURE
C DROP.
      READ 20001,TMAX,PDROPMX
      PRINT 20011,TMAX,PDROPMX
215   INDYMAX=LEVELG*LEVELT*LEVELH
C INITIALIZE THE COST INTERPOLATION GRID TO ABSURD VALUES.
      DO 9 I=1,LEVELM
      DO 7 J=1,LEVELTH
      DO 5 K=1,INDYMAX
5     OPTCOST(I,J,K)=10.E25
7     CONTINUE
9     CONTINUE
C READ IN VALUES FROM HEATER OUTPUT AND BY COMPARISON, FILL THE GRID WITH OPT-
C IMAL COSTS.
11    READ(37,30000)IM,ITH,IND(3),IND(2),IND(1),EXPENSE
      IF(EOF(37))20,15
15    IM=IM-MINDXM
      ITH=ITH-MINDXTH
      IND(3)=IND(3)-MINDXG
      IND(2)=IND(2)-MINDXT
      IND(1)=IND(1)-MINDXH
      CALL IENCODE(INDY,IND,MAX,NDIM)
      IF(EXPENSE.LT.OPTCOST(IM,ITH,INDY))16,17
16    OPTCOST(IM,ITH,INDY)=EXPENSE
17    GOTO 11
C APPLY IN TURN ALL AIRFLOW RATE, INLET AIR TEMPERATURE, AND HUMIDITY COMBIN-
C ATIONS WITHIN THE GRID.
20    DO 190 IG=1,LEVELG
      IGG=LEVELG-IG+1$IND(3)=IGG
      GIN=GLOW+FLOAT(IGG-1)*DELG
      IND(3)=IGG
      DO 180 IT=1,LEVELT
      ITT=LEVELT-IT+1
      TIN=TLOW+FLOAT(ITT-1)*DELT$IND(2)=ITT
      DO 170 IH=1,LEVELH
      HIN=HLOW+FLOAT(IH-1)*DELH$IND(1)=IH
      ATIN=TIN+459.69
      RHIN=RHDBHA(ATIN,HIN)
      EQM=EMC(RHIN,TIN)
C APPLY ALL DISCRETE DEPTH CONTROLS IN TURN.
      DO 160 IDEPTH=1,LEVELD
      IDD=LEVELD-IDEPTH+1
      DEPTH=DLOW+FLOAT(IDD-1)*DELD
      ATEMP=TIN+459.69
      QAPPROX=GIN*VSDBHA(ATEMP,HIN)/60.
      PDROP=DEPTH*(QAPPROX/58.)**1.528
C CHECK MAXIMUM AIR TEMPERATURE AND PRESSURE DROP CONSTRAINTS.
      IF(PDROP.GT.PDROPMX)GOTO160
      DELTADD=.37854889*PDROP/FANETA/EMETA
      SUMTEMP=TIN+DELTADD
      IF(SUMTEMP.GT.TMAX)GOTO160
      ASUM=SUMTEMP+459.69
      Q=GIN*VSDBHA(ASUM,HIN)/60.
C CALL DRYER SIMULATION TO PREDICT OUTLET GRAIN MC AND TEMPERATURE AND CHECK IF
C RESULT FALLS WITHIN GRID.
      CALL DRYSIM(1)
      IF(XM.GT.XMHIGH)GOTO170
      IF(XM.LT.XMLOW)GOTO160
```

```
      IF(THETA.LT.THLOW)GOTO160
      IF(THETA.GT.THHIGH)GOTO170
C APPLY SATURATION, ADSORPTION, AND EQUILIBRIUM CHECKS TO OUTLET AIR TEMP AND
C HUMIDITY PREDICTIONS OF DRYER MODEL.
      CALL DRYSIM(2)
      ATOUT=TOUT+459.69
      RHOUT=RHDBHA(ATOUT,HOUT)
      IF(RHOUT.GT.1.)GOTO160
      IF(((XM-EQM)/(XMCIN-EQM)).GT.1.)GOTO160
      EQMOUT=EMC(RHOUT,TOUT)
      IF(EQMOUT.GE.XM)GOTO160
C SET UP INTERPOLATION PROCEDURE AND INTERPOLATE.
      CALL IENCODE(INDY,IND,MAX,NDIM)
      INDEX(1)=IFIX((XM-XMLOW)/DELM)+1
      INDEX(2)=IFIX((THETA-THLOW)/DELTH)+1
      INDEX(3)=INDY
      XNEW(1)=XM$XNEW(2)=THETA
      NSTORE=0
      DO 85 I=1,NDIM
85    ICECREM(I)=0
      XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      IF(ABS(XVAL-XM).LT.10.E-8)90,100
90    NSTORE=NSTORE+1$ICECREM(NSTORE)=1
100   XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      IF(ABS(XVAL-THETA).LT.10.E-8)110,120
110   NSTORE=NSTORE+1$ICECREM(NSTORE)=2
120   NSTORE=NSTORE+1$ICECREM(NSTORE)=3
      IF(NSTORE.EQ.NDIM)130,140
130   PARCOST=OPTCOST(INDEX(1),INDEX(2),INDEX(3))
      GOTO 145
140   CALL INTERP(PARCOST,NDIM,INDEX,ICECREM,NSTORE,DRYHEAT)
145   IF(PARCOST.LT.1.E-8.OR.PARCOST.GT.10.E22)GOTO160
C EVALUATE COST OF DRYING AND ADD TO INTERPOLATED COST.  COMPARE THE RESULT TO
C THE CURRENT OPTIMAL COST.
      CALL DRYCOST(COST,PDROP,Q)
      TRYCOST=PARCOST+COST
      IF(TRYCOST.LT.BESTCST )150,160
C IF CURRENT TOTAL COST IS OPTIMAL, REPLACE THE PREVIOUS BEST COST AND PRINT THE
COPTIMAL COST AND CORRESPONDING DRYER INLET AIR AND OUTLET GRAIN CONDITIONS PLUS
C OPTIMAL DEPTH.
150   BESTCST=TRYCOST$BESTDPH=DEPTH$BESTG=GIN$BESTT=TIN$BESTH=HIN$BESTM=
     1XM$BESTTH=THETA
      PRINT 20013,BESTCST,BESTM,BESTTH
      PRINT 20014,BESTDPH,BESTG,BESTT,BESTH
160   CONTINUE
170   CONTINUE
180   CONTINUE
190   CONTINUE
10000 FORMAT(8I10)
10002 FORMAT(8F10.3)
20001 FORMAT(1H1*FEASIBLE LEVELS REMAINING IN EACH DIMENSION-MC=*I3* PRO
     1D TEMP=*I3* AIRFLOW RATE=*I3* AIR TEMP=*I3* HUMIDITY=*I3,//)
20002 FORMAT(1HO*MOISTURE CONTENT FEASIBLE BOUNDS-LOWER=*F7.4*  UPPER=*F
     17.4*  INCREMENT=*F7.4,//)
20003 FORMAT(1HO*PRODUCT TEMP FEASIBLE BOUNDS-LOWER=*F6.2*  UPPER=*F6.2*
     1  INCREMENT=*F6.2,//)
20004 FORMAT(1HO*LOWER FEASIBLE AIRFLOW RATE BOUND=*F6.2*  INCREMENT=*F6
     1.2*  INCREMENT=*F6.2,//)
20005 FORMAT(1HO*LOWER FEASIBLE AIR TEMP BOUND=*F6.2*  INCREMENT=*F6.2,/
     1/)
```

