PERFORMANCE EVALUATION OF A COUNTERFLOW PARTICLE-TO-PARTICLE HEAT EXCHANGER

By

HAYWARD RAY SIMONTON, JR

Bachelor of Science in Agricultural Engineering
University of Georgia
Athens, Georgia
1979

Master of Science
University of Georgia
Athens, Georgia
1981

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 1986

PERFORMANCE EVALUATION OF A COUNTERFLOW PARTICLE-TO-PARTICLE HEAT EXCHANGER

Thesis Approved:

Marvin L Stone
Thesis Adviser

Glenn Kranzler

C. Haan

James R. Rowland

Gerald H. Brusewitz

Norman N. Durham
Dean of the Graduate College

ACKNOWLEDGEMENTS

Undoubtedly no dissertation is solely the result of one person's work and this study, on the performance of counterflow solid particle-to-particle heat exchange, is no exception. I would first like to thank my adviser, Dr. Marvin Stone, for the ingenuity to come up with the initial idea for the project and for the patience to contend with a stubborn and hard-headed person like myself. Each of my committee members performed a different and necessary role in my PhD program and I would like to thank them also. Members of my committee other than Dr. Stone were Dr. Tom Haan, Dr. Gerald Brusewitz, Dr. Glenn Kranzler, and Dr. James Rowland. Special thanks go to Dr. Rowland for remaining on the committee after leaving Oklahoma State to accept the Department Head position at the University of Kansas.

Since the construction of a prototype was an integral portion of the project, a great deal of work was done at the Agricultural Engineering Laboratory. I would like to thank the following people for their large part in the success of this study: Mark Marston, for helping in whatever was required; Robert Harrington, for construction of the manually rotating test stand; Van Swift, for assisting with the materials handling; Clifford Riley, for the huge

task of construction of the prototype; Mark Appleman, for the initial "startup" of the CMS computer, for handling the software bugs which appeared periodically, and for designing/wiring the rear computer interface panel; Bruce Lambert, for development and calibration of the many electronic interface circuits which were required; and Norvil Cole, for managing whatever I needed whenever I needed it. I would like to thank Richard Greenwell, an undergraduate and master siever, for his help on the prototype development, testing, and maintenance, along with data entry on the computer. Thanks also to Missie Murnan for typing numerous letters, papers, etc.

Friends and family always play a large role in any long-term effort such as this. I feel I have made many friends during my stay here and I appreciate the support of these friendships. With respect to family, pursuing a PhD puts more pressure, other than perhaps winning a trip to the South Pole on the Wheel of Fortune or the adoption of a pet elephant (African), on a marriage than any other situation. I want to deeply thank my wife, Gina, for her love, encouragement, and support throughout our stay in Stillwater.

TABLE OF CONTENTS

Page

APPROVAL ii

ACKNOWLEDGEMENTS iii

LIST OF TABLES vii

LIST OF FIGURES viii

LIST OF SYMBOLS xi

CHAPTER I

INTRODUCTION 1

CHAPTER II

OBJECTIVES 7

CHAPTER III

REVIEW OF LITERATURE 8

Chemical Disinfestation 8

Radiation and Thermal-Related Processes for Disinfestation 15

Potential Use of Solid Heat Transfer Media as a Mechanism for Thermal Disinfestation 32

Adaptive Control as a Means of Improving the Performance of Nonlinear/Time-Varying Systems . . 45

CHAPTER IV

MODELING AND SIMULATION OF A COUNTERFLOW PARTICLE-TO-PARTICLE HEAT EXCHANGER 52

Heat Transfer Testing and Results 52
Heat Exchanger Simulation 61
Simulation Results 67

Page

CHAPTER V

STEADY-STATE PERFORMANCE EVALUATION OF A PROTOTYPE . . 76

Solid Particle Heat Exchanger Design 76
Separation Efficiency and Medium Distribution . . 91
Heat Exchanger Testing and Results 94

CHAPTER VI

PERFORMANCE ENHANCEMENT USING ADAPTIVE CONTROL 109

Determination of Baseline Performance Level . . . 109
Parameter Estimation 122
Adaptive control 135

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS 157

Conclusions 158
Recommendations 160

BIBLIOGRAPHY 161

APPENDIX - SOURCE LISTINGS FOR SOFTWARE 170

CFLOW - Process Simulation 171
SUPER - Open-loop Control of Prototype 173
ADAPT - Complete Prototype Control 177
TUNSECN - Controller Simulation 189

VITA

LIST OF TABLES

Table Page

I. Worldwide Stored-grain Insects 18

II. Heat Transfer Coefficient Results 59

III. Time Constant Results 59

IV. Steady-state Heating Performance 97

V. Heat Recovery Ratios 105

VI. Tuning Results for Proportional Controller . . 118

VII. Tuning Results for Proportional-Integral Controller 118

VIII. Various Performance Measures for the Proportional-Integral Controller 121

IX. Regression Results for Initial Estimates . . . 125

X. Parameter Estimation Results Using a Sixth-Order Predictor 127

XI. Parameter Estimation Results Using a Fourth-Order Predictor 129

XII. Poles and Zero from the Fourth-Order Predictor 129

XIII. Pole-Placement Simulation Results for Prediction of Optimal Startup Performance in the Presence of Limited Input Control Energy . . 146

XIV. Adaptive Controller Tuning Results for the Machine Using a Constant Speed 147

XV. Adaptive Controller Tuning Results for the Machine Using Slow-speed Startup 152

LIST OF FIGURES

Figure Page

1. Temperature Profile of a Concurrent Flow Process . 42
2. Temperature Profile of a Counterflow Process . . . 43
3. Temperature Profile of a Dual-stage Counterflow Process for Heat Recovery 44
4. Heat Transfer Testing Apparatus 54
5. Results of Time Constant Analysis 58
6. Discrete Counterflow Particle-to-Particle Heat Exchange 62
7. Temperature Profile of Two Mixed Solids 63
8. Mass Ratio Effect on Energy Input 68
9. Temperature Profile Within a 5 Cell/Stage Machine 70
10. Temperature Profile Within a 15 Cell/Stage Machine 71
11. Effect of Cell Number on Energy Input 72
12. Effect of Heat Capacity Ratio on Energy Input . . 74
13. Effect of Outlet Wheat Temperature on Energy Input 75
14. Process Flow Diagram of Heat Exchanger 77
15. First Machine Cross-section Depicting Solids Flow 79
16. Mixture Beginning to Flow Out of Holding Area . . 80
17. Mixture Flowing Towards Inclined Screen 81

Figure Page

18. Mixture Flowing Along Screen Resulting in Separation 82

19. Separated Material Flowing in Opposite Directions 83

20. Top and Front Views of Machine 86

21. Temperature Measurement Circuitry 89

22. Heater Control Circuitry 90

23. Energy Measurement Circuitry 92

24. Wheat Temperature Profiles Using Salt and Various Levels of Energy Input 99

25. Wheat Temperature Profiles Using Interprop and Various Levels of Energy Input 100

26. Energy Utilization Performance Using Salt 101

27. Energy Utilization Performance Using Interprop . . 102

28. Effect of Speed on Performance Using Salt and a Constant Energy Input 104

29. Temperature Profile Comparison Between Results from the Prototype Testing and that Predicted by the Simulation 106

30. Step-test to Determine Approximate Process Parameters 115

31. Temperature Response and Control Effort for Optimal Proportional Controller 117

32. Temperature Response and Control Effort for Optimal Proportional-Integral Controller 119

33. Tracking Capability of the Fourth-Order Predictor 131

34. Fluctuation of Output Variable Coefficients in Fourth-Order Predictor 132

35. Fluctuation of Manipulated Variable Coefficients in Fourth-Order Predictor 133

36. Fluctuation of Gain of Predicted Process Transfer Function 134

Figure Page

37. Block Diagram of Closed-Loop System 137

38. Effect of an Unstable Zero on the Manipulated Variable 141

39. Effect of Noise on the Manipulated and Output Variables 142

40. Overshoot and Manipulated Variable Saturation Caused by Poor Pole-placement 145

41. Optimal Manipulated and Output Variable Trajectories for the Adaptive Controller (Constant Speed) 148

42. Optimal Manipulated and Output Variable Trajectories for the Adaptive Controller (Slow-Speed Startup) 151

43. Base Error Gain During Optimal Adaptive Test . . . 153

44. Estimated Process Parameters Used as Gains During Optimal Adaptive Test 154

45. Gain History of First Two Input Coefficients During Optimal Adaptive Test 155

46. Gain History of Second Two Input Coefficients During Optimal Adaptive Test 156

LIST OF SYMBOLS

a1,2	output variable coefficients in predictor
Aw	surface area of wheat sample (m^2)
b0,1	manipulated variable coefficients in predictor
c	coefficient on integral of energy input in performance measure
cp	specific heat of wheat (kJ/kg °C)
d	process deadtime (min)
e	base of natural logarithm
e(t)	discrete-time error used in controller (°C)
eoff	error in predictor (°C)
Gc(z)	transfer function of controller
Gp(z)	transfer function of process
h	heat transfer coefficient (W/m^2 °C)
hrr	heat recovery ratio (W/W)
i/f	initial or final condition
I	identity matrix
Iae	integral of the absolute error
Iein	integral of the energy input
j	output of energy measurement circuit (pulses/rev)
J	control system performance measure
k	inverse of theoretical time constant (1/sec)
K	base error gain in adaptive control law

Kc	proportional gain in conventional controller
ke	inverse of empirical time constant (1/sec)
m	order of numerator dynamics in process transfer function
M	intermediate variable in recursive least squares algorithm
mcm	heat capacity of heat transfer medium (kJ/^{o}C)
mcw	heat capacity of wheat (kJ/^{o}C)
mr	mass flowrate (kg/sec)
n	order of denominator dynamics in process transfer function
p	calibration constant for energy measurement circuit (amps-min/pulse)
p1,2	desired closed-loop poles of the discrete characteristic equation
P	covariance matrix in recursive least squares algorithm
qin	energy input rate to heaters (W)
qls	rate of energy lost by exiting medium (W)
qlw	rate of energy lost by exiting wheat (W)
qs	energy absorption rate by medium in cooling stage (W)
qw	energy absorption rate by wheat in heating stage (W)
r(t)	input to closed-loop system; physically is the required tracking or setpoint temperature (^{o}C)
s	rotational speed of machine (rpm); also used as the Laplace transform variable
t	time (min); used as both continuous and discrete based
td	derivative time in conventional controller (min)

ti	integral time in conventional controller (min)
tp	approximate first-order time constant of the heat exchanger (min)
T	sampling rate of controller (min)
Tm	medium temperature (°C)
Tfilt	output of digital filter for wheat temperature in heated end section (°C)
Tmax	setpoint temperature of wheat (°C)
Tout	exiting wheat temperature (°C)
Tw	wheat temperature (°C)
Twhs	temperature of wheat in heating section (°C)
u(t)	manipulated variable (energy input rate) in controllers and predictor; used as a range (0-2000) or percent of full-range effort (0-100%)
ua	overall heat transfer coefficient (kJ/°C)
v	ac line voltage (Vac)
vp	manipulated variable in positional form of conventional (PID) controller
x	asymptote offset parameter (dimensionless)
y(t)	system output variable (wheat temperature in the heated end section; sometimes referred to as controlled variable) (°C)
z	heat balance efficiency (W/W); also used as the z-transform variable
θ	parameter vector in the recursive least squares algorithm
ϕ	vector of inputs and outputs in the recursive least squares algorithm

CHA

CHAPTER I

INTRODUCTION

Each year a large part of America's grain crop is put into storage. The time span of the storage may range from two weeks to two years. Unfortunately, the longer the storage time, the greater the potential for loss of quality in the grain. The decrease in quality is due to such factors as insect infestation, fungi damage, and moisture-related spoilage. The task of accurately pin-pointing the amount of grain lost during storage, either by volume or by monetary worth, is almost impossible. Figures from four to five and one-half billion dollars (Edwards, 1973; Gillet, 1970) have been used to describe the economic loss due to agricultural pests, in field and in storage, in America alone. Storey and Bulla (1978) estimate that as much as one-third of the world's harvest is lost during storage. The actual value of the figures is not as important as the magnitude of the problem that they suggest.

A considerable portion of the grain damage incurred during storage is due to insect infestation. Current practice, including grain elevator operations, milling operations, and farm operations, all rely extensively on chemical treatment to both protect and disinfest stored crops. Losses due to stored grain insects could run as high as 20% (Gillet, 1970) if insecticides are not used. Pliny the Elder, a Roman living in the first century, recommended the use of arsenic as an insecticide (Edwards, 1973), and countless other chemicals have been used since then. In the past thirty years, many new chemical groups have been found to be extremely effective insecticidal agents; among these are certain organochlorides, organophosphates, and bromides.

Although the correct and safe use of these chemicals has helped maintain grain quality throughout long storage periods, dependence on chemical disinfestation has its own set of problems. Often, chemical treatment is a batch operation requiring holding the grain within a single storage facility for 20 hours (Dermott and Evans, 1978a) or more, which is a problem if the grain is in transit. Most of the chemicals used to disinfest grain are hazardous and thus require a trained operator for application (Johnson and Townsend, 1981; SDS, 1980; USDA, 1969). Insects are biologically able to adapt to changing environmental conditions, and this mechanism sometimes leads to pesticide resistance forming within a certain insect population (Upitis, et al. 1973; Bell, et al. 1977). The most serious problem, however, is the effect of insecticides on man and the environment. Undesirable environmental effects include excessive mortality and reduced reproductive capacity in wildlife, and the general pollution of our natural resources (Geissbuhler, 1978). Man faces short term hazards such as toxicity (SDS, 1980) and long term hazards such as mutagenicity and carcinogenicity (Nielson, 1980; Soderman, 1982; Sontag, 1981). Generally, the long term hazards are the most dangerous, and the most difficult to evalute.

The debate over the use versus the banning of currently used insecticides has "high emotional impact" (Edwards, 1973). The ecologists wish to curtail the use of the chemicals, and the agriculturalists maintain that said usage is essential to the large-scale production of food. A middle ground is required: to control pests by alternative means when possible; and to minimize the use of persistent and potentially harmful pesticides. In "World Food Production Environment - Pesticides" (Geissbuhler, 1978), three causes are given as to the general nature of the problem. These causes were listed as a limited capacity to predict environmental hazards of chemical usage, a low priority given to evaluation of current environmental problems, and a lack of safe and effective alternatives for pest control. It is the third point which this project intends to address: develop and evaluate an efficient and effective chemically independent process to disinfest stored feed and seed grains.

Many alternatives to current chemical usage have been investigated. Some are management techniques, such as using refrigerated or oxygen-deficient storage facilities, or mixing insect growth regulators with the grain. Another category includes microbial control, which utilizes bacteria, fungi, or a virus as insect pathogens. Still other alternatives include the use of radiation. Gamma rays, x-rays, and dielectric heating with microwaves or RF band energy fit into this disinfestaion group. The last category relies on basic heating agents such as particle conduction, air convection, or infrared radiation. Each of the above management techniques/processes has both advantages and disadvantages. Examining the benefits of current chemical controls shows that most are relatively inexpensive, effective in destroying insect populations from egg to adult, and acceptable in usage to farmers, millers, etc. An alternative to chemical treatment, therefore, must at least match if not exceed the above benefits to be accepted and utilized in a widespread manner. However, no single alternative is going to match all the requirements for all the disinfestation needs. Different methods are going to be applied to different situations.

The alternative that appears to offer the most promise in regard to wheat, corn, and flour treatment is the use of heat to destroy all elements of an insect population. Thermal disinfestation has already been shown to be an effective treatment process (Grossman, 1931; Kenagen and Fletcher, 1947; Kirkpatrick and Tilton, 1973; Dermott and Evans, 1978b; Mittal, et al. 1981). The approximate temperature levels required to eradicate an infestation, and the holding times, are known. What is not known is what type of heat exchange process is best suited for the job. Of the processes designed thus far, none has concentrated on efficient utilization of energy. Although a full scale thermal disinfestation plant has been built in Australia, the capital outlay for that design far exceeds what an average elevator operator can be expected to pay.

To be a viable alternative, a thermal disinfestation system must be extremely high in energy efficiency to keep operating costs low and compact to keep down capital investment. The system must offer these capabilities while performing the stated job: destroying the insect population without decreasing grain quality attributes such as germination and baking quality. Solid particle heat exchangers have been shown to have high heat transfer coefficients (Lapp and Manchur, 1974; Raghavan and Harper, 1974) and offer potential to reduce both heat exchanger size and operating costs. Several patents exist on machines to perform solid particle-to-particle heat exhange, including those by Aspegran (1959), Benson (1966), and Bateson and Harper (1973). Each of these machines, along with others built by Lapp and Manchur (1974) and Khan et al. (1974), are concurrent flow machines. Although taking advantage of the high heat transfer coefficients of solid particle heat exchange, these machines do not have significant heat recovery potential due to the inherent temperature profiles of concurrent flow processes.

Counterflow heat exchangers do have the potential for heat recovery and such use is common with standard shell-and-tube heat exchangers. By exploiting the temperature profiles of a counterflow process and the high heat transfer coefficients of solid particle heat exchange, a unique machine might be developed which would greatly decrease energy required to operate a heating process for grains. This research investigates the concept of solid particle heat exchange as a process for performing thermal disinfestation.

CHAPTER II

OBJECTIVES

The major objective of the research may be stated as follows:

To develop and evaluate a system using counterflow particle-to-particle heat exchange for potential use as a thermal disinfestation process.

The specific objectives were:

To develop a database on potential process characteristics;

To determine steady-state prototype characteristics and performance;

To design and evaluate various control strategies for near-optimal operation of the heat exchange process.

CHAPTER III

REVIEW OF LITERATURE

Chemical Disinfestation

Chemical insecticides can be classified into two basic groups according to usage: protection and disinfestation. Protectants are liquids or dusts which are applied to and around grain storage areas, and to the grain itself as it enters into storage. The object is to guard against any insect population developing. Grain protectants include malathion, methoxychlor, and pyrethrins (Johnson and Townsend, 1981). Disinfestation is generally achieved by fumigation, a process which may use solid, liquid, or gaseous compounds. The process in consideration for this project is for disinfestation. Therefore, the chemicals reviewed are only those compounds curently being used as fumigants.

Bromides, Organophosophates, and Carbon Tetrachloride-based Fumigants

Two bromide-hydrocarbon compounds are currently being used as grain or flour disinfestation agents: ethylene dibromide and methyl bromide. Ethylene dibromide has a boiling point of 131.5 °C and therefore is used in the liquid state (Torkelson, et al. 1966). The vapors of this compound are heavier than air, detectable by smell in concentrations above 25 ppm (Lauhoff), but do not penetrate the grain kernel as effectively as many other fumigants (SDS, 1980). Due to the latter limitation, ethylene dibromide has been used in conjunction with other chemicals such as ethylene dichloride and carbon tetrachloride (Torkelson, et al. 1966). Recently, ethylene dibromide has been banned from many agricultural applications, including use on citrus fruit.

Methyl bromide is a much more volatile compound and is used as a gas. It offers good grain penetration (SDS, 1980) and is not flammable. Various food and agricultural industries rely on methyl bromide to keep damaging insect populations to a minimum (Lauhoff, 1978; Great Lakes, 1978). When severe problems have occurred, entire farms (Moulden, 1979) have been fumigated with methyl bromide to rid an area of a particularly dangerous insect. Methyl bromide is also used to kill certain stored grain fungi (Paster, et al. 1979). Both methyl bromide and ethylene dibromide require 24-48 hours (Great Lakes, 1978) of exposure time to be effective and must be allowed to dissipate after this period before the grain can be used.

Organophosphates are a very important chemical group in both protection and disinfestation. Various representatives of this group began to be used as insecticides in Europe at the end of World War II (Davidson and Lyon, 1979). The organophosphates essentially took the place of the organochlorides, such as DDT, when the latter became restricted in use by law (Ware, 1980). Phosphine, also known as hydrogen phosphide, is the most widely used of the organophosphate fumigants. Due partly to good grain penetration, phosphine is an excellent insectide (SDS, 1980). The boiling point of phosphine is -87 ^{o}C (Torkelson, et al. 1966), but it is used in the solid form as aluminum phosphide. This solid breaks down in the presence of atmospheric moisture, and the toxic phosphine gas is released. "Phostoxin" pellets, a brand of aluminum phosphide, take 12-18 hrs to break down depending on temperature and humidity; tablets will decompose in 48-72 hours (Phostoxin booklet, 1972). Warehouses (Kumar, et al. 1981; Wyckoff and Anderson, 1971), bins (Singh and Srivastava, 1980), and ship cargo holds (Gillenwater, et al. 1981) have all been successfully treated with phosphine.

Although usage is decreasing, carbon tetrachloride-based fumigants are still common in liquid mixtures (Goodship, et al. 1982). According to the Oklahoma Cooperative Extension Service (1984), one of the most widely used on-farm fumigants is an 80%-20% mixture of carbon tetrachloride and carbon disulphide. A USDA pamphlet (#553) lists three common liquid fumigants, and carbon tetrachloride is a 25%-80% constituent in each. Due to the hazardous nature of this compound, research has been done to find a similar, but safer, replacement for tetrachloride. Goodship, Scudamore, and Hann (1982) have evaluated 1,1,1-Trichloroethane as a possible candidate.

Problems Associated with Chemically-dependent Disinfestation Procedures

Basic problems that are encountered in the long-term use of chemical disinfestation include batch operation, development of insect resistance, and hazards to man. The general nature of the fumigation process leads to the batch operation difficulty. While perhaps an advantage on the farm, elevator operations and seaboard shipping points must contend with grain shipment delay due to a need for the grain to be treated.

Since the early 1970's, resistance developed by insects against certain insecticides has become an important concern. The Food and Agriculture Organization, FAO, undertook a worldwide survey of the distribution of pesticide resistance (Champ and Dyte, 1976). Eight species of stored product beetles were found to have developed resistance in varying degrees, and this estimate was soon altered to eleven beetles and five moths. According to Bhatia (1978), the majority of the resistance has developed against the contact insecticides such as malathion, with Tribolium castaneum (Herbst) showing resistance in 70 countries. Due to this, Bhatia recommends more fumigation and less contact insecticide usage. However, numerous authors (Bell, et al. 1977; Nakakita and Winks, 1981; Upitis, et al. 1973; Bond and Upitis, 1972) have discovered resistance to such common fumigants as methyl bromide, phosphine, and ethylene dibromide.

Upitis et al. (1973), by selective breeding and sublethal exposure to methyl bromide, increased the tolerance of Sitophilus granarius (L.) adults seven to eight times the normal maximum level. Bell et al. (1977) examined ten Rhyzopertha dominica (F.) strains which showed some resistance in the adult stage to phosphine. Of the ten strains, nine were also resistant in the egg stage. Nakakita and Winks (1981) did a similar sudy on phosphine-resistant adults of a Tribolium castaneum (Herbst) strain. The authors found the highest resistance occurring in the early and mid-pupae stages. The genetic conditioning studies above reveal the survival-adeptness of the stored grain insects, and the difficulties of long-term use of a particular chemical against a particular specie of insect.

The third major grain-related pesticide problem is the hazardous nature of the compounds. Environmental concerns are important, and unlike other pollutants such as industrial waste products, insecticide usage puts the hazards immediately by, around, and in the foodstream of the general population. Shirasu et al. (1977) wrote that pesticides are potential hazards to the health of large populations when, as residues in food, carcinogenic and mutagenic effects exist. This group tested 193 pesticides and found 15 of them to be mutagenic. Pesticides can be hazardous in three ways: as a toxicant; as a mutagen; and as a carcinogen. Grierson (1978) demonstrates, by citing examples, that some of the publicized problems with agricultural chemicals are inaccurate and/or sensationalized. However, Grierson concludes that current pesticide usage is safe (long-term) almost solely on the fact that, with the exception of lung cancer, all forms of cancer in America are currently dropping in number of cases.

A growing amount of scientific data is becoming available showing the potential dangers associated with modern chemical pesticides (Shirasu, et al. 1977; Nielson, 1980; Soderman, 1982; Sontag, 1981; Hayes, 1982). Ethylene dibromide use came under fire in early 1984 in Hawaii, Florida, and California. This fumigant is a highly toxic material that is hazardous by ingestion, inhalation, and skin absorption (Nielson, 1980). High concentrations, ingested or inhaled, can lead to liver and kidney damage (Torkelson et al, 1966). According to Soderman (1982), ethylene dibromide is a carcinogen for both rats and mice. Nielson (1980) lists three experiments by the National Cancer Institute (NCI) in which ethylene dibromide caused tumors to develop in lab animals, and shows the American Conference on Governmental Industrial Hygienists (ACGIH) putting the chemical in the suspected human carcinogen category.

A related fumigant, methyl bromide, is also highly toxic (Torkelson et al, 1966). The toxic reaction is due to action on certain enzymes (Hayes, 1982) and ultimately the central nervous system (Great Lakes, 1978). One hundred ppm, seven hours per day, can produce serious poisoning, and 1000 ppm for 30-60 minutes can be dangerous to life (Great Lakes Chemical Co, 1978). It is generally acknowledged that low exposures to either methyl bromide or ethylene dibromide over a period of time are cumulative in the toxic effect (Torkelson, et al. 1966), (Great Lakes Chemical Co, 1978). The human body cannot expel inorganic bromide. Blood bromide concentrations rise in the presence of methyl bromide. No data was found on the carcinogenicity, or lack of, for methyl bromide.

Phosphine-based fumigants can be lethal in 400-600 ppm in thirty minutes. The National Institute of Occupational Safety and Health (NIOSH) shows 200+ ppm to be immediately dangerous to life and health (Nielson, 1980). According to Davidson and Lyon (1979), phosphine's toxicity comes from its destruction of the enzyme cholinesterase. This enzyme is the counterpart to acetylcholine in the nerve impulse mechanism. While being the most toxic widely used fumigant (Davidson and Lyon, 1979), phosphine leaves no dangerous residues and is not thought to be carcinogenic.

Both ACGIH and NIOSH recommend that carbon tetrachloride formulations be labeled as a suspected human carcinogen (Nielson, 1980). Liver and kidney damage can result from acute or chronic exposure, including inhalation, ingestion, and skin absorption. The toxicity of carbon tetrachloride is also markedly increased by a synergistic effect with alcohol (Torkelson, et al. 1966; Nielson, 1980). Carbon disulphide, also contained in several fumigant mixtures, has very low concentration recommendations with respect to vapors in open air. The Occupational Safety and Health Administration (OSHA) gives 20 ppm for eight hours, and 100 ppm for 30 minutes. NIOSH recommends dropping this level to 10 ppm in 15 minutes. Chronic exposure can lead to kidney, liver, and vision problems (Nielson, 1980). Sittag (1981) also noted that long-term exposure can bring about psychological and behavioral disorders.

Radiation and Thermal-related Processes for Disinfestation

Gamma Radiation and Radiofrequency, Including Microwave, Methods

Although the use of chemical disinfestation far outweighs other methods of disinfestation, various alternative strategies have been investigated. Methods based on exposing grain to different forms of radiation have been successfully used to perform disinfestation. Giddings and Welt (1982) review the current technical and political status of food irradiation in the United States, concentrating on gamma radiation. Gamma rays are shorter wavelength electromagnetic radiation as compared to x-rays, and lie in the 10^{19} - 10^{21} cycles per second range of the spectrum. The Food and Drug Administration, through an act of Congress, approved irradiation of wheat and flour in 1963. Unlike Japan and the Netherlands, however, the United States has not employed commercial gamma radiation of foodstuffs. Possible reasons for this hesitation, as given by Giddings and Welt, include absence of a pressing need, lack of broad clearance for other foods, and a concern about consumer resistance. The fact that the Food and Drug Administration, the United States Department of Agriculture, the Department of Energy, the Environmental Protection Agency, and other federal and state agencies all claim some jurisdiction in this area may lead to further delays in the implementation of gamma radiation as a disinfestation agent.

Research on food irradiation began in 1943 at MIT (Giddings and Welt, 1982), but did not pick up in ernest country-wide until radioisotopes became readily available in the 1950's (Tilton and Burditt, 1983). Since that time, two main types of radiation sources have been proposed: electron accelerators and radioisotopic sources such as Cobalt-60. The latter type has dominated the research effort. The output of these sources is generally measured in rads or Gray units. Rad, an acronym for radiation absorbed dose, is equivalent to 100 ergs/gram.

One Gray unit equals 100 rads (Giddings and Welt, 1982; Nielson, 1981).

The USDA Stored-Product Insects Research and Development Laboratory in Savannah, Georgia evaluated dosage responses of insects to Cobalt-60 irradiation (Giddings and Welt, 1982). In one report (Brower and Tilton, 1971), this research team used doses of 10, 20, and 40 krad on Plodia interpunctella (Hubner), Tribolium castaneum (Herbst), and Oryzaephilus surinamensis (L.) (Table I) in whole peanuts and shelled peanuts. The results indicated that 20 krad was sufficient to eliminate beetle populations, but that 40 krad was required for moths. Sterilty was found to occur in the insects at sub-lethal exposures. The dosage rate in the above experiment was 2.4 krad/min, therefore the longest exposure was for 16.67 minutes. By the time this research effort was completed, 30 species of insects had been tested for elimination in bulk and packaged grain and flour. All populations were found to be destroyed with a dosage of 50 krad. A table of sterilization doses for 28 stored-product insects was developed by Tilton and Burditt (1983), showing doses in the range of 10 to 100 krad.

From approximately 10^3 to 10^{11} cycles per second lies the radiofrequency band, with the 10^9 - 10^{11} area referred to as microwaves. Microwave and longer wavelength radiation have been used in research to disinfest small

TABLE I

WORLDWIDE STORED-PRODUCT INSECTS

Scientific name	Common name
Tribolium castaneum (Hbst) ------	Red flour beetle *
Sitophilus oryzae (L.) ----------	Rice weevil
Oryzaephilus surinamensis (L.) --	Sawtoothed grain beetle
Cryptolestes ferrugineus (St.) --	Rusty grain beetle *
Ahasveras advena (Waltl) --------	Foreign grain beetle
Tribolium confusum (J. du V.) ---	Confused flour beetle *
Rhyzopertha dominica (F.) -------	Lesser grain borer *
Cryptolestes pusilus ------------	Flat grain beetle *
Sitophilus granarius (L.) -------	Granary weevil
Tenebrio molitor (L.) -----------	Yellow mealworm
Sitophilus zeamis (Mothsch.) ----	Maize weevil
Trogoderma granarium (Everts) ---	Khapra beetle
Plodia interpunctella (Hubner) --	Indian meal moth
Oryzaephilus mercator (Fauvel) --	Merchant grain beetle
Cathartus quardricollis (Gm) ----	Squarenecked grain beetle
Sitotroga cerealla (Oliv.) ------	Angoumois grain moth
Trogoderma variabile ------------	Warehouse beetle

Note: * indicates particular importance in Oklahoma.

grains and flour (Nelson, et al. 1966). In this process, a material absorbs some of the radiation energy internally due to its dielectric loss factor (Nelson, 1973). Baker et al. (1956) used microwaves at 2.45 x 10^9 cycles per second on eggs, larvae, and adults of Tribolium confusum (Duv.) and Sitophilus granarius (L.). Various infested grain samples were exposed from 3 to 21 seconds to the radiation. This range in time corresponded to a range in temperature of the grain/insect mass of 90-178 °F in the T. confusum, and 89-195 °F in the S. granarius. The authors found that a maximum temperature of 165 °F with an exposure of 21 seconds was lethal to 100% of the adult population, with as much as 23% of the eggs surviving to hatch.

Nelson, et al. (1966) examined the physical factors that influence the efficiency and effectiveness of radiofrequency-based disinfestation. Among the most important were frequency, field intensity, and rate of dielectric heating, all of which are interrelated. The authors noted the difference in insect susceptibilty to radiofrequency treatment, with Rhyzopertha dominica (F.) offering the most resistance. Nelson and Charity (1972) attempted to exploit an important dielectric heating property for disinfestation purposes. Different materials absorb energy from RF fields in varying degrees according to each materials' dielectric loss factors at a particular frequency. Nelson and Charity searched through the RF portion of the spectrun for the highest ratio of Sitophilus oryzae (L.) loss factor versus the hard red winter wheat loss factor. The authors found the 10^7 - 10^8 cycles per second region to contain this maximum efficiency. Nelson (1973) wrote of the state-of-the-art of RF heating for disinfestation and concluded then that application of the technology was questionable from an economic standpoint, even though the process had been proven to be effective.

Infrared, Conduction, and Convection Heating Methods

Heat as a disinfestation process has been employed since primitive times (Cotton, 1963) when the sun or an open fire served as the energy source. Elevated insect body temperatures cause death by coagulation of soluble proteins, injury to certain enzymes, and/or dessication (Cotton, 1963). This method, termed thermal disinfestation, has been investigated scientifically as far back as 1911 (Dean, 1912). One of the methods for heating that has been researched is infrared radiation. With the introduction in the late 1950's of the gas-fired ceramic panel infrared heater, research began on utilizing infrared radiation in both drying and disinfestation of grains. Schroeder and Tilton (1961) utilized a heater of this type to determine mortality criteria of Sitophilus oryzae (L.) and Rhyzopertha dominica (F.) in rice. The authors reported complete control when the rice was heated to the 56-68 °C range. Rate of change of temperature also appeared to be an important factor. Continuing their work, Tilton and Schroeder (1963) studied Sitotroga cerealla (Oliv.), along with the above mentioned species. Grain mass distance from the infrared source, irradiation time, insect age, and grain temperature were all considered in the experiments. Total mortality could be expected, the authors reported, with a grain temperature in the 65-70 °C range. Higher radiation intensities tended to decrease the required temperatures. S. cerealla was the most heat resistant insect in the immature stage of all three tested, and S. oryzae was the least resistant. R. dominica was the most resistant as an adult.

Kirkpatrick and Tilton (1972) examined the effects of infrared radiation on 12 stored-grain insects. Wheat temperatures were raised to 49, 57, and 65 °C using 20, 32, and 40 second exposures. At 49 °C, all mortalities were above 93%, and at 65 °C, virtually a complete kill was obtained. This work was done with young adults of each of the species tested, and mortality was measured 72 hours after exposure. Kirkpatrick, et al. (1972) compared microwave and infrared treatments on Sitophilus oryzae (L.) in wheat. Tight control on insect age distribution per sample was used to allow investigation of the age-versus-resistance correlation. Both processes were set to give a sample temperature of 54 °C. Results showed that temperatures required for the same degree of control were higher for the microwave treatment when compared to the infrared treatment. Infrared reduced first generation emergence from treated samples from 9 to 18% more than microwave with respect to the untreated (control) samples.

Thermal disinfestation using a low-grade energy source such as heated air is another external heating process. Reports of use date back to 1911, when Dean (1912) raised the temperature inside a flour mill in an attempt to eradicate an insect population. Munro (1966) reports this same procedure being used in another mill in 1914. In Florida, Grossman (1931) exposed Tribolium castaneum (Herbst), Sitophilus oryzae (L.), Cathartus quadricollis (Gm), and Sitotroga cerealla (Oliv.) to 122 °F for one hour. All stages of all insects were reportedly killed. Oryzaephilus surinamensis (L.), S. oryzae, T. confusum, and Rhyzopertha dominica (F.) were exposed by Kenagen and Fletcher (1942) to an environment in which the temperature was raised from 80 to 105 °F. Some mortality was reported in the first two species listed above, but none in the latter two. Munro (1966) recommended a temperature in the range of 120-130 °F for 10 to 12 hours to destroy an insect population in a flour mill. Cotton (1963) and Davidson and Lyon (1979) reported that exposure to 140 °F for 10 minutes would be fatal to all stored-grain insects, and would not impair seed germination.

Only three research efforts have recently investigated the use of conduction or convection for thermal disinfestation as a possibility for modern insect control:

Australia's Commonwealth Scientific and Industrial Research Organization (CSIRO); USDA's Stored-Product Insects Research and Development Laboratory; and the Agricultural Engineering Department at the University of Manitoba.

CSIRO has done the most work in this area. Dermott and Evans (1978a,b) began studying a physical process that could be used at export points to continuously disinfest grain. Primarily due to being a proven device with high heat transfer rates, fluidized beds were chosen as the process to implement. In the initial Dermott and Evans work, Sitophilus oryzae (L.), Rhyzopertha dominica (F.), and Sitotroga cerealla (Oliv.) were mixed with wheat and placed in a batch fluidized bed. Samples of 500 or 1000 grams were exposed to heated air at 60, 70, and 80 °C. Post-exposure treatment was in a cooling fluidized bed. Lethal times for 50% and 99% of the insect populations were calculated from the experimental results. R. dominica appeared to be the most heat resistant, with a lethal time to 99% of the population (LT99) of approximately 10 minutes at an air temperature of 60 °C, and 4 minutes at 80 °C.