```
20006 FORMAT(1HO*LOWER FEASIBLE HUMIDITY BOUND=*F6.4*  INCREMENT=*F6.4,/
     1/)
20007 FORMAT(1HO*LOWER BOUNDS BEFORE ADJUSTMENT--MC*F5.3*  PRODUCT TEMP=
     1*F5.2*  AIRFLOW RATE=*F5.2*  AIR TEMP=*F5.2*  HUMIDITY=*F5.4,/7)
20008 FORMAT(1HO*DRYER DEPTH--LOWER BOUND=*F5.3*  UPPER BOUND=*F5.3*  NU
     1MBER OF LEVELS=*I3,//)
20009 FORMAT(1HO*INLET CONDITIONS TO DRYER--GRAIN MC=*F5.3*  GRAIN TEMP=
     1*F6.4*  ATMOSPHERIC PRESSURE=*F6.4,//)
20010 FORMAT(1HO*ECONOMIC FACTORS--TIME ON WHICH OPTIMIZATION IS BASED=*
     1F4.2*  ELECTRIC PRICE=*F4.2*  EFFICIENCIES-MOTOR=*F4.3*  FAN=*F4.3
     1,//)
20011 FORMAT(1HO,*CONSTRAINTS--MAX AIR TEMP=*F6.2* MAX PRESSURE DROP=*F6
     1.2,//)
20013 FORMAT(1HO*CURRENT OPTIMAL COST=*F10.5*  OUTLET MC=*F5.4*  OUTLET
     1GRAIN TEMPERATURE=*F6.2)
20014 FORMAT(1HO*CURRENT OPTIMAL DEPTH=*F4.2*  AIR CONDITIONS--FLOW RATE
     1=*F6.2*  TEMPERATURE=*F6.2*  HUMIDITY=*F6.4,///)
30000 FORMAT(5I5,E22.15,44X)
99999 FORMAT(2F10.0,I10)
      END
```

```
      PROGRAM RECOVRY(INPUT,OUTPUT)
C PROGRAM RECOVERY IS THE FINAL PROGRAM TO BE RUN IN THE 4-PROGRAM SEQUENCE FOR
C OPTIMAL DESIGN OF AN AIR-RECYCLE TYPE CONCURRENT DRYER-COUNTERFLOW COOLER
C SYSTEM.  ASSOCIATED SUBPROGRAMS REQUIRED-COOLSIM,DRYSIM,HWDEFPR,VSDBHA,ZEROIN,
      COMMON/PRESS/PATM/IN/THEAT,BESTH,Q,BESTDPH/OUT/Z,YZ,TOUTDRY,HOUTDR
     1Y/INCOOL/BESTM,BESTTH,GINHEAT,XDP/OUTCOOL/HOUT,TOUT,THOUT,XMOUT/FI
     2XED/TAMB,HAMB,ATAMB,SPVLCON/AITCH/HINCOOL
      COMMON/VALUES/XMCIN,THIN,GP,SA,CA,CP,CV
      EXTERNAL HOWDEFP
      DATA EPS/.01/
C READ (FROM DRYER OUTPUT) OPTIMAL AIRFLOW RATE, INLET AIR TEMP, INLET HUMIDITY
C DRYER DEPTH, PLUS CORRESPONDING OUTLET MC AND PRODUCT TEMP.
      READ 10000,BESTG,BESTT,BESTH,BESTDPH,BESTM,BESTTH
C READ (FROM HEATER OUTPUT) THE BASE VALUES OF MC, PRODUCT TEMP, AIRFLOW RATE,
C AND HEAT ADDED NEEDED FOR INTERPOLATION.
      READ 10000,BASEM,BASETH,BASEGAD,BASEDHT
C READ (FROM HEATER OUTPUT) THE VALUES OF MC INCREMENT, AIRFLOW RATE, AND HEAT
C ADDED NEEDED FOR INTERPOLATION IN THE MC DIMENSION.
      READ 10000,DELM,PLUSGAM,PLUSHTM
C READ (FROM HEATER OUTPUT) THE VALUES OF PRODUCT TEMP INCREMENT, AIRFLOW RATE,
C AND HEAT ADDED NEEDED FOR INTERPOLATION IN THE PRODUCT TEMPERATURE DIMENSION.
      READ 10000,DELTH,PLUSGAT,PLUSTHT
C READ INLET MC AND PRODUCT TEMPERATURE TO DRYER.
      READ 10000,XMCIN,THIN
C READ TEMP AND HUMIDITY OF INLET AIR TO HEATER, PLUS FAN AND MOTOR EFFICIENCIES
      READ 10000,TINHEAT,HINHEAT,FANETA,EMETA
C READ TEMP AND HUMIDITY OF INLET AIR TO COOLER, PLUS MAX AND MIN BOUNDS ON
C COOLER DEPTH, AND ATMOSPHERIC PRESSURE.
      READ 10000,TAMB,HAMB,DPTHMAX,DPTHMIN,PATM
C INTERPOLATE TO FIND OPTIMAL AMOUNTS OF ADDED AIR AND HEAT.
      BMP1=BASEM+DELM$BTHP1=BASETH+DELTH
      GAD1=BASEGAD+(PLUSGAM-BASEGAD)*(BMP1-BASEM)/DELM
      GAD2=BASEGAD+(PLUSGAT-BASEGAD)*(BTHP1-BASETH)/DELTH
      GADDED=(GAD1+GAD2)/2.
      DHT1=BASEDHT+(PLUSHTM-BASEDHT)*(BMP1-BASEM)/DELM
```

```
      DHT2=BASEDHT+(PLUSTHT-BASEDHT)*(BTHP1-BASETH)/DELTH
      DELHEAT=(DHT1+DHT2)/2.
C CALCULATE AIRFLOW RATE THROUGH COOLER.
      GINHEAT=BESTG-GADDED
C CALCULATE OUTLET AIR HUMIDITY FROM COOLER.
      HINCOOL=(BESTG*BESTH-GADDED*HINHEAT)/GINHEAT
C CALCULATE OUTLET AIR TEMP FROM COOLER.
      ABESTT=BESTT+459.69
      QAPPROX=BESTG*VSDBHA(ABESTT,BESTH)/60.
      PDROP=BESTDPH*(QAPPROX/58.)**1.528
      TAPPROX=BESTT+.37854889*PDROP/FANETA/EMETA
      ANEWT=TAPPROX+459.69
      Q=BESTG*VSDBHA(ANEWT,BESTH)/60.
      PDROP=BESTDPH*(Q/58.)**1.528
      THEAT=BESTT+.37854889*PDROP/FANETA/EMETA
C SIMULATE DRYER TO CALCULATE OUTLET AIR TEMP AND HUMIDITY.
      CALL DRYSIM(2)
      PRINT 20000,TOUTDRY,HOUTDRY,XMCIN,THIN,BESTDPH
      PRINT 30000,THEAT,BESTH,BESTG,BESTM,BESTTH
      PRINT 40000,GADDED,DELHEAT,TINHEAT,HINHEAT
      ATAMB=TAMB+459.69
      SPVLCON=VSDBHA(ATAMB,HAMB)/60.
C SEARCH FOR OPTIMAL COOLER DEPTH.
      GUESS1=DPTHMIN$GUESS2=DPTHMAX
      CALL ZEROIN(GUESS1,GUESS2,EPS,HOWDEEP)
      DEPTH=(GUESS1+GUESS2)/2.
      CALL COOLSIM(4)
      PRINT 50000,TOUT,HOUT,GINHEAT,BESTM,BESTTH
      PRINT 60000,TAMB,HAMB,DEPTH,XMOUT,THOUT
10000 FORMAT(8F10.0)
20000 FORMAT(1H1*DRYER SPECS-OUTLET AIR TEMP=*F6.2*  HUMIDITY=*F6.4*  IN
     1LET GRAIN MC=*F5.3*  PROD TEMP=*F5.1*  DRYER DEPTH=*F4.2,//)
30000 FORMAT(1X*DRYER-HEATER INTERFACE AIR TEMP=*F6.2*  HUMIDITY=*F6.4,
     1*AIRFLOW=*F6.2*  MC=*F5.3*  PROD TEMP=*F5.1,//)
40000 FORMAT(1X*HEATER SPECS-ADDED AIR=*F6.2*  ADDED HEAT=*F10.2*  TEMP
     1INLET AIR=*F6.2*  INLET HUMIDITY=*F6.4,//)
50000 FORMAT(1X*HEATER-COOLER INTERFACE AIR TEMP=*F6.2*  HUMIDITY=*F6.4*
     1  AIRFLOW RATE=*F6.2*  MC=*F4.3*  PROD TEMP=*F6.2,//)
60000 FORMAT(1X*COOLER SPECS-INLET AIR TEMP=*F6.2*  HUMIDITY=*F6.5*  DEP
     1TH=*F6.2*  OUTLET MC=*F6.4*  OUTLET PROD TEMP=*F6.2)
      END
```

APPENDIX C