Evans and Dermott (1981) continued the research by examining the influence of air inlet temperature, bed depth, and grain temperature on the required exposure time for a particular mortality level. Rhyzopertha dominica (F.) were exposed to 60 to 80 °C inlet air temperatures in varying loads of wheat. Lethal times were determined at the 50% and 99% mortality levels for different loads and inlet temperatures. Evans and Dermott reported that exposure time was inversely related to air temperature, and directly related to load. Taking the project a step further, Evans et al. (1983) designed and tested a continuous flow thermal disinfestation system. The unit consisted of two fluidized beds, one for heating and one for cooling, and treated up to 500 kg/hr of wheat. In general, air temperatures of 80 to 90 oC, air flowrates of 0.25-0.32 kg/sec, and residence times of 1-2 minutes were used. The authors reported complete success with any treatment method that forced the grain surface temperature to reach or exceed 65 oC. Grain dispersion was thoroughly investigated and found not to be a problem. Niro Atomizer (Bulletin No. 69) reported a 50 tonne per hour high temperature fluidized bed thermal disinfestation plant in Dunolly, Victoria, Australia (Davidson, 1983). The design was based solely on the CSIRO research, and the plant was built for and purchased by the Grain Elevators Board of Victoria. The plant is equipped with research-type monitoring equipment for aid in studying the long-term operation.

The Stored-Product Insects Research and Development Laboratory (Kirkpatrick and Tilton, 1973) investigated thermal disinfestation using low-grade energy sources as a part of their continued efforts on various physical processes to control insects. As mentioned previously, this group investigated gamma radiation, microwave, and infrared radiation methods of disinfestation. Kirkpatrick and Tilton mixed Sitophlus oryzae (L.) and Rhyzopertha dominica (F.) with wheat, and placed the samples in an incubator. Conditions used in the experiments included 39 °C at 60-75% relative humidity for S. oryzae, and 39 °C at 60%, or 43 °C at 50% for R. dominica. Exposure time was four days. Mortality rate for adult S. oryzae was only 1.8% for the 39 °C, 75% treatment, but rose to 100% when the 39 °C, 60% treatment was used. A mortality rate of 99.7% was reported for R. dominica in the treatment of 43 °C, 60% relative humidity. The authors also found that, for the two species tested, the life cycle stages varied in resistance. The difference was not significant in S. oryzae, but R. dominica showed increasing resistance in the following order: larva; egg; pupa; and adult.

Vardell and Tilton (1981) duplicated the work done with fluidized beds by CSIRO. The only difference in the USDA process with respect to that used by CSIRO was that no cooling section was used; the grain samples were allowed to steep after heat exposure. The fluidizing air temperature was 80 °C. Samples were kept at 27 °C and 80% relative humidity for six weeks after exposure to monitor progeny emergence. The results matched or exceeded those obtained by CSIRO. Complete control of R. dominica occurred when the sample was allowed to reach 67 °C, corresponding to a residence time of 4.75 minutes in the fluidized bed. Similarly, complete control was obtained for S. oryzae at 60 °C.

The Agricultural Engineering Department at the University of Manitoba has been investigating drying small grains using a solid particle heat transfer process since 1972 (Lapp and Manchur, 1974). Both batch (Lapp, et al. 1975; Mittal, et al. 1983) and concurrent flow (Lapp and Manchur, 1974; Lapp, et al. 1976a,b; Lapp, et al. 1977) processes have been used to dry wheat and rice. Granular salt, steel balls, and various textures of sand have all been utilized as the heat transfer medium (Lapp, et al. 1976a). Mittal et al. (1981) conducted experiments using both the batch and concurrent flow equipment to analyze the ability to disinfest wheat. Various life stages of Crytolestes ferrugineus (Stephens) were used in the tests. In the batch tests, 450 grams of clean wheat was mixed with 50 grams of infested wheat, then placed in a container with a measured amount of hot sand. The sand-to-grain ratio varied from 4:1 to 4.5:1, with sand temperatures of 91 to 106 °C. Residence times of 1-2 minutes were used. Adult insect survival was measured 24 hours after exposure. Post-treatment environmental conditions were not reported. Samples were kept for six weeks to check emergence.

Virtually 100% kill of all stages was achieved in each of the batch tests. The continuous flow treatments did not do as well. The same ratios of infested grain, clean grain, and hot sand were used, but sand temperatures were slightly higher, averaging 105-115 °C. The problem with the concurrent flow experiments was procedure: the infested grain portion was placed into bags, then fed into the system. Actual heat transfer to the infested grain was therefore greatly diminished. The data reported on these tests was unclear. It was apparent that 100% mortality was achieved on the adult stage. However, younger life cycles had much lower mortality rates, the extent of which could not be discerned from the report.

Potential Problems Associated with Thermal Disinfestation

The application of a thermal-based process as a method for disinfesting grain is not "problem-free." Although none of the potential difficulties should prevent investigation of the concept, research performed with the intent to formulate a practical process must acknowledge and deal with them. At present, it appears that there are three areas of concern: thermal acclimatization of insects; output grain quality; and practical economics.

Acclimation describes the changes that occur within an organism due to changes in the organism's environment. Thermal acclimatization refers to an organism's compensation for change in temperature in a natural environment (Ernst, 1968). Research on cooling grain to inhibit stored-grain insect population growth has fostered a good deal of work on insect response to low temperatures (Edwards, 1958; Nuttal, 1970; Howe, 1965). Cold acclimation has been shown to lower the chill coma temperatures of several stored-grain insects, including Tenebrio molitor (L.) and Tribolium confusum (duVal) (Evans, 1981b). Investigations on dispersal (Ernst, 1968) and oxygen consumption (Evans, 1981b) in relation to acclimation have also been done. Much less research, however, has been done on acclimatization in insects due to exposure to sub-lethal high temperatures.

The possibilty of insects surviving a thermal disinfestation process and thus being "heat-treated" may be remote, but acclimation in insects is not well-defined. For example, one would expect that an insect kept at higher than normal temperatures would exhibit a higher thermal-death point. This has been demonstrated, for instance, in Tenebrio molitor (L.) larvae (Ernst, 1968). Larvae kept at 30 °C had a thermal-death point of 42 °C, while those kept at 37 °C for 24 hours had a thermal-death point of 44 °C. However, Edwards (1958) kept Tribolium confusum (duVal) samples at 18, 30, and 38 °C for six months, then exposed them to a 40 °C environment. The 30 °C sample was considered the control group. The 18 °C sample had a higher survival rate in the new environment than did the control, and the control surpassed the 38 °C sample. Obviously, acclimation is a complex process; heat resistance can be increased through cold acclimation or heat acclimation, depending on the species of the insect.

The fact that thermal disinfestation processes require operation in the 55-70 °C range for a short period of time should make it difficult for an insect or its progeny to acclimate. A related heat-resistant problem which is more likely to affect disinfestation efficiency is that of diapause. The state of diapause is similar to that of hibernation; it is a dormant state, species specific, and induced by environmental changes. Diapausing larvae of some stored-grain insects have been reported to be more heat tolerant (Battu et al., 1975). Bell (1983) examined the effects of high temperature (40-45 °C) on diapausing Ephestia elutella (Hb.) larvae. Bell demonstrated that, like induction, the termination of diapause is also stress related. Short, repeated exposures were much less effective in terms of mortality in the diapausing insects. The severe rate of heating in thermal disinfestation is such that diapause will most likely not be induced. However, the survival of some insects that are already in the diapause state, as demonstrated by Bell (1983), is a possibility.

A second potential problem area in thermal disinfestation is that of the elevated temperatures causing heat damage to the grain. Evans and Dermott (1978b) recognized this possibility when they first began using the fluidized bed concept. Wheat quality, in terms of baking and dough characteristics, was tested, and only those samples which were exposed to 100 °C air showed significant damage. Ghaly and Taylor (1982) examined the effects of using 60-120 °C air for 2-5 minutes on two wheat varieties, Olympic and Spica, each at 12 and 14% moisture content. The results showed both varieties withstanding 60 °C air for up to 2 hours with no decrease in germination or baking quality. Exposure of the hard variety (Spica) at 14% to air at 80 °C for 15 minutes decreased overall quality considerably. The same exposure to the Olympic variety at both moisture contents, and to Spica at 12%, did not significantly damage the grain.

Ghaly and Van der Touw (1982) used Teal, Condor, and Eagle wheat varieties at 12% and 14% in similar tests. While the hard variety sustained the most damage in the previous tests, the soft variety (Teal) showed the least heat resistance in this set of testing. The authors reported that protein content may be more of a factor than hardness in specific variety tolerance. As in the previous tests, wheat at 14% sustained more damage than did the wheat at 12%. The above investigations seem to indicate a safety line drawn at approximately 70 °C for 12% wheat.

Agricultural production and distribution of non-value-added commodities are most often low margin operations. Therefore, any new concept, process, etc. should fit within this economic constraint if it is to be considered for widespread use. Fumigation currently costs $ 0.88 to $ 1.32 per tonne (OSU Cooperative Extension

Service estimates, 1984), and a process to implement thermal disinfestation would most probably have to match or lower that range. Dermott and Evans (1978b) estimated the energy required per tonne of wheat to run their fluidized beds. They reported 69 MJ for heating (grain heated from 25 to 65 °C), 120 Kw for 4 minutes for fluidizing the heated bed, and the same for fluidizing the cooling bed. Capital costs were not considered. Using $0.005 per cubic foot for natural gas, and $ 0.065 per kilowatt-hour for electricity, this corresponds to $ 0.33 per tonne for heating and $ 0.52 per tonne for fluidization in each of the beds. The total is therefore $ 1.37 per tonne, with fluidization taking 61% of the cost in the heating section, and 76% overall.

Current chemical fumigation requires little in the way of capital, but in a fluidization process the initial investment may be considerable. Since the operational costs for the CSIRO system are at least $ 0.05 per tonne more than conventional methods, there is no impetus for investment. A new method that works within the economic constraints is therefore required. A possibility exists for considering thermally disinfested grain a value-added commodity, particularly on the international market. If so, more detailed economic analyses would be necessary to determine the potential of the investment.

Potential Use of Solid Heat Transfer Media as a Mechanism for Thermal Disinfestation

Devices Based on Solid Heat Transfer Media Used to Process Particulate Solids

The heating and cooling of particulate solids is a common process throughout the food and chemical industries. Some of the physical devices which accomplish the required heating or cooling actions are based on solid-to-solid heat transfer. Of these devices, a great majority are of the indirect heat-processing type (i.e. transfer of energy through a wall) (Holt, 1967).

One of the earlier indirect heating machines is the agitated pan, principally used to dry material with a heated steel plate. Another device, called rotating horizontal tube-bundle and shell, forces liquid through tubes to serve as the heat transfer medium, and the solids flow around the tubes to complete the process (Holt, 1967). Specially-designed screw conveyor equipment is also used for heat processing of particulate solids. Many of these are hollow, multi-shaft machines which utilize the blades as the heat transfer mechanism. The screw is jacketed and the heat transfer occurs during tranporting of material (Katatkin et al., 1964). Screw conveyors of this type are used for preheating, drying and melting, and all have high heat transfer coefficients (Uhl and Root, 1967).

Solid-to-solid heat transfer has been investigated by several individuals. Aspegran (1959), in U.S. Patent Number 2,872,386 , depicts a direct solid-to-solid heat transfer mechanism. Aspegran's machine takes in heated balls and ambient particle solids continuously, rotates for mixing and conveyance, and separates the solids at the output, recirculating and reheating the balls. A different sort of mechanism involves rolling mills. Roll mills are common in the food processing industry for handling such tasks as flaking, bumping, sheeting, and forming, but they are also used for heating (Holt, 1967). Huber (1955), in U.S. Patent Number 2,701,200, presents a process to puff cereal products using a roll mill as the heat transfer mechanism.

Benson (1966) designed a particle-to-particle, concurrent flow device for processing cereal grains. The machine, described in U.S. Patent Number 3,253,533, uses an inclined barrel with an internal perforated screw attached. The heat transfer medium, in this case salt, stays within the device at all times and is heated through the shell of the barrel. Grain is input to the lower end of the barrel, then transported out by the screw, with the salt supposedly falling back through the screw perforations. Raghavan and Harper (1974) built and tested a machine based on the design by Benson. Salt was used as the heat transfer medium, and the performance was based on drying corn. Salt bed temperatures were varied from 274 °F to 525 °F, and the residence times from 4.2 to 21.8 seconds. The latter figures are approximations since, due to the nature of the device, exact residence times cannot be determined. Stress cracking and heat transfer uniformity were found to be problems; however, the device could dry corn successfully to 12-14% moisture content, wet basis, using 450 °F salt.

Lapp and Manchur (1974) built a particle-to-particle heat transfer device to investigate more efficient drying of cereal grains. The machine metered both sand and grain into a rotating barrel. The mixture was then conveyed along to the center section, at which point the heat transfer medium was separated from the mixture and returned to a heated bin. The output section cooled the grain with ambient air. Good drying results were reported for tests with rapeseed. Khan et al. (1974) designed and built a particle-to-particle heat transfer device to dry paddy. This device is similar to the one described above, pushing the heat transfer medium and the grain down a barrel. However, no external heating bin was used. After separation at the end of the device, external sweeps on the barrel push the heat transfer medium back to the front, reheating along the way. No cooling section was used.

Bateson and Harper (1973) obtained U.S. Patent Number 3,746,546 for a machine designed to puff food products. The device also used particle-to-particle heat transfer as in the above examples, but it differs in the separation and reheating sections. A mixing barrel was used, and the mixture was dumped onto a vibratory, screened conveyor. This configuration separated the heat transfer medium, which was conveyed by a screw back up to the barrel. Reheating was done in the screw using electrical resistance heating bands. Chancellor (1974) designed laboratory and full-scale dryers which used a horizontal metal surface transmitting heat to a stirred bed of grain. The design was for developing countries, and utilized animal power for stirring and crop residue for fuel. Successful drying was reported, but seed germination was destroyed.

Models Applicable to Particulate Solids

Heat Transfer

Mathematical models are important tools for design and analysis. Theoretical and/or empirical models are particularly useful to process engineers in that a certain idea or operation can be analyzed prior to physical experimentation. The use of solid heat transfer media as a mechanism for thermal disinfestation is a prime target for initial investigation through established model analysis. Although the number of research efforts on solid heat transfer media modeling is small, there has been some success in the area. Uhl and Root (1967), in an overview of practical granular solids heat transfer, reported on several models. One of the more accurate theoretical models, by Otake and Tone (1960), uses the concept of "effective" thermal conductivity. The model is based on the assumption that there are two heat transfer processes operating in parallel: one, from a solid, heated barrier to a bed of solid particles; two, from the heated particles to non-heated particles. The use of this model is limited since the equations apply only to a specific device and geometry. Uhl and Root described two more batch models, one based on a Bessel function solution of a differential heat balance equation, and another based on dimensional analysis. The only non-steam continous model described in this reference was one credited to M.S. Mery. Mery developed an empirical model for a cut-flight, hollow screw-type agitator. The relation, in English units, was given as follows:

$$h = 21.5\ k^{0.532}\ p^{0.473} \tag{3.1}$$

where h = heat transfer coefficient from granular solid to the wall of the blade (Btu/hr ft^2 ^{o}F);

k = "effective" thermal conductivity of granular solid bed (Btu/hr ft ^{o}F);

p = bulk density of granular solids (lb/ft^3).

The device for which the above equation was developed enhances the heat transfer rate by a phenomenon referred to as "backmixing" (Uhl and Root, 1967).

Raghavan et al. (1974) developed both theoretical and empirical models for particle-to-particle heat transfer.

The theoretical model was based on the assumption that the heat transfer rate is proportional to the number of impacts between particles. The following relationships were derived:

$$h = \frac{(\ 2\ D\ p\ c\ )}{(\ 3\ v\ r\ )} \frac{dT}{dt} \qquad (3.2)$$

$$h' = h/\text{delta } T \qquad (3.3)$$

where h = heat transfer coefficient per unit mass (Btu/lb);

h' = heat transfer coefficient (Btu/lb ^{o}F);

T = temperature of large particle (^{o}F);

t = time (sec);

D = diameter of large particle (ft);

p = bulk density of large particle (lb/ft^{3});

r = bulk density of small particle (lb/ft^{3});

c = specific heat of large particle (Btu/lb ^{o}F);

v = relative velocity between large and small particles (ft/sec).

Experimental data were taken using a metal ball as the large particle and salt as the small particle. A regression model was derived from the data which showed a linear correlation between the rate of heat transfer and the inverse of "v" above. The authors concluded that the model had been validated, but this was unclear.

Sullivan and Sabersky (1975) studied the heat transfer mechanism between granular solids and other adjacent objects. Several assumptions were made in the analytical development of the first model, including infinite particle thermal conductivity and "orderly" heat flow. The result of the work was an equation giving dimensionless temperature of a particle in space in terms of location and modified Bessel functions. An equation for the Nusselt number for an individual particle was also derived. The second model considered the bed of particles as a one-component continuum, a much simpler formulation, leading to the following:

$$h = \frac{k}{((\text{pi}\ a\ x)/v))^{0.5}} \tag{3.4}$$

where

h = film heat transfer coefficient (Btu/hr ft2 °F);

a = thermal diffusivity (ft2/hr);

k = thermal conductivity of plate (Btu/hr ft °F);

x = coordinate in direction of flow (ft);

v = velocity of particles (ft/hr).

Sullivan and Sabersky found that the simpler model appeared to be as accurate as the discrete particle model provided certain physical constraints were met.

In an extensive National Science Foundation report, Downs et al. (1977) presented work that led to four dimensionless empirical equations for particle-to-particle heat transfer. Metal balls were used to transfer heat to sand, salt, and glass beads. Data were taken over a variety of conditions, and dimensional analysis was used to develop the equations. The equations gave a theoretical Nusselt number in terms of a group of dimensionless terms and their individual exponents. Goodness of fit varied from an r-square value of 0.477 for sand to 0.946 for the glass beads. The overall r-square was 0.707. The shape of the sand particle appeared to be the factor in negating better results. The diameter ratio was the highest correlating dimensionless term, with the internal angle of friction and shape factors being significant also. The poorest correlation was for the Froude number. The lack of velocity effect as evidenced by the low correlation of the Froude number is in conflict with the results reported by Sullivan and Sabersky (1975) and Raghavan et al. (1974).

Richard and Raghavan (1980) examined the heat transfer process between flowing particle solids and objects within the flow path. An analytical equation was derived for flow past a flat surface. The result was identical to that reported by Sullivan and Sabersky (1975). Another equation was developed for flow past a sphere, and it was in the same basic form as the first. A third model was derived based on the assumption that contact resistance between the particles and the object was due to a gas film which had its own inherent thermal properties. A regression equation for the heat transfer coefficient was developed using data reported by other authors:

$$h^{-1} = B1(t^{0.5}) + B0 \quad (3.5)$$

$$B1 = c^{-1} (\text{pi } D/4)^{0.5} \quad (3.6)$$

$$B0 = j\, x^{-1} \quad (3.7)$$

where

h = heat transfer coefficient (W/m^2 ^{o}C);

B1 = first regression constant;

B0 = second regression constant;

t = time (s);

c = thermal conductivity of particles (W/m ^{o}C);

x = thermal conductivity of gas layer (W/m ^{o}C);

D = thermal diffusivity of particles (m^2/s);

j = constant relating particle size and gas film thickness.

The equations for "B1"and "B0" show the theoretical basis of the regression equation as chosen by Richard and Raghavan. Goodness of fit ranged from r-square values of 0.84 to 0.99 .

Potential of Counterflow Processes

While the overall heat transfer coefficient can be used to compare the performance of a given system or process to another, an equally important performance criterion is based on heat recovery capability. The difference between the two involves more than semantics. A high heat transfer coefficient indicates a good rate of transfer of energy. A high heat recovery implies economy in the use of input energy. Consider a concurrent flow heat exchange system, as shown in Figure 1. Both the heat transfer medium and material to be processed are conveyed in the same direction. The heat transfer coefficient of such a system may be extremely high, but the possibility of heat recovery is limited. Note that as the two materials are conveyed over a longer and longer distance, the exit temperatures will approach equlibrium (McCabe and Smith, 1967). The heat transfer medium can be recycled, maintaining the energy not transferred to the processed material. However, the temperature of the processed material, being approximately equal to that of the heat transfer media, does not allow for the necessary temperature difference required for recovery heat flow.

The use of a counterflow process, also called countercurrent flow, allows for the possibility of recoverying the energy from both the heat transfer medium and the processed material. The maximum temperature change in a counterflow heat exchange process is limited only by the outlet temperatures equilibrating with the inlet temperatures of the other stream, as shown in Figure 2. Therefore, a counterflow process is more efficient, with respect to potential heat recovery, than a concurrent flow process (Bell, 1983). Figure 3 depicts a counterflow arrangement for raising the temperature of a material to a prescribed level, then allowing it to cool, reclaiming the energy by way of the heat transfer medium. As shown, the heat necessary to sustain the process is input to the

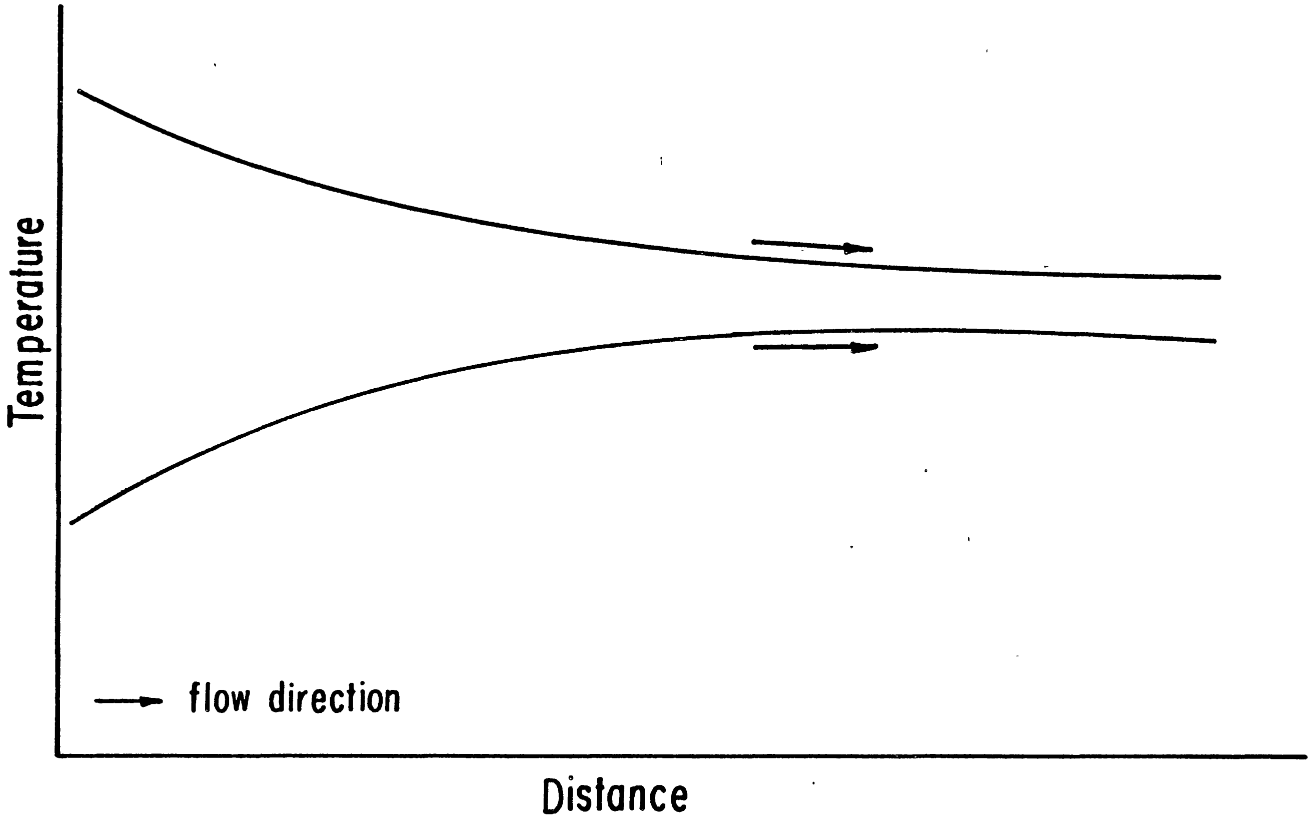


Figure 1. Temperature Profile of a Concurrent Flow Process

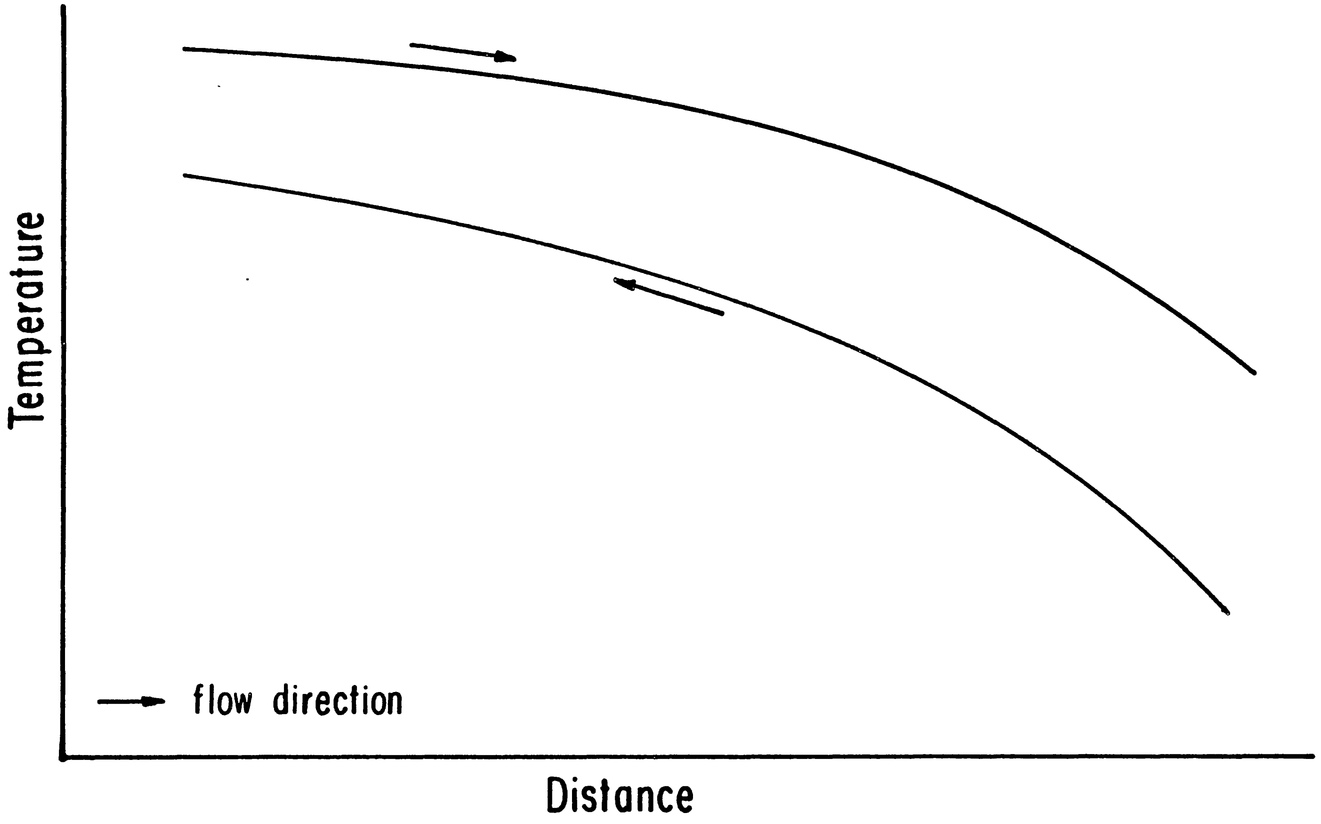


Figure 2. Temperature Profile of a Counterflow Process

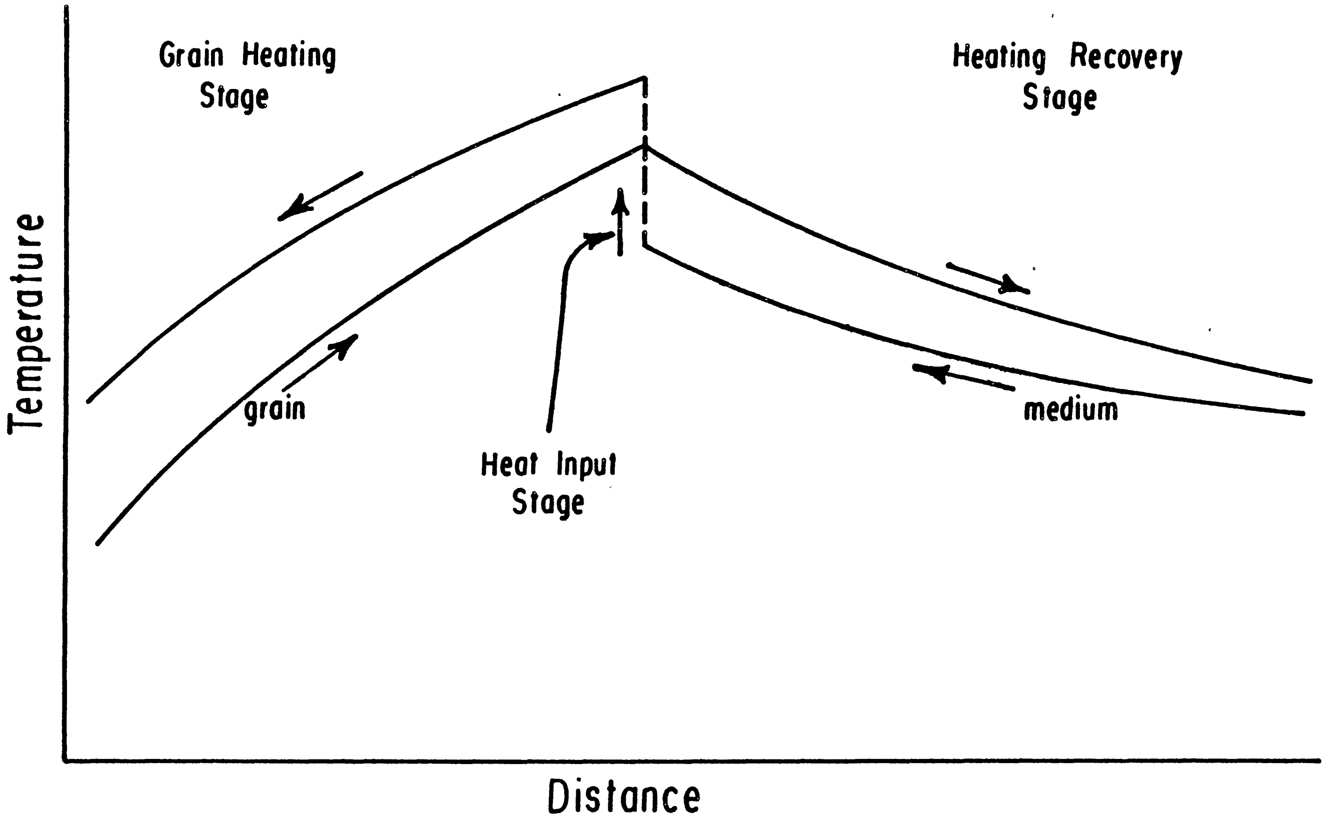


Figure 3. Temperature Profile of a Dual-stage Counterflow Process for Heat Recovery

center section. The dual-stage counterflow process has excellent heat recovery potential. To take advantage of this potential, using solid heat transfer media on solid particles such as small grains, amounts to a difficult mechanical task. However, if a process such as this could be designed, the operational costs of the system would most definitely enhance the acceptance and usability of the system for grain processing.

Adaptive Control as a Means of Improving the Performance of Nonlinear/Time-varying Systems

Background

Many processing applications require the use of feedback control systems, also termed closed-loop control. Examples include temperature control within a cooker or dryer and flow control in a liquids handling system. A general closed-loop system may be defined as a process in which system outputs have an effect on system inputs. In a closed-loop control system, this relationship is utilized for regulatory purposes. The principal advantage of using such a system is improved performance in the presence of unknown disturbance inputs and unknown system parameters (Rowland, 1983). The basic objectives of closed-loop control systems are to maintain certain key process outputs at or near the desired values while also providing some optimal response.

From the early 1930's with the work on servomechanisms and feedback amplifiers (Black, 1934) to the present, feedback control theory has progressed steadily. Many techniques for control system design have been developed for processes which have a known dynamic model. However if the model is unknown, and in particular time-varying and/or nonlinear, an optimum design becomes very difficult. This condition spawned the field of adaptive controller design. Adaptive systems are those which automatically adjust controller settings to compensate for changes in the process or environment (Seborg, et al. 1983).

Early development in the field of adaptive systems was prompted by a need in the design of autopilots for high performance aircraft and rockets in the 1950's. During this time period there was little theoretical base for the control engineer (Seborg, et al. 1983). In general, efforts were not particularly successful. With the advent of the microprocessor and the beginnings of a strong theoretical base (Astrom and Wittenmark, 1973), the study and application of adaptive control systems have become much more prominent.

An adaptive technique initially proposed by Kalman (1958) provided the basis for the self-tuning regulator fundamentals developed by Astrom and Wittenmark (1973). Much of the subsequent work on adaptive control systems utilizes the latter authors work as a foundation. The strategy requires estimation of the dynamic parameters of the process using natural input and output data, then incorporating the estimated parameters into a feedback control law. Thus as the process dynamics change, so do the controller dynamics. Equally important is the fact that the parameter estimation algorithm can be designed separately from the control algorithm with no impact on stability. This is termed "certainty equivalence" (Goodwin and Sin, 1984).

Adaptive Control

The significant works on adaptive control in the 1970's and 1980's have dealt predominately with algorithms for recursive parameter estimation, controller design strategies, and stability. Recursive least squares algorithms have been the most widely used technique for parameter estimation (Seborg, et al. 1983). The general form of the least squares algorithm minimizes a cost function (between predicted and actual system output) while remaining somewhat insensitive to both process and measurement noise (Goodwin and Sin, 1984). Recursive least squares algorithms tend to "turn off" over time and several authors (Young, 1969; Goodwin, et al. 1983; Clarke, 1981) have designed methods of ensuring the algorithm remains active, at least periodically. A data weighting factor or a resetting algorithm are the most commonly recommended methods.

The advent of the microprocessor has played a critical role in the development of control laws in adaptive configurations. Virtually all of the algorithms currently considered and/or applied are discrete both in initial development and in final form (Seborg, et al. 1983). Also, the use of parameter estimation as an adapting mechanism implies the use of process parameters in the control law. Therefore, most control laws used in adaptive control systems are based on assuming the following input-output difference equation can model the dynamics of the process:

$$\begin{aligned} y^*(t) = {} & (ao)y(t) + \dots + (an)y(t-n) + \\ & (bo)u(t-d) + \dots + (bm)u(t-d-m) \end{aligned} \tag{3.8}$$

where y = discrete time process output;
u = discrete time process input;
n = order of the process dynamics;
m = order of input dynamics;
d = number of discrete deadtime elements.

To apply (3.8) in an adaptive framework requires using the estimated parameters in place of the actual (unknown) parameters in a control law. Astrom and Wittenmark (1973), using as a basis (3.8), are credited with the first rigorous development of an adaptive control law. Other classes of control laws exist, such as the self-tuning controller

(Clarke and Gawthorp, 1975), closed-loop pole placement (Astrom and Wittenmark, 1980), and model reference (Landau, 1979). Although the several classes of algorithms have been designed under differing frameworks, research has shown that all of the techniques are closely related (Egardt, 1980; Ljung and Landau, 1978).

Robustness

A control system which continues to regulate a process satisfactorily in the presence of a disturbance is said to have good disturbance rejection. Likewise, if a control system regulates a process satisfactorily as the process parameters change, the system is said to have low sensitivity to those parameters. A robust system is one which has both good disturbance rejection and low sensitivity to changing process parameters (Franklin and Powell, 1980). Developing an adaptive system which is only stable does not ensure robustness. However, inherent in the use or application of an adaptive controller lies a need for robustness. In general, the form of the control law is fixed and its effect on robustness is minimal compared to the effect of the parameter estimation scheme (Seborg, et al. 1983).