```
      SUBROUTINE COOLERC(AVERAGE,KIP,LMNOP,INDEX)
C COOLERC IS CALLED FROM INTERP TO INTERPOLATE COOLER COST.
      DIMENSION INDEX(2),LMNOP(2)
      COMMON/RECALL/OPTCOST(  ,  ),GCOOL(  ,  ),DCOOL(  ,  ),XNEW(2),DXI
     1NV(2),XMLOW,DELM,THLOW,DELTH
      STINTRP(XONE,XTWO,XX,XXX,DX)=XONE+(XTWO-XONE)*(XX-XXX)*DX
      LMNOP(KIP)=INDEX(KIP)
      XONE=OPTCOST(LMNOP(1),LMNOP(2))
      LMNOP(KIP)=INDEX(KIP)+1
      XTWO=OPTCOST(LMNOP(1),LMNOP(2))
      GOTO(10,20),KIP
10    XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      AVERAGE=STINTRP(XONE,XTWO,XNEW(1),XVAL,DXINV(1))
      RETURN
20    XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      AVERAGE=STINTRP(XONE,XTWO,XNEW(2),XVAL,DXINV(2))
      RETURN
      END


      SUBROUTINE COOLERD(AVERAGE,KIP,LMNOP,INDEX)
C COOLERD IS CALLED FROM INTERP TO INTERPOLATE COOLER DEPTH.
      DIMENSION INDEX(2),LMNOP(2)
      COMMON/RECALL/OPTCOST(  ,  ),GCOOL(  ,  ),DCOOL(  ,  ),XNEW(2),DXI
     1NV(2),XMLOW,DELM,THLOW,DELTH
      STINTRP(XONE,XTWO,XX,XXX,DX)=XONE+(XTWO-XONE)*(XX-XXX)*DX
      LMNOP(KIP)=INDEX(KIP)
      XONE=DCOOL(LMNOP(1),LMNOP(2))
      LMNOP(KIP)=INDEX(KIP)+1
      XTWO=DCOOL(LMNOP(1),LMNOP(2))
      GOTO(10,20),KIP
10    XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      AVERAGE=STINTRP(XONE,XTWO,XNEW(1),XVAL,DXINV(1))
      RETURN
20    XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      AVERAGE=STINTRP(XONE,XTWO,XNEW(2),XVAL,DXINV(2))
      RETURN
      END


      SUBROUTINE COOLERG(AVERAGE,KIP,LMNOP,INDEX)
C COOLERG IS CALLED FROM INTERP TO INTERPOLATE COOLER AIRFLOW RATE.
      DIMENSION INDEX(2),LMNOP(2)
      COMMON/RECALL/OPTCOST(  ,  ),GCOOL(  ,  ),DCOOL(  ,  ),XNEW(2),DXI
     1NV(2),XMLOW,DELM,THLOW,DELTH
      STINTRP(XONE,XTWO,XX,XXX,DX)=XONE+(XTWO-XONE)*(XX-XXX)*DX
      LMNOP(KIP)=INDEX(KIP)
      XONE=GCOOL(LMNOP(1),LMNOP(2))
      LMNOP(KIP)=INDEX(KIP)+1
      XTWO=GCOOL(LMNOP(1),LMNOP(2))
      GOTO(10,20),KIP
10    XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      AVERAGE=STINTRP(XONE,XTWO,XNEW(1),XVAL,DXINV(1))
      RETURN
```

```
20    XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      AVERAGE=STINTRP(XONE,XTWO,XNEW(2),XVAL,DXINV(2))
      RETURN
      END
```

```
      SUBROUTINE COOLSIM(IGO)
C COOLSIM PREDICTS OUTLET PRODUCT TEMPERATURE, HUMIDITY, AIR TEMP, AND MOISTURE
C CONTENT FOR THE COUNTERFLOW COOLER AS A FUNCTION OF INLET CONDITIONS AND BED
C DEPTH.
      COMMON/FIXED/TAMB,HAMB,ATAMB,SPVLCON/INCOOL/A5,A6,XGA,A8/OUTCOOL/A
     11,A2,A3,A4
      DELTH(X)=X-140.
      DELT(X)=X-45.
      DELH(X)=X-.0035
      DELM(X)=X-.21
      DELCFM(X)=X-100.
      DELX(X)=X-.4
      A7=XGA*SPVLCON
      GOTO(10,15,20,10),IGO
C PRODUCT TEMPERATURE CALCULATIONS.
10    DTH=.3833*DELTH(A6)
      DT=.6097*DELT(TAMB)
      DH=264.9*DELH(HAMB)
      DM=71.93*DELM(A5)
      DCFM=-.4452*DELCFM(A7)
      DXL=-18.88*DELX(A8)
      A3=DTH+DT+DH+DM+DCFM+DXL+80.459
      IF(IGO.NE.4)RETURN
C HUMIDITY CALCULATIONS.
15    DTH=.00003667*DELTH(A6)
      DT=.0001568*DELT(TAMB)
      DH=.9129*DELH(HAMB)
      DM=.02305*DELM(A5)
      DCFM=5.474E-7*DELCFM(A7)**2-6.115E-5*DELCFM(A7)
      DXL=.0007893*DELX(A8)
      A1=DTH+DT+DH+DM+DCFM+DXL+.00772
      IF(IGO.NE.4)RETURN
C AIR TEMPERATURE CALCULATIONS.
      DTH=.952*DELTH(A6)
      DCFM=2.035*(1.-EXP(.02190*DELCFM(A7)))
      A2=DTH+DCFM+137.77
C MC CALCULATIONS.
20    DM=.9635*DELM(A5)
      DXL=-.00826*DELX(A8)
      A4=DM+DXL+.20555
      RETURN
      END
```

```
      SUBROUTINE COSTFUN(COST,TINLET,Q)
C  SUBROUTINE COSTFUN EVALUATES  THE PROCESS COST.
      COMMON/ECON/FAMETA,EMETA,THERETA,ELPRICE,FUPRICE,TAMB,FACTOR,ELCOS
     1T,FULCOST/DT/DEPTH,TIME
```

```
20     PDROP=DEPTH*(Q/58.)**1.528$DELT=.378548895899*PDROP/FANETA/EMETA
       ELCOST=Q*PDROP*TIME*ELPRICE*1.1748031496O6E-4/EMETA/FANETA
       FULCOST=FUPRICE*Q*FACTOR*(TINLET-TAMB-DELT)*TIME/THERETA*60$
       COST=ELCOST+FULCOST$RETURN
       END




       SUBROUTINE COSTONE(COST,PDROP)
C COSTONE EVALUATES THE COST OF THE DRYING PROCESS.
       COMMON/DOLLAR/TIME,ELPRICE,FUELPRI,FUELHT,EMETA,FANETA,THERETA/SIM
      1COST/Q,TAMB,TIN/IN/SUMTEMP,HAMB,GIN,DEPTH/VALUES/XMCIN,THIN,GP,SA,
      2CA,CP,CV
       ELCOST=Q*PDROP*TIME*ELPRICE*1.17480314960 6E-4/EMETA/FANETA
       FULCOST=GIN*(CA+CV*HAMB)*(TIN-TAMB)/FUELHT/THERETA*FUELPRI*TIME
       COST=ELCOST+FULCOST
       RETURN
       END




       FUNCTION COSTRT(Q)
C  FUNCTION COSTRT IS USED BY OPTWHIZ TO DETERMINE ISOCOST SLOPE.
       COMMON/ECON/FANETA,EMETA,THERETA,ELPRICE,FUPRICE,TAMB,FACTOR,ELCOS
      1T,FULCOST/DT/DEPTH,TIME/CT/COSTLFT,TOFF
       PDROP=DEPTH*(Q/58.)**1.528$DELT=.378548895899*PDROP/FANETA/EMETA
       ELCOST=Q*PDROP*TIME*ELPRICE*1.1748031496O6E-4/EMETA/FANETA
       FULCOST=FUPRICE*Q*FACTOR*(TOFF   -TAMB-DELT)*TIME/THERETA*60$
       COSTRT=COSTLFT-ELCOST-FULCOST$RETURN
       END




       FUNCTION COVER(T)
C  FUNCTION COVER IS USED BY OPTWHIZ TO FIND THE INTERSECTION OF THE
C  ISOSTERE WITH A TEMPERATURE BOUND.
       COMMON/FINAL/XMFINAL/QT/QUE,TEE
       CALL QUICK(AVEMCDB,T,QUE)$COVER=AVEMCDB*XMFINAL$RETURN
       END