Several authors have dealt with improving the robustness of adaptive controllers by modifying the estimation and control calculation relationship. The main difficulty is determining when a set of estimated parameters is "good" and thus can be used in the control law. It is important not to update the controller with parameter estimates which are grossly in error. Goodwin and Teoh (1983) recommended the use of two time frames, one for sampling data and updating the parameter estimates, and the other for updating the control law. Vogel (1982) suggested monitoring the gain of the predicted process transfer function. If the gain was determined to be unreasonable, the controller parameters should not be updated. Whatever technique is utilized, robustness is an important design consideration when developing an adaptive controller.

Applications

Although much work has been done on the development of adaptive controllers, usage in the private sector is limited. Seborg, et al. (1983) lists over 70 recent applications in such diverse areas as cement raw material blending, distillation columns, paper machines, and power stations. However, less than half were on full-scale equipment and virtually all were in the experimental stage. The majority were in Europe and Canada.

Several successes have been reported recently. Anex and Hubbard (1984) applied an adaptive controller to a laboratory three degree-of-freedom robot. An LSI 11/23 computer was used to sample joint positions and torques at 530 Hz. Although certain nonlinear affects such as static friction proved a hindrance, the arm trajectory was controlled successfully. Harrell (1984) applied a model reference adaptive controller to a solar fruit juice pasteurization process. Using a Motorola M6800 microprocessor-based system, Harrell reported accurate control in the presence of substantial load variations. Van Amerongen (1984) reported successfully implementing an adaptive controller in place of a standard autopilot for the steering of a Royal Netherlands Navy supply ship.

Two "intelligent" control systems are currently being marketed in the United States. Both are being touted as expert systems rather than adaptive controllers. "Picon", from Lisp Machine Incorporated, is designed to supervise existing control loops and advise operators on alarms and process disturbances. "Exact", from the Foxboro Company, does actually tune loops on-line using over 100 tuning rules developed from pattern recognition techniques (Chowdhury, 1985). Although these two systems have been developed with a different philosophy than the adaptive control systems discussed above, they do represent the trend towards application of sophisticated devices for process control.

CHAPTER IV

MODELING AND SIMULATION OF A COUNTERFLOW PARTICLE-TO-PARTICLE HEAT EXCHANGER

Heat Transfer Testing and Results

Methods and Materials

As shown in the review of literature, the concept of counterflow particle-to-particle heat exchange appears promising with respect to heat recovery capability. Little is known, however, about potential performance characteristics of a device which accomplishes this type of heat transfer. In order to predict the effects of such variables as heat transfer medium-to-grain mass ratio, heat exchanger length, and outlet wheat temperature on heat exchanger performance, an analytical tool of some nature is required. For this application, a digital computer simulation based on a mathematical model was chosen. Due to a lack of consensus on solid particle heat transfer model structure and the unique nature of this process, a decision was made to develop a new model for this work.

A practical simulation of the discrete solid particle heat exchange process required information on process parameters such as heat transfer coefficients and heating efficiency. Based on this need, a set of experiments was designed to examine the heating process as it might occur in a prototype device. A calorimeter was built following the general size and shape of a physical model which had been constructed to test the discrete counterflow concept (see Chapter V for details of physical model). The calorimeter was utilized to determine basic heat transfer information.

The 10.2 cm x 15.2 cm x 15.2 cm calorimeter was constructed of 0.635 cm plywood and 2.54 cm polyurethane sheets (Figure 4). On two opposite sides of the box, insulated plates were mounted to help simulate two particle masses flowing together. An insulated separation chamber was constructed using the polyurethane sheet material. The chamber was in two sections, each divided by a 10 mesh screen with 0.0635 cm wire. Five Type T thermocouples were placed near the base of each section. These thermocouples, along with others for measuring ambient conditions and initial particle temperatures, were connected to a data recorder with digital temperature and time output.

Wheat was used as the grain, and table salt and Norton-Alcoa Interprop were used as the heat transfer media. Common table salt has a bulk density of 2178 kg/m^3 and a specific heat of 0.837 kJ/kg ^{o}C. Interprop, an

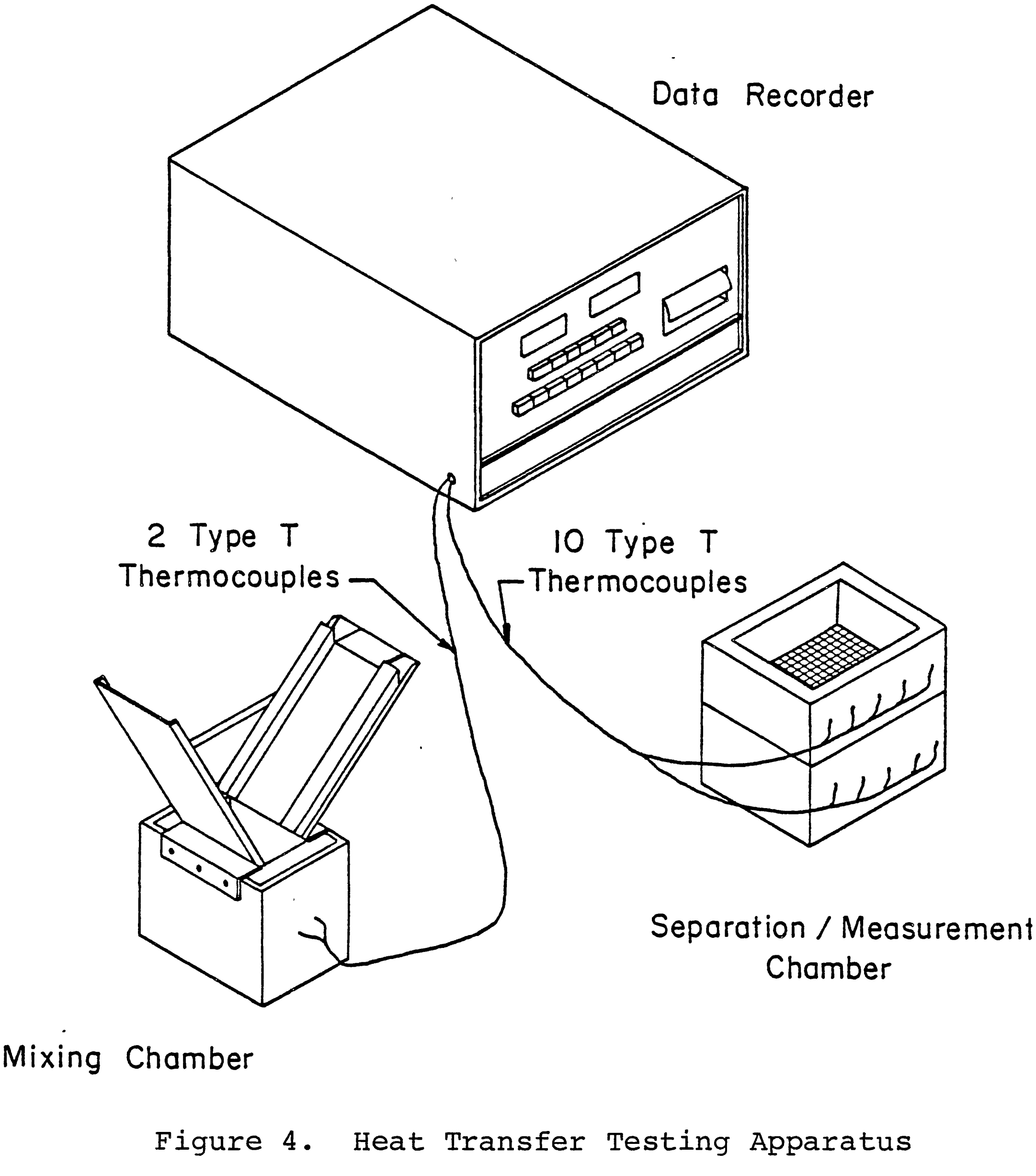


Figure 4. Heat Transfer Testing Apparatus

aluminum oxide product with a bulk density of approximately 3204 kg/m^3 and specific heat of 1.005 kJ/kg ^{o}C, is utilized by the oil industry in well fracturing. Interprop has a sphericity of 0.95 and a particle size of 90% in the 20-40 mesh range. Heat transfer media-to-grain mass ratios of 2:1, 3:1, 4:1, and 5:1 were used.

The following test procedure was used. Samples of heat transfer media, from 0.9 to 2.3 kg, were placed in an oven set at approximately 80 ^{o}C. Thermocouples monitored sample temperatures until they reached 74 to 76 ^{o}C. Each media sample was removed from the oven and its temperature recorded, along with the initial temperature of a 0.454 kg wheat sample. The two materials were then poured simultaneously into the calorimeter. No mixing was performed. For the majority of the tests, the mixture was allowed to remain in the calorimeter for 60 seconds. Other tests were conducted using 20 and 30 second holding times. Thermocouples were used to continuously record the mixture temperature. Due to the uneven temperature distribution within the mixture, the data from these thermocouples were not used in calculations. After the prescribed heating time, the mixture was poured into the separation chamber and manually sieved. Thermocouple readings from each chamber section were recorded during this sieving process to obtain final temperatures of both materials.

The following equation, derived by Downs et al.

(1977) using the classical Newtonian cooling equation, was utilized to calculate a heat transfer coefficient for each media/mass ratio pair:

$$h = \frac{-\ln\left\{1 - \dfrac{(mcm + mcw)(Twi - Twf)}{mcm(Twi - Tmi)}\right\}}{\dfrac{Aw\ (mcm + mcw)\ t}{(mcm)(mcw)}} \qquad (4.1)$$

A more convenient parameter for use in the simulation is the overall heat transfer term, ua, calculated as follows:

$$ua = Aw(h)(tf) \qquad (4.2)$$

where tf = 60 seconds

Aw = surface area of wheat sample (m2)

Note that the final time, "tf", is not the same as the test time, "t". The final time is an estimate of when the materials reach the final steady-state temperature level. Using the time constants measured in the experiments, the salt and Interprop have attained 97.7% and 99.3%, respectively, of their temperature change within a 60-second period. Therefore, for convenience, 60 seconds was used for tf in (4.2).

Results

As described above, most tests were run using 60 second holding times in the calorimeter. However, when tests using shorter holding times (20 and 30 seconds) were conducted, it became apparent that the wheat had attained its largest temperature increase before 60 seconds. Figure 5 demonstrates how the heat loss to the environment begins affecting the heat gained by the grain sample within a 60 second period. The data show the majority of heat transfer occurs within the first 30 seconds. In order to improve the accuracy of the heat transfer coefficient calculation, a "t" of 30 seconds was used in (4.1).

Heat transfer coefficient and efficiency results can be seen in Table II. Efficiency was calculated as the amount of energy absorbed by the wheat divided by the amount of energy released by the heat transfer medium. Time constant results can be seen in Table III. The time constant was calculated by fitting the two summarized data points to a first-order response curve.

Salt effectiveness in heating increased as mass ratio increased, yet the highest heat transfer coefficient attained in the Interprop tests occurred with the lowest mass ratio. The higher volume of salt resulting in better mixing is most likely the reason for the former phenomenon, but the explanation of the latter is not clear. Based on observation during the test procedure, the large coefficients of variation are due to nonuniformity in mixing.

The work by Downs, et al. (1977) resulted in an empirical model for agitated particle-to-particle heat transfer. To further investigate the mixing affect, this model was used to calculate the heat transfer

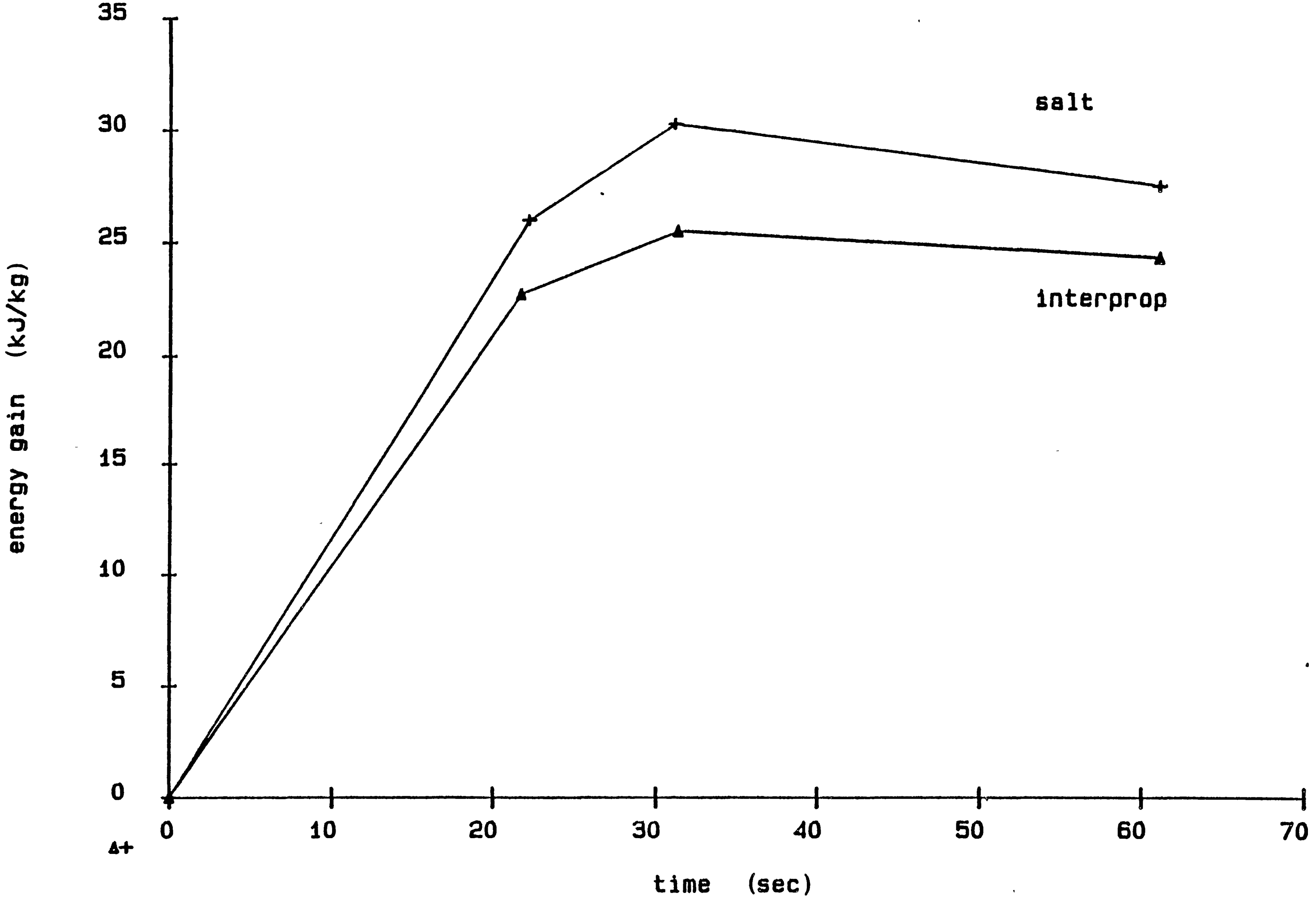


Figure 5. Results of Time Constant Analysis

TABLE II

HEAT TRANSFER COEFFICIENT RESULTS

Media	Mass Ratio (m:w)	q (kJ/kg)	eff (%)	h (W/m^2 °C)	ua (kJ/°C)	cv
Interprop	3:1	25.8	90.5	17.8	0.746	22%
	4:1	24.2	81.1	13.5	0.560	18%
	5:1	25.8	74.9	13.8	0.579	21%
Salt	2:1	16.7	74.8	9.9	0.412	38%
	3:1	24.0	69.8	13.9	0.581	26%
	4:1	27.4	72.2	16.9	0.710	38%
	5:1	36.3	82.3	24.8	1.309	32%

TABLE III

TIME CONSTANT RESULTS

Media	Mass Ratio (m:w)	t1 (sec)	q1 (kJ/kg)	t2 (sec)	q2 (kJ/kg)	tau (sec)
Interprop	4:1	22.2	25.9	31.2	30.2	15.9
Salt	4:1	21.7	22.6	31.3	25.4	12.0

coefficient for a salt/wheat mixture with a mass ratio of 4:1. The following equation was used.

$$Nu = 18.9(Dw,e/Dm,e)^{0.828}(u)^{4.11}(w/m)^{-0.388} \qquad (4.3)$$

where Nu = Nusselt number

Dm,e = dimensionless shape factor of salt

Dw,e = dimensionless shape factor of wheat

um = angle of repose of salt (rad)

w = density of wheat (g/cm^{3})

m = density of salt (g/cm^{3})

Properties of salt were obtained from Downs, et al. (1977), while wheat properties were obtained from Mohsenin (1979). Note that (4.3), in its original form, also contained the Froude number. However, this term was not utilized in the calculation since the particle-to-particle velocity in this case is zero. Although some error is surely introduced by not using the Froude number term, Downs, et al. concluded that due to the small exponent (0.077) it may be negligible. The heat transfer coefficient predicted by the Downs model was 326.5 W/m^{2} ^{o}C. In the above tests with no mixing, however, the heat transfer coefficient was measured as 16.9 W/m^{2} ^{o}C. The significant contribution that agitation makes to heating effectiveness is apparent in this comparison.

Heat Exchanger Simulation

A mathematical model was developed that predicts both the steady-state and transient responses of the temperature profiles of two solid particle masses held in an insulated cell. The model was then incorporated into a simulation for calculating the time-temperature profiles within a discrete particle-to-particle counterflow process. Figure 6 depicts such a process.

Model Development

The data from the heat transfer tests clearly show that a simple first-order model is inadequate in predicting the dynamic response of the process. The temperatures of the the two materials are initially approaching separate asymptotes and not an equilibrium point (Figure 7). The data suggest that heat transfer is more rapid from the salt to the wheat than through the salt. If the equilibrium temperature of a mixture of two solids was the same as the ambient condition, then the process would appear as in Figure 7, with one time constant much smaller than the second. Balancing the mixing or holding time of each discrete heat transfer element versus the amount of energy transferable during that time indicates that the heating/cooling duration should be much closer to the smaller time constant. Therefore, a first-order model may be sufficient if the following constraints are met: (1) the model must utilize

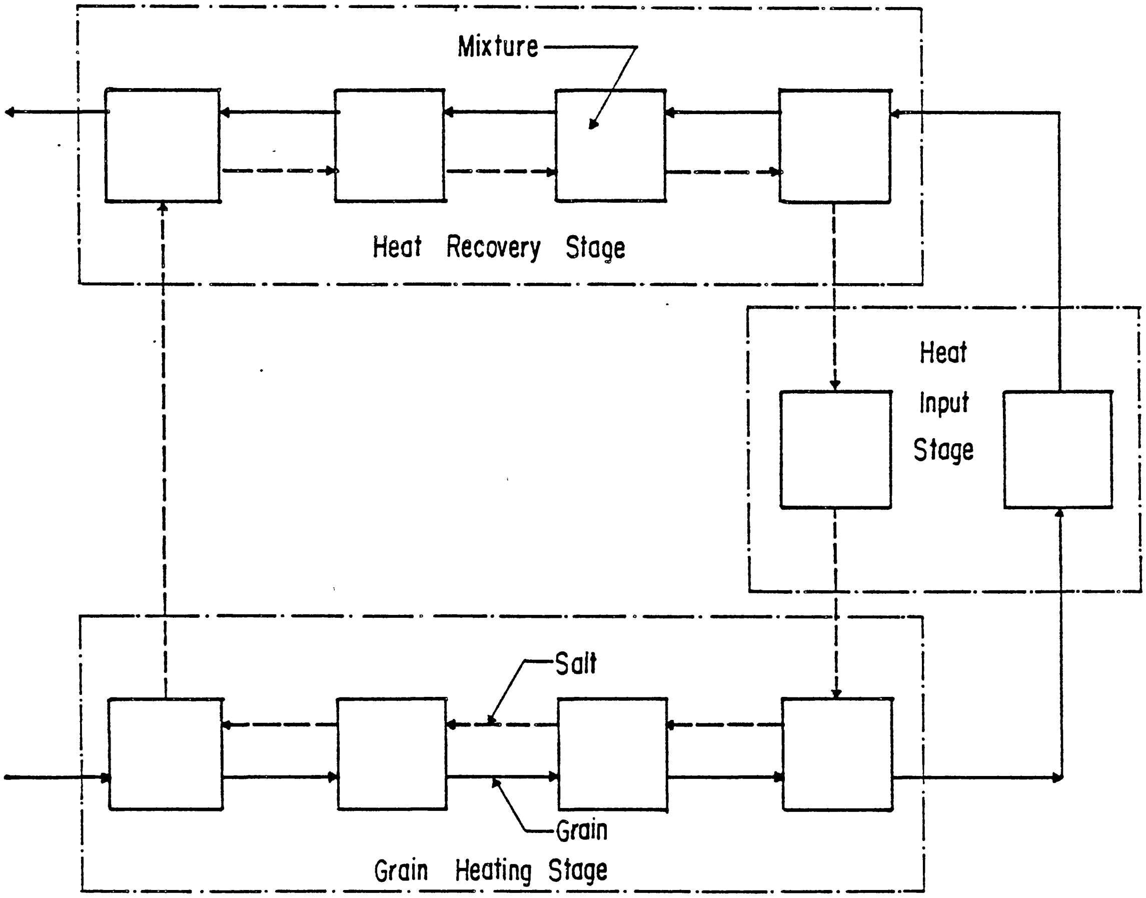


Figure 6. Discrete Counterflow Particle-to-Particle Heat Exchange

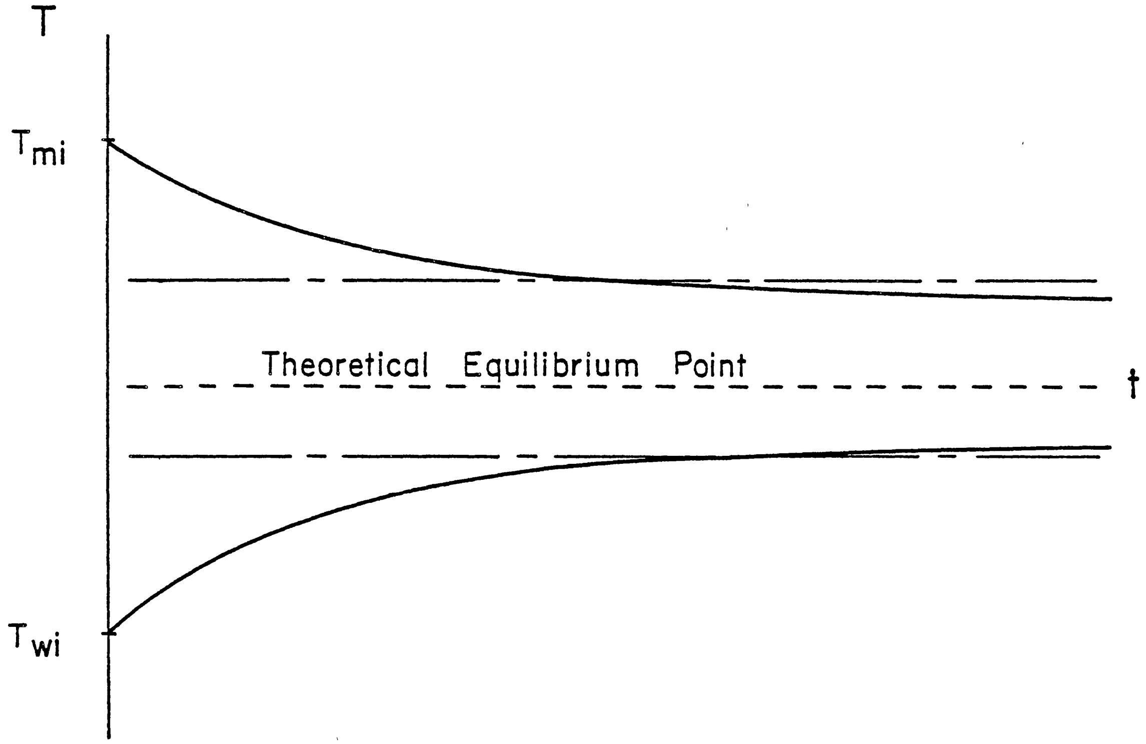


Figure 7. Temperature Profile of Two Mixed Solids

the heat transfer coefficients, time constants, and media-to-wheat energy efficiencies determined in the tests; (2) the model must predict different temperature equilibria (i.e. asymptotes) for the two materials.

Beginning with Newton's law of cooling, the following equations can be written for an idealized case:

$$\frac{dTw}{dt} = \frac{h(Aw)(Tm - Tw)}{mcw} \qquad (4.4)$$

$$\frac{dTm}{dt} = \frac{h(Aw)(Tm - Tw)}{mcm} \qquad (4.5)$$

The above equations alone do not fit the constraints since a common equilibrium point is predicted. Subtracting the above equations yields

$$\frac{d(Tm-Tw)}{dt} = \frac{-h(Aw)(mcw + mcm)(Tm - Tw)}{(mcw)(mcm)} \qquad (4.6)$$

which has the solution

$$(Tm-Tw) = (Tmi-Twi)e^{kt} \qquad (4.7)$$

where $k = \frac{-h(Aw)(mcw + mcm)}{(mcw)(mcm)}$

Assuming a finite time interval and a constant heat transfer coefficient over that interval, taking the logarithm of the above yields

$$\ln\left\{\frac{(Tmf - Twi)}{(Tmi - Twi)}\right\} = \frac{-ua(mcw + mcm)}{(mcw)(mcm)} \qquad (4.8)$$

where ua=h(Aw)tf.

Note that for the final salt temperature to approach the final wheat temperature, "ua" must be infinitely large. A heat balance, including an efficiency term (z) which represents the portion of energy transfered from the medium to the grain, gives

$$(Twf - Twi) = \frac{z(mcm)(Tmi - Tmf)}{(mcw)} \tag{4.9}$$

Equations (4.8) and (4.9) constitute a two equation/two unknown set, with the initial conditions known and the final conditions unknown. Solving for the final temperatures,

$$Twf = \frac{z(mcm)(1 - e^{-x})}{z(mcm) + mcw} Tmi + \frac{(mcw + z(mcm)(e^{-x})}{z(mcm) + mcw} Twi \tag{4.10}$$

$$Tmf = \frac{(zmcm + (mw)e^{-x})}{z(mcm) + mcw} Tmi + \frac{mcw(1 - e^{-x})}{z(mcm) + mcw} Twi \tag{4.11}$$

where $x=ua(1/mcw + 1/z(mcm))$

Finally, utilizing (4.10) and (4.11) as the asymptotes of a first order response and substituting into a standard solution to a first order differential equation yields:

$$Tw(t) = Twi + \frac{z(mcm)(1 - e^{-x})(Tmi - Twi)(1 - e^{-ket})}{z(mcm) + mcw} \tag{4.12}$$

$$Tm(t) = Tmi + \frac{mcw(1 - e^{-x})(Tmi - Twi)(1 - e^{-ket})}{mcw + z(mcm)} \tag{4.13}$$

where

ke = inverse of time constant.

Note that the only difference between the model made up of (4.12) and (4.13) and a model made up of the solution to

(4.4) and (4.5) is the $(1-e^{-x})$ term, which shifts the equilibrium points of the two materials.

Simulation Algorithm

The discrete counterflow heat exchanger is assumed to have two stages, one for heating the grain and the other for reclaiming energy. Each stage is made of a number of cells (Figure 6). All heat transfer is assumed to take place within the cells and not during material flow. Each cell initially contains a mixture of wheat and the heat transfer medium. The centermost medium temperature, located in the energy input section in Figure 6, is held constant at some elevated setpoint and the medium recirculates throughout the machine. Wheat enters the first (heating) stage and exits the second (cooling) stage.

The steps in the algorithm are as follows. The state (temperature) of each material within each cell is calculated using equations (4.12) and (4.13). The states of the cells are shifted, wheat in one direction and heat transfer medium in the other. The states are then recalculated. This process continues until a steady-state temperature profile is reached.

The algorithm was implemented in Turbo Pascal on an IBM PC-XT microcomputer. The software, CFLOW, can be found in the Appendix. All heat transfer results (heat transfer coefficients and time constants) were incorporated in the model. Inputs include medium type, mass ratio, and number of cells per stage. The program output gives the state of each material in each cell over time.

Simulation Results

The simulation was used to examine the effects of medium/grain mass ratio, medium/grain heat capacity ratio, number of cells per stage, and outlet wheat temperature on energy input. It is presumed that the major performance measure of either a practical or theoretical machine is the minimization of input energy to the device while maintaining the setpoint temperature of the grain.

Factors held constant throughout all runs were: wheat temperature setpoint = 65.5 ^{o}C; initial wheat temperature = 21.1 ^{o}C; heat transfer efficiency within a cell = 80%; holding time within each cell = 30 seconds. In all the runs, the required media temperature in the center section to maintain the wheat setpoint within plus or minus 0.5 ^{o}C was found iteratively. Energy input efficiency to the heat transfer medium was assumed to be 100%. Although not realistic, this assumption only affects the actual amount of energy input required to raise the medium temperature to its desired value in the center section.

The first parameter investigated was the medium/mass ratio. For both Interprop and salt, required energy input increased as the medium-to-grain-mass ratio increased (Figure 8). Machine configuration was set at 10 cells per

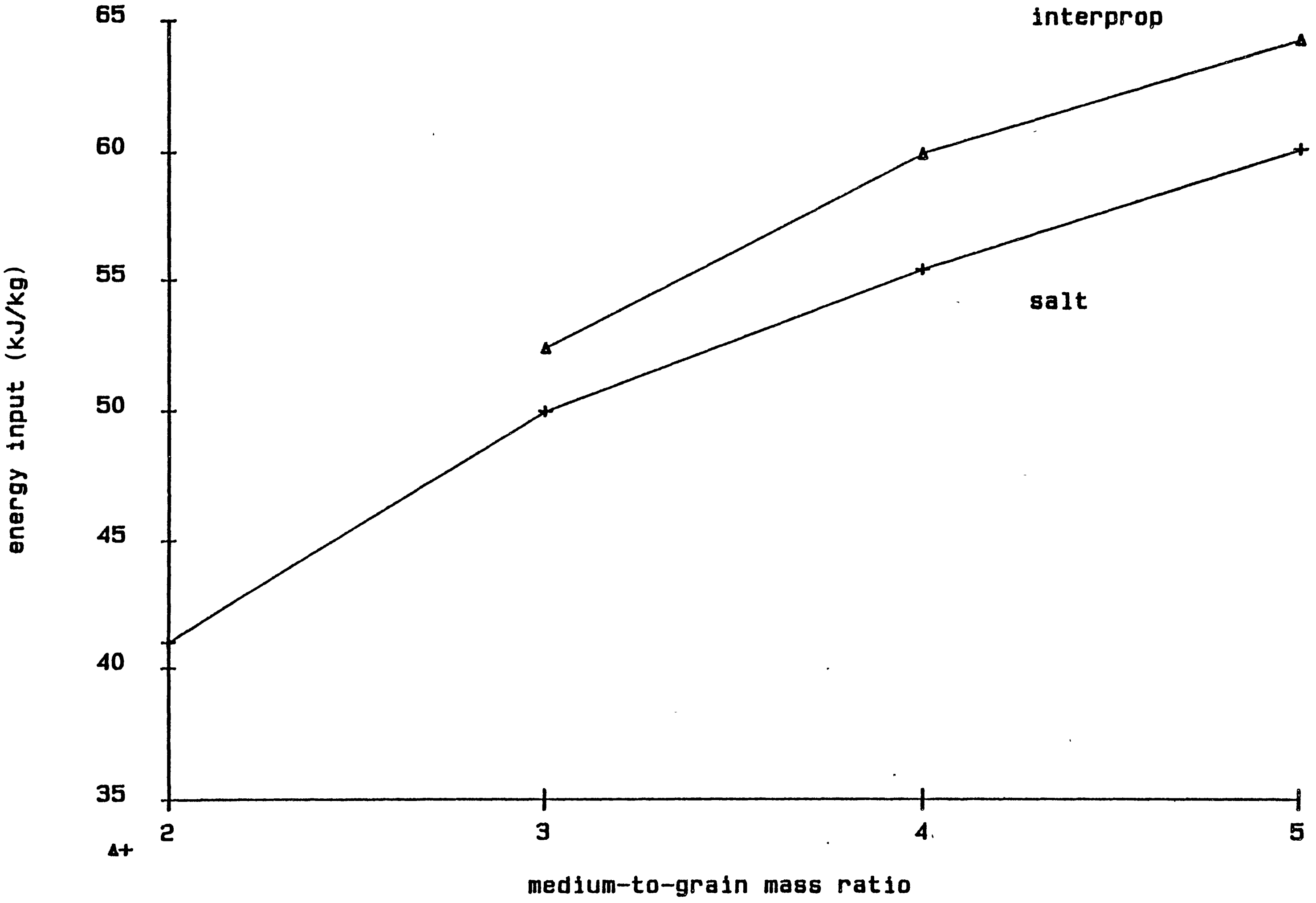


Figure 8. Mass Ratio Effect on Energy Input

bank. This relationship seems to occur in spite of the fact that, for salt, heat transfer coefficient increases as the mass ratio increases. However, the data indicate a penalty for high heat capacity of the medium and the greater heat transfer coefficient of the salt was unable to overcome this penalty.

The number of cells significantly affects machine performance. Examples of the temperature profiles predicted by the simulation can be seen in Figures 9 and 10. The difference in required maximum medium temperature to achieve the same wheat temperature at the center of the machine can easily be seen. Salt at a mass ratio of 3:1 was used. Cell numbers ranged from 5 to 30 cells per stage. Figure 11 shows that as the number of cells increased, required energy decreased. However the simulation does not take into account possible greater heat loss due to larger machine size. The decrease in required energy is great from the 5 cell/stage machine to the 15 cell/stage machine, but greater cell numbers produce a declining benefit.

To investigate the effect of medium heat capacity on energy input, data was obtained by running the simulation with various medium-to-wheat heat capacity ratios. Ten cells per bank were used in all runs. Medium-to-wheat heat capacity ratios were varied from 0.5 to 1.50. An overall heat transfer coefficient of 0.57 kJ/C, approximately that of salt at a mass ratio of 3:1, was used. The simulation

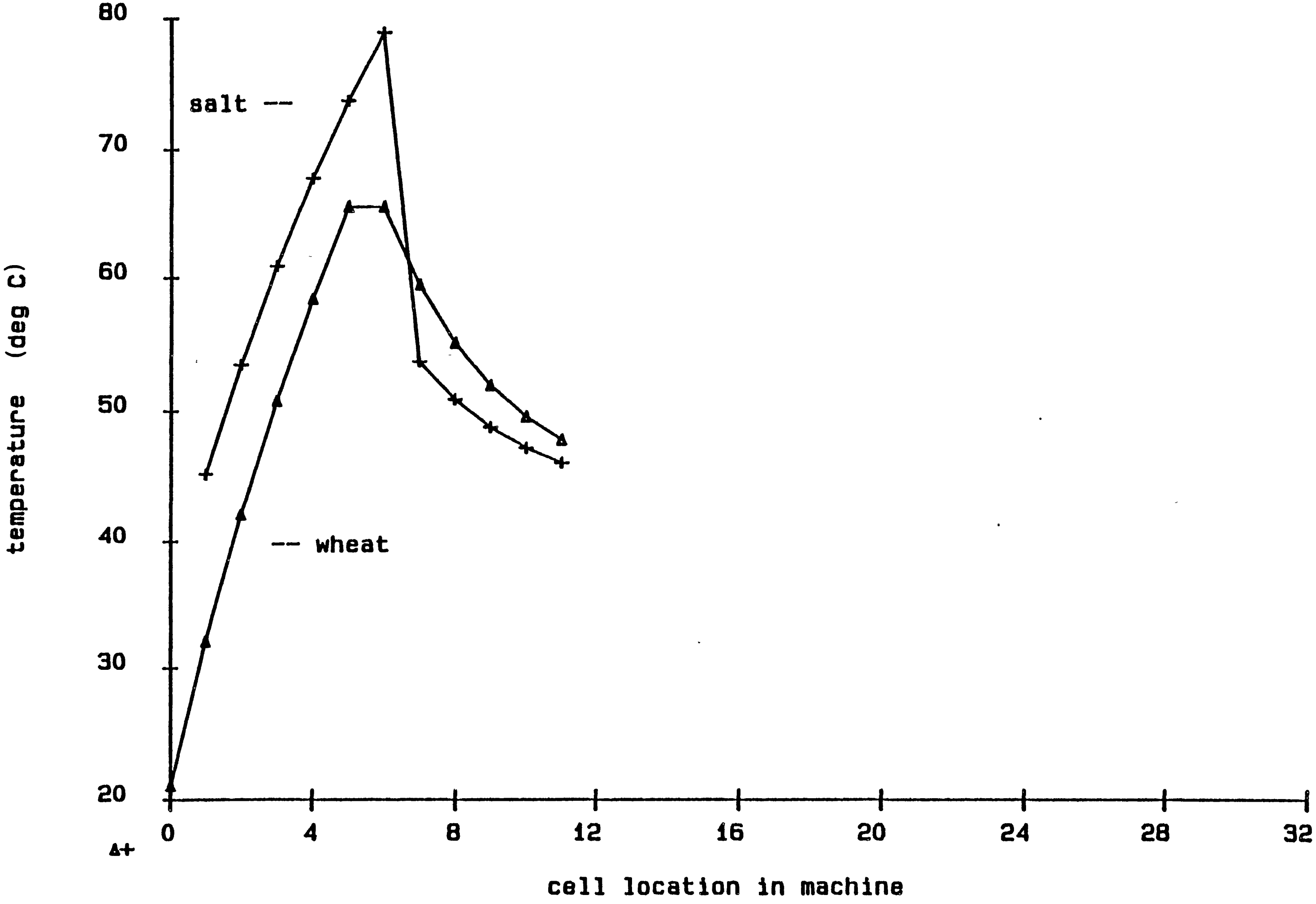


Figure 9. Temperature Profile Within a 5 Cell/Stage Machine

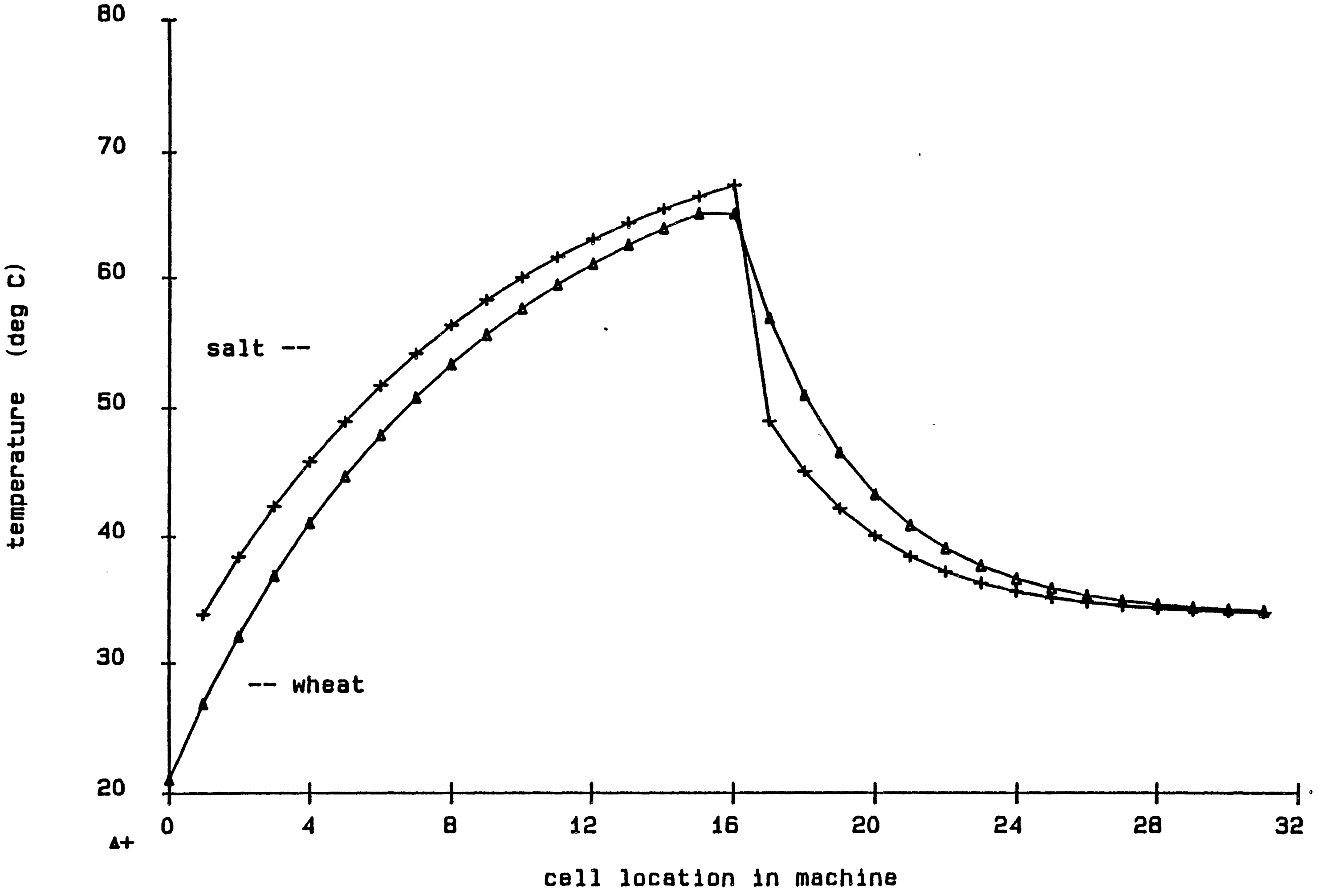


Figure 10. Temperature Profile Within a 15 Cell/Stage Machine

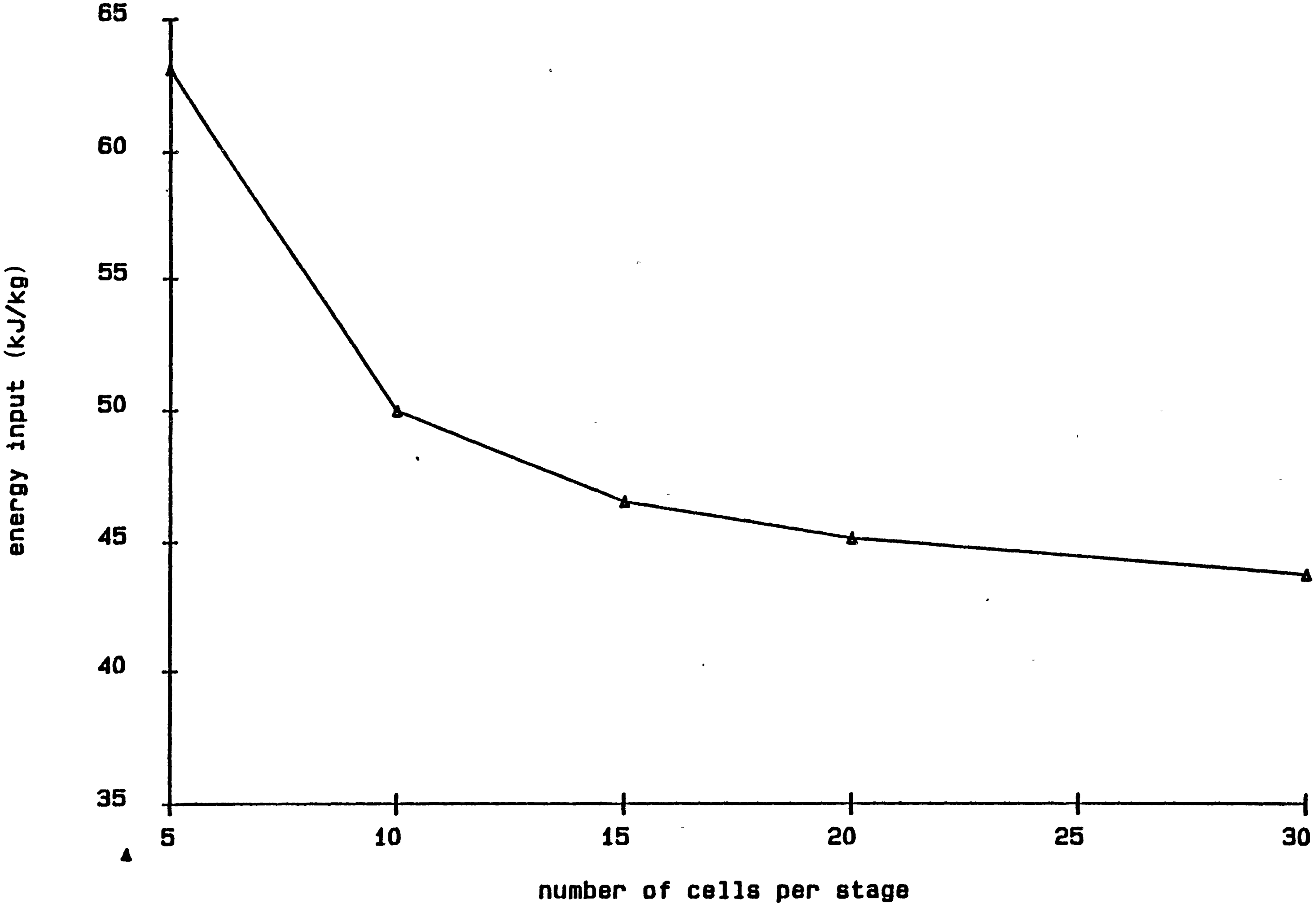


Figure 11. Effect of Cell Number on Energy Input

indicates that optimal medium heat capacity is approximately the same as that of the grain (Figure 12). For instance, if the overall heat transfer coefficient (ua) is relatively constant for a particular medium in the lower mass ratios, the optimal mass ratio would be equal to the ratio of the medium-to-wheat specific heat. For salt, this value would be approximately 2:1. However, at low mass ratios poor mixing is most likely to cause lower values of the heat transfer coefficient.

The relationship between outlet wheat temperature and energy input was investigated. A data set was formed consisting of the majority of previous runs. These included tests with salt, Interprop, and the variable heat capacity. Neither cell numbers nor mass ratios were held constant. Seventeen data points were used. Figure 13 shows the linear relationship. The following heat balance verifies the result:

$$E = \frac{cp(Tmax - Twi)}{z} - cp(Tmax - Tout)z \qquad (4.14)$$

The above equation relates the difference in energy required to heat the grain (first stage) and the energy released by the grain (second stage). Manipulating (4.14) to fit a standard linear form yields:

$$E = \frac{cp(Tmax)(1 - z^2) - cp(Twi)}{z} + cp(Tout)z \qquad (4.15)$$

Energy input is thus minimized by minimizing the wheat temperature exiting the machine.

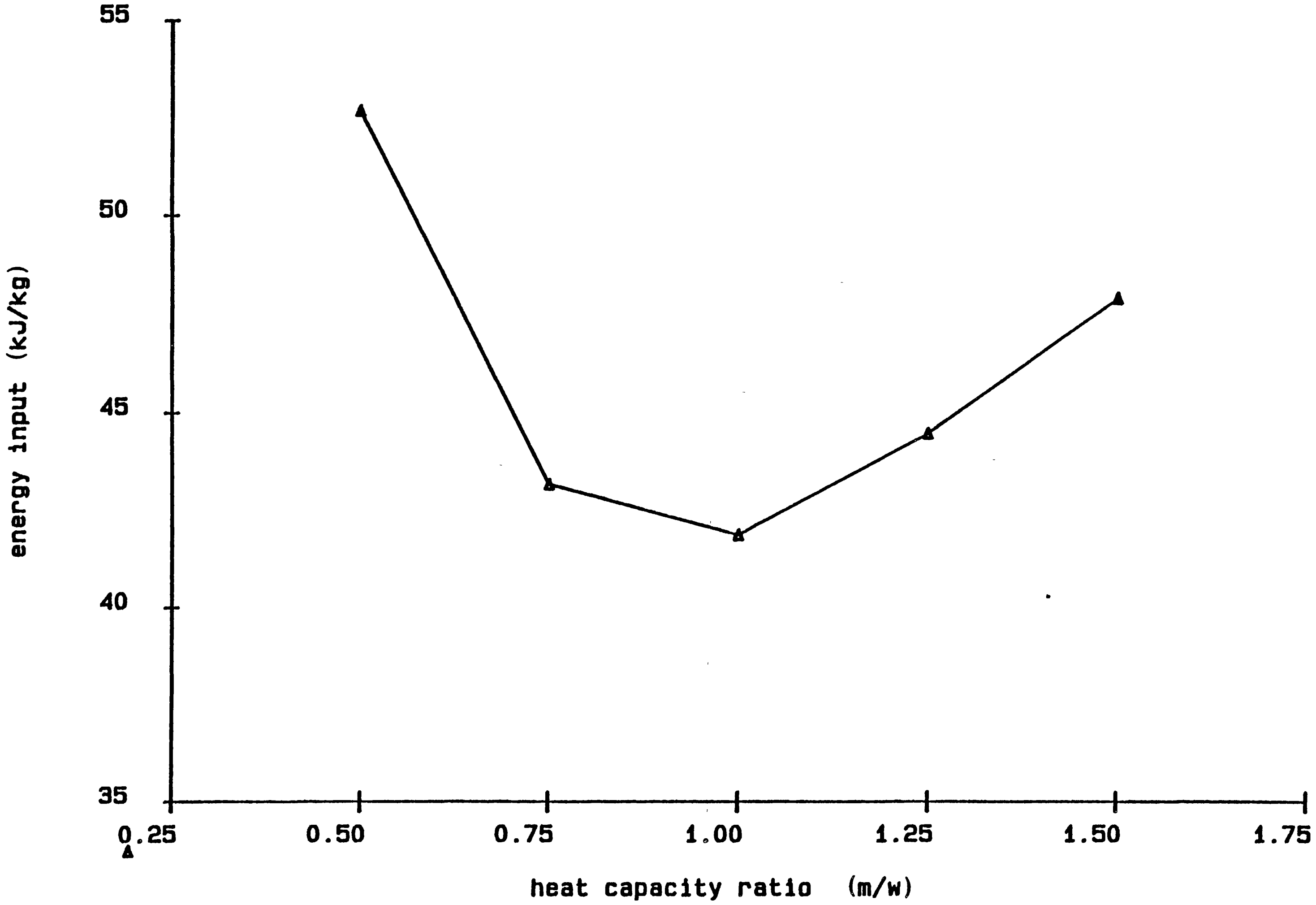


Figure 12. Effect of Heat Capacity Ratio on Energy Input

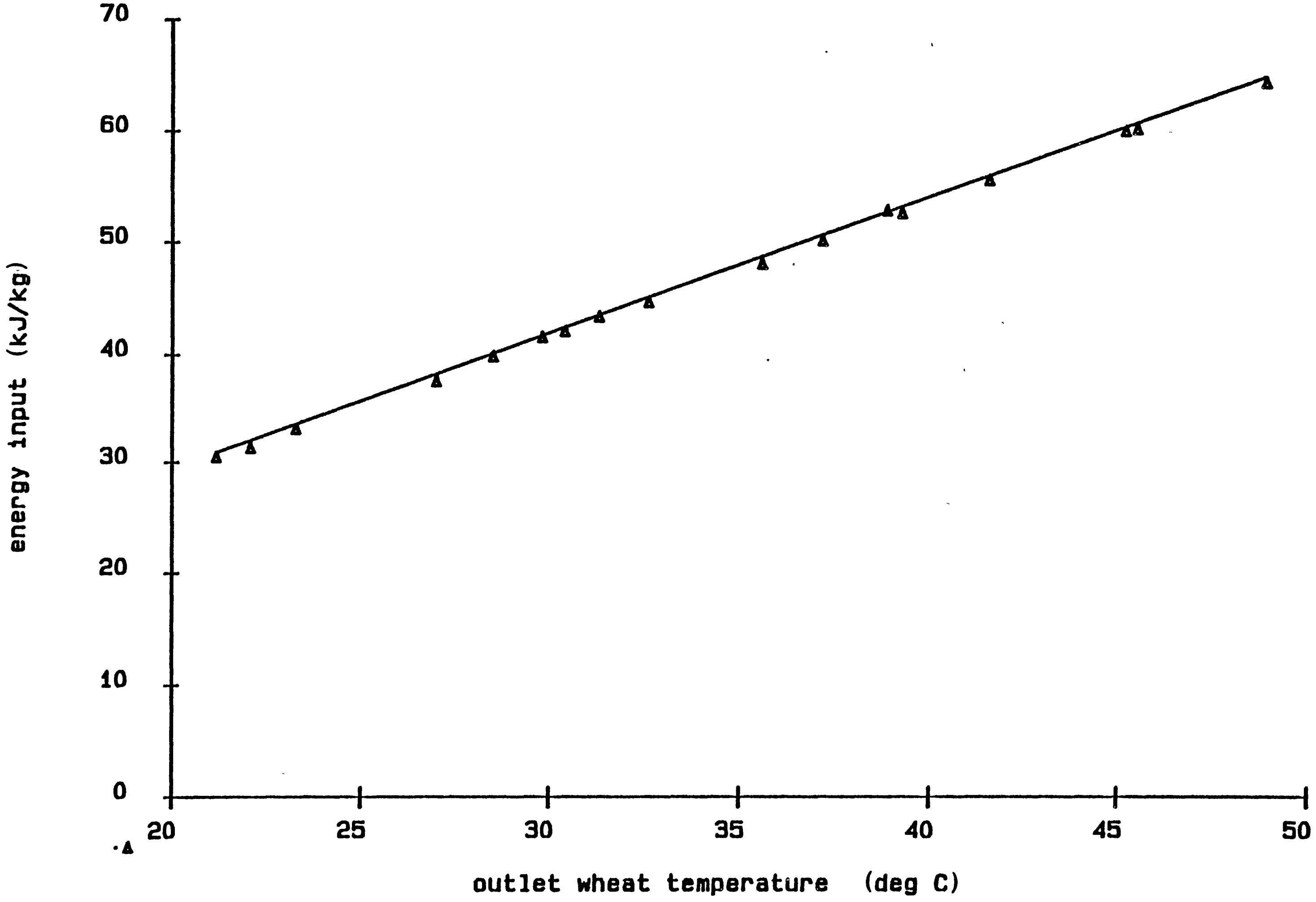


Figure 13. Effect of Outlet Wheat Temperature on Energy Input

CHAPTER V

STEADY-STATE PERFORMANCE EVALUATION OF A PROTOTYPE

Solid Particle Heat Exchanger Design

In Chapter IV, the potential performance of a solid particle counterflow heat exchanger was explored. Due to the positive results predicted by the simulation, the next step would naturally be to attempt to construct a prototype and analyze its operational characteristics. This chapter covers the design, construction, and testing of a prototype.

Basic Operation

To handle the dual-purpose task of heating the grain and then reclaiming the energy, the heat exchanger must have two separate banks, or stages (Figure 14). The temperature profile of such a device using a recirculating heat transfer medium can be seen in Figure 3. Energy is added to the center portion of the heat exchanger located between the two stages. This energy replaces only heat lost due to inefficiency in the system, i.e. that amount of energy not reclaimed in the second stage.

The approach taken to implement a counterflow heat

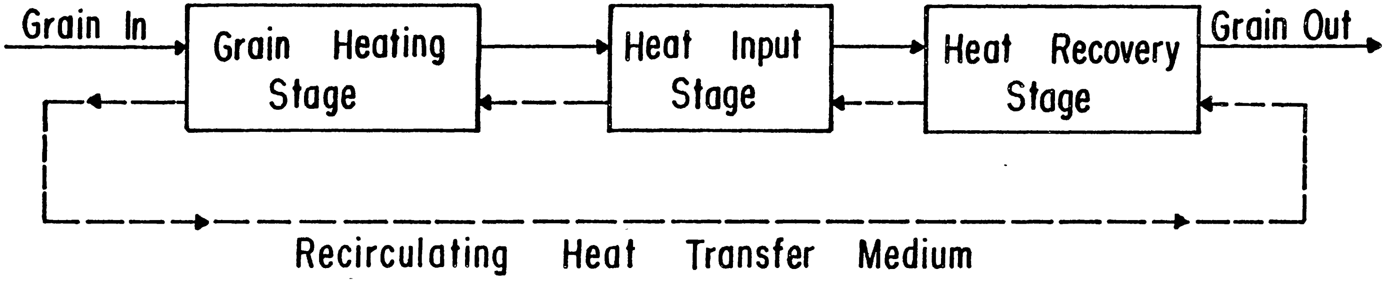


Figure 14. Process Flow Diagram of Heat Exchanger

exchange temperature profile on particulate solids was that of discrete and repeated unit operations of mixing, heating, and separating. Many possible mechanical configurations were considered. A mechanism which utilizes gravity-induced flow over an inclined screen was chosen for the separating portion of the unit due to a lack of moving parts required.

Design Concept

Figures 15 through 19 show cross-sections of the design and depict internal flow. White areas are flowpaths for the grain and heat transfer medium, gray areas are solid and exclude flow, and dashed lines represent screens. In Figure 15, the machine is upright. A holding area, located at the bottom, is filled with a mixture of a heat transfer medium and grain. As the machine rotates about its horizontal axis and the back of the machine reaches the mixture's angle of repose, the mixture begins to flow (Figure 16). The mixture flows downward, encounters a partition, and flows along the inclined screen (Figure 17). Figure 18 shows the separation. The smaller and more dense particles making up the heat transfer medium (salt) flow through the screen, and the grain flows along the screen. Ultimately, the mixture is separated into its two components, each having been forced to flow in opposite directions (Figure 19). As this process occurs in the cells shown, an identical process occurs on either side of the viewed

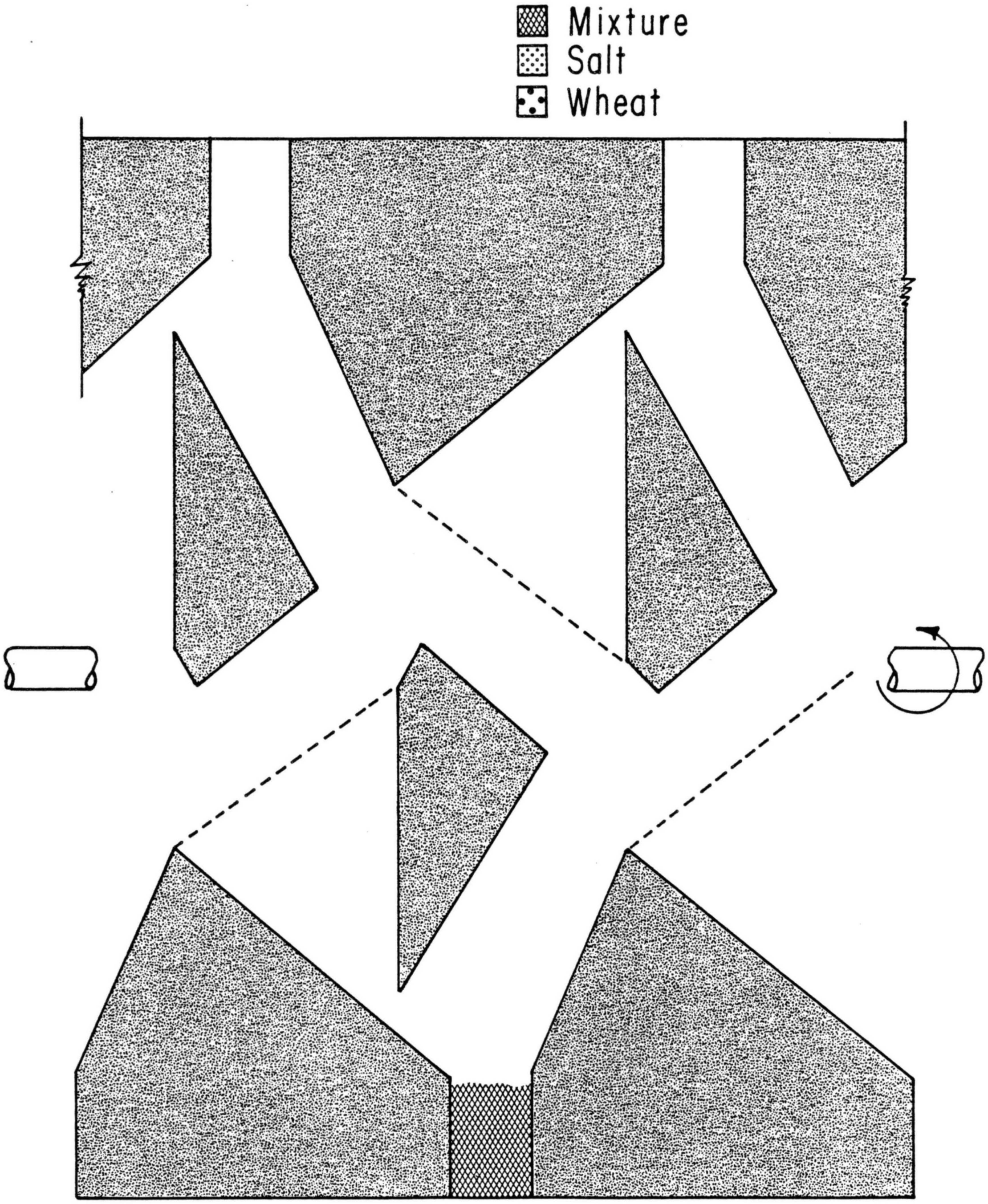


Figure 15. First Machine Cross-section Depicting Solids Flow

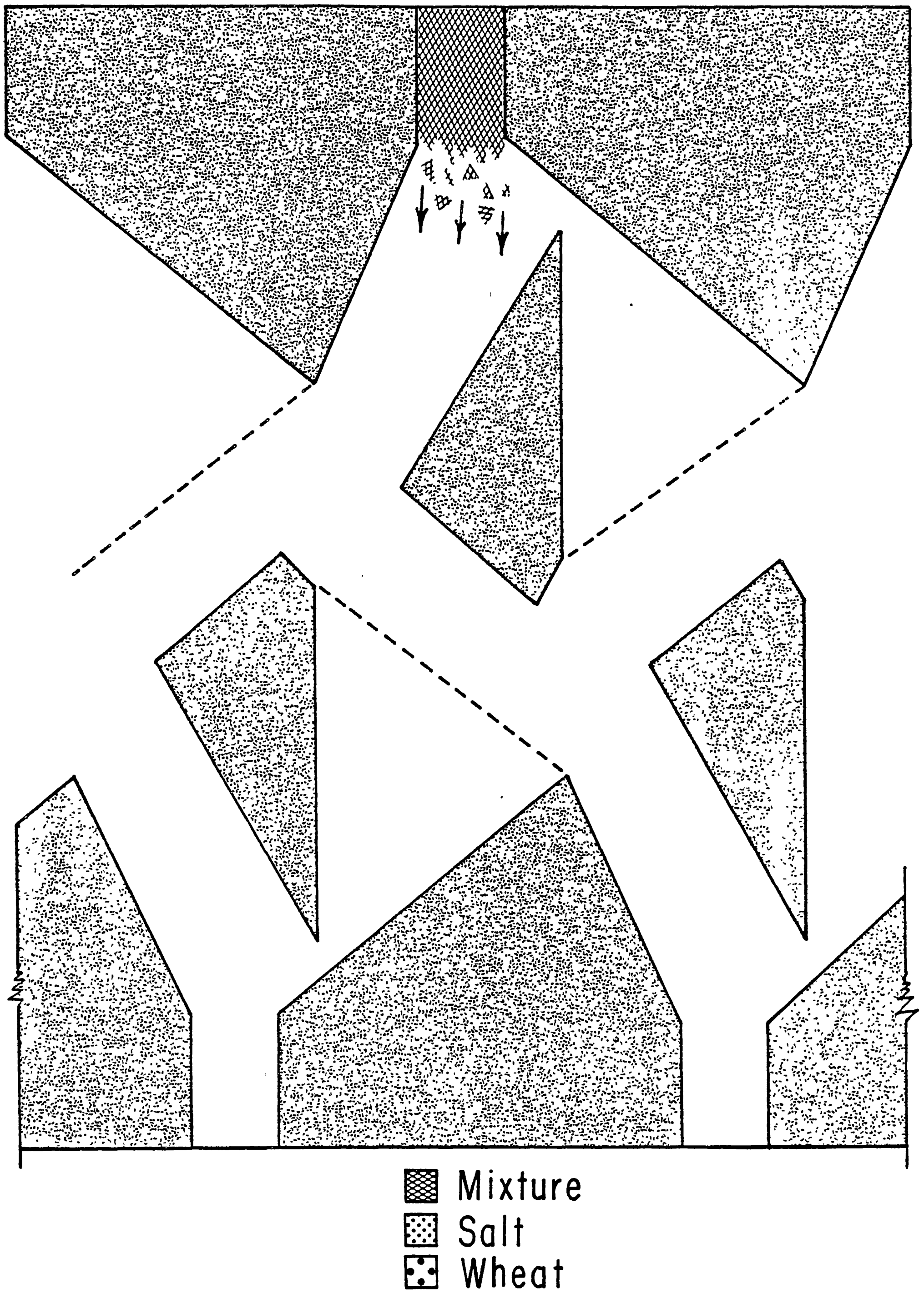


Figure 16. Mixture Beginning to Flow Out of Holding Area

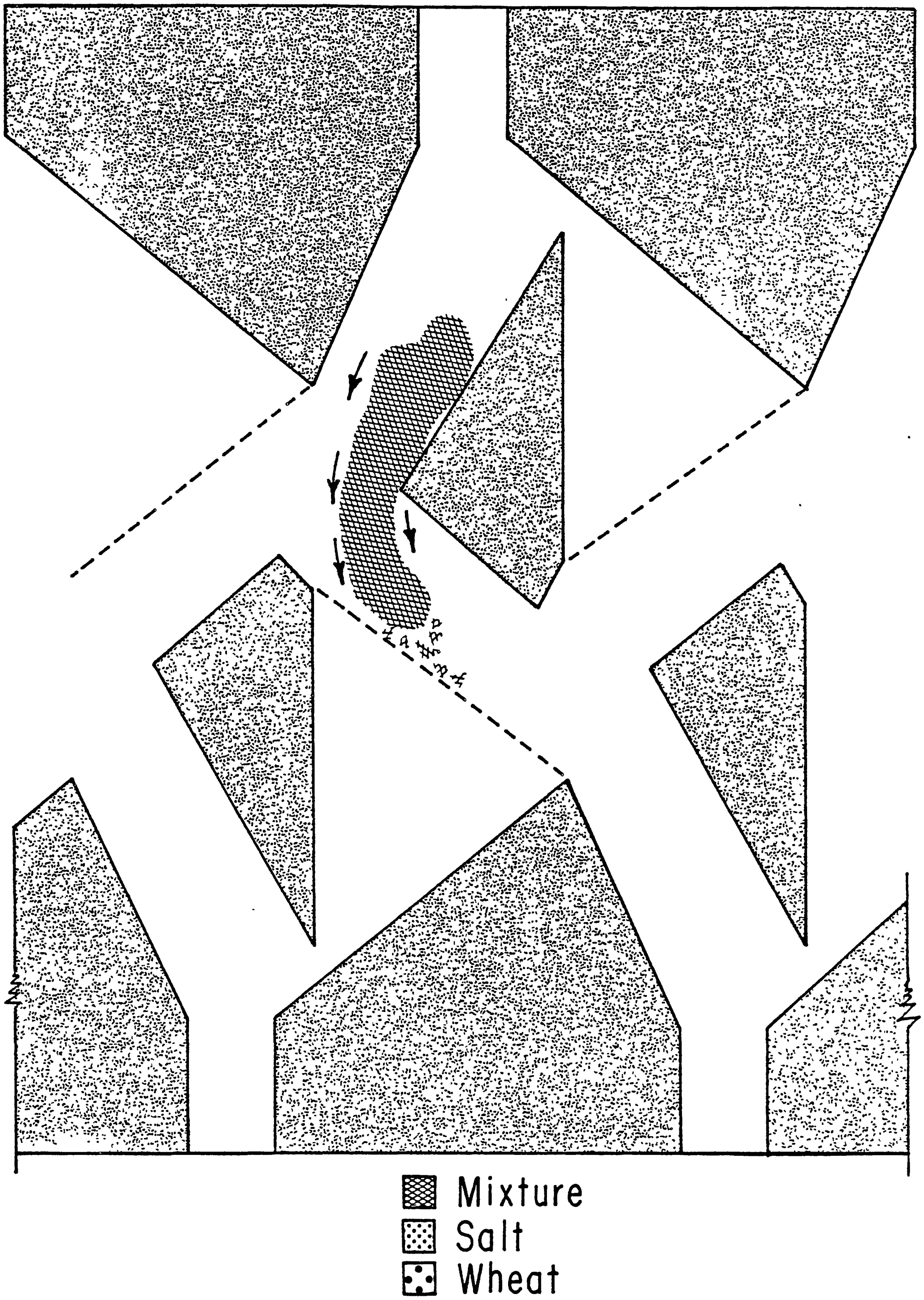


Figure 17. Mixture Flowing Towards Inclined Screen

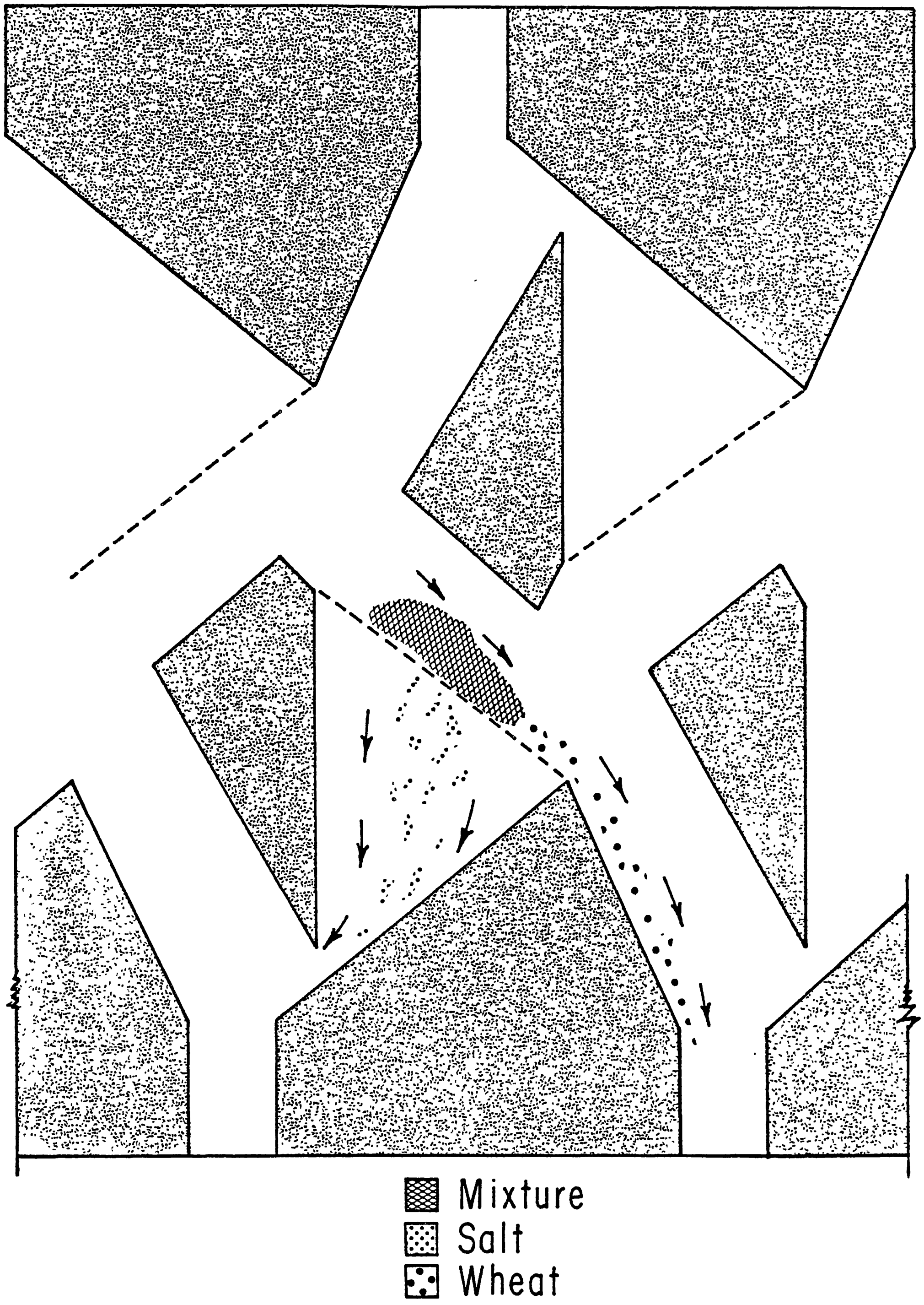


Figure 18. Mixture Flowing Along Screen Resulting in Separation

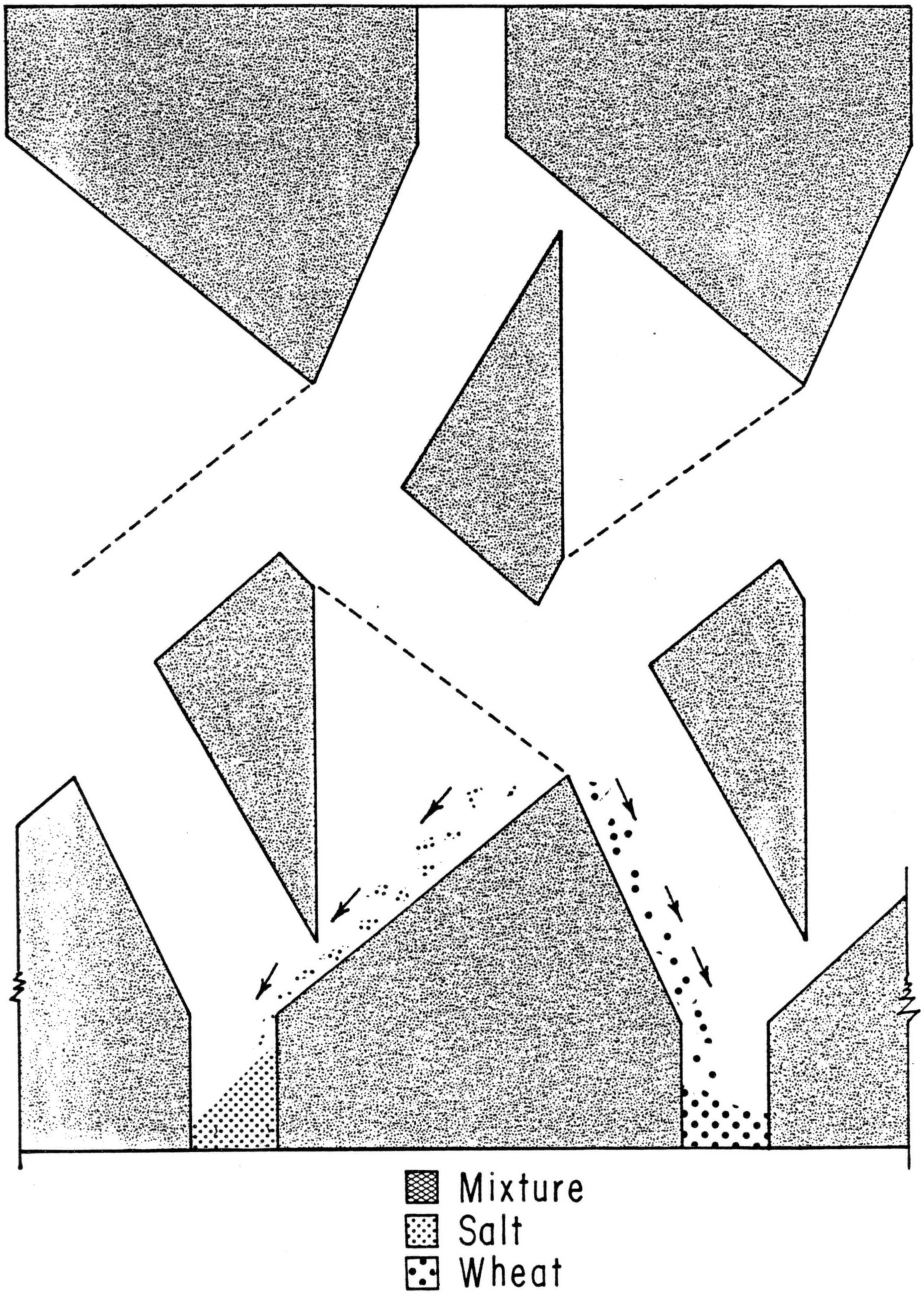


Figure 19. Separated Materials Flowing in Opposite Directions

cross-section. Therefore, the heat transfer medium which moved to the leftmost cell is mixed with grain coming in from the left. Likewise, the grain which moved to the rightmost cell is mixed with heat transfer media coming in from the right.