       SUBROUTINE CSTCOOL(COST)
C CSTCOOL EVALUATES THE COST OF THE AIRFLOW THRU THE COUNTERFLOW COOLER.
       COMMON/FIXED/TAMB,HAMB,ATAMB,SPVLCON/PRICE/ELPRICE,TIME,FANETA,EMETA
      1TA/INCOOL/PRCUM,THETA,GA,DEPTH
       Q=GA*SPVLCON
       PDROP=DEPTH*(Q/58.)**1.528
       COST=Q*PDROP*TIME*ELPRICE*1.74803149606E-4/EMETA/FANETA
       RETURN
       END
```

```
      FUNCTION DEEPMNM(D)
C DEEPMNM IS USED IN A 1-D SEARCH TO FIND THE BED DEPTH AT WHICH THE MAXIMUM
C ALLOWABLE OUTLET MC OCCURS.
      COMMON/TRIAL/THOUTMX,XMOUTMX/INCOOL/XMOI,XTH,XGA,XDP/OUTCOOL/HOUT,
     1TOUT,THOUT,XMOUT
      XDP=D$CALL COOLSIM(3)
      DEEPMNM=XMOUT-XMOUTMX
      RETURN
      END



      FUNCTION DEEPMNT(D)
C DEEPMNT IS USED IN A 1-D SEARCH TO FIND THE BED DEPTH AT WHICH THE MAXIMUM
C ALLOWABLE OUTLET PRODUCT TEMPERATURE OCCURS.
      COMMON/TRIAL/THOUTMX,XMOUTMX/INCOOL/XMOI,XTH,XGA,XDP/OUTCOOL/HOUT,
     1TOUT,THOUT,XMOUT
      XDP=D$CALLCOOLSIM(1)
      DEEPMNT=THOUT-THOUTMX
      RETURN
      END



      FUNCTION DIAG(T)
C  FUNCTION DIAG IS USED BY OPTWHIZ TO LOCATE THE INTERSECTION OF THE
C  ISOSTERE WITH THE DIAGONAL OF THE FEASIBLE REGION.
      COMMON/FINAL/XMFINAL/STUFF/DIAGSLP,QMIN,TMIN,QMAX,TMAX
      Q=DIAGSLP*(T-TMIN)+QMIN$CALL QUICK(AVEMCDB,T,Q)
      DIAG=AVEMCDB-XMFINAL$RETURN
      END



      SUBROUTINE DRYCOST(COST,PDROP)
C DRYCOST EVALUATES THE COST OF THE DRYING PROCESS.
      COMMON/DOLLAR/TIME,ELPRICE,EMETA,FANETA/SIMCOST/Q,ZZ,ZZZ
      COST=Q*PDROP*TIME*ELPRICE*1.174803149606E-4/EMETA/FANETA
      RETURN
      END



      SUBROUTINE DRYHEAT(AVERAGE,KIP,LMNOP,INDEX)
C DRYHEAT IS CALLED FROM INTERP TO PERFORM A LINEAR INTERPOLATION IN ONE DIM-
C ENSION.
      DIMENSION INDEX(3),LMNOP(3)
      COMMON/RECALL/OPTCOST(   ,   ,   ),XNEW(2),DXINV(2),XMLOW,DELM,THLOW
     1,DELTH
      STINTRP(XONE,XTWO,XX,XXX,DX)=XONE+(XTWO-XONE)*(XX-XXX)*DX
      LMNOP(KIP)=INDEX(KIP)
      XONE=OPTCOST(LMNOP(1),LMNOP(2),LMNOP(3))
```

```
      LMNOP(KIP)=INDEX(KIP)+1
      XTWO=OPTCOST(LMNOP(1),LMNOP(2),LMNOP(3))
      GOTO(10,20),KIP
10    XVAL=XMLOW+FLOAT(INDEX(1)-1)*DELM
      AVERAGE=STINTRP(XONE,XTWO,XNEW(1),XVAL,DXINV(1))
      RETURN
20    XVAL=THLOW+FLOAT(INDEX(2)-1)*DELTH
      AVERAGE=STINTRP(XONE,XTWO,XNEW(2),XVAL,DXINV(2))
      RETURN
      END



      SUBROUTINE DRYSIM(IGO)
C DRYSIM PREDICTS THE OUTLET GRAIN MOISTURE CONTENT AND TEMP PLUS THE OUTLET
C AIR TEMP AND HUMIDITY.
      COMMON/CON/CON1,CON2/VALUES/XMCIN,THIN,GP,SA,CA,CP,CV/SIMCOST/AF,Z
     1Z,ZZZ/IN/TIN,HIN,GA,DEPTH/OUT/XM,THETA,TOUT,HOUT
      GOTO(10,210),IGO
C OUTLET MC MODEL.
10    HC=.69*GA**.49
      DF=HC*CON1
      DM=1.-(HC*SA*DEPTH/GA/CA/(10.**(-1.531*ALOG10(DF)+5.92)))**((( 3.8
     177-ALOG10(GA))/1.073)*(77.36/(DF+111.9))*(825.95-TIN)/428.5*(1.12*
     2XMCIN+.59)*(432.68-THIN)/384.61)
C OUTLET PRODUCT TEMPERATURE MODEL.
      ATIN=TIN+459.69
      RHIN=RHDBHA(ATIN,HIN)
      XME=EMC(RHIN,TIN)
      XM=DM*(XMCIN-XME)+XME
      THETA=150.89085*DEPTH**(-0.13458986)
      IF(THIN.LE.75.)GOTO20
      SLOPE=0.46836638*DEPTH**(-0.28550254)
      GOTO40
   20 SLOPE=0.46493212*DEPTH**(-0.26856385)
   40 DELTEMP=SLOPE*(THIN-75.)
      THETA=THETA+DELTEMP
      IF(TIN.LE.350.)GOTO60
      SLOPE=0.19723675*DEPTH**(-0.3155996)
      GOTO80
   60 SLOPE=0.25301944*DEPTH**(-0.26230775)
   80 DELTEMP=SLOPE*(TIN-350.)
      THETA=THETA+DELTEMP
      IF(XMCIN.LE.0.30)GOTO100
      DELTEMP=-128.5*(XMCIN-0.30)
      GO TO 120
  100 DELTEMP=-183.*(XMCIN-0.30)
  120 THETA=THETA+DELTEMP
      IF(HIN.LE.0.004)GOTO140
      DELTEMP=157.2727*(HIN-0.004)
      GOTO160
  140 DELTEMP=205.*(HIN-0.004)
  160 THETA=THETA+DELTEMP
      IF(AF.GT.150.)GOTO180
      SLOPE=0.43671501*DEPTH**(-0.28665312)
      GOTO200
  180 SLOPE=0.33195071*DEPTH**(-0.16581073)
  200 DELTEMP=SLOPE*(AF-150.)
```

```
      THETA=THETA+DELTEMP
      DELTEMP=-15.713684*DEPTH**(-0.37998875)
      THETA=THETA+DELTEMP
      RETURN
C OUTLET AIR TEMPERATURE MODEL.
210   TOUT=110.2+45.74*EXP(-0.5492*DEPTH)+37.39*(EXP(-3.351*(XMCIN-0.3))-1.)+0.2
     1-1.)+0.2028*EXP(-.1275*DEPTH)*(TIN-350.)+.6535*EXP(-0.1348*DEPTH)*(THIN-75
     2(THIN-75.)+.4050*EXP(-0.1575*DEPTH)*(AF-150.)
C OUTLET HUMIDITY MODEL.
      TEMPORY=0.0121*ALOG(DEPTH/0.178)+(0.0055*DEPTH+.41)*(XMCIN-0.3)+(-
     14.496E-6*(DEPTH-4.631)**2+1.145E-4)*(TIN-350.)+1.208E-4*DEPTH**0.5
     2431*(THIN-75.)+0.9006(HIN-.0035)
      IF(AF.GT.150.)GOTO220
      HOUT=TEMPORY+.378E-4*(AF-150.)
      RETURN
220   HOUT=TEMPORY+(-4.15E-6*(DEPTH+1.155)**2+9.594E-5)*(AF-150.)
      RETURN
      END