Development

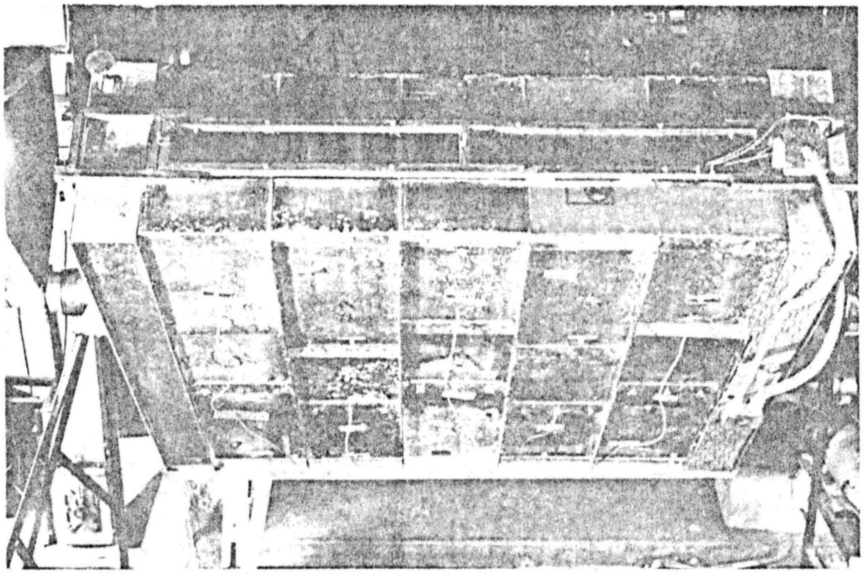
Tests were conducted using wheat and salt to determine proper screen angle, mesh, and wire size for maximum separation of a mixture. Replications included the following: screen angles of 30, 35, 40, and 45 degrees from the horizontal; salt-to-wheat mass ratios of 0:1, 2:1, 3:1, and 4:1; and six different screens. Results indicated an 8x8 mesh screen with 0.071 cm wire at 40 degrees from the horizontal should be utilized.

Based on the above results and machine cross-section, a single-stage three cell unit was designed and built. This manually rotated test unit was approximately 152 cm wide, 132 cm high, and 15 cm deep. Both salt and Interprop were tested in the unit for separation and mixing effectiveness with wheat. In general, good results were achieved. For salt, 91% flowed in the proper direction, as opposed to 98% for Interprop and wheat. Various adjustments were made, and observations of the flow patterns resulted in slight design alterations to be incorporated in a prototype.

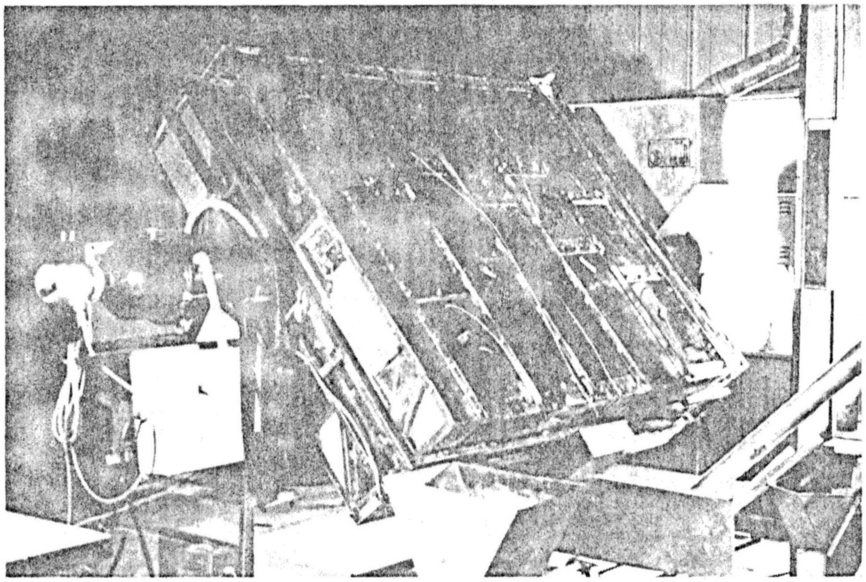
Prototype

The development work resulted in the design and construction of a two-stage particle-to-particle heat exchanger with five cells per stage (Figure 20). Sizing of internal geometry was done to allow 454 g/cell of wheat along with 1816 g/cell of salt. Intricate inlet and heating cells were designed to allow the two stages to be mounted back to back. Basic construction materials consisted of 24 and 28 gauge galvanized sheet metal, 2.54 cm x 2.54 cm steel tubing (14 gauge), 32 kg/m^3 urethane block, open cell sponge rubber, screen material, and a high temperature silicone sealant. The design was modular so that more cells could be added if necessary. Each module is 30.5 cm x 147.3 cm x 15.2 cm and is insulated using 1.9 cm insulation board with an R-value of 2.6 m $^{o}C/W$ (4.5 h ft $^{o}F/Btu$).

A 0.56 kW dc motor, reduced through gearbox and chain drive, rotates the machine. Electrical strip and cartridge heaters, rated at a total of 2.2 kilowatts at 240 volts, are located in the heating section. The required wattage is a function of the heat transfer medium's mass, specific heat, temperature increase, and allowable heating time. Size constraints due to internal flowpaths within the machine and standard electrical heaters limit the wattage which can actually be placed in the device. Only the heat transfer medium comes in contact with the heaters. Two sets of slip rings are mounted on



(a)



(b)

Figure 20. Top (a) and Front (b) Views of Machine

the heating end of the machine. One is used to transmit power to the heaters while the other is used to transmit instrumentation signals.

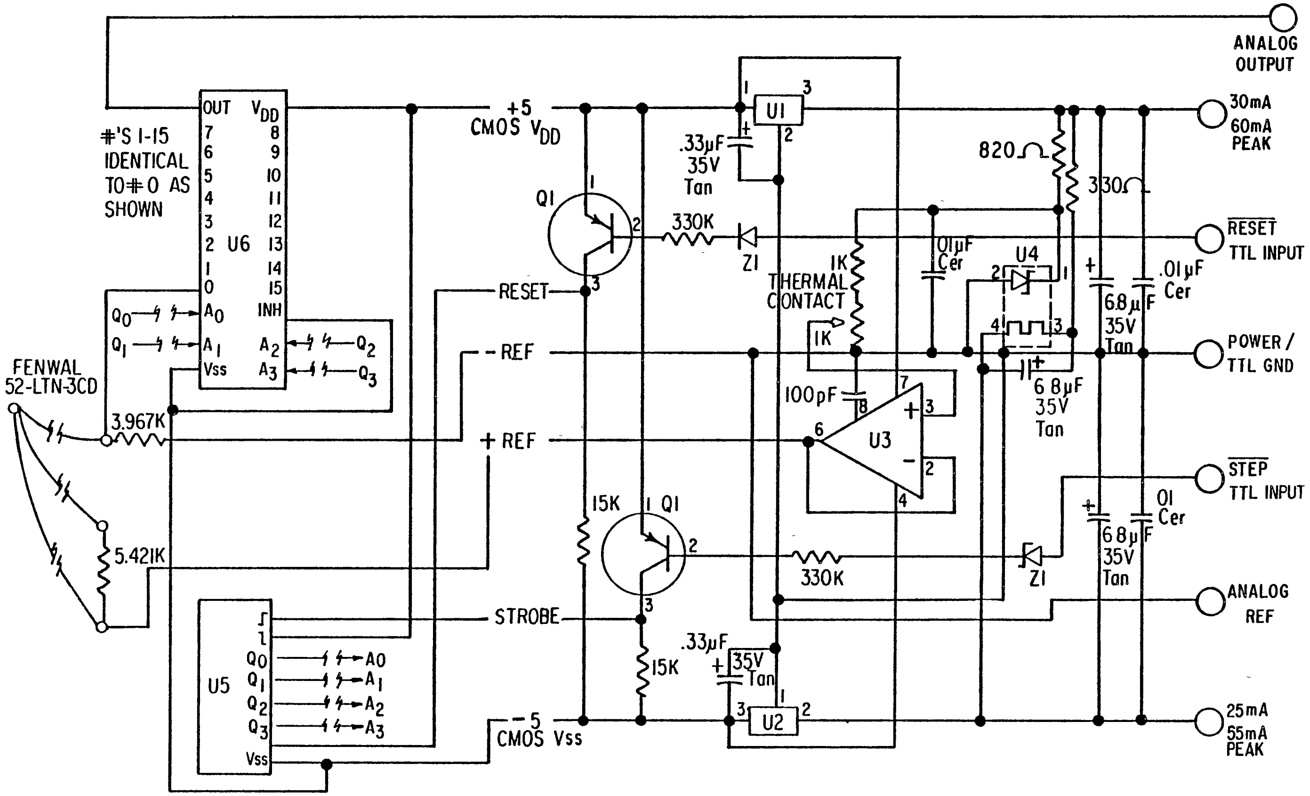
Electronic Requirements

Operation of the heat exchanger was controlled by a MC6809-based microcomputer. The central processing unit (cpu) was a CMS (Creative Micro-Systems) 9619A board with a custom PAL for 2K bytes of input/output space. A CMS 9671 disk controller coupled to two Mitsubishi double-sided, double-density disk drives served to provide permanent storage capabitility. For RAM, a CMS 9638-1 board with 64k bytes of available static memory was utilized. An Analog Devices RTI-1231-R board was used for analog input. Digital input and output capability came from two MC6821 peripheral interface adapters located on the cpu board. A CMS 9640A board with 8 MC6840 programmable timers provided timing capability, along with a real-time clock on the cpu board. The FLEX^{TM} operating system was used as the major software shell for the system. Programming capabilities included 6809 assembler, Basic, and Pascal.

Sixteen thermistors were mounted within the heat exchanger. One thermistor is placed on the leading edge of each screen to monitor wheat temperature as the grain flows from one cell to the next. A V-shaped cup, with a capacity of approximately 15-20 grams of wheat, was also installed on the end of each screen to collect wheat samples for temperature measurement. Salt and wheat temperatures in the heating and inlet sections were each monitored with a pair of thermistors. Thermistor signals were brought out from the heat exchanger through a reset/pulse analog multiplexer (Figure 21) and a set of slip rings.

Pulse width modulation, using MC6840 programmable timers controlling zero-crossover solid state relays at a frequency of 1 Hz, was used to control electrical energy input (Figure 22). A single revolution of the heat exchanger was divided into two energy input components: one portion of the revolution to pulse the set of preheaters (strip heaters); the second to pulse the main heaters (cartridge heaters). The preheaters are used during the first-half revolution when the heat transfer medium enters the heating section. The main heaters are used during the second-half revolution.

Due to the geometry of the heating section, at any one point in time one set of heaters is covered with heat transfer medium and the other is exposed to free convection. The dual cycling of energy input allowed for maximum heating efficiency while protecting the set of heaters not covered with medium. A limit switch connected to the interrupt line of a MC6821 peripheral interface adapter was utilized to mark the orientation of the heat exchanger during each revolution. The interrupt generated by the limit switch started the cycle for 40% to 100% duty cycle pulsing of the main heaters. A programmable timer



U1 : MC78L05
U2 : MC79L05
U3 : LM208A
U4 : LM299
Q1 : RS-2023
Z1 : 4.6v
U5 : 4520
U6 : 4067

Figure 21. Temperature Measurement Circuitry

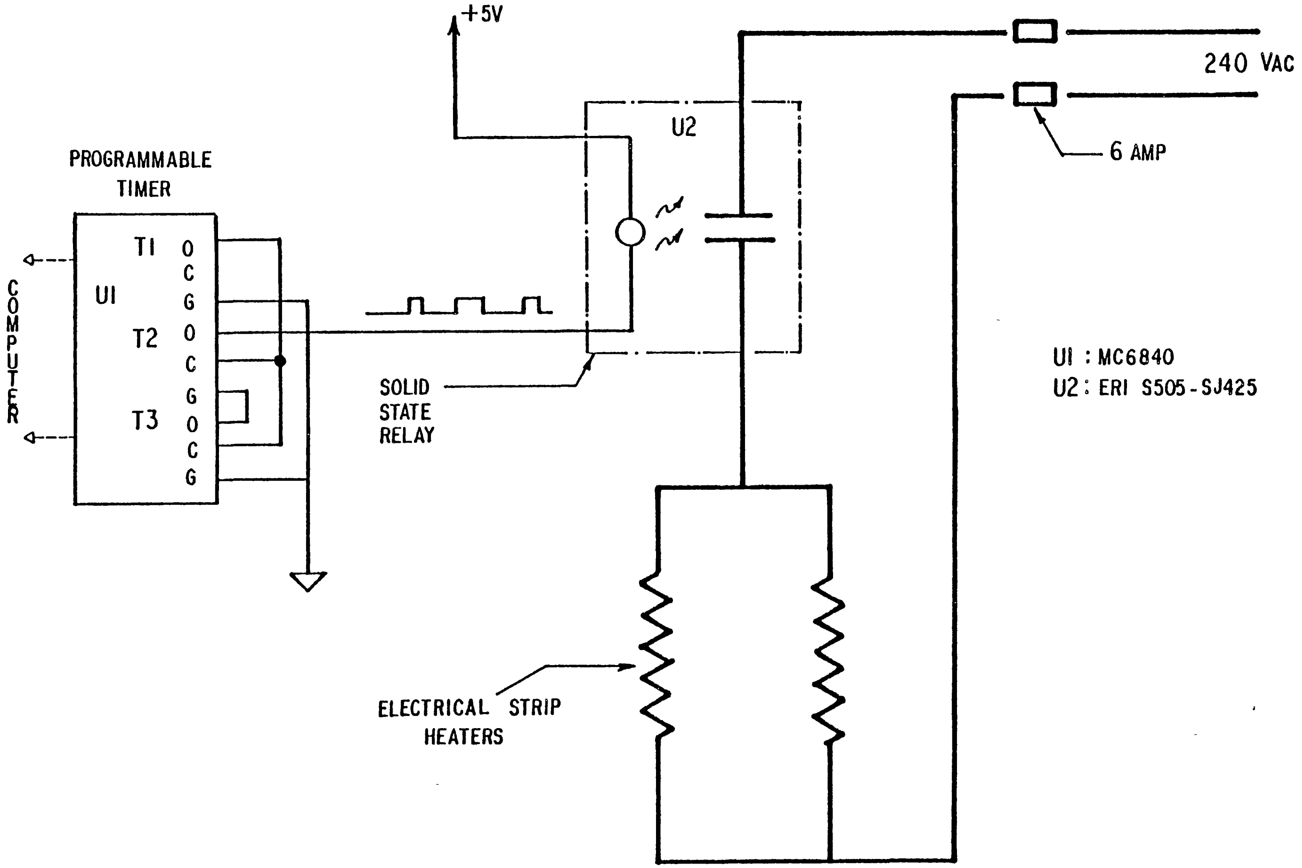


Figure 22. Heater Control Circuitry

was used to start the opposite cycle, that of pulsing the preheaters.

In order to accurately monitor heat exchanger performance, electrical energy input was measured (Figure 23). Ac input current over the pulsed time intervals was measured and integrated. Ac input voltage was periodically recorded manually. The output of a current transformer is converted first to a low voltage ac signal, then to a dc voltage. The ac/dc conversion was accomplished through a linear operational amplifier rectification circuit. A voltage-to-frequency converter outputs a pulse train, and a MC6840 programmable timer was used to count the pulses. The resulting count is proportional to the integration of the ac current drawn by the heaters during test period.

Separation Efficiency and Medium Distribution

For proper operation, the prototype should have the required mixing, separation, and flow capabilities. Tests were conducted which examined the effects of medium, mass of material, and speed of rotation on separation efficiency and distribution. The amount of medium exiting with the wheat during each revolution was used as the performance measure for evaluating separation. Medium distribution was evaluated by measuring the amount of material in each of the seven discrete components at the end of the test period.

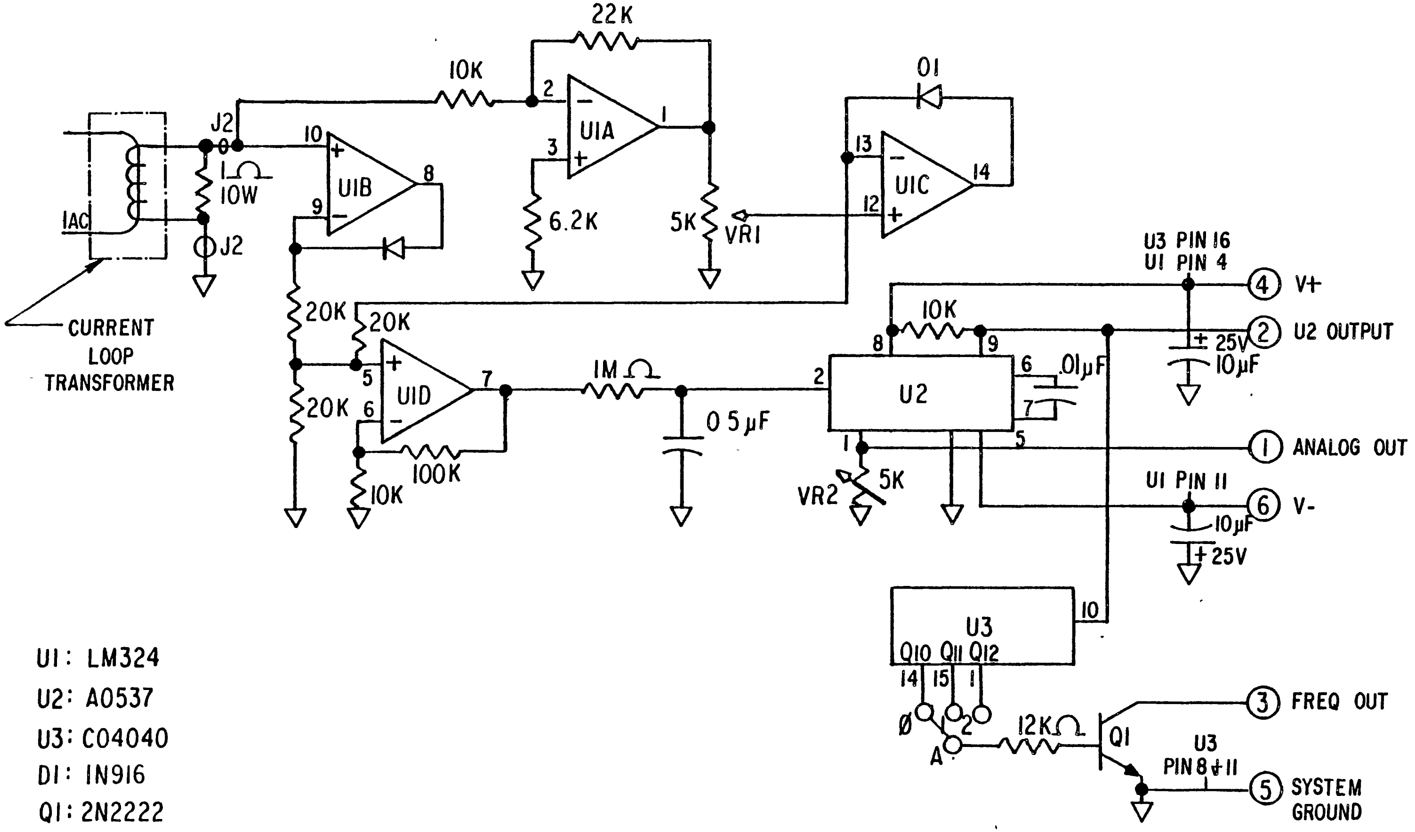


Figure 23. Energy Measurement Circuitry

The test procedure went as follows. The machine was manually loaded with 908, 1362, or 1816 g/cell. Wheat was augered into the machine continuously tp give 454 g/cell. Three speeds of rotation were utilized: 0.5, 1.0, and 1.5 rpm. These speeds were chosen based on the simulation results of Chapter IV. The machine was rotated until filled with material, then rotated for 30 revolutions to constitute the test period. Exiting material samples were taken every sixth revolution. At the end of the test period, each component of medium was emptied into a separate container, sieved, and weighed.

The results indicated that, while not ideal, separation efficiency was satisfactory. Due to a shorter flow period along the screens, medium loss increased as the machine rotational speed increased. Salt, in general, separated more easily than Interprop. However, 1816 g/cell of salt gave very poor results due to apparent volumetric overloading within the cells and consequently a full set of data was not taken at this rate of loading. The average losses of medium per revolution for salt were 11, 17, and 53 g/cell at 0.5, 1.0, and 1.5 rpm respectively. This compares to average losses for Interprop of 22, 37, and 52 g/cell at the same speeds.

With respect to medium distribution, salt tended to remain more evenly distributed than did Interprop. While Interprop had an average peak-to-peak difference in mass of 12.4% from cell to cell, salt averaged 8.6%. However, both media maintained a satisfactory distribution throughout and did not show any tendency to become concentrated in any area. Thus, it was concluded that the separation efficiency of the machine and medium distribution within the machine was sufficient for counterflow movement of the two particulates.

Heat Exchanger Testing and Results

Procedure

Steady-state heating tests were designed and run in order to evaluate the machine's overall effectiveness as a heat exchanger. Wheat temperature and efficiency of heat transfer were chosen as output variables of primary importance. Variable process inputs included heat exchanger rotational speed, heat transfer medium, and electrical energy input to the heaters.

The Basic program SUPER, shown in the Appendix, was written to control the operation of the heat exchanger during these tests. All device control performed by SUPER was open-loop, thus relying on calibration equations and an operator to ensure proper feeder and drive speeds. SUPER is menu-driven and has several embedded interlocks to prevent problems such as illegal setpoints.

The following test procedure was used. Line input voltage to the heaters was manually recorded. The machine was loaded with heat transfer medium, either salt or Interprop, at 1362 grams per cell. The amount of medium utilized per cell was based on the separation and distribution testing along with preliminary heating tests. The machine speed was set and the feedrate was adjusted, using SUPER, to allow 454 grams of wheat per cell. The heat exchanger was always at or near the ambient temperature at the beginning of each test. Input power (duty cycle level) for the heaters was then set and the heaters were turned on.

Temperature data on the ambient condition, inlet wheat, and all thermistor locations were recorded every 15 minutes. The time was also recorded, along with true machine speed, energy input in a revolution, and exiting material composition. Each test was continued until a steady-state temperature profile was reached within the heat exchanger. In general, this requirement took approximately two hours. Line input voltage was again recorded at the end of the test.

Preliminary testing of the heat exchanger revealed several adjustments that could be made in order to improve performance. Among these were insulating the end sections and adding silica gel desiccant pockets in the heated section to absorb moisture and prevent condensation. Initially, low duty cycle pulsing (20%) of the heaters during the portion of the revolution when one of the sets of heaters was uncovered was utilized. This technique proved to preheat the machine more rapidly but decreased efficiency once the machine reached some steady-state condition. Therefore, for the steady-state performance tests, the low duty cycle pulsing was not used.

Steady-state Performance Results

After the preliminary tests were completed and the noted adjustments made, eleven tests were conducted to evaluate the device. Three tests were made with Interprop at three different rates of energy input (60, 80, and 100% duty cycle) all at a rotational speed of 1 rpm. Eight tests with salt were made: four rates of energy input (40, 60, 80, 100% duty cycle) at 1 rpm, and four other speeds (0.75, 1.20, 1.45 and 1.75 rpm) at the 80% energy input level. Results of the tests are shown in Table IV.

In Table IV, Twhs is the wheat temperature in the centermost portion (heated end section) of the machine. The rate of energy exchange by the processed wheat (qw), by the heat transfer medium (qs), by the exiting wheat (qlw), and by the small amount of heat transfer medium exiting the machine (qls) were all calculated using q=mr*c*(Tf-Ti). Energy input (qin) was calculated using the following equation:

$$qin = (p)(j)(s)(v) \quad (W) \tag{5.1}$$

where j = pulses/rev from energy measurement circuit;

p = 0.0333 = (10 amps/5 pulses/sec)*(1 min/60 sec).

For direct test comparisons, the energy input variable (qin) and wheat energy absorption variable (qw) were normalized to an amount of energy per unit mass of wheat

TABLE IV

STEADY-STATE HEATING PERFORMANCE

Test		Duty Cycle	Speed	Twhs	q_w		q_{in}		q_s	q_{lw}	q_{ls}
#	Medium	of htrs (%)	(rpm)	(°C)	(W)	(kJ/kg)	(W)	(kJ/kg)	(W)	(W)	(W)
1	Interprop	60	1.0	44.4	228.4	32.1	428.2	60.3	465.8	84.9	4.0
2	Interprop	80	1.0	49.4	288.4	39.9	549.1	75.9	731.1	123.3	9.3
3	Interprop	100	1.0	55.6	368.9	49.5	678.7	91.1	852.8	93.9	11.6
4	Salt	40	1.0	40.6	199.8	26.0	288.2	37.5	328.8	79.7	4.2
5	Salt	60	1.0	49.4	266.9	39.0	420.3	61.4	503.4	89.0	3.7
6	Salt	80	1.0	57.2	380.5	51.7	548.8	74.5	550.4	107.4	3.9
7	Salt	80	0.75	58.9	281.5	54.1	535.1	102.4	503.6	95.0	6.9
8	Salt	80	1.20	56.6	434.0	50.3	558.5	64.8	533.3	126.6	8.4
9	Salt	80	1.45	54.1	474.9	46.8	565.3	55.7	744.3	133.6	9.5
10	Salt	80	1.75	48.6	476.2	39.5	569.7	47.3	756.1	111.7	8.6
11	Salt	100	1.0	62.8	457.5	59.5	685.6	89.2	788.1	146.8	3.5

(i.e. kJ/kg). This was necessary due to wheat loading rates (material per cell) varying approximately 10% from test to test.

The data in Table IV reveal several important aspects of the heat exchanger performance. Salt was more effective in achieving a higher wheat temperature for a given amount of energy input than was Interprop. This agrees with simulation results in Chapter IV. However, the heat loss variables qls and qlw show that the major heat loss of the prototype is not due to exiting material but due to heat flow from the machine to the surrounding environment. A large (machine mass)/(medium+grain mass) ratio plus insufficient insulation are the reasons for this phenomenon.

Figures 24 and 25 show resulting steady-state temperature profiles within the heat exchanger using various levels of energy input while the heat exchanger rotated at a speed of 1 rpm. The counterflow and heat recovery aspect of the machine can be seen in these temperature profiles, along with the increased performance of salt relative to Interprop. The somewhat uneven temperature increments from cell to cell shown in Figures 24 and 25 are due to a combination of uneven mixing and difficulty in measuring the true wheat temperature as the grain moves from cell to cell.

The effect of input energy on maximum wheat temperature and overall efficiency while the heat exchanger rotated at a speed of 1 rpm can be seen in Figures 26 and 27. The

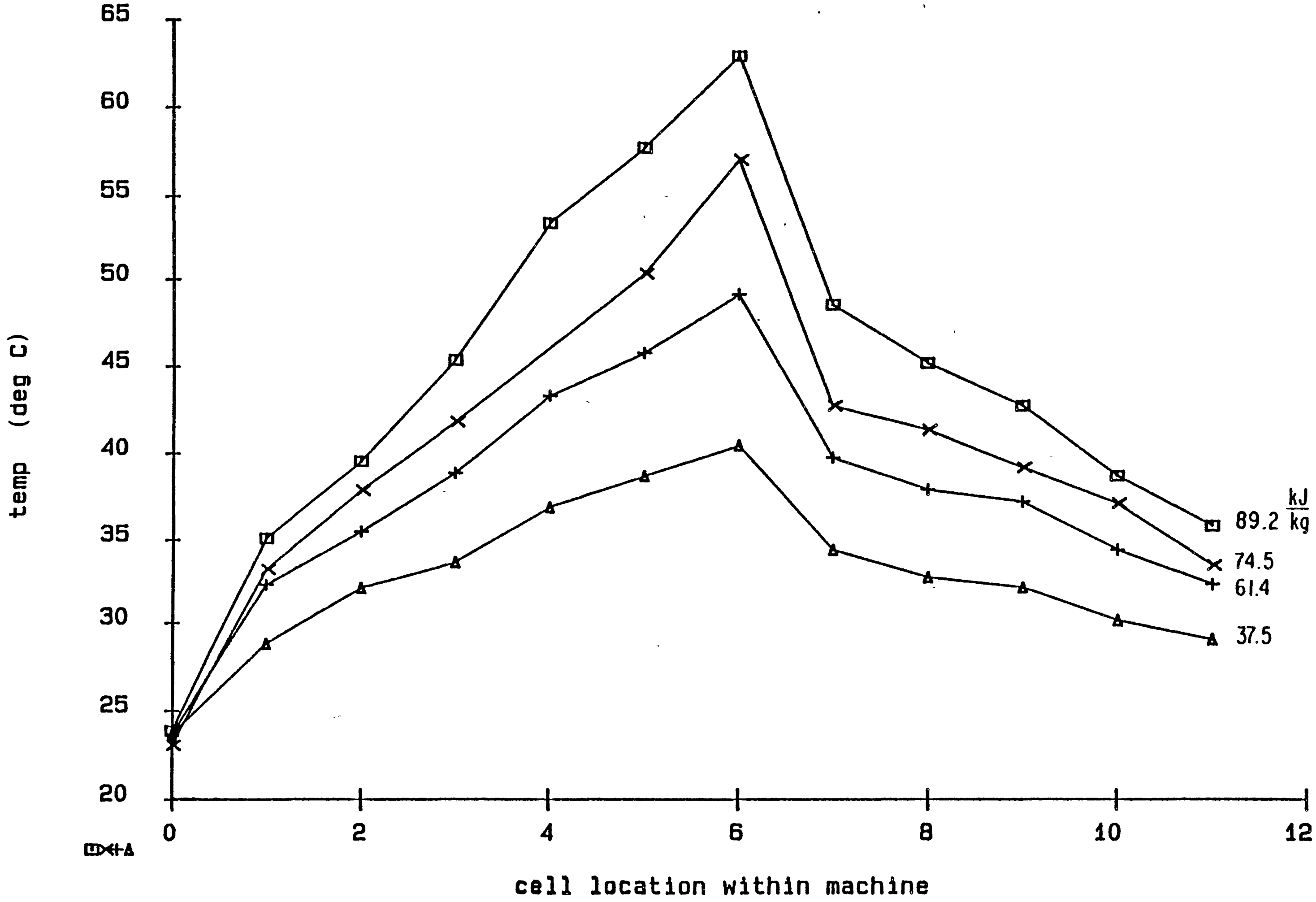


Figure 24. Wheat Temperature Profiles Using Salt and Various Levels of Energy Input

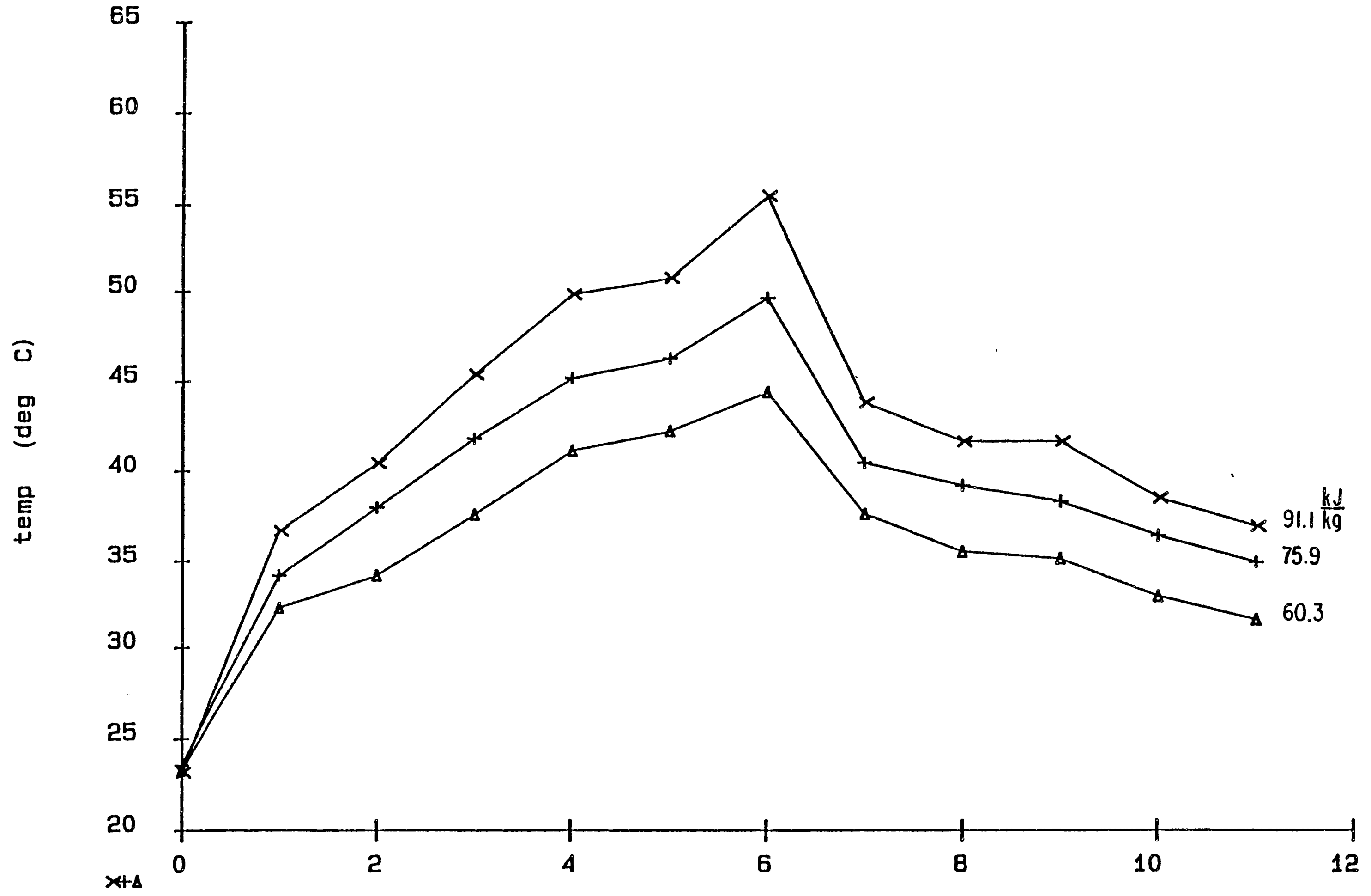


Figure 25. Wheat Temperature Profiles Using Interprop and Various Levels of Energy Input

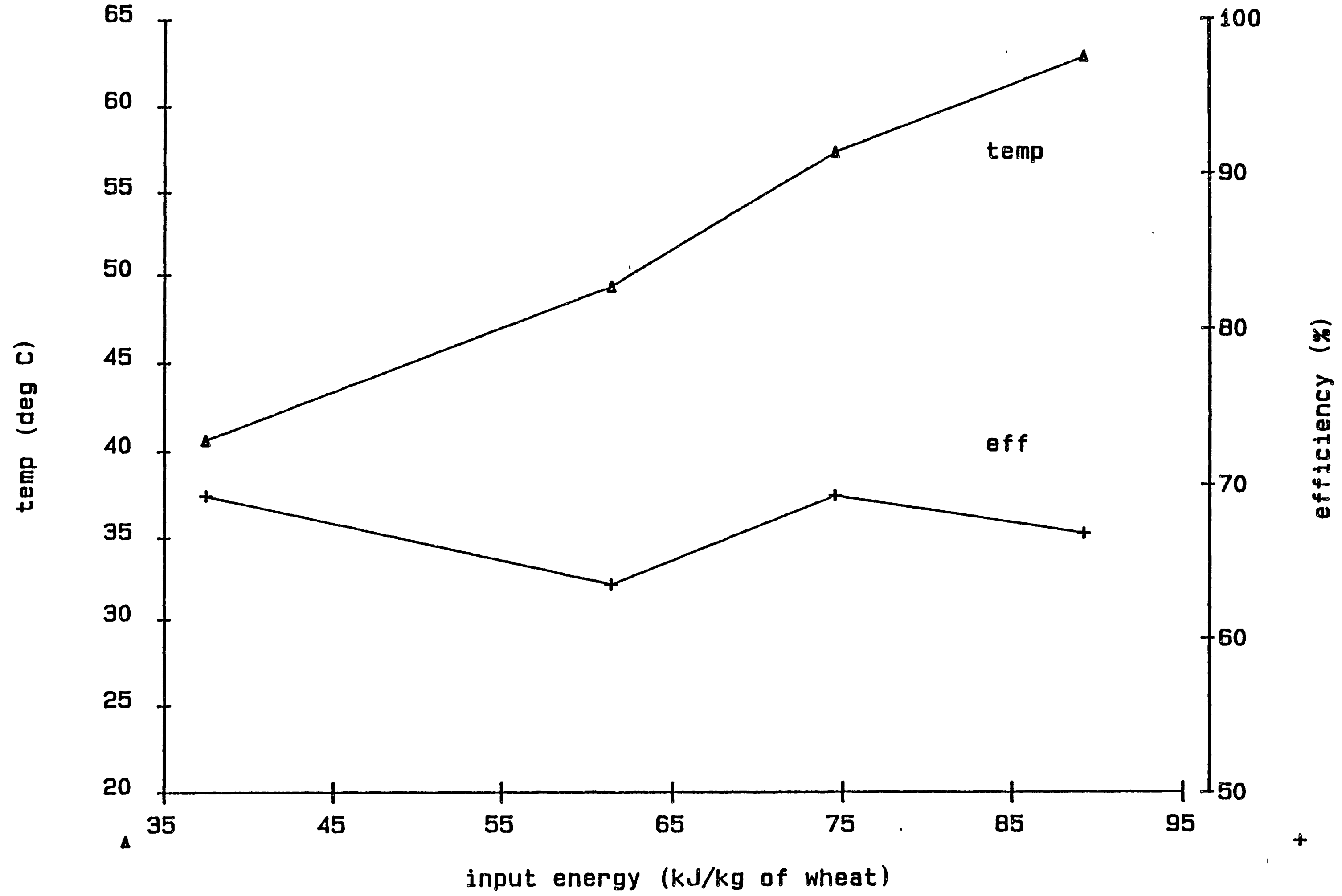


Figure 26. Energy Utilization Performance Using Salt

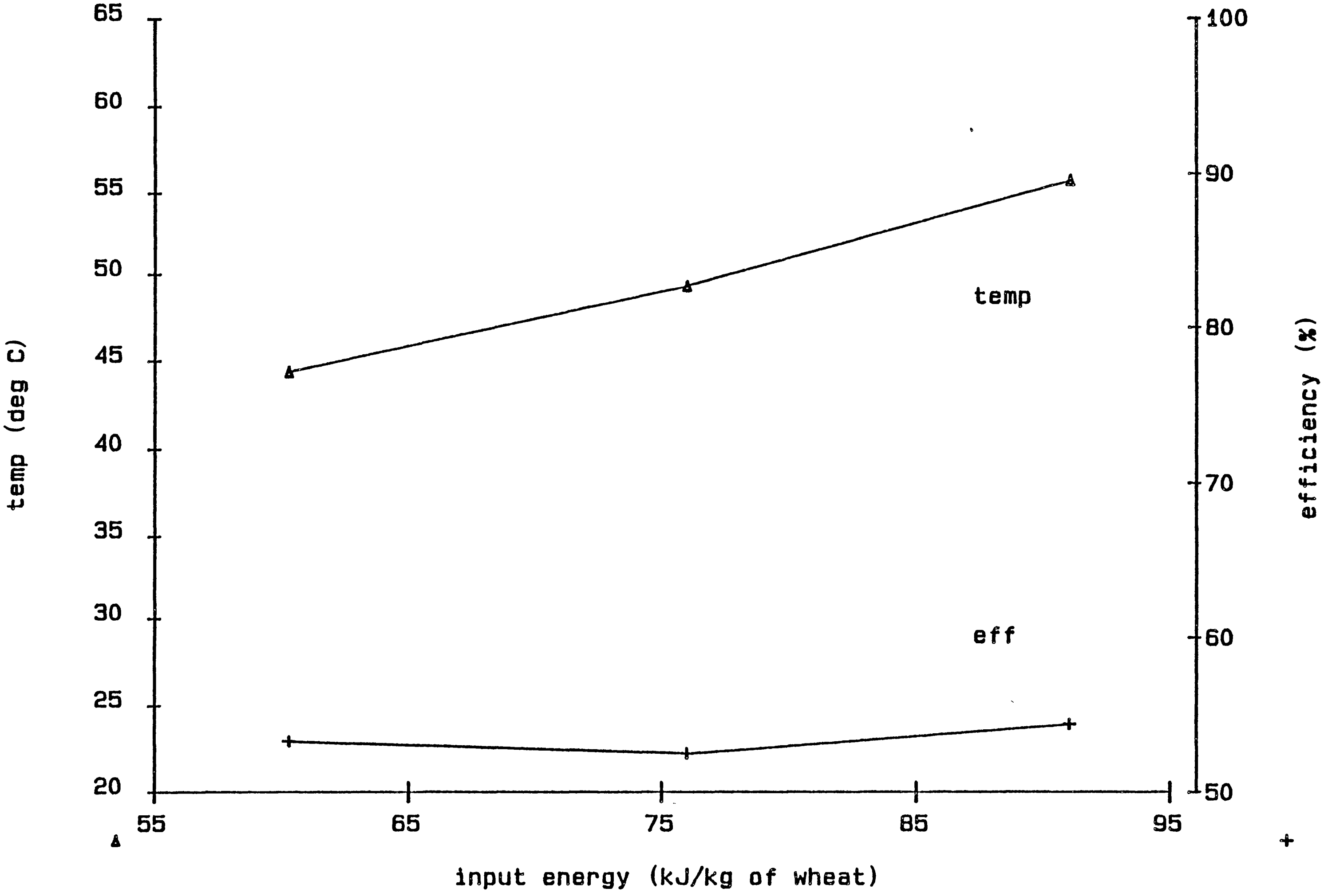


Figure 27. Energy Utilization Performance Using Interprop

efficiency is calculated as amount of energy absorbed by the wheat in the 5 cell heating stage divided by the amount of energy input during one revolution at steady-state. Salt tests had an average efficiency of 67.2% for the four tests at 1 rpm, while Interprop averaged 53.4%.

The effect of heat exchanger rotational speed on maximum wheat temperature and overall efficiency can be seen in Figure 28. Salt and an 80% duty cycle were used in each of the tests. While an increase in temperature was realized by reducing speed from 1.0 to 0.75 rpm, efficiency decreased due to a reduction in wheat flow per time (wheat flow per cell held constant) and the salt losing a greater percentage of its energy to the machine structure during the longer period of heat transfer. Conversely, increasing the speed from 1.0 rpm up to 1.45 rpm increased efficiency. Maximum wheat temperatures decreased, but the total amount of energy actually transferred to the grain was greater (Table IV). The peak efficiency, 84%, occurred at 1.45 rpm. The final speed tested, 1.75 rpm, resulted in a sharp drop in maximum wheat temperature, but efficiency remained high, decreasing slightly to 83.5%.

Heat Recovery Capability

Heat exchanger performance has been evaluated in terms of temperature and utilization of input energy. A more important criterion for prototype evaluation may be heat recovery capability. A measure of the heat recovery

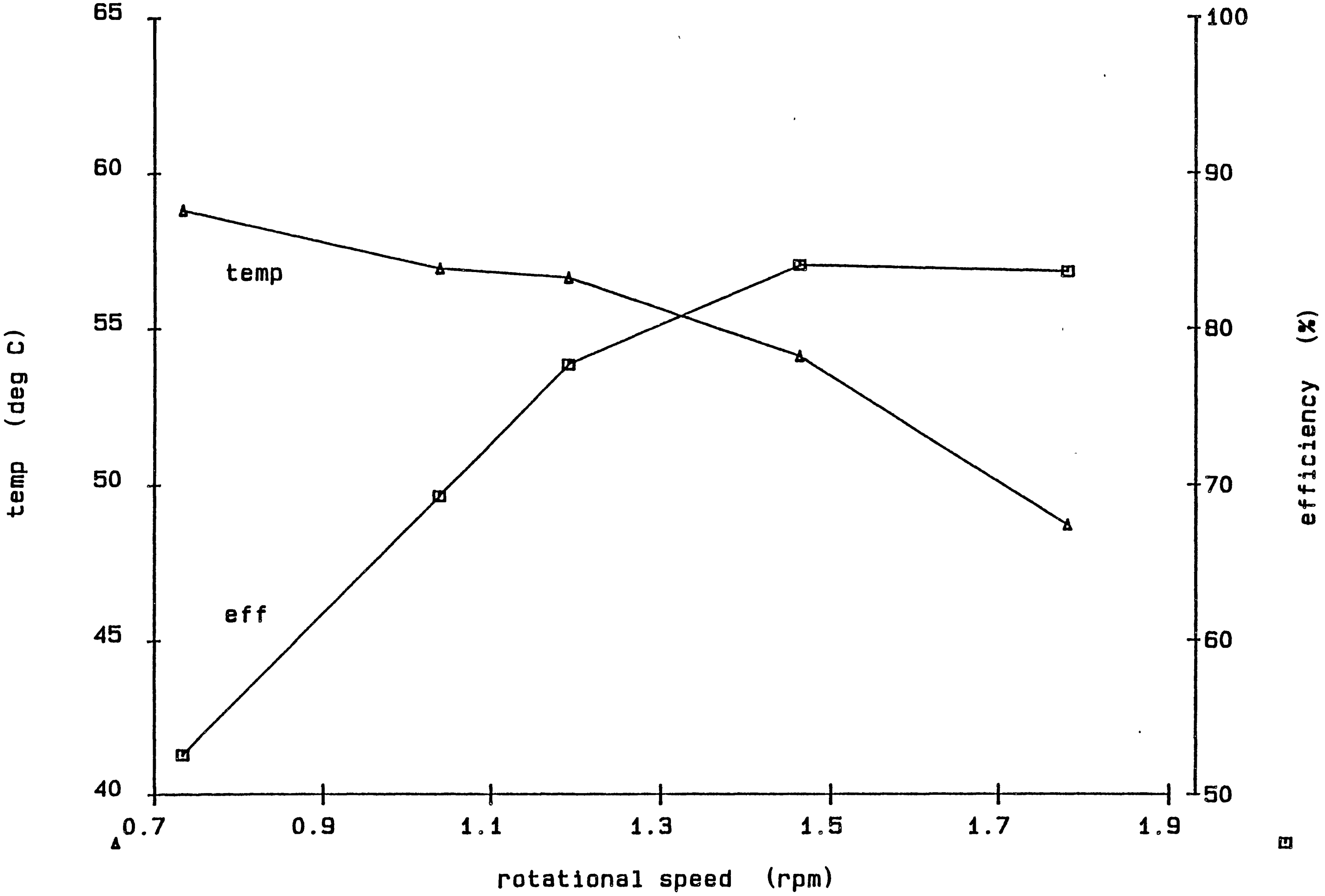


Figure 28. Effect of Speed on Performance Using Salt and a Constant Energy Input

capability of the heat exchanger was defined as a ratio of the amount of energy lost by the wheat in the cooling stage to the amount of energy gained by the wheat in the heating stage. These ratios can be seen in Table V. The heat recovery ratio was calculated as

$$hrr = (qw-qlw)/qw \ . \tag{5.2}$$

The average heat recovery ratio for all the tests was 0.679. This average value indicates a strong capability of the prototype to reclaim energy from the heated grain. Over two-thirds of the energy absorbed by the grain in the heating stage was recovered, both by the media and by the machine structure, before it exited the cooling stage.

TABLE V

HEAT RECOVERY RATIOS

Test	hrr	Test	hrr	Test	hrr
1	0.628	5	0.666	9	0.719
2	0.573	6	0.718	10	0.766
3	0.745	7	0.663	11	0.679
4	0.601	8	0.708		

Comparison with Simulation

In Figure 29, a comparison is made between an actual

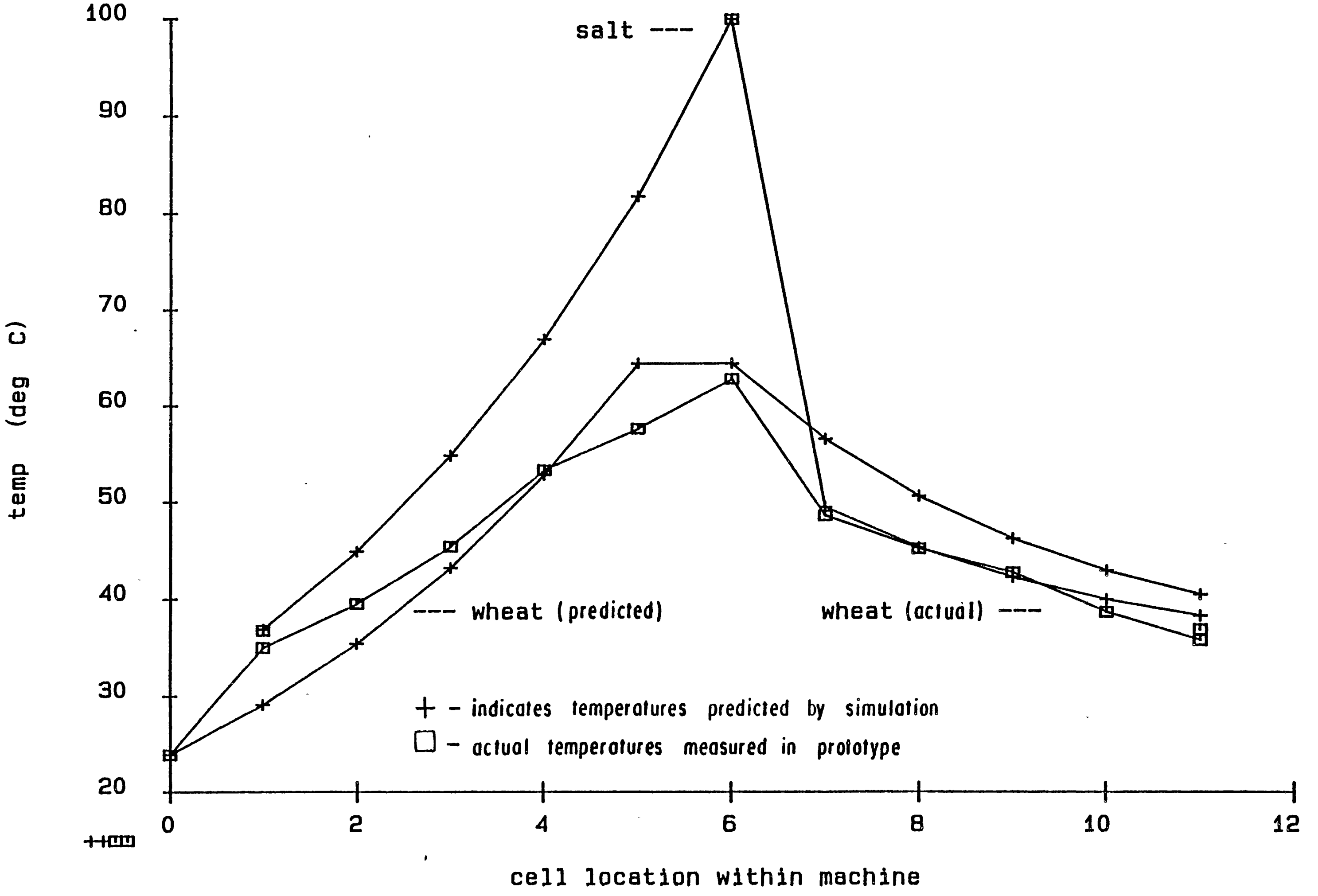


Figure 29. Temperature Profile Comparison Between Results from the Prototype Testing and that Predicted by the Simulation

temperature profile measured in the prototype and that predicted by simulation. The predicted profile was generated using wheat inlet temperature, maximum salt temperature, and heat transfer efficiency measured in the test as inputs to the simulation. The resulting comparison shows good agreement between actual and predicted total temperature differences of both the wheat and the salt. However, the wheat temperature profiles do not appear to match well. The differences between the simulation and the prototype account for this: (1) the simulation assumes idealized conditions in which machine heat capacity and conduction effects are not considered; (2) inadequate mixing in the cells located on either side of the heated end section, as seen in the decrease in temperature rise in cell number 5 and the increased heat loss in cell number 7, adversely affect the temperature profile; and (3) the prototype, due to a gravity-flow based geometry, contains only 7 discrete units of heat transfer medium for a 10 cell unit, while the simulation assumes all cells are occupied.

Brief economic analysis

The following analysis was performed assuming \$ 0.005 per cubic foot for natural gas and \$ 0.065 per kilowatt-hour for electricity. Tests conducted on the counterflow solid particle heat exchange prototype showed that an average of 68% of the energy added to the wheat in the heating section was recovered in the cooling section. Assuming the same ability in a full-scale machine and that 50% of the energy recovered is usable in the heating section, 39 MJ of heat energy would be required to process one tonne of wheat. Using an 85% input efficiency, this corresponds to $0.83 per tonne using electrical energy or $0.22 per tonne using natural gas. Note that the use of natural gas would most likely entail higher capital costs. Energy required to drive a full-scale device would be approximately 9.8 MJ per tonne costing $0.18 for the electrical energy.

Therefore, total operational costs would be approximately $1.01 per tonne using all electrical energy or $0.40 per tonne for a combined natural gas/electical unit. These figures can be compared to the current estimates for fumigation costs, which are $0.88 to $1.32 per tonne. A possibility also exists for considering thermally disinfested grain a value-added commodity, particularly on the international market. If so, more detailed economic analyses would be necessary to determine the potential of the investment.

CHAPTER VI

PERFORMANCE ENHANCEMENT USING ADAPTIVE CONTROL

Determination of Baseline Performance Level

Importance

The solid particle heat exchanger has been shown to be both effective and well-behaved from a processing standpoint. Given a certain input energy rate, the wheat being passed through the exchanger will in time reach some steady-state temperature profile from cell-to-cell and therefore attain some maximum temperature in the heated end section. However, there is considerable lag time before this state is reached and, assuming a desired maximum wheat temperature, a considerable amount of material is processed incorrectly.

A very desirable condition would be for the wheat to come up to setpoint temperature relatively quickly while not having to preheat the heat exchanger. Another way of stating this is to require minimizing the setpoint temperature deviation of grain being processed while at the same time minimizing the energy requirement. The difficulty in satisfying these performance criteria lies in the control of the heat exchanger during startup. That feedback control is required to accomplish this is obvious. What is not obvious is the type of controller best suited for the task. This chapter examines the use of a conventional versus an adaptive controller to optimize the performance of the solid particle heat exchanger during startup.

Derivation of Conventional Controller

A conventional control law used in many feedback control systems is the proportional-integral-derivative (PID) controller. The continuous-time PID equation is generally written as

$$vp = vref + Kc\{ e + (1/ti)\int e\,dt + (td)de/dt \}. \qquad (6.1)$$

The use of (6.1) in a digital control system requires a discrete-time equivalent. One of the commonly used difference equations used to approximate (6.1) can be derived using Laplace and z-transforms. Substituting "u" for "vp-vref" and transforming to the Laplace domain yields

$$U(s)/E(s) = Kc\ \{ 1 + 1/(ti)s + (td)s \} \qquad (6.2)$$

where "s" is the Laplace variable.

Now, (6.2) can be transformed to the z-domain with or without the use of the zero-order hold. In this case, the zero-order hold is not used to maintain the desired phase relationship between the integral term (1/ti) and the current error, and the result is

$$\frac{U(z)}{E(z)} = Kc \left\{ 1 + \frac{Tz}{ti(z-1)} + \frac{td(z-1)}{Tz} \right\}. \qquad (6.3)$$

Solving for the transfer function yields

$$U(z)/E(z) =$$

$$\frac{Kc \{(1+(T/ti)+(td/T))z^2 - (1+2td/T)z + (td/T)\}}{z(z-1)} \qquad (6.4)$$

Inverting (6.4) to the discrete-time domain for the required controller algorithm in difference equation form gives

$$u(t) = u(t-1) + Kc \{ (1+(T/ti)+(td/T))\, e(t) \qquad (6.5)$$
$$- (1+(td/T))\, e(t-1) + (td/T)\, e(t-2) \}.$$

Due to the sluggish nature of the solid particle heat exchanger and the limited control effort (input energy rate) available, derivative action was not used. Without this gain, (6.5) is a PI controller which can be written as

$$u(t) = u(t-1) + Kc \{ (1+(T/ti))\, e(t) - e(t-1) \}. \qquad (6.6)$$

PI Control in the Presence of Limited Control Effort

Integral action is important for removing steady-state error and increasing the response of a controller to some disturbance. However, integral action can also lead to saturation of the manipulated variable u(t), particularly in processes with very limited control effort. Such is the case with the solid particle heat exchanger. Only 1600 Watts are available for heating, far below what is required for rapid temperature elevation of either the heat transfer media or the wheat.

Saturation caused by integral action is a particular problem in digital controllers. The predicted numerical manipulated variable may far exceed the actual control effort available. If this condition occurs primarily due to integral action, a large overshoot of the controlled variable, i.e. wheat temperature, is likely.

Simply clamping the maximum allowable u(t) in a digital PI controller is not the solution. Clamping not only inserts a non-linearity into the controller but ultimately limits the proportional gain in a properly tuned control system. To alleviate the saturation problem on the solid particle heat exchanger control system, the integral action was turned off whenever the manipulated variable exceeded 100% of the available control effort. Proportional action was allowed to function as always, as shown in the following equation.

$$u(t) = u(t-1) + Kc \{ e(t) - e(t-1) \} \qquad (6.7)$$

Therefore, the controller utilized (6.6) until u(t) reached a value greater than 100% of the available control effort, then used (6.7) until u(t) returned to the 0-100% range.

PI Controller Tuning Procedure

Assuming that the sampling period "T" is already set based on current system requirements or limitations, only two free parameters are available for adjusting a digital PI controller to fit a particular process: Kc and ti. These parameters may be adjusted, or tuned, to obtain some desired system performance. One method of tuning relies on adjusting Kc while the integral action is turned off to find the optimal proportional gain, then running the controller with various levels of integral action to obtain the ultimate optimum. This was the method utilized on the solid particle heat exchanger controller. However, rather than cutting off the integral action completely, a small amount (equal in all tests) was used during the proportional gain tuning.

Each tuning test was run using a wheat setpoint temperature of 54.4 °C, a rotational speed of 1.25 rpm, and a duration of 2 hours. The setpoint was chosen low enough such that the steady-state energy requirement would not be higher than 80%; the speed was chosen as the optimal speed based on the steady-state heating tests (Chapter V); and the duration was chosen sufficiently long to include oscillations that may occur.

A version of the heat exchanger control program ADAPT, seen in the Appendix, was used for all the tuning tests. The software performs feedback control on the main drive, feeder drive, make-up media drive, and heaters. The drive loops are sampled and updated on approximately 70 millisecond periods. The heater loop is serviced once per revolution. Temperature and control effort data were dumped to a printer approximately every six minutes.

In order to limit the number of tests required for tuning, equations (6.8) and (6.9) (Rovira, et al. 1969) were used to predict the optimal Kc and ti. These equations were derived for setpoint changes on first-order processes and are based on an integral of the absolute error performance measure.

$$Kp\ Kc = 0.758(d/tp)-0.861 \qquad (6.8)$$

$$tp/ti = 1.02 - 0.323(d/tp) \qquad (6.9)$$

To obtain values for Kp and tp to use in the above equations, a step test was run on the heat exchanger (Figure 30). Using 80% of the available control effort, salt, and a rotational speed of 1 rpm, approximations for Kp (0.0375) and tp (23 minutes) were found. The deadtime "d" was estimated to be 2.5 revolutions, the number required for grain to makes its way along one stage of the heat exchanger. Sub- stituting these values into (6.8) and (6.9) yields predicted values for Kc (136.6) and ti (23.35). These values were used to predict the ranges required for the tuning tests.

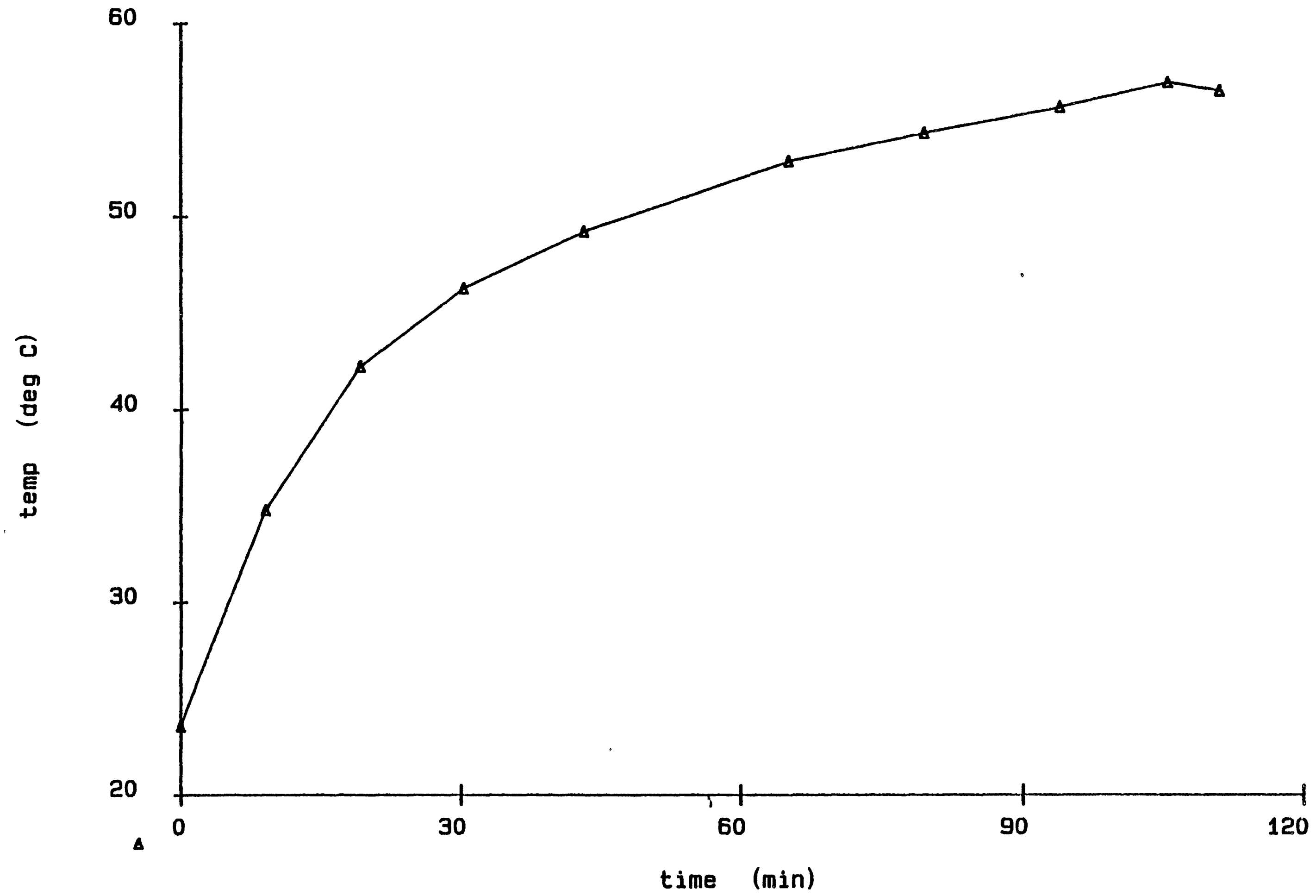


Figure 30. Step-test to Determine Approximate Process Parameters

Judging Performance

The actual basis on judging control system performance, the performance measure(s), is derived relative to the predicted system requirements. For instance settling time, percent overshoot, integral of the absolute error, and a combined integral of error and control effort are all valid performance measures. In the case of the solid particle heat exchanger, minimizing the amount of wheat processed incorrectly and minimizing energy requirements during startup are important. Therefore, both integral of the absolute error (where error = setpoint - output) and integral of energy input were monitored to follow system performance.

Tuning Results

The results of the proportional gain tuning can be found in Table VI. A gain of 90.0 yielded the minimum error performance, closely followed by a gain of 136.6. Although the energy-based performance measure was somewhat higher for the latter, a gain of 136.6 was chosen as the optimal over 90.0 for the following reasons: (1) both the inlet wheat temperature and the ambient temperature were lower during the Kc=136.6 test, thus requiring more energy to reach and maintain the setpoint; and (2) the higher gain should give more sensitivity to process disturbances. Response, shown as a deviation variable, and control trajectories are shown in Figure 31.