      FUNCTION EMC(RH,T)
C  FUNCTION EMC IS USED TO DETERMINE EQUILIBRIUM MOISTURE CONTENT AS A
C  FUNCTION OF TEMPERATURE AND RH.
      IF(RH-.50)300,300,309
300   F1=-.0003922*T+.13$F2=-.0004353*T+.1328$F3=-.0005359*T+.1646$S1=13.
     1838*(-9.*F1+6.*F2-F3)$S2=13.838*(4.*F3-9.*F2+6.*F1)$B=RH-.17
      IF(B)301,301,302
  301 EMC =(S1*RH*RH*RH/1.02+(F1/.17-S1*.02833)*RH)$RETURN
  302 IF(RH-.34)303,303,304
  303 A = .34-RH
      EMC =(S1*A*A*A/1.02+S2*B*B*B/1.02+(F2/.17-S2*.02833)*B+(F1/.17-S1*
     1.02833) *A)$RETURN
  304 A=.51-RH
      EMC = S2*A*A*A/1.02+(F3/.17)*(RH-.34)+(F2/.17-S2*.028333)*A$RETURN
309   F0=-.0005373*T+.1624$F1=-.0007075*T+.2075$F2=-.0007449*T+.2532
      F3=-.001071*T+.3931$S1=13.838*(4.*F0-9.*F1+6.*F2-F3)
      S2 = 13.838 *(4.*F3-9.*F2+6.*F1-F0)$B=RH-.66$IF(B)305,305,306
  305 A=RH-.49
      EMC=S1*A*A*A/1.02+(F1/.17-S1*.02833)*A+(F0/.17)*(.66-RH)$RETURN
  306 IF(RH-.83)307,307,308
  307 A=.83-RH
      EMC=S1*A*A*A/1.02+S2*B*B*B/1.02+(F2/.17-S2*.02833)*B+(F1/.17-S1*
     1.028333)*A $RETURN
  308 A=1.0-RH
      EMC=S2*A*A*A/1.02+(F3/.17)*(RH-.83)+(F2/.17-S2*.028333)*A$RETURN
      END




      FUNCTION HADBRH(DB,RH)
C HADBRH IS USED TO FIND ABSOLUTE HUMIDITY,GIVEN THE DRY BULB TEMP AND RH.
      HADBRH=HAPV(RH*PSDB(DB))    $RETURN
      END
```

```
      FUNCTION HAPV(PV)
C HAPV IS USED TO FIND ABSOLUTE HUMIDITY, GIVEN THE VAPOR PRESSURE,
      COMMON/PRESS/PATM
      HAPV=,6219*PV/(PATM-PV)  $ RETURN
      END
```

```
      FUNCTION HLDB (DB)
C HLDB IS USED TO FIND LATENT HEAT, GIVEN THE DRY BULB TEMP,
      IF(DB-491,69) 1,2,2
    1 HLDB=1220,884-,05077*(DB-459,69)$RETURN
    2 IF(DB=609,69) 3,4,4
    3 HLDB=1075,8965-,56983*(DB-459.69)$RETURN
    4 HLDB=SQRT(1354673,214-,9125275587*DB*DB)$RETURN
      END
```

```
      FUNCTION HOWDEEP(D)
C HOWDEEP IS USED IN A 1-D SEARCH TO FIND THE BED DEPTH AT WHICH THE CURRENTLY
C SPECIFIED HUMIDITY OCCURS,
      COMMON/OUTCOOL/HOUT,TOUT,THOUT,XMOUT/AITCH/HNOW/INCOOL/PRODM,THETA
     1,GA,DEPTH
      DEPTH=D$CALL COOLSIM(2)
      HOWDEEP=HOUT-HNOW
      RETURN
      END
```

```
      SUBROUTINE IENCODE(INDY,IND,MAX,NDIM)
C IENCODE IS USED TO COMBINE THREE INDICES INTO A SINGLE INDEX;
      DIMENSION IND(3),MAX(3)
      INDY=IND(1)
      MAXSIZE=MAX(1)
      DO 30 N=2, NDIM
      INDY=IND(N)*MAXSIZE-MAXSIZE+INDY
   30 MAXSIZE=MAXSIZE*MAX(N)
      RETURN
      END
```

```
      SUBROUTINE INTERP(VALUE,NDIM,INDEX,ICECREM,NSTORE,DUMMY)
C INTERP IS USED TO SET UP INTERPOLATION IN FOUR DIMENSIONS OR LESS;  FOR THIS
C PROGRAM, TWO DIMENSIONS ARE INTERPOLATED,
      DIMENSION INDEX(3),ICECREM(3),LMNOP(3)
      SUM=0,0
      DO 175 KIP=1,NDIM
      DO 126 N=1,NDIM
126   LMNOP(N)=0
```

```
      K=1
      MATSTOP=MRKSTOP=LUKSTOP=0
      DO 129 JJ=1,NDIM
      IF(JJ-ICECREM(K))129,127,129
 127  IF(JJ-KIP)128,175,128
 128  LMNOP(JJ)=INDEX(JJ)
      IF(K-NSTORE)1285,130,1285
1285  K=K+1
 129  CONTINUE
 130  DO 174  I=1,2
 131  IF(I-2)132,135,132
 132  MATTHEW=0
      K=1
 1325 IF(LMNOP(K)-INDEX(K))133,137,133
 133  IF(K-KIP)134,137,134
 134  MATTHEW=K
 135  LMNOP(MATTHEW)=INDEX(MATTHEW)-I+2
      IF(MATSTOP-1)136,1355,136
1355  CALL DUMMY(AVERAGE,KIP,LMNOP,INDEX)
      IF(AVERAGE-10.E23)1357,174,174
 1357 IF(AVERAGE-0.0)1745,174,1745
 136  IF(I-2)137,142,137
137   IF(K-NDIM)139,138,139
 138  MATSTOP=1
      CALL DUMMY(AVERAGE,KIP,LMNOP,INDEX)
      IF(AVERAGE-10.E23)1385,174,174
 1385 IF(AVERAGE-0.0)1745,174,1745
 139  IF(MATTHEW-0)141,140,141
 140  K=K+1
      GO TO 1325
 141  K=K+1
 142  DO 173  II=1,2
 143  IF(I-2)144,152,144
 144  IF(II-2)145,152,145
 145  MARK=0
 146  IF(LMNOP(K)-INDEX(K))150,147,150
147   IF(K-NDIM)148,149,148
 148  K=K+1
      GO TO 146
 149  MATSTOP=1
      GO TO 174
 150  IF(K-KIP)151,154,151
 151  MARK=K
 152  LMNOP(MARK)=INDEX(MARK)-II+2
      IF(MRKSTOP-1)153,1525,153
1525  CALL DUMMY(AVERAGE,KIP,LMNOP,INDEX)
      IF(AVERAGE-10.E23)1527,173,173
 1527  IF(AVERAGE-0.0)1745,173,1745
 153  IF(II-2)154,159,154
154   IF(K-NDIM)156,155,156
 155  MRKSTOP=1
      CALL DUMMY(AVERAGE,KIP,LMNOP,INDEX)
      IF(AVERAGE-10.E23)1555,173,173
 1555 IF(AVERAGE-0.0)1745,173,1745
 156  IF(MARK-0)158,157,158
 157  K=K+1
      GO TO 146
 158  K=K+1
 159  DO 172  III=1,2
 160  IF(I-2)1601,165,1601
```

```
 1601 IF(II-2)1602,165,1602
 1602 IF(III-2)1603,165,1603
 1603 LUKE=0
 1605 IF(LMNOP(K)-INDEX(K))164,161,164
161    IF(K-NDIM)162,163,162
 162   K=K+1
       GO TO 1605
 163   MRKSTOP=1
       GO TO 173
 164   IF(K-KIP)1645,166,1645
 1645 LUKE=K
 165   LMNOP(LUKE)=INDEX(LUKE)-III+2
       IF(LUKSTOP-1)166,169,166
166    IF(K-NDIM)167,168,167
 167   K=K+1
       GO TO 1605
168    LUKSTOP=1
169    CALL DUMMY(AVERAGE,KIP,LMNOP,INDEX)
       IF(AVERAGE-10.E23)171,172,172
 171   IF(AVERAGE-0.0)1745,172,1745
 172   CONTINUE
 173   CONTINUE
 174   CONTINUE
       IF(AVERAGE-0.0)1742,1741,1742
1741   VALUE=0.0
       RETURN
1742   VALUE=0.0
       RETURN
1745   SUM=SUM+AVERAGE
175    CONTINUE
       VALUE=SUM/FLOAT(NDIM-NSTORE)
176    RETURN
       END



       FUNCTION OFFDIAG(Q)
C   FUNCTION OFFDIAG IS USED BY OPTWHIZ TO DETERMINE ISOSTERE SLOPE.
       COMMON/FINAL/XMFINAL/ENOT/TONE
       CALL QUICK(AVEMCDB,TONE,Q)$OFFDIAG=XMFINAL-AVEMCDB$RETURN
       END



       SUBROUTINE OPTWHIZ
C   SUBROUTINE OPTWHIZ CONTROLS 2-DIMENSIONAL OPTIMIZATION
       COMMON/ICHECK/IFLAG/DT/DEPTH,TIME/FINAL/XMFINAL/QT/QUE,TEE/STUFF/D
     1IAGSLP,QMIN,TMIN,QMAX,TMAX/CT/COSTLFT,TOFF/BOUND/XMBND,BNDM/ENOT/
     2TONE/ECON/FAMETA,EMETA,THERETA,ELPRICE,FUPRICE,TAMB,FACTOR,ELCOST,
     4FULCOST
       EXTERNAL COVER,SIDE,DIAG,OFFDIAG,COSTRT$KOUNT=0$DIFF=100.$EPS=1.E-
     17$EPS2=1.E-4
C      SURVEY MOISTURE CONTENTS AT EXTREME POINTS
       CALL QUICK(AVEMCDB,TMIN,QMIN)$TMNQMN=AVEMCDB$PRINT 20001,TMIN,QMIN
     1,AVEMCDB
       CALL QUICK(AVEMCDB,TMAX,QMIN)$TMXQMN=AVEMCDB$PRINT 20001,TMAX,QMIN
```

```
      1,AVEMCDB$CALL QUICK(AVEMCDB,TMIN,QMAX)$TMNQMX=AVEMCDB$PRINT 20001,
      2TMIN,QMAX,AVEMCDB$CALL QUICK(AVEMCDB,TMAX,QMAX)$TMXQMX=AVEMCDB
C      CHECK LH,TOP,RH,BOTTOM BOUNDS SEQUENTIALLY FOR FINAL AVEMC ISOSTER
       PRINT 20001,TMAX,QMAX,AVEMCDB$IF(XMFINAL.GT.TMNQMN.OR.XMFINAL.LT.T
      1MNQMX)GO TO 20 $GUESS1=QMIN$GUESS2=QMAX$TEE=TMIN$CALL ZEROINA(GUES
      2S1,GUESS2,EPS,SIDE)$QZERO=(GUESS1+GUESS2)/2.$CALL QUICK(AVEMCDB,TM
      3IN,QZERO)$KOUNT=KOUNT+1$DELT=.1
12     TEE=TMIN+DELT$GUESS1=QMIN$GUESS2=QMAX$CALL ZEROINA(GUESS1,GUESS2,E
      1PS,SIDE)$IF(IFLAG.NE.1)GOTO 14$DELT=DELT/2.$GOTO 12
14     QONE=(GUESS1+GUESS2)/2.$SLOPEM=(QONE-QZERO)/DELT$DELT=.1$CALLCOSTF
      1UN(COSTLFT,TMIN,QZERO)
155    TOFF=TMIN+DELT$GUESS1=QMAX$GUESS2=QMIN$CALL ZEROINA(GUESS1,GUESS2,
      1EPS,COSTRT)$IF(IFLAG.NE.1)GOTO 17$DELT=DELT/2.$GOTO 155
17     QRIGHT=(GUESS1+GUESS2)/2.$SLOPEC=(QRIGHT-QZERO)/DELT
       PRINT 20002,TMIN,QZERO
       PRINT 30002,SLOPEM,SLOPEC,COSTLFT
       IF(SLOPEC.GT.SLOPEM)GO TO 20$IF(BNDM.LE.XMBND)GO TO 195$PRINT 2001
      10,BNDM,XMBND
195    COSTPBU=COSTLFT/DEPTH*1.25$PRINT 20004,COSTLFT,COSTPBU,TMIN,QZERO
       GO TO 70
20     IF(XMFINAL.GT.TMNQMX.OR.XMFINAL.LT.TMXQMX)GO TO 70$GUESS1=TMIN$GUE
      1SS2=TMAX$QUE=QMAX$CALL ZEROINA(GUESS1,GUESS2,EPS,COVER)$TZERO=(GUE
      2SS1+GUESS2)/2.$CALL QUICK(AVEMCDB,TZERO,QUE)$KOUNT=KOUNT+1$DELQ=.1
31     QUE=QMAX-DELQ$GUESS1=TMIN$GUESS2=TMAX$CALL ZEROINA(GUESS1,GUESS2,E
      1PS,COVER)$IF(IFLAG.NE.1)GOTO 33$DELQ=DELQ/2.$GOTO 31
33     TONE=(GUESS1+GUESS2)/2.$SLOPEM=DELQ/(TZERO-TONE)$DELT=.1$CALL COST
      1FUN(COSTLFT,TZERO,QMAX)
334    TOFF=TZERO+DELT$GUESS1=QMAX$GUESS2=QMIN$CALLZEROINA(GUESS1,GUESS2,
      1EPS,COSTRT)$IF(IFLAG.NE.1)GOTO337$DELT=DELT/2.$GOTO 334
337    QRIGHT=(GUESS1+GUESS2)/2.$SLOPEC=(QRIGHT-QMAX)/DELT$PRINT 20002,TZ
      1ERO,QMAX$PRINT30002,SLOPEM,SLOPEC,COSTLFT$IF(SLOPEC.GT.SLOPEM)GO
      1TO 70
       IF(BNDM.LE.XMBND)GO TO 60$PRINT 20010,BNDM,XMBND$STOP
60     COSTPBU=COSTLFT/DEPTH*1.25$PRINT 20004,COSTLFT,COSTPBU,TZERO,QMAX
70     IF(XMFINAL.GT.TMXQMN.OR.XMFINAL.LT.TMXQMX)GO TO 130$GUESS1=QMIN
       GUESS2=QMAX$TEE=TMAX$CALL ZEROINA(GUESS1,GUESS2,EPS,SIDE)$QZERO=
      1(GUESS1+GUESS2)/2.$CALL QUICK(AVEMCDB,TEE,QZERO)$KOUNT=KOUNT+1
       GUESS1=QMIN$GUESS2=QMAX$DELT=.1
81     TEE=TMAX-DELT $CALL ZEROINA(GUESS1,GUESS2,EPS,SIDE)$IF(IFLAG.NE.1)
      1GO TO 83 $ DELT=DELT/2. $ GO TO 81
83     QONE=(GUESS1+GUESS2)/2.$SLOPEM=(QZERO-QONE)/DELT$DELT=.1
       CALL COSTFUN(COSTLFT,TMAX,QZERO)
84     TOFF=TMAX+DELT$GUESS1=QMAX$GUESS2=QMIN$CALL ZEROINA(GUESS1,GUESS2,
      1EPS,COSTRT)$IF(IFLAG.NE.1)GOTO 86$DELT=DELT/2.$GOTO 84
86     QRIGHT=(GUESS1+GUESS2)/2.$SLOPEC=(QRIGHT-QZERO)/DELT
       PRINT 20002,TMAX,QZERO
       PRINT 30002,SLOPEM,SLOPEC,COSTLFT
       IF(SLOPEC.LT.SLOPEM)GO TO 130$IF(BNDM.LE.XMBND)GO TO 120
       PRINT 20003,BNDM,XMBND$STOP
120    COSTPBU=COSTLFT/DEPTH*1.25
       PRINT 20004,COSTLFT,COSTPBU,TMAX,QZERO$STOP
130    IF(XMFINAL.GT.TMNQMN.OR.XMFINAL.LT.TMXQMN)GO TO 190$GUESS1=TMIN
       GUESS2=TMAX$QUE=QMIN$CALL ZEROINA(GUESS1,GUESS2,EPS,COVER)$TZERO=
      1(GUESS1+GUESS2)/2.$CALL QUICK(AVEMCDB,TZERO,QUE)$KOUNT=KOUNT+1
       DELQ=.1  $PRINT 20002,TZERO,QMIN
141    QUE=QMIN+DELQ$GUESS1=TMIN$GUESS2=TMAX$CALL ZEROINA(GUESS1,GUESS2,
      1EPS,COVER)$IF(IFLAG.NE.1)GOTO 143$DELQ=DELQ/2.$GOTO 141
143    TONE=(GUESS1+GUESS2)/2.$SLOPEM=DELQ/(TONE-TZERO)$DELT=.1
       CALL COSTFUN(COSTLFT,TZERO,QMIN)
144    TOFF=TZERO-DELT$GUESS1=QMAX$GUESS2=QMIN$CALL ZEROINA(GUESS1,GUESS2
```