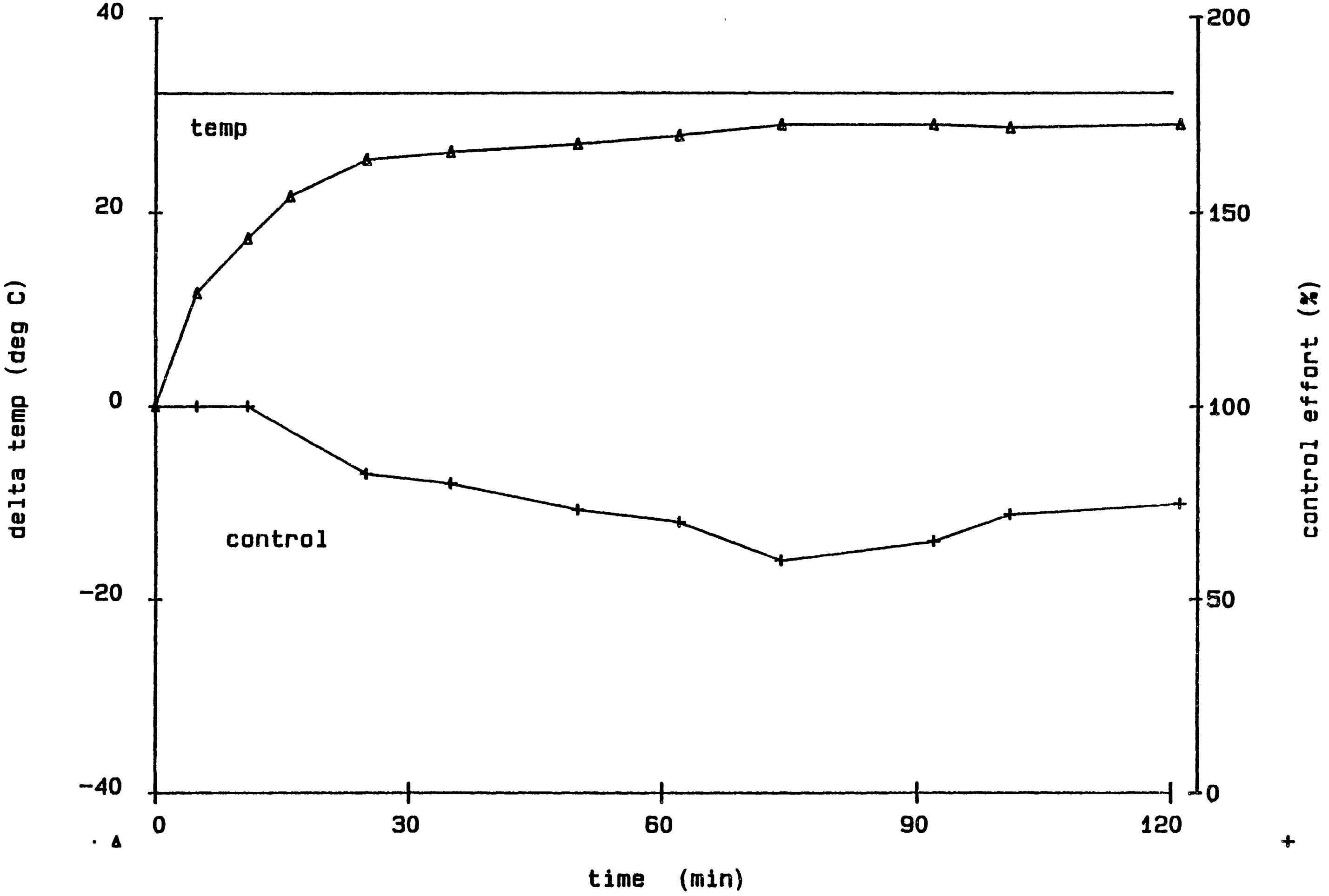


Figure 31. Temperature Response and Control Effort for Optimal Proportional Controller

TABLE VI

TUNING RESULTS FOR PROPORTIONAL CONTROLLER

Proportional Gain	Integral of Absolute Error	Integral of Energy Input
40.0	1906	3475
50.0	1747	3483
70.0	1581	3618
90.0	1436	3657
136.6	1489	3899

Table VII shows the results of the integral gain tuning, all done with a proportional gain of 136.6. The minimum error-based performance measure occurred with an integral time of 4.6 minutes, which corresponds to an integral gain of 23.76. Based on the time-temperature profile (Figure 32) and the low error-based performance measure, these gains were selected as optimal values for the

TABLE VII

TUNING RESULTS FOR PROPORTIONAL-INTEGRAL CONTROLLER

Integral Gain { Kc(T/ti) }	Integral of Absolute Error	Integral of Energy Input
0.533	1489	3899
4.68	934	3097
23.76	916	4228
42.86	917	4164
61.74	1142	4294

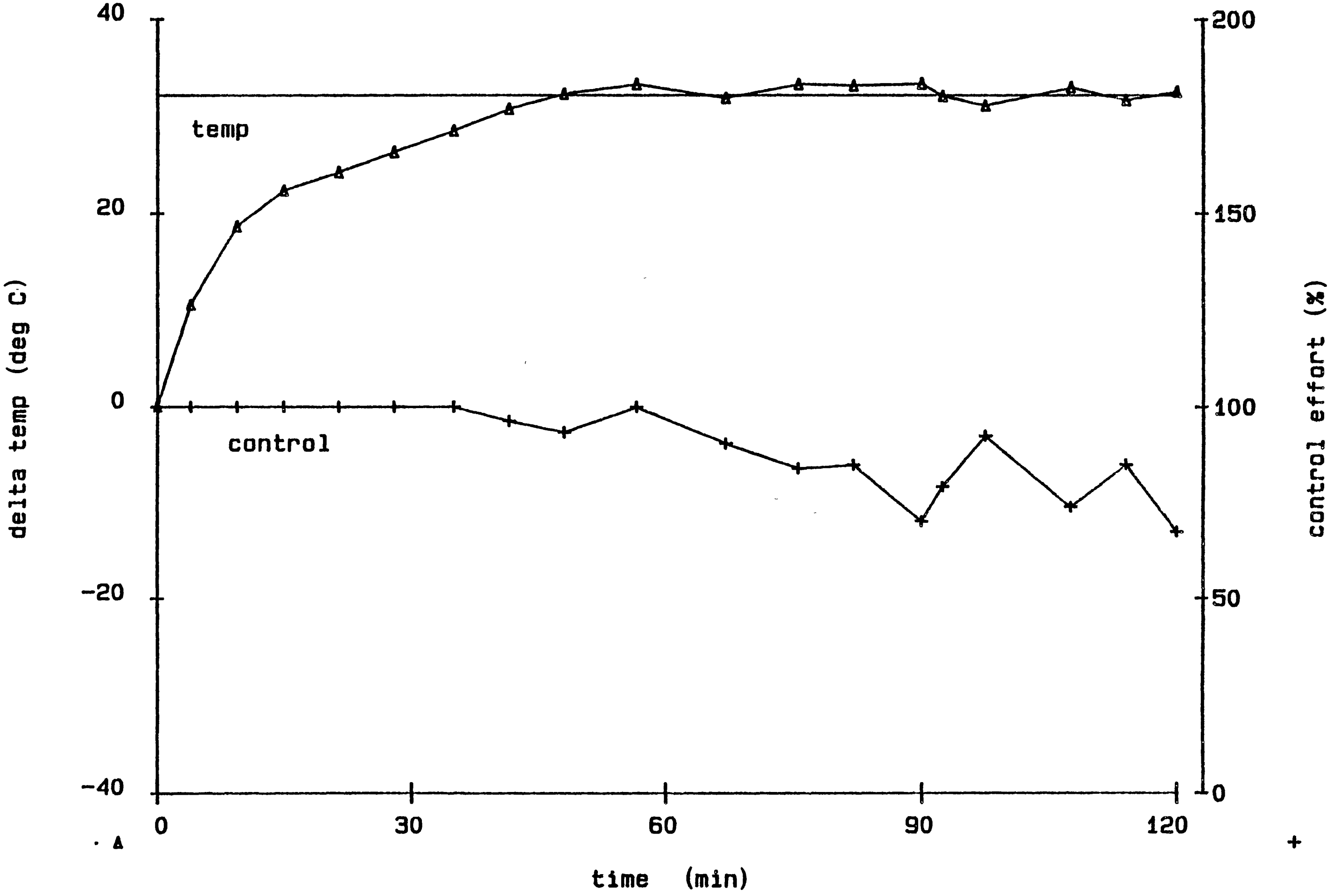


Figure 32. Temperature Response and Control Effort for Optimal Proportional-Integral Controller

PI controller. A noise disturbance during this test at approximately t=80 minutes caused the control effort to rise and the resultant energy-based performance measure to be skewed relative to the other tests.

Derivation of Performance Measure

The objective of the PI controller tuning procedure was not just to find the "best" performance parameters but to quantify this performance such that it could be used as a baseline for comparison to the adaptive controller(s). Therefore a performance measure consisting of the combined error-based and energy-based criteria was formed as follows:

$$J = Iae + c(Iein) \tag{6.10}$$

where

$$0.0 < c < 1.0.$$

Table VIII shows how the rankings of the optimal integral gain shift as "c", the energy integral coefficient, is varied. The rankings for c equal to 0.25 indicate a dual minimum. However, given the characteristics of the heat exchanger, a dual minimum is not likely to actually occur. Observing that the skewed energy-based performance measure was causing "J" to be skewed for the integral gain of 23.76, an interpolated value for Iein was used. This led to the rankings being shifted using an energy integral coefficient of 0.25, giving a minimum performance measure of 1948. This "J", along with the accompanying time-temperature profile, will serve as the baseline performance.

TABLE VIII

VARIOUS PERFORMANCE MEASURES FOR THE PROPORTIONAL-INTEGRAL CONTROLLER

Integral Gain {Kc(T/taui)}	Energy Integral Coefficient "C"	Performance Measure "J"	Rank (*)
0.533	0.50	3438	5
4.68		2983	1
23.76		3030	3
42.86		2999	2
61.74		3290	4
0.533	0.33	2789	5
4.68		2300	1
23.76		2325	3
42.86		2305	2
61.74		2574	4
0.533	0.25	2464	5
4.68		1958	2
23.76		1973	3
42.86		1957	1
61.74		2216	4
0.533	0.25	2464	5
4.68		1958	3
23.76 (**)		1948	1
42.86		1957	2
61.74		2216	4

Note: (*) "1" is best, "5" is worst, etc.
(**) Iein used was an interpolated value.

Parameter Estimation

Adaptive Control Foundation

An adaptive control law is based on an algorithm that either periodically or continuously changes the controller dynamics. Modern adaptive controllers rely on the use of on-line system identification as a mechanism for updating the control law. Performing system identification requires estimation of the dynamic parameters of the process using natural system inputs and outputs. These parameters are then incorporated into a feedback control law. Most parameter estimation techniques predict the coefficients of the numerator and denominator polynomials of the process transfer function. Such is the case with the recursive least squares algorithm.

Parameter Estimation Using Recursive Least Squares

To use the least squares algorithm, it must be assumed that the dynamics of the process can be modeled, if only over a narrow range, by a linear difference equation such as

$$\begin{aligned} y(t) = a1\ y(t-1) + a2\ y(t-2) + \ldots + an\ y(t-n) \\ + b0\ u(t-d) + \ldots + bm\ u(t-d-m). \end{aligned} \tag{6.11}$$

The parameters a1, a2, etc may or may not be time-varying. A predictor in vector form for (6.11) can be written as

$$y(t) = \phi^{T}(t-1)\ \theta(t-1) \tag{6.12}$$

where

$$\phi(t-1) = [y(t-1)\ \ldots\ y(t-n)\ u(t-d)\ \ldots\ u(t-d-m)]^{T}$$

$$\theta(t-1) = [\ a1(t-1)\ \ldots\ an(t-n)\ b0(t-d)\ \ldots\ bm(t-d-m)]^{T}$$

According to Goodwin and Sin (1984), the recursive least squares algorithm can be written as follows:

$$\theta(t) = \theta(t-1) + \frac{P(t-2)\ \phi(t-1)\ \{\ y(t) - \phi^{T}(t-1)\ \theta(t-1)\ \}}{a + M} \tag{6.13}$$

$$P(t-1) = 1/a\ \{\ P(t-2) + \frac{P(t-2)\ \phi(t-1)\ \phi^{T}(t-1)\ P(t-2)\ \}}{a + M} \tag{6.14}$$

where

$$M = \phi^{T}(t-1)\ P(t-2)\ \phi(t-1).$$

The variable "a" is equal to 1.0 in the standard version of the algorithm. However, it may also be allowed to be less than 1.0, but generally greater than 0.9, to weight recent data more heavily. This is termed exponential data weighting. P(t) is called the covariance matrix and generally decreases over time. P(0) is always a scalar multiple of the identity matrix, I. Elements on the diagonal may initially range from 2 to 10,000 depending on such factors as accuracy of initial estimates, sampling rate, and the time-varying nature of the process.

Initial Parameter Estimates

In order to begin the recursive estimation, initial guesses of the elements of the parameter vector $\theta(t)$ must be made. Since the period of concern for the heat exchanger control is on startup, the initial estimates $\theta(0)$ need to be relatively accurate. An off-line analysis using multiple regression was used to predict a satisfactory set of initial parameters.

A dataset of temperature and control history was obtained for the analysis by allowing the heat exchanger to come to equilibrium at 40.0 °C and then shifting the setpoint to 43.3 °C while under PI control. High controller gains were used to ensure an overshoot condition. The small step change in setpoint was made to keep the manipulated variable within the 0-100% control effort range. These conditions are both important in the attempt to accurately perform system identification.

In the analysis the highest order of the estimated parameter vector was six while the lowest was three. A sixth-order parameter vector corresponds to third-order denominator dynamics and second-order numerator dynamics in the discrete-time transfer function, as can be seen in Equations (6.11) and (6.12). Deadtimes of one, two, and three revolutions were considered. Results of the regression can be found in Table IX. A "good" set of parameters is one with a high coefficient of determination, a low standard deviation, low order, and low offset. The sixth and fourth-order models with a deadtime of 2 revolutions were chosen to be used as initial estimates in trial runs on the heat exchanger.

TABLE IX

REGRESSION RESULTS FOR INITIAL ESTIMATES

Dead-time	Offset	Difference Equation Coefficients θ						Goodness of Fit	
								cod	std
1	2.2	0.718	0.226	0.021	0.0043	0.0236	0.0091	0.972	0.611
	0.7	0.775	0.205	-----	0.0040	0.0309	------	0.971	0.590
	-6.4	0.918	0.126	-----	0.0346	------	------	0.966	0.634
2	7.7	0.631	0.094	0.186	0.0358	-0.0249	0.0349	0.979	0.528
	8.4	0.573	0.153	0.180	0.0233	------	0.0288	0.977	0.538
	3.5	0.726	0.228	-----	0.0267	0.0083	------	0.972	0.587
3	17.6	0.570	0.213	0.041	0.0108	0.0169	0.0118	0.973	0.600
	15.5	0.669	0.176	-----	0.0069	0.0274	------	0.973	0.578

Note: "-----" represents deletion of term during regression;
"cod" represents coefficient of determination;
"std" represents standard deviation.

Procedure for Testing Estimator

Equations (6.13) and (6.14) were incorporated into the version of ADAPT used in the PI controller tests. Interprop was used as the heat transfer medium to protect the internal integrity of the heat exchanger. A rotational speed of 1.0 rpm was used to minimize the amount of grain required per test. The gains of the PI controller were set high to ensure at least some oscillation during operation. Preliminary tests were run to determine the maximum setpoint change that could be made without the manipulated variable exceeding the maximum control effort. The maximum change was found to be approximately 4.5 °C.

The test procedure was then set as follows. ADAPT was initialized to bring the wheat temperature (in the heated end section) up to 33.3 °C. The setpoint was then stepped by 4.5 °C and the estimator initialized and turned on. Data on temperatures, control effort, predicted parameters and the associated predicted process transfer function gain were output each revolution to a printer. To restrict the effect of noise on the controlled variable, which is wheat temperature, a digital first-order filter was utilized.

$$Tfilt(t) = 0.25\ Tfilt(t-1) + 0.75\ Twhs(t) \qquad (6.15)$$

Tfilt was used as the input to the estimator. Once the new setpoint had been reached and the oscillations had settled, the process was repeated. A new setpoint was given and the estimator was re-initialized with new predicted parameters.

Estimation Results

Overall, the recursive least squares algorithm functioned well. Table X shows data from a sixth-order test. The first line in each range represents the initial estimates used. The second line represents the final estimates. Iterations varied from 32 to 41 within each range. The process transfer function gain appears to decrease as the machine warms up. Oscillations in the predicted parameters were not severe provided that no large noise measurements occured.

TABLE X

PARAMETER ESTIMATION RESULTS USING A SIXTH-ORDER PREDICTOR

Range (°C)	Gain Kp	Difference Equation Coefficients θ					
33.3-37.8	0.76	0.640	0.100	0.200	0.036	-0.025	0.035
	2.22	0.645	0.310	0.027	0.035	0.008	-0.003
37.8-42.2	2.13	0.640	0.299	0.039	0.035	0.008	-0.003
	1.96	0.466	0.349	0.158	0.044	0.023	-0.013
42.2-46.7	2.13	0.640	0.299	0.039	0.035	0.008	-0.003
	1.85	0.789	0.197	-0.007	0.029	0.021	-0.013
46.7-51.1	1.75	0.803	0.181	-0.006	0.029	0.022	-0.012
	1.56	0.530	0.330	0.109	0.029	0.024	-0.005

Tables XI and XII show results from a fourth-order test. Although no on-line error analysis was performed, it appeared that the fourth-order estimator was able to track the process as well as the sixth-order. Table XII shows that a dominant pole was very consistent while the second pole was somewhat erratic. Observations indicated that the estimator had much more trouble predicting the zero than was the case with the two poles.

Resetting

As noted previously, the effect of the covariance matrix P(t) decreases over time. Therefore, resetting of P(t) is required periodically to keep up with the time-varying nature of the heat exchanger. Several choices or combinations of choices exist for an algorithm to determine the proper time for resetting: (1) on a constant period; (2) when the trace (summation of elements on the diagonal) of P(t) goes below a specified limit; and (3) when the prediction error exceeds some specified limit. P(t) is best reset to a similar condition as at t=0, and that being a scalar multiple of the identity matrix.

The following logic was used in a test to determine the validity of a resetting procedure:

```
if (trace(P)<0.5) and
   (eoff(t)+eoff(t-1)+eoff(t-2)+eoff(t-3))>4.0 then
      P(t) = ko(I)  where 2.0<=ko<=5.0.
```

TABLE XI

PARAMETER ESTIMATION RESULTS USING A FOURTH-ORDER PREDICTOR

Range (^{o}C)	Gain Kp	Difference Equation Coefficients θ			
33.3-37.8	1.95	0.754	0.228	0.0267	0.0083
	1.93	0.661	0.314	0.1122	-0.0639
37.8-42.2	2.34	0.680	0.300	0.1130	-0.0066
	1.83	0.420	0.548	0.0588	-0.0002
42.2-46.7	1.91	0.430	0.540	0.0610	-0.0036
	1.66	0.803	0.171	0.0584	-0.0154
46.7-51.1	1.43	0.800	0.170	0.0580	-0.0150
	1.70	0.682	0.291	0.0229	0.0230

TABLE XII

POLES AND ZERO FROM THE FOURTH-ORDER PREDICTOR

Range (^{o}C)	Poles		Zero
33.3-37.8	0.981	-0.320	-0.570
37.8-42.2	0.979	-0.559	0.003
42.2-46.7	0.978	-0.175	-0.264
46.7-51.1	0.979	-0.297	1.004

All limits were determined through testing of the estimator. The small values of "ko" were due to the initial estimates predicting well and therefore the estimator gain not needing to be very large. The fact that the initial estimates did predict well allowed for the inclusion of an automatic "turn-on" logic which allowed the estimator to run and track the wheat temperature without updating the estimated process parameters until the above error summation was reached.

During the resetting test, the heat exchanger was allowed to stabilize at a wheat temperature of 40.0 ^{o}C under PI control as before. The setpoint was then shifted to 46.1 ^{o}C and the estimator initialized. Once the wheat temperature began to stabilize at the new setpoint, the setpoint was again shifted, this time to 54.4^{o}C.

As can be seen in Figure 33, the estimator tracked the actual wheat temperature quite well. The spike on the predicted temperature occurred when the manipulated variable suddenly changed due to the change in setpoint. The process characteristic equation parameters (Figure 34) seemed to mirror one another, thus holding the dominant pole close as was seen in Table XII. The transfer function numerator dynamics (Figure 35) appeared to maintain a steady zero while oscillating. Except shortly after the second and last setpoint change, the predicted process transfer function gain was relatively stable (Figure 36).

Five automatic initializations, or resets, occurred

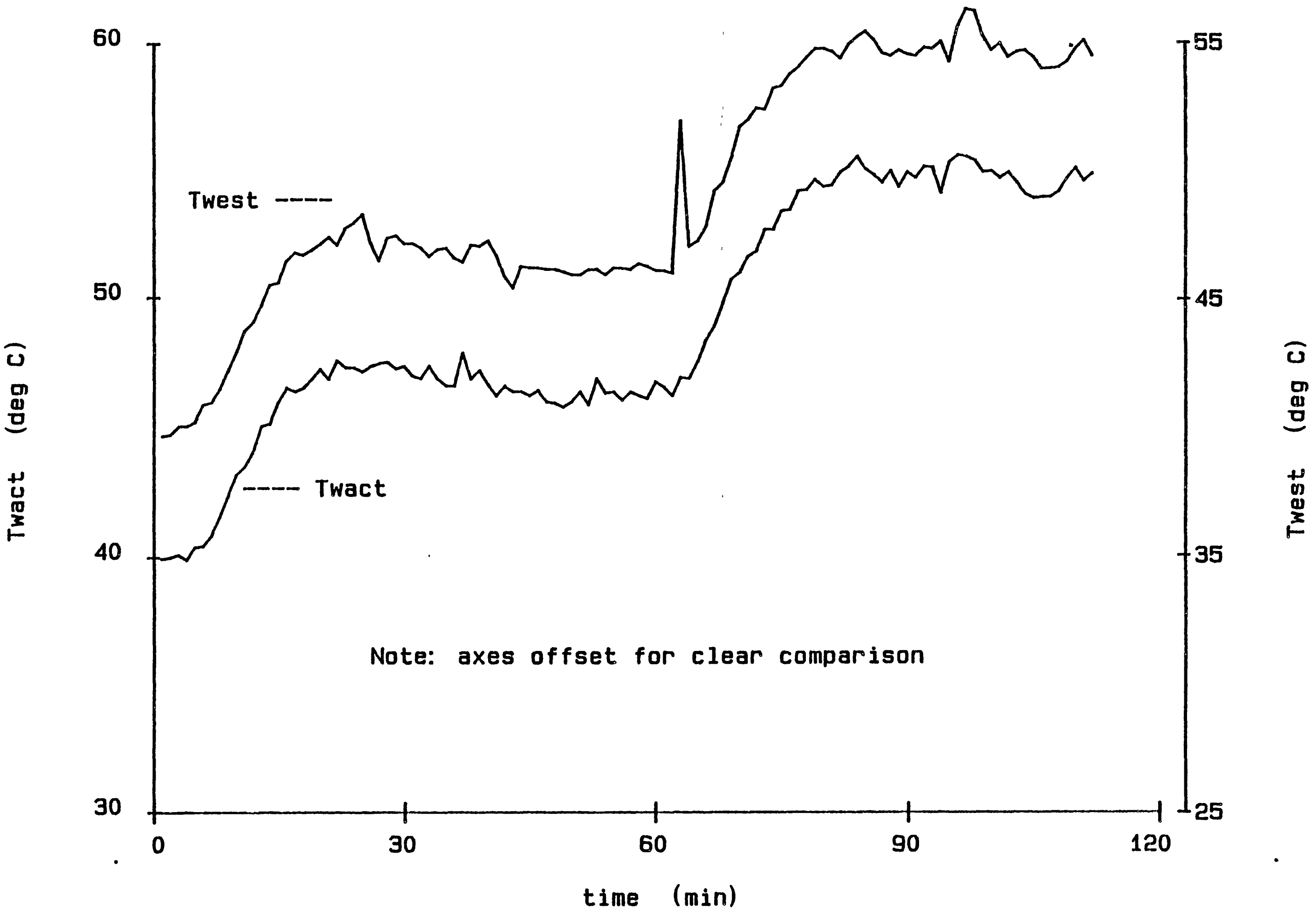


Figure 33. Tracking Capability of the Fourth-Order Predictor

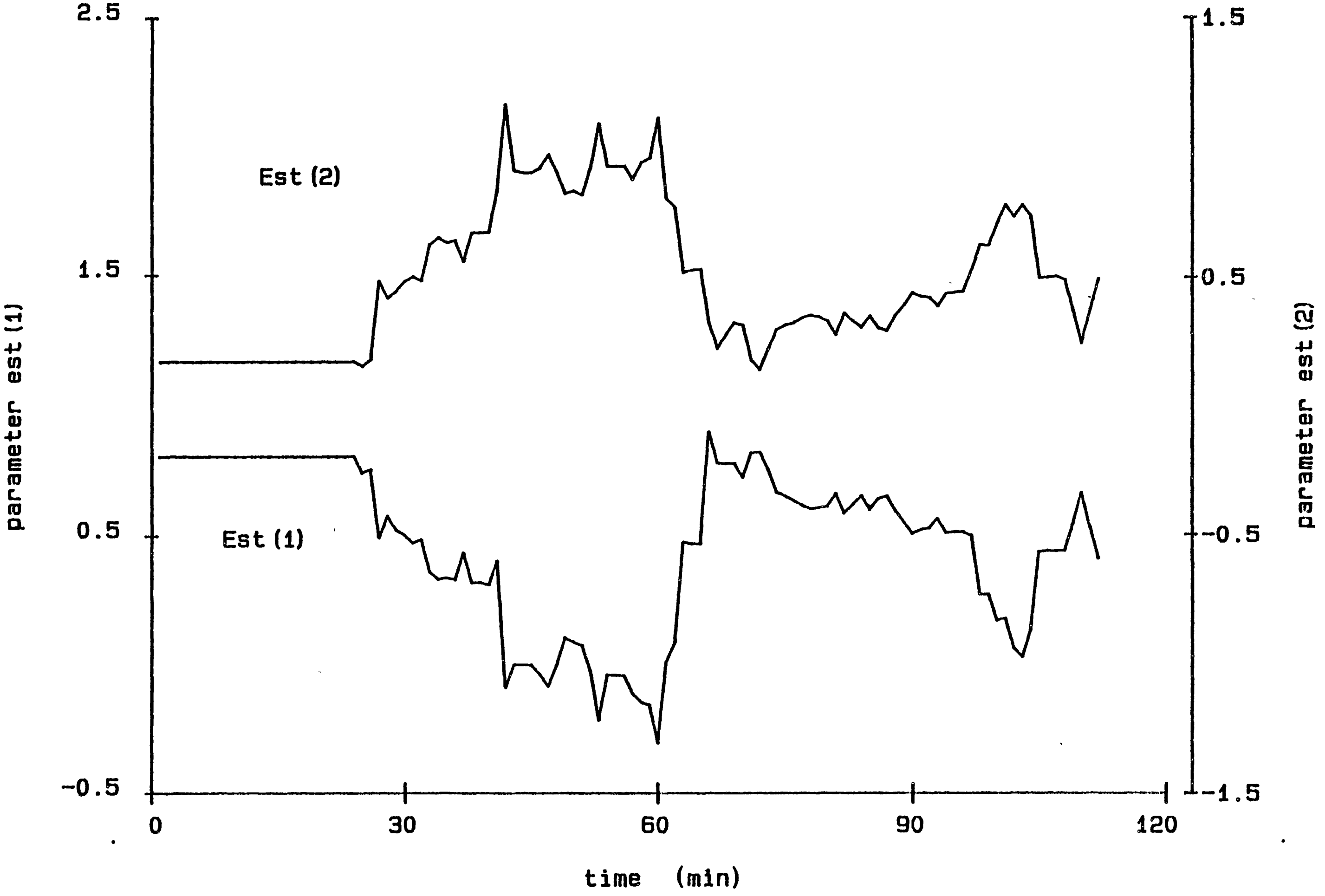


Figure 34. Fluctuation of Output Variable Coefficients in Fourth-Order Predictor

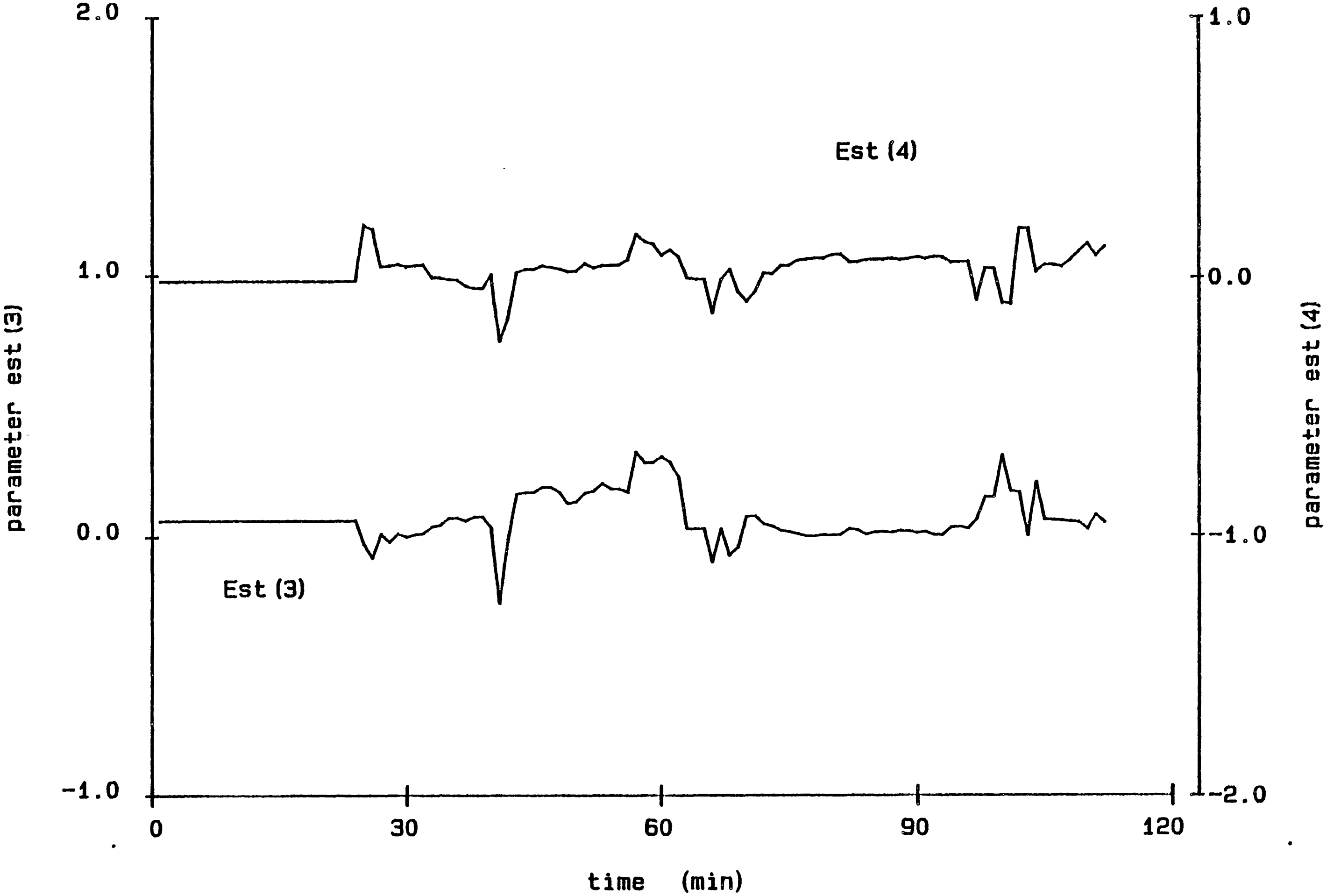


Figure 35. Fluctuation of Manipulated Variable Coefficients in Fourth-Order Predictor

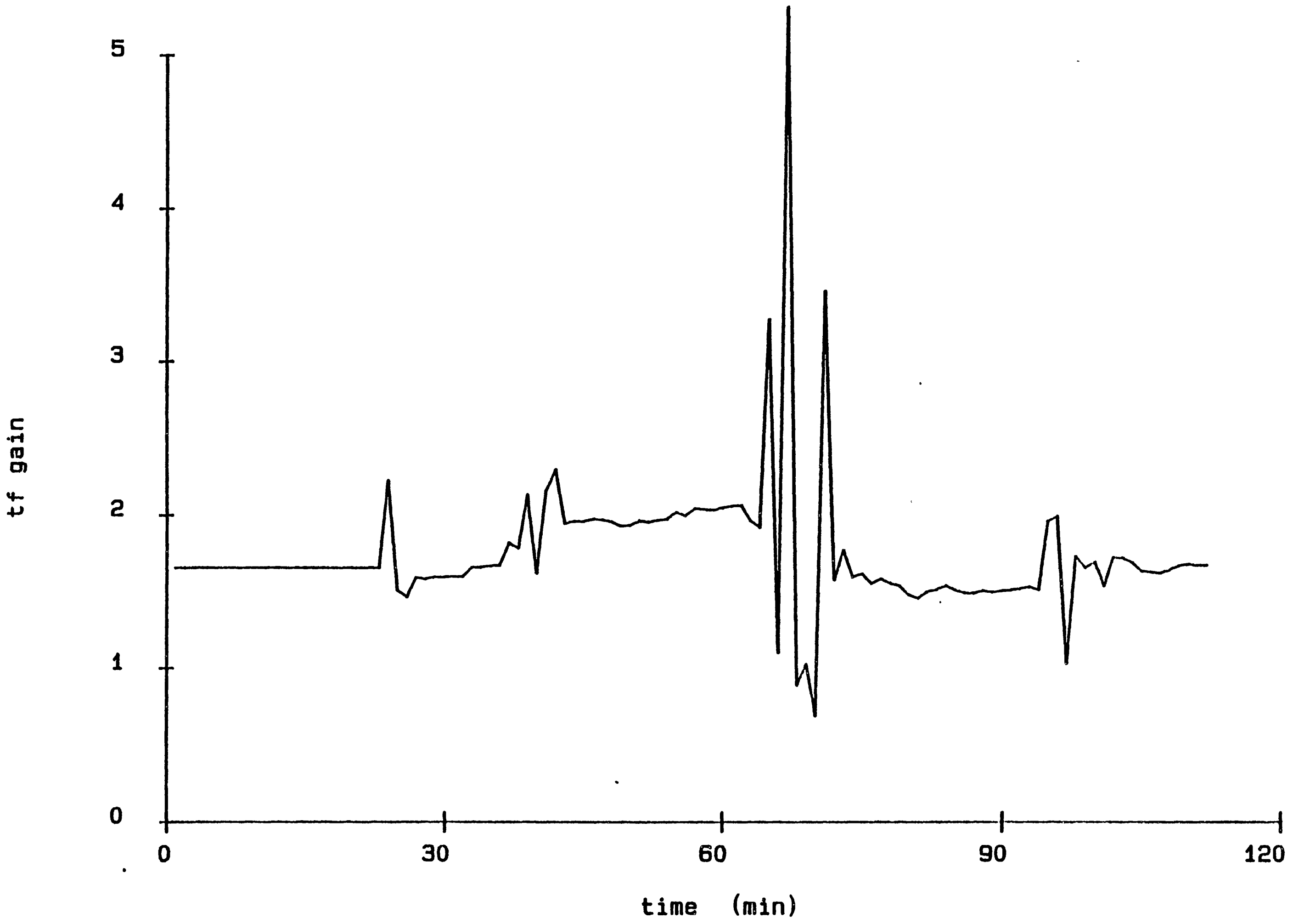


Figure 36. Fluctuation of Gain of Predicted Process Transfer Function

during the test. The first was due to the self-initialization, or automatic turn-on feature, at t=23 minutes; the remainder occurred at various times when the specified constraints were met. Based on the results of the test, the reset mechanism appeared satisfactory.

Adaptive Control

Derivation of Control Law

A digital control law based on pole placement, setting desired system response characteristics by specification of closed-loop poles, was derived for the adaptive heat exchanger controller. While all control laws are in fact pole placement based, those specified as such directly manipulate gains to achieve some desired closed-loop characteristic equation. Several methods exist, including phase variable feedback, to determine the required control law. The most popular method in digital control is controller synthesis. Well-known digital algorithms such as the deadbeat and Dahlin's algorithms were formulated using the controller synthesis approach. This method is particularly useful in the ability to specify a desired closed-loop characteristic equation and also incorporate any deadtime which may be present.

Based on the parameter estimation work, it was assumed that the process could be modeled by a difference equation of second-order and discrete deadtime of two as:

$$y(t) = a1\ y(t-1) + a2\ y(t-2) + b0\ u(t-3) + b1\ u(t-4). \qquad (6.16)$$

Transforming to the z-domain and solving for the process transfer function yields:

$$Gp(z) = \frac{b0\ z + b1\ z}{z2(z2 - a1\ z - a2\ z)} \qquad (6.17)$$

where

$$Gp(z) = Y(z)/U(z).$$

It was also assumed the desired closed-loop characteristic equation could be written as:

$$\frac{Y(z)}{R(z)} = \frac{z\ (1 - p1)\ (1 - p2)}{z2\ (z - p1)\ (z - p2)}\ . \qquad (6.18)$$

To begin the controller synthesis, the transfer function of the controller "Gc(z)" was derived in terms of the other transfer functions of the system (Figure 37).

$$Gc(z) = \frac{(\ Y(z)/R(z)\ )}{Gp(z)\ (\ 1 - Y(z)/R(z)\ )} \qquad (6.19)$$

where

$$Gc(z) = U(z)/E(z)$$

Substituting (6.17) and (6.18) into (6.19) and simplifying yields the transfer function of the controller.

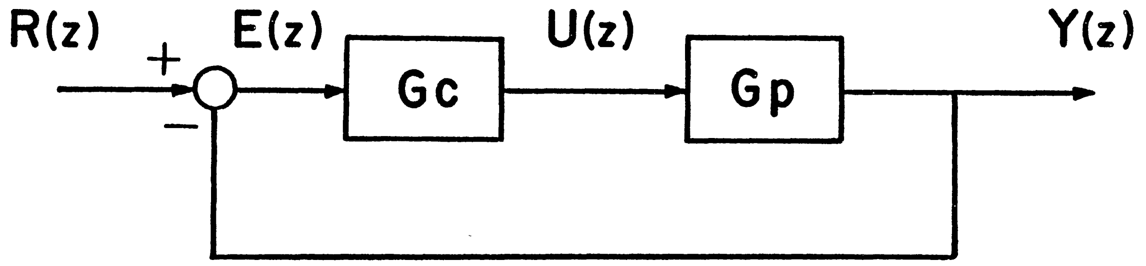


Figure 37. Block Diagram of Closed-Loop System

$$Gc(z) = \frac{z2\ (z2 - (a1)z - (a2))\ (1-p1)\ (1-p2)}{\begin{gathered}\{\ (b0)z4 + (b1 - b0(p1+p2))z3 + (b0p1p2 - b1(p1+p2))z2 \\ + (b1p1p2 - b0(1-p1)(1-p2))z - b1(1-p1)(1-p2)\ \}\end{gathered}} \qquad (6.20)$$

Lastly, inverting back to the discrete time domain and solving for the manipulated variable yields the following control law.

$$\begin{aligned} u(t) = K\ \{\ e(t) - (a1)e(t-1) - (a2)e(t-2)\ \} & \qquad (6.21) \\ - \{(b1-b0(p1+p2))/b0\}\ u(t-1) & \\ - \{(b0p1p2-b1(p1+p2))/b0\}\ u(t-2) & \\ - \{b1p1p2-b0(1-p1)(1-p2)/b0\}\ u(t-3) & \\ + \{b1(1-p1)(1-p2)/b0\}\ u(t-4) & \end{aligned}$$

where

$$K = (1-p1)(1-p2)/b0$$

The above control law is not adaptive. However, the process parameters a1,a2,b0, and b1 are required for the gain calculations. By using the estimated process parameters from the recursive least squares algorithm, the gains become time-varying and thus (6.21) along with (6.13) and (6.14) formulate an adaptive controller.

Preliminary Results

Equation (6.21) was incorporated into ADAPT (see Appendix) and preliminary tests were conducted to determine basic information on ability to control to a specified setpoint, stability, and noise immunity. Several problem areas were discovered in these tests. The first problem concerned when to update the gains in the controller.