```
     1,EPS,COSTRT)$IF(IFLAG.NE.1)GOTO 146$DELT=DELT/2.$GOTO144
146   QLEFT=(GUESS1+GUESS2)/2.$SLOPEC=(QMIN-QLEFT)/DELT
      PRINT 30002,SLOPEM,SLOPEC,COSTLFT$IF(SLOPEC.LT.                     SLO
     1PEM)GO TO 190$IF(BNDM.LE.XMBND)GO TO 180$PRINT 20003,BNDM,XMBND
      STOP
180   COSTPBU=COSTLFT/DEPTH*1.25$PRINT 20004,COSTLFT,COSTPBU,TZERO,QMIN
      STOP
190   IF(KOUNT.EQ.0)200,210
200   PRINT 20005
      STOP
210   IF(KOUNT.NE.2)220,230
220   PRINT 30003$STOP
C       THERE IS AN INTERIOR OPTIMUM, BEGIN SEARCH FOR IT.
230   GUESS1=TMIN$GUESS2=TMAX$DIAGSLP=(QMAX-QMIN)/(TMAX-TMIN)
      CALL ZEROINA(GUESS1,GUESS2,EPS,DIAG)$IF(IFLAG.NE.1)GO TO 2302$STOP
2302  TZERO=(GUESS1+GUESS2)/2.$QZERO=DIAGSLP*(TZERO-TMIN)+QMIN$DELT=.1
      CALL QUICK(AVEMCDB,TZERO,QZERO)$IF(BNDM.LE.XMBND)GOTO 231
      QMIN=QZERO$TMAX=TZERO$GOTO230
231   GUESS1=QMIN$GUESS2=QMAX$TONE=TZERO +DELT$CALL ZEROINA(GUESS1,
     1GUESS2,EPS,OFFDIAG)
      IF(IFLAG.NE.1)GO TO 233$DELT=DELT/2.$GO TO 231
233   QONE=(GUESS1+GUESS2)/2.$SLOPEM=(QONE-QZERO)/(TONE-TZERO)$CALL COST
     1FUN(COSTLFT,TZERO,QZERO)$DELT=.1
234   TOFF=TZERO+DELT$GUESS1=QMAX$GUESS2=QMIN$CALL ZEROINA(GUESS1,GUESS2
     1,EPS,COSTRT)$IF(IFLAG.NE.1)GOTO 237$DELT=DELT/2.$GOTO 234
237   QRIGHT=(GUESS1+GUESS2)/2.$SLOPEC=(QRIGHT-QZERO)/DELT
      IF(SLOPEC-SLOPEM)272,275,270
270   PRINT 20008,TZERO,QZERO,SLOPEM,SLOPEC,COSTLFT
      TMIN=TZERO$QMAX=QZERO$GO TO 274
272   PRINT 20009,TZERO,QZERO,SLOPEM,SLOPEC,COSTLFT
      TMAX=TZERO$QMIN=QZERO
274   CURDIFF=ABS(SLOPEC-SLOPEM)$IF(ABS(CURDIFF-DIFF).GT.EPS2)GO TO 278
275   COSTPBU=COSTLFT/DEPTH*1.25$PRINT 20011,TZERO,QZERO
      PRINT 30011,COSTLFT,ELCOST,FULCOST,COSTPBU
      STOP
278   DIFF=CURDIFF$GOTO230
20001 FORMAT(3X,*INLET AIRTEMP,DEGF = *,F8.4,3X,*AIRFLOW RATE,CFM/FT2 =*
     1,F7.4,3X,*FINAL AVE.MC.,DEC.D.B. =*,F7.3)
20002 FORMAT(3X,*FINAL AVE MC ISOSTERE CROSSES BND AT INLET TEMP = *,F8.
     14,3X,*AIRFLOW RATE =*,F7.4)
20003 FORMAT(3X,*THE MC OF THE TOP LAYER,*,F7.4,2X,*HAS NOT REACHED THE
     1MAX. PERMISSIBLE MC,*,F7.4)
20004 FORMAT(3X,*AN OPTIMUM COST,*,F7.4,2X,*CTS/FT2 OR*,F7.4,2X,*CTS/BU
     1OCCURS AT INLET TEMP=*,F8.4,2X,*AIRFLOW RATE =*,F8.4,*CFM/FT2*)
20005 FORMAT(3X,*IT IS IMPOSSIBLE TO REACH THE DESIRED MC WITHIN THE STA
     1TED BOUNDS*)
20008 FORMAT(3X,*THE OPTIMUM OCCURS AT T.GT.*,F8.4,*,Q.LT.*,F8.4,*,MC SL
     1OPE =*,F8.4,*,COST SLOPE =*,F8.4,*,COST =*,F8.4)
20009 FORMAT(3X,*THE OPTIMUM OCCURS AT T.LT.*,F8.4,*,Q.GT.*,F8.4,*,MC SL
     1OPE =*,F8.4,*,COST SLOPE =*,F8.4,*,COST = *,F8.4)
20010 FORMAT(3X,*THE MC OF THE TOP LAYER,*,F7.4,2X,*CANNOT REACH THE MAX
     1, PERMISSIBLE MC,*,F7.4,*WITHIN THE STATED BOUNDS*)
20011 FORMAT(3X,*THE OPTIMUM OCCURS AT T =*,F8.4,*,Q =*,F8.4,*CFM/FT2.*)
30002 FORMAT(3X,*ISOSTERE SLOPE = *,F7.4,2X,*ISOCOST SLOPE =*,F7.4,2X,
     1*ISOCOST VALUE,CTS = *,F7.4)
30003 FORMAT(3X,*THE FINAL MC ISOSTERE CROSSES OTHER THAN 2 BOUNDS-.THE
     1METHOD FAILS*)
30011 FORMAT(3X,*TOTAL COST/FT2= *,F8.4,*  ELECT.COMPONENT= *,F8.4,*,  F
     1UEL COMPONENT *,F8.4 , * COST/BU =*,F8.4, *CTS*)
      END
```

```
      FUNCTION PSDB (DB)
C PSDB IS USED TO FIND THE SATURATION VAPOR PRESSURE, GIVEN THE DRY BULB TEMP.
      DATA R,A,B,C,D,E,F,G/,3206182232E04,-.2740552558361426E05,.54189607
     A6328951E02,-.4513788384112655E-1,.2153211916363354E-4,-.462026656819
     B9862E-8,.2416127209874E01,.1215465167060555E-2/
      IF(DB-491,69) 1,2,2
    1 PSDB=EXP(23,3924-11286,6489/DB-.46057*ALOG(DB))$RETURN
    2 PSDB=R*EXP((A+DB*(B+DB*(C+DB*(D+DB*E))))/(DB*(F-G*DB)))$RETURN
      END



      FUNCTION PVDBWB(DB,WB)
C PVDBWB IS USED TO FIND THE VAPOR PRESSURE, GIVEN THE DRY BULB AND WET BULB
C TEMPS.
      COMMON/PRESS/PATM
      A=PSDB(WB) $ B=.62194*HLDB(WB)*PATM $ C=.2405*(A-PATM)*(WB-DB)
      PVDBWB=(A*B-C*PATM)/(B+.15577*C) $RETURN
      END



      FUNCTION PVHA (HA)
C PVHA IS USED TO FIND THE VAPOR PRESSURE, GIVEN THE ABS. HUMIDITY.
      COMMON/PRESS/PATM
      PVHA=HA*PATM/(.6219+HA)  $RETURN
      END



      FUNCTION PVTG(TRYPV)
C PVTG IS USED TO LOCATE THE VAPOR PRESSURE AT WHICH EQUILIBRIUM MC OF THE GRAIN
C OCCURS.
      COMMON/PRESS/PATM/AMB/PSWB,HPFG,ATWB,XMZERO
      TEMP=ATWB-(PSWB-TRYPV)*HPFG*2,586029106029/(PSWB-PATM)/(1.+0.15577
     1*TRYPV/PATM)$TRYRH=TRYPV/PSDB(TEMP)$TEMPF=TEMP-459.69
      PVTG=XMZERO-EMC(TRYRH,TEMPF)$RETURN
      END



      SUBROUTINE QUICK(AVEMCDB,TINLET,Q)
C SUBROUTINE QUICK SIMULATES THE BATCH-IN-BIN DRYING PROCESS
      DIMENSION XMCDB(21)
      COMMON/PRESS/PATM/ICHECK/IFLAG/DT/DEPTH,TIME/BOUND/XMBND,BNUM/AMB/
     1PSWB,HPFG,ATWB,XMZERO/OTHER/DELX,PV
      EXTERNAL PVTG
      ATINLET=TINLET+459.69$PSAT=PSDB(ATINLET)$RH=PV/PSAT$HABS=HADBRH(AT
     1INLET,RH)$ATWB=WBDBHAS(ATINLET,HABS,495.,550.,.01)
      TWB=ATWB-459.69 $PSWB=PSDB(ATWB) $ HPFG=HLDB(ATWB)
      IF(EMC(1.,TWB).LE.XMZERO)1,2
1     TG=TWB   $   GO TO 3
```

```
2     GUESS1=.1$GUESS2=PSWB$CALL ZEROIN(GUESS1,GUESS2,1,E-8,PVTG)
      IF(IFLAG.EQ.1)25,26
25    PRINT 10001
      STOP
26    PVATTG=(GUESS1+GUESS2)/2.
      TG=TWB-(PSWB-PVATTG)*HPFG*2.586029106029/(PSWB-PATM)/(1.+0.15577
     1*PVATTG/PATM)
3     XME=EMC(RH,TINLET)
      DELM=XMZERO-XME $V=(1094.-0.57*TINLET)*(1.+4.35*EXP(-1412.5*DELM))
      VONE=V+(TINLET-70.)*(1.+0.2/DELM)
      XK=-.04564539+.0055836759*PSAT**.33804141*Q**.94061342
      Y=XK*TIME $ X=0.0 $ DO 65  KK=1,21
      DD=VONE*XK*DELM*X/Q/(TINLET-TG)*22.19316276535
      RATIO=EXP(DD)/(EXP(DD)+EXP(Y)-1.) $ XMCDB(KK)=RATIO*DELM+XME
65    X=X+DELX$BNDM=XMCDB(21)$SUM=(XMCDB(1)+XMCDB(21))/2.$DO850LL=2,20
850   SUM=SUM+XMCDB(LL)$AVEMCDB=SUM*DELX/DEPTH$RETURN
10001 FORMAT(3X,*THE MODEL CANNOT HANDLE THIS CASE*)
      END



      FUNCTION RHPSPV(D1,D2)
C RHPSPV IS USED TO FIND THE RH, GIVEN THE PARTIAL AND SATURATION VAPOR PRESSURE
      A=D1 $ B=D2 $ GO TO 1
      ENTRY RHDBHA
      A=PSDB(D1) $ B=PVHA(D2)
    1 RHPSPV=B/A
      RETURN
      END



      FUNCTION SIDE(Q)
C  FUNCTION SIDE IS USED BY OPTWHIZ TO FIND THE INTERSECTION OF THE
C  ISOSTERE WITH A TEMPERATURE BOUND,
      COMMON/FINAL/XMFINAL/QT/QUE,TEE
      CALL QUICK(AVEMCDB,TEE,Q)$SIDE=AVEMCDB-XMFINAL$RETURN
      END



      FUNCTION VSDBHA (DB,HA)
C VSDBHA IS USED TO FIND THE SPECIFIC VOLUME, GIVEN THE DRYBULB TEMP AND ABS,
C HUMIDITY,
      COMMON /PRESS/PATM
      VSDBHA=53.35*DB*(.6219+HA)/144./.6219/PATM$RETURN
      END
```

```
      FUNCTION WBDBHAS(DB,HA,G1,G2,EPS)
C WBDBHAS IS USED TO FIND THE WET BULB TEMP., GIVEN THE DRY BULB TEMP. AND ABS.
C HUMIDITY.
      EXTERNAL WBL
      COMMON /SPECIAL/PV,TB
      A=G1$B=G2$TB=DB$PV=PVHA(HA)$CALL ZEROIN(A,B,EPS,WBL)$WBDBHAS=(A+B
     1)/2.$RETURN
      END



      FUNCTION WBL(TWB)
C WBL IS THE FUNCTION USED TO DESCRIBE THE WET BULB LINE.
      COMMON/PRESS/PATM/SPECIAL/PV,DB
      PWB=PSDB(TWB)
      WBL=TWB-DB-(PWB-PV)/(.2405*(PWB-PATM)*(1.+.15577*PV/PATM))*(.62194
     1*HLDB(TWB)) $RETURN
      END



      SUBROUTINE ZEROIN(A,B,EPS,FUNC)
C  SUBROUTINE ZEROIN IS THE 1-DIMENSIONAL ZERO-FINDING ROUTINE USED BY
C BOTH PARTS OF THE THESIS.
      COMMON/ICHECK/IFLAG
      REAL I,M
      FA=FUNC(A) $ FB=FUNC(B) $ FC=FA $ C=A $ IFLAG=0
      IF(SIGN(1.,FB).EQ.SIGN(1.,FC))400,1
400   IFLAG=1          $ RETURN
1     IF(ABS(FC)-ABS(FB)) 2,3,3
2     C=B $ B=A $ A=C $ FC=FB $ FB=FA $ FA=FC
3     IF(ABS(C-B)-2.*EPS) 12,12,4
4     I=(B-A)*FB/(FB-FA) $ J=LEGVAR(I) $ M=(C+B)/2. $ IF(J-0) 7,5,7
5     I=-I+B $ CHINT=(M-I)*(M-I) $ IF(CHINT) 8,8,7
7     I=M
8     IF(ABS(B-I)-EPS) 9,10,10
9     I=SIGN(1.,(C-B))*EPS+B
10    A=B $ B=I $ FA=FB $ FB=FUNC(B)
      IF(SIGN(1.,FB)-SIGN(1.,FC)) 1,11,1
11    C=A$ FC=FA $ GO TO 1
12    A=(C+B)/2. $ FA=FUNC(A)
      IF(SIGN(1.,FA).EQ.SIGN(1.,FB)) B=C      $ RETURN
      END



      SUBROUTINE ZEROINA (A,B,EPS,FUNC)
C  SUBROUTINE ZEROINA IS THE 1-DIMENSIONAL ZERO-FINDING ROUTINE USED BY
C  OPTWHIZ AND CONTROL.
      COMMON/ICHECK/IFLAG
      REAL I,M
      FA=FUNC(A) $ FB=FUNC(B) $ FC=FA $ C=A $ IFLAG=0
      IF(SIGN(1.,FB).EQ.SIGN(1.,FC))400,1
400   IFLAG=1          $ RETURN
```

```
1       IF(ABS(FC)-ABS(FB)) 2,3,3
2       C=B $ B=A $ A=C $ FC=FB $ FB=FA $ FA=FC
3       IF(ABS(C-B)-2.*EPS) 12,12,4
4       I=(B-A)*FB/(FB-FA) $ J=LEGVAR(I) $ M=(C+B)/2. $ IF(J-0) 7,5,7
5       I=-I+B $ CHINT=(B-I)*(M-I) $ IF(CHINT) 8,8,7
7       I=M
8       IF(ABS(B-I)-EPS) 9,10,10
9       I=SIGN(1.,(C-B))*EPS+B
10      A=B $ B=I $ FA=FB $ FB=FUNC(B)
        IF(SIGN(1.,FB)-SIGN(1.,FC)) 1,11,1
11      C=A$ FC=FA $ GO TO 1
12      A=(C+B)/2. $ FA=FUNC(A)
        IF(SIGN(1.,FA).EQ.SIGN(1.,FB)) B=C        $ RETURN
        END
```