Although the estimation algorithm may update the process parameters each revolution, it became obvious that the controller gains need not and should not be updated as frequently. Since the major objective of the control system was to start-up the process in the most effective manner, the controller could not operate with a "poor" set of gains at any time. To solve this problem, the estimator and controller were decoupled. A set of constraints was developed which attempt to ensure that the controller gains will not be updated when the predicted process transfer function is inaccurate.

One possible constraint was found in the literature (Seborg, et al. 1983). The authors suggested monitoring the predicted process transfer function gain and updating the controller gains only if the former was reasonable. Although this constraint was implemented, others were required also. For instance, during some of the parameter estimation tests, the manipulated variable coefficient "b0" would at times go negative, which is not realistic for a positive input - positive output system.

Another example is the predicted zero ("b0/bl") in the process transfer function. In digital control theory, stable roots (either poles or zeros) are those which lie within a region of the z-plane called the unit circle. A control law derived using the controller synthesis approach is only stable for those processes with stable zeros. If the estimated zero falls outside the unit circle, the new calculated gains will cause a ringing condition. This phenomenon can be seen in Figure 38, where the plus symbols indicate times which the gains were updated when the estimator predicted a zero outside the unit circle. Obviously such a ringing effect is undesirable.

Two other constraints were also added which restricted controller gain updating. One specified a maximum bound on the predictor error, the difference between the actual (filtered) wheat temperature and the predicted wheat temperature, to be within 1 °C. The last constraint placed a bound on the coefficient "b0", necessary since as the estimated b0 increased, the base error gain K (see equation (6.21)) sensitivity to controller error decreased.

Another problem encountered was that of noise immunity. Noise, either of process or measurement origin, was generally insignificant. However, perhaps one to three instances per test, a noisy measurement would occur which caused the controller to react adversely. Figure 39 shows a test where this occurred. Although the controller did stabilize, wheat temperature dropped and did not reach setpoint during the remaining 30 minutes of the test. To help alleviate the noise problem, a clamp was placed on the derivative of the measured error (setpoint minus actual temperature). Since the sampling rate was much greater than the dominant natural frequency of the process, the bounded derivative only affected the controller when a large noise

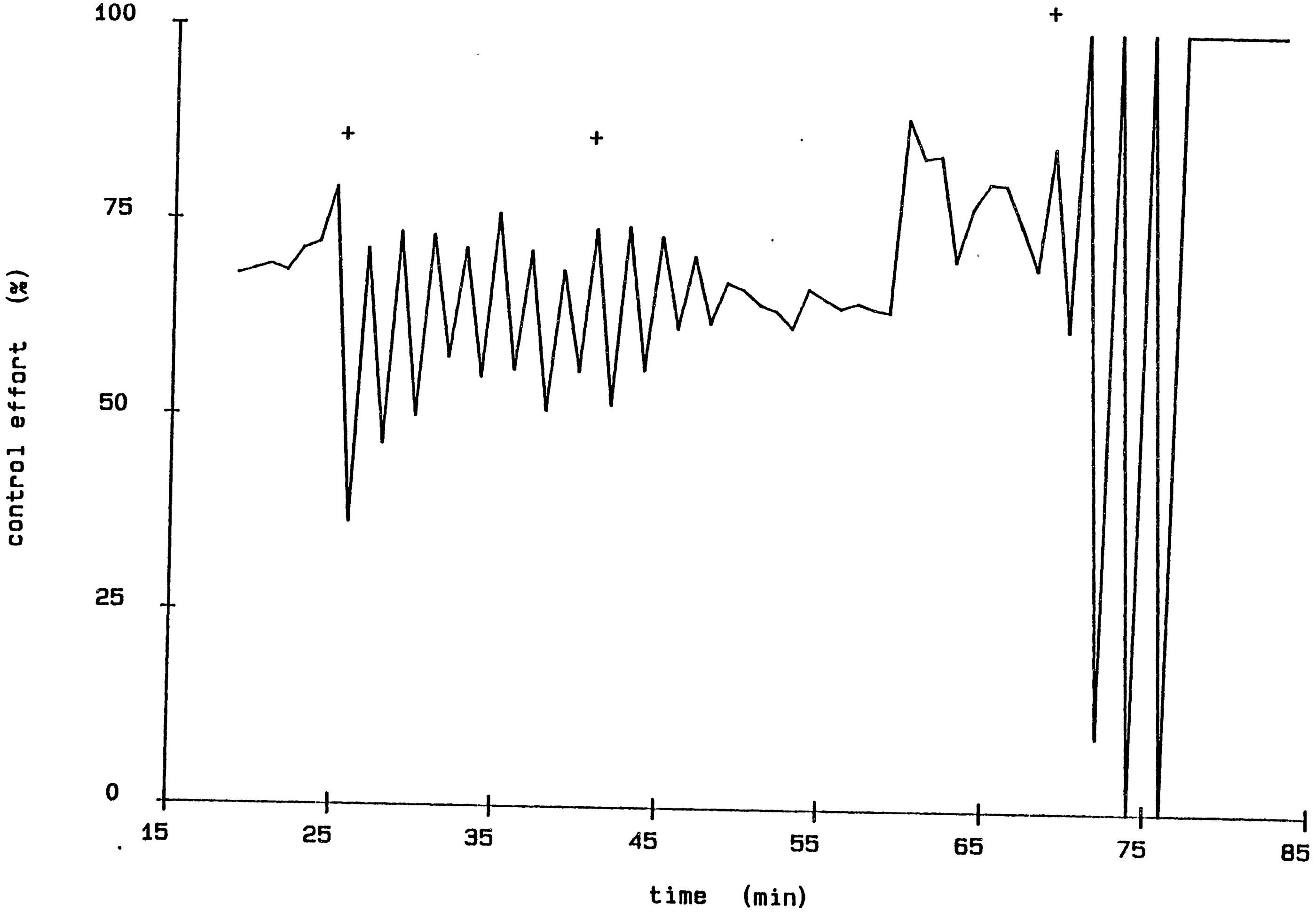


Figure 38. Effect of Unstable Zero on the Manipulated Variable

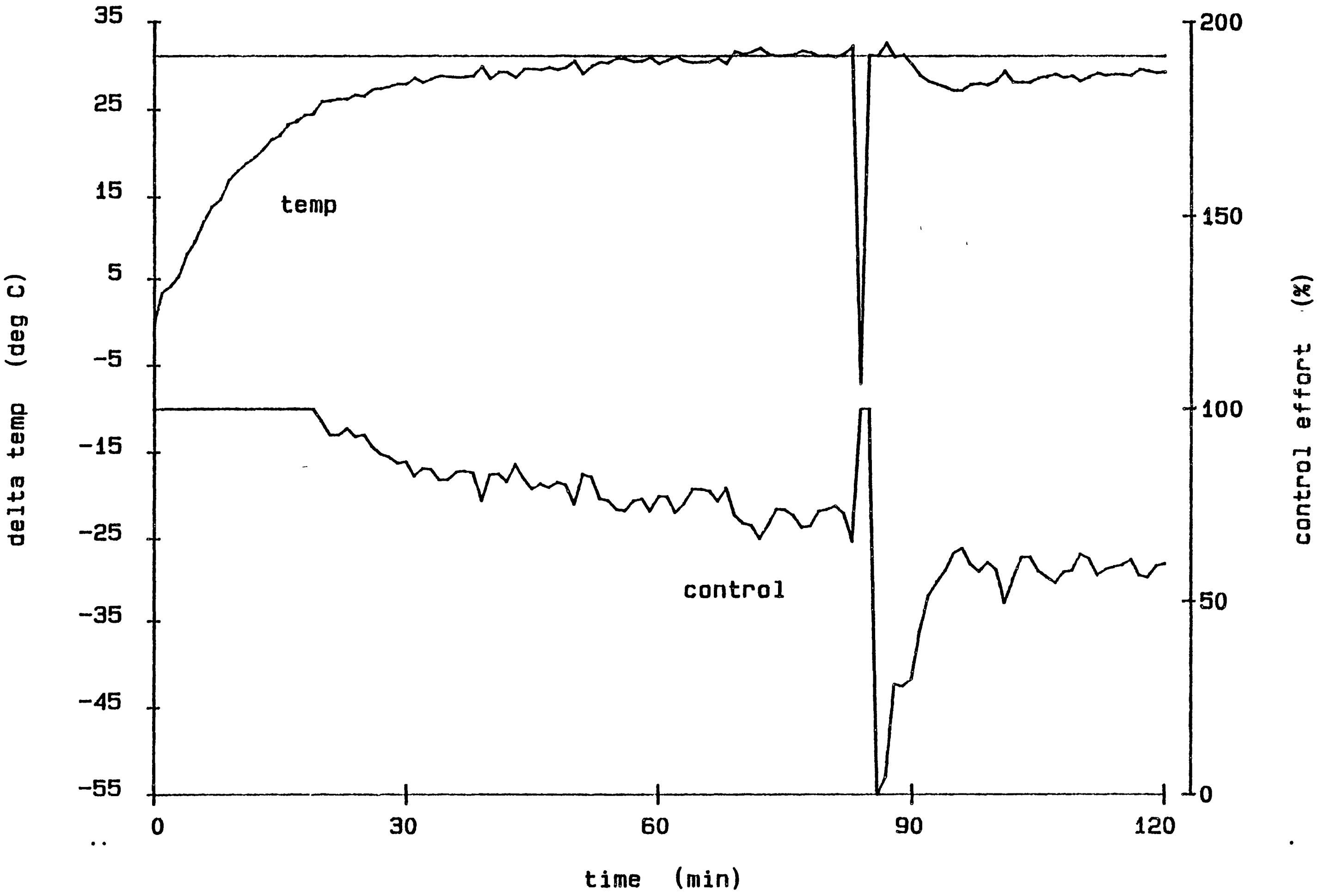


Figure 39. Effect of Noise on the Manipulated and Output Variables

in measurement occurred.

The last problem area concerned the base error gain K in equation (6.21) and its associated sensitivity to the controller error. The structure of the control law (6.21) is heavily influenced by the deadtime compensation. However, if K is updated and falls below approximately 1.5, the controller's ability to respond to error and drive the wheat temperature to the setpoint is severly impaired. As noted above, this problem was partially alleviated by placing a constraint on the controller gains being updated. To further ensure that K remained in a desirable range, an upper and lower limit was used. The lower limit was equivalent to approximately 30% of the conventional controller's optimal proportional gain and the upper limit equivalent to 120%.

Tuning

As in the case of the conventional controller, the adaptive controller must be adjusted, or optimized, to fit the particular process of concern and to obtain the desired system response characteristics. Provided that sufficient control effort is available and a model of the process transfer function is known, analytical methods can be used as the basis for tuning the controller. However, the solid particle heat exchanger has very limited control effort capability and more empirical methods of adjustment were required.

The free parameters available for tuning with the controller structure of (6.21) are the two poles of the desired closed-loop characteristic equation. Figure 40 demonstrates the effect of poor specification of the closed-loop poles. At the time the wheat temperature had reached the setpoint, the calculated control effort was more than 150% of the actual available control effort in the system. Had the test been continued, a severe overshoot of the controlled variable would have occurred.

Pole-Placement Simulation

To minimize the number of tests required for optimal specification of the closed-loop poles, a simulation was written (TUNESECN, see Appendix). The results from the parameter estimation tests were utilized to construct an approximate dynamic model of the process in difference equation form. Using the controller (6.21) in a non-adaptive framework, the simulation was allowed to perform 150 iterations. This would be analogous to 150 revolutions, or two hours of operation at 1.25 rpm. The closed-loop poles were varied and the integral of absolute error was used as the performance measure. The actual available control effort to the process was clamped as in the real physical system. Table XIII summarizes the results of the simulation runs. A penalty can be observed when K is large during startup, in large part due to the limited control effort available.

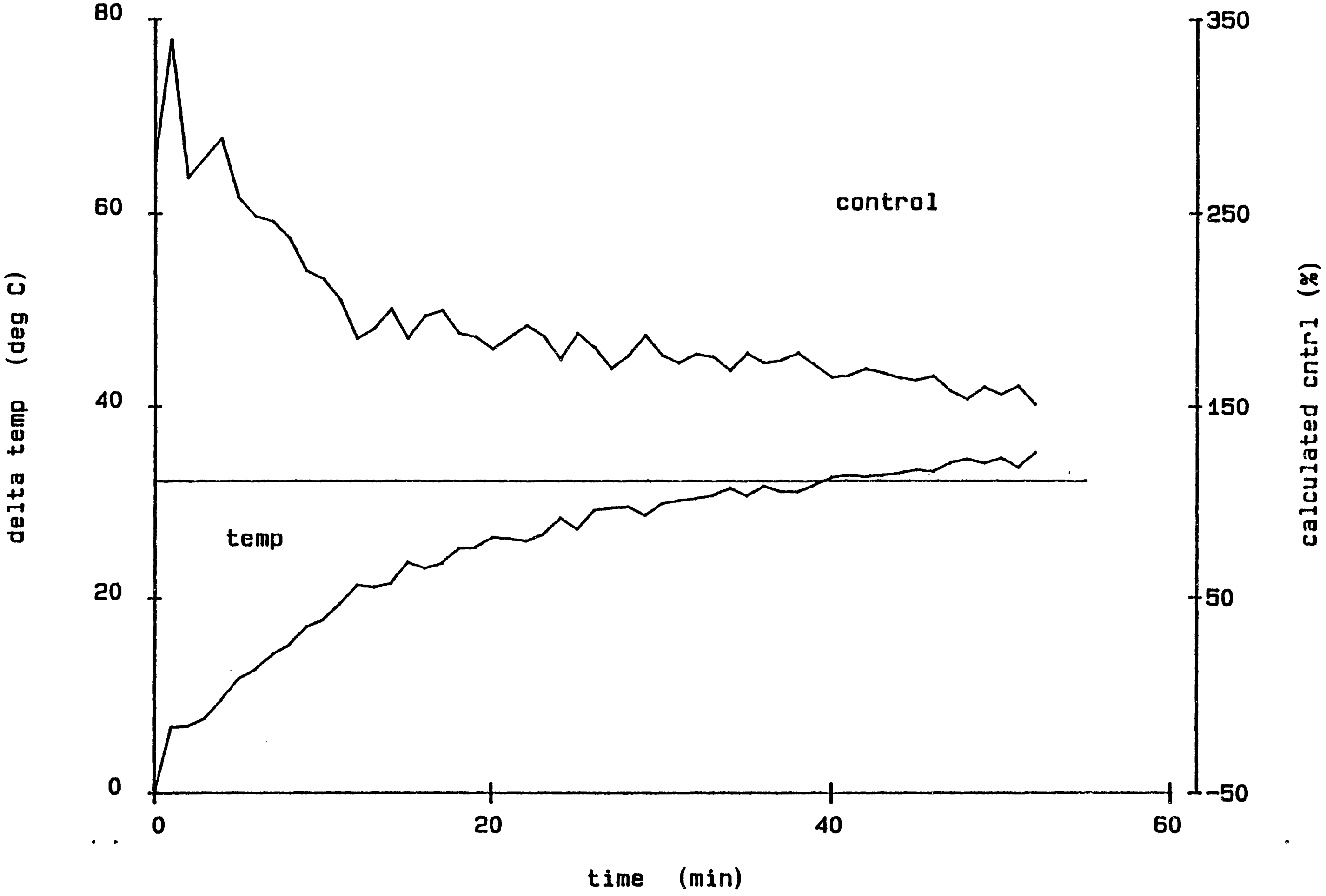


Figure 40. Overshoot and Manipulated Variable Saturation Caused by Poor Pole-placement

From the above results, the pole pairs (0.90,0.10), (0.85,0.15), and (0.80,0.20) were chosen for investigation in the heat exchanger controller. The adaptive tests were run under conditions held as closely as possible to the conventional controller tuning tests. Due to fluctuations in the ambient condition and inlet wheat temperature, the wheat setpoint temperature was always set 32.2 °C above the initial inlet wheat temperature. The environment was generally 2-3 °C lower than during the conventional controller tests.

Tuning Results

The adaptive control system performed well. The system was effective in causing the wheat temperature to reach the

TABLE XIII

POLE-PLACEMENT SIMULATION RESULTS FOR PREDICTION OF OPTIMAL STARTUP PERFORMANCE IN THE PRESENCE OF LIMITED INPUT CONTROL ENERGY

Poles	"K"	Controller Gains				Performance Measure
0.90,0.40	1.03	1.99	-1.25	0.30	-0.04	2711
0.90,0.10	1.54	1.68	-0.77	0.15	-0.06	1657
0.85,0.15	2.18	1.68	-0.81	0.22	-0.09	1369
0.80,0.20	2.74	1.68	-0.84	0.27	-0.11	1518
0.70,0.30	3.60	1.68	-0.89	0.35	-0.14	1630
0.60,0.40	4.11	1.68	-0.92	0.40	-0.16	1659
0.50,0.50	4.28	1.68	-0.93	0.42	-0.17	1665
0.60,0.10	6.16	1.38	-0.54	0.40	-0.24	1682

setpoint. Results of the adaptive controller tuning tests can be found in Table XIV. Figure 41 shows the temperature and control trajectories for the test using the optimal pair of closed-loop poles. The optimal pair of the closed-loop pole specification, using the performance measure J, was (0.85,0.15).

During the optimal adaptive run, a sharp noise measurement occurred at the 65 minute mark yet had a minimal effect on the system due to the error derivative clamp. Note that in post-test data analysis the noisy measurements, those which were significantly in error from the true condition, were removed from the error integral calculation and replaced with interpolated values. This was necessary due to a varying number of noise occurrences per test.

TABLE XIV

ADAPTIVE CONTROLLER TUNING RESULTS FOR THE MACHINE USING A CONSTANT SPEED

Poles	Integral of Absolute Error	Integral of Energy Input	Performance Measure
0.90,0.10	1126	3790	2074
0.85,0.15	909	3798	1859
0.80,0.20	910	4188	1957

The performance measure of the optimal adaptive

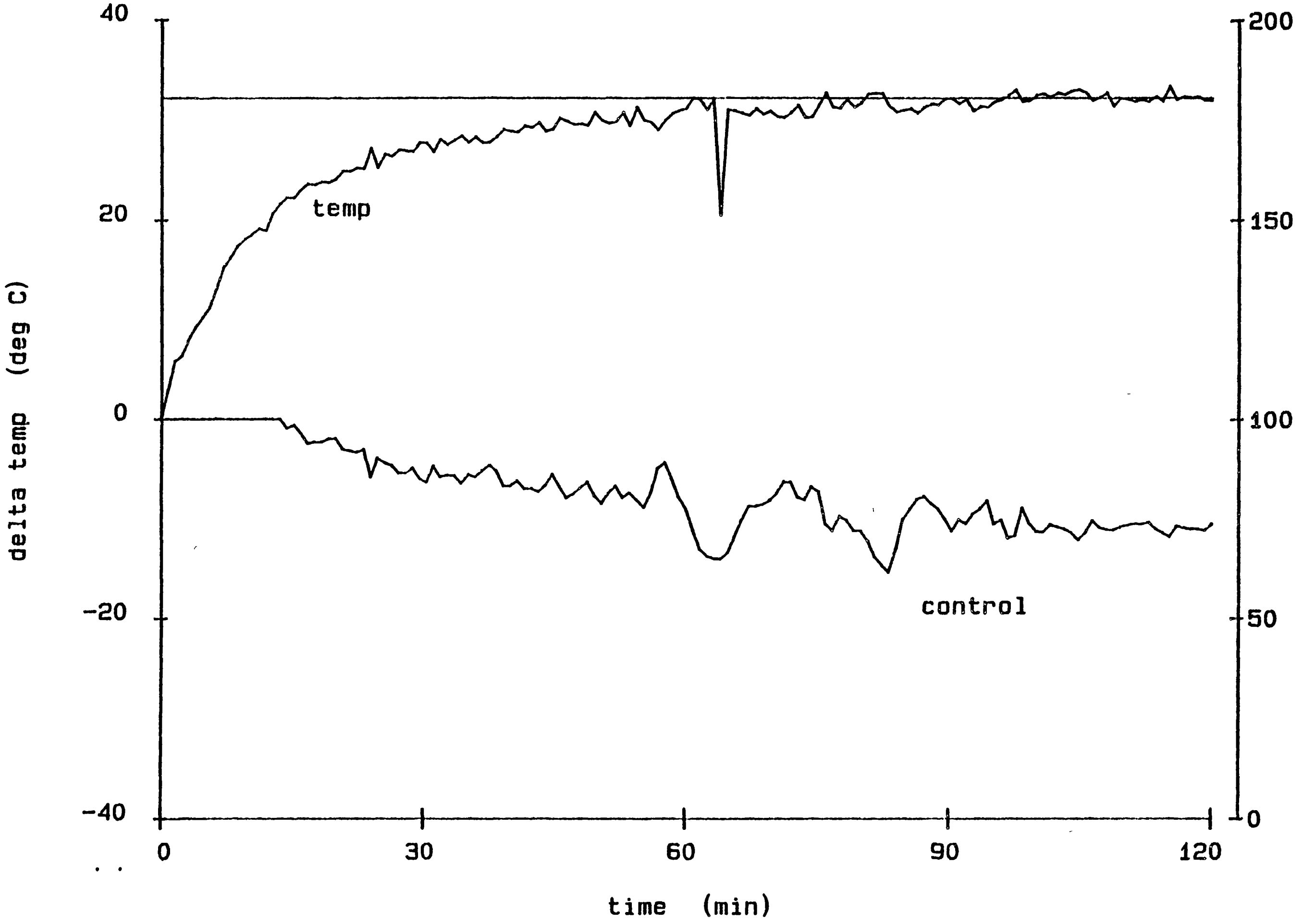


Figure 41. Optimal Manipulated and Output Variable Trajectories for the Adaptive Controller (Constant Speed)

controller was 4.6% less than that of the optimal conventional controller. Although it is difficult to compare Figures 41 and 32 due to the difference in data sampling rates, it would appear the adaptive controller caused less oscillation about the setpoint than did the conventional controller. The reduction in oscillation can be attributed to compensation for deadtime in the adaptive controller while the conventional controller does not.

Slow Speed Start-Up

The majority of the error in wheat temperature during startup can be attributed to the "rolling off" of the temperature profile as the control effort decreased from 100%. The rate of increase of the wheat temperature during startup is heavily influenced by the transit time of the grain and therefore by the rotational speed of the machine. To take advantage of this factor, three tests were run which started the heat exchanger at a slower speed and increased the speed as the wheat temperature setpoint was reached.

The first attempt was made using the pole pair (0.85,0.15) and an initial speed of 0.65 rpm. The algorithm was set to increase the speed by 0.2 rpm each time the setpoint was reached to a maximum of 1.25 rpm. During the two hour run, the speed never exceeded a setpoint of 0.85 rpm. The controller gains, particularly "K", were too low for the reduced speed.

The second test used as desired poles (0.70,0.30) and an initial speed of 0.85 rpm. The algorithm was again set to increment the speed setpoint by 0.2 rpm. The resulting performance was better than the optimal adaptive run but an overshoot condition occurred. Also, the in-between speed step to 1.05 rpm did not appear to have any real effect on performance.

Therefore, a third and final run was made using as poles (0.80,0.20) at an initial speed of 0.85 rpm. The algorithm was set to increase the speed directly to 1.25 rpm once the wheat temperature setpoint was reached. Excellent results were obtained. Figure 42 shows that the setpoint was reached at the 43 minute mark as opposed to the 62 minute mark during the optimal single-speed adaptive run. The plus sign on Figure 42 shows the point at which the heat exchanger rotational speed was changed to 1.25 rpm. The wheat temperature dropped off but came back to the setpoint quickly. The three noisy measurements did not seem to have a significant affect. Table XV, summarizing the results, shows the optimal slow speed adaptive controller was a 6.9% improvement over the single speed adaptive controller and an 11.1% improvement over the conventional controller with respect to the performance measure J.

Final Discussion on the Adaptive Implementation

Although the adaptive controller(s) did outperform the conventional controller during heat exchanger startup with

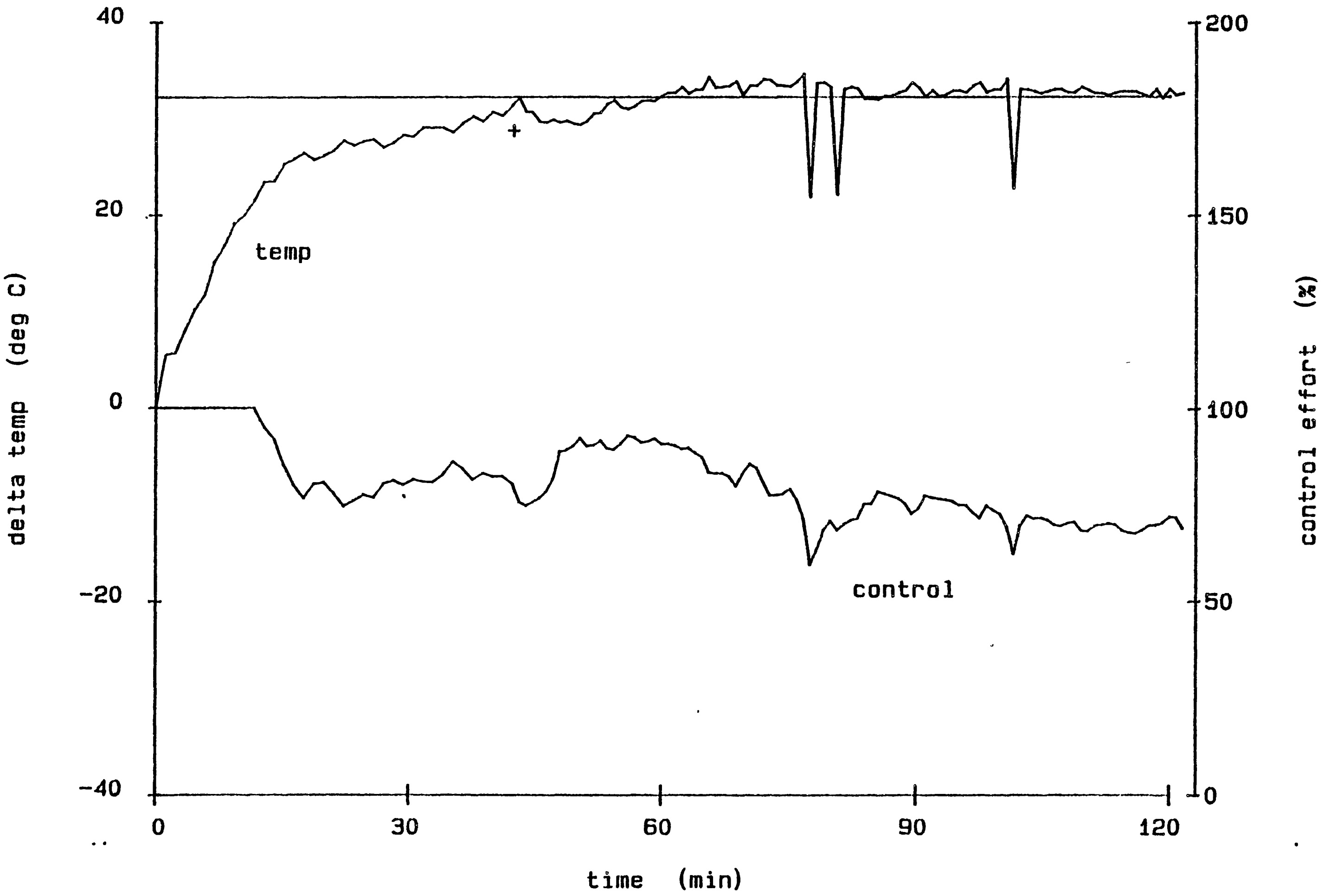


Figure 42. Optimal Manipulated and Output Variable Trajectories for the Adaptive Controller (Slow-Speed Startup)

TABLE XV

ADAPTIVE CONTROLLER TUNING RESULTS FOR THE MACHINE USING SLOW-SPEED START-UP

Test Conditions Poles	Speeds	Integrals of Absolute Error	Energy Input	Performance Measure "J"
0.85,0.15	0.65,0.85	1150	3009	1902
0.80,0.20	0.85,1.25	784	3789	1731
0.70,0.30	0.85,1.05,1.25	826	3938	1810

respect to the performance measure J, a question exists as to why. For instance, did the increased performance come as a result of the adaptive nature of the control law, or the structure of the control law, or some combination of both? Figures 43-46 depict the time histories of all the controller gains associated with the optimal single speed adaptive controller test. The mirrored images noted on Figures 44-46 and the relatively small changes in "K" as seen on Figure 43 might suggest that the controller structure of (6.21), derived with both the ability to perform derivative action and to compensate for deadtime, might alone be responsible for the increased performance relative to the conventional proportional-integral controller. More testing would have to be performed to adequately answer this question.

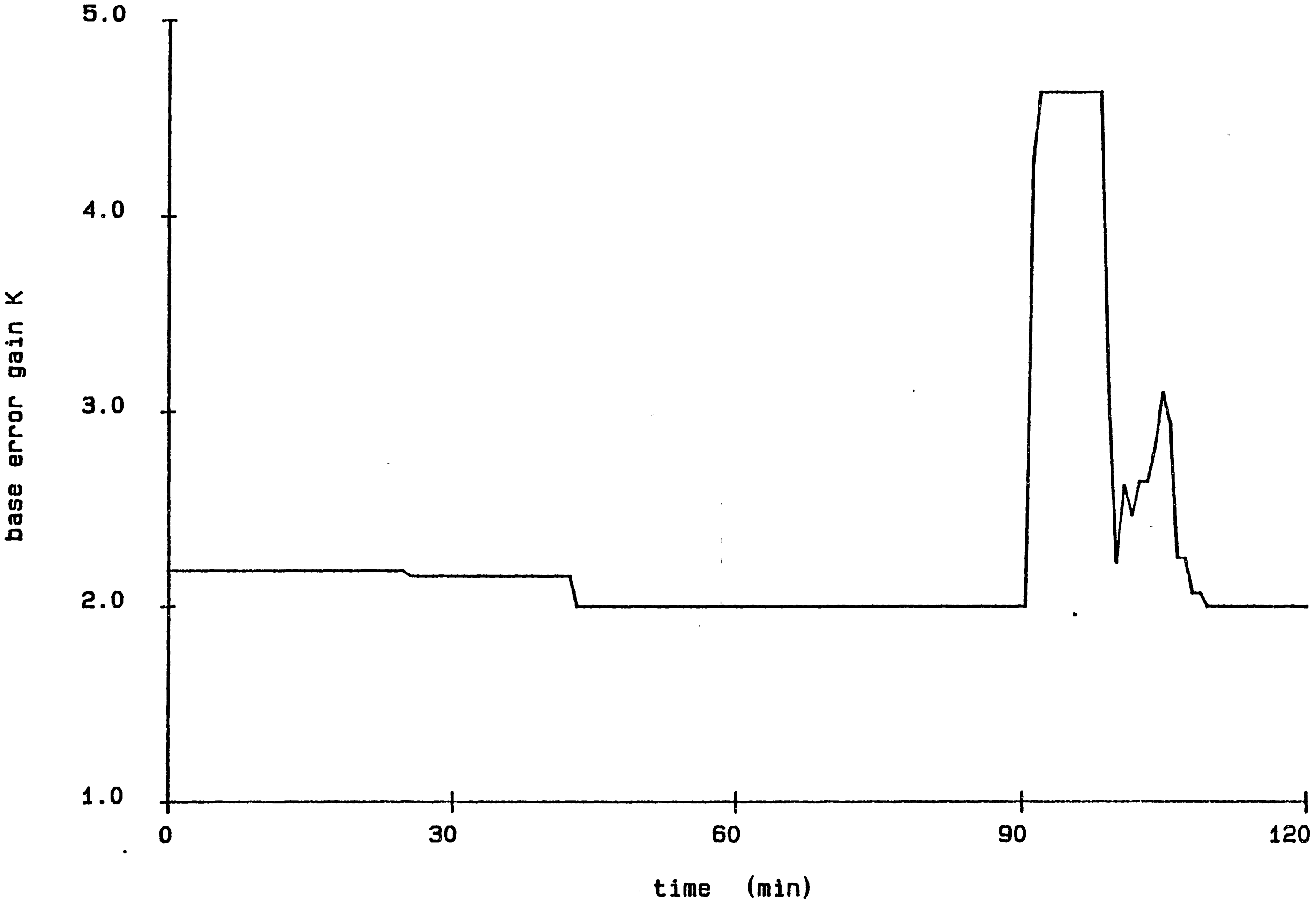


Figure 43. Base Error Gain During Optimal Adaptive Test

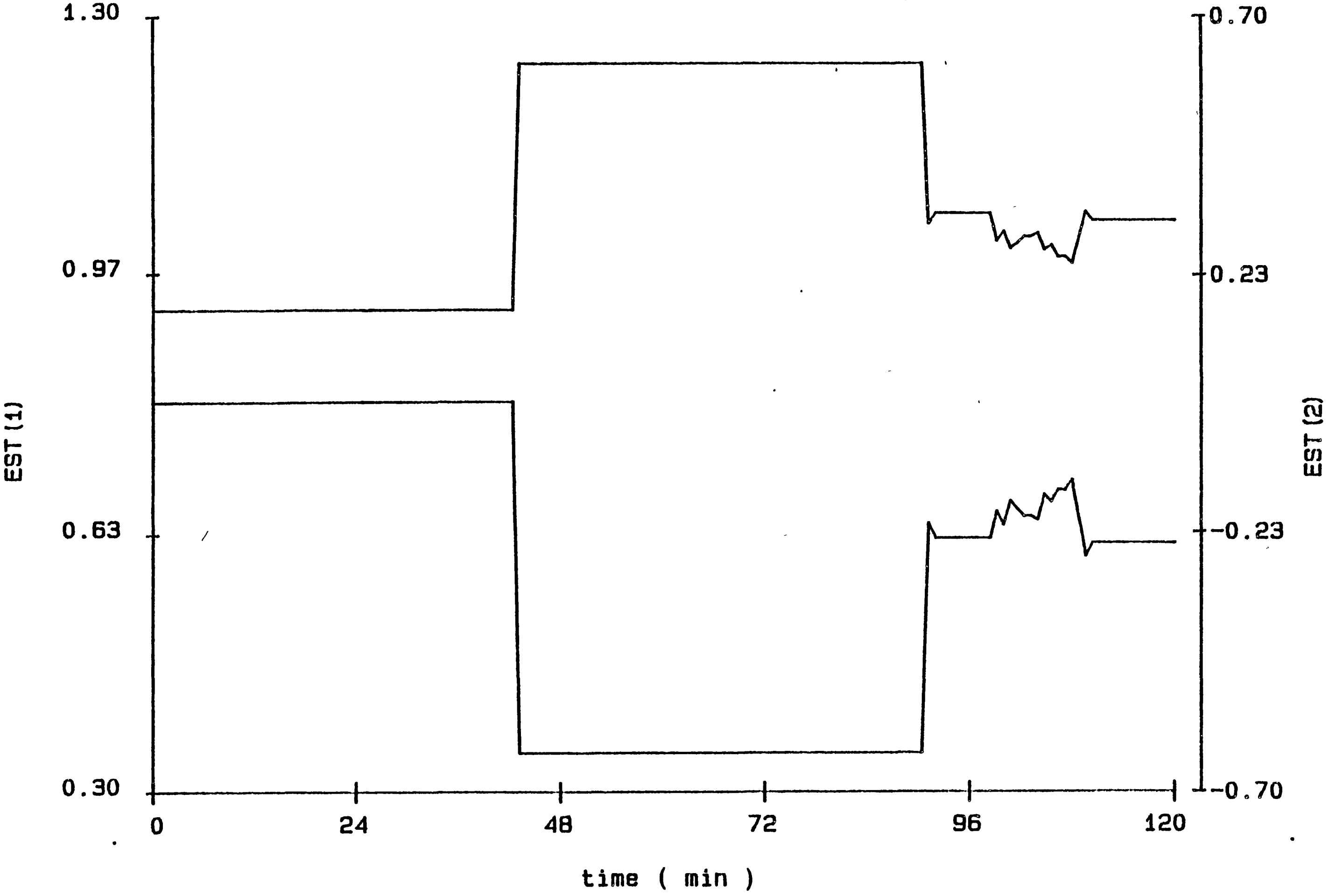


Figure 44. Estimated Process Parameters Used as Gains During Optimal Adaptive Test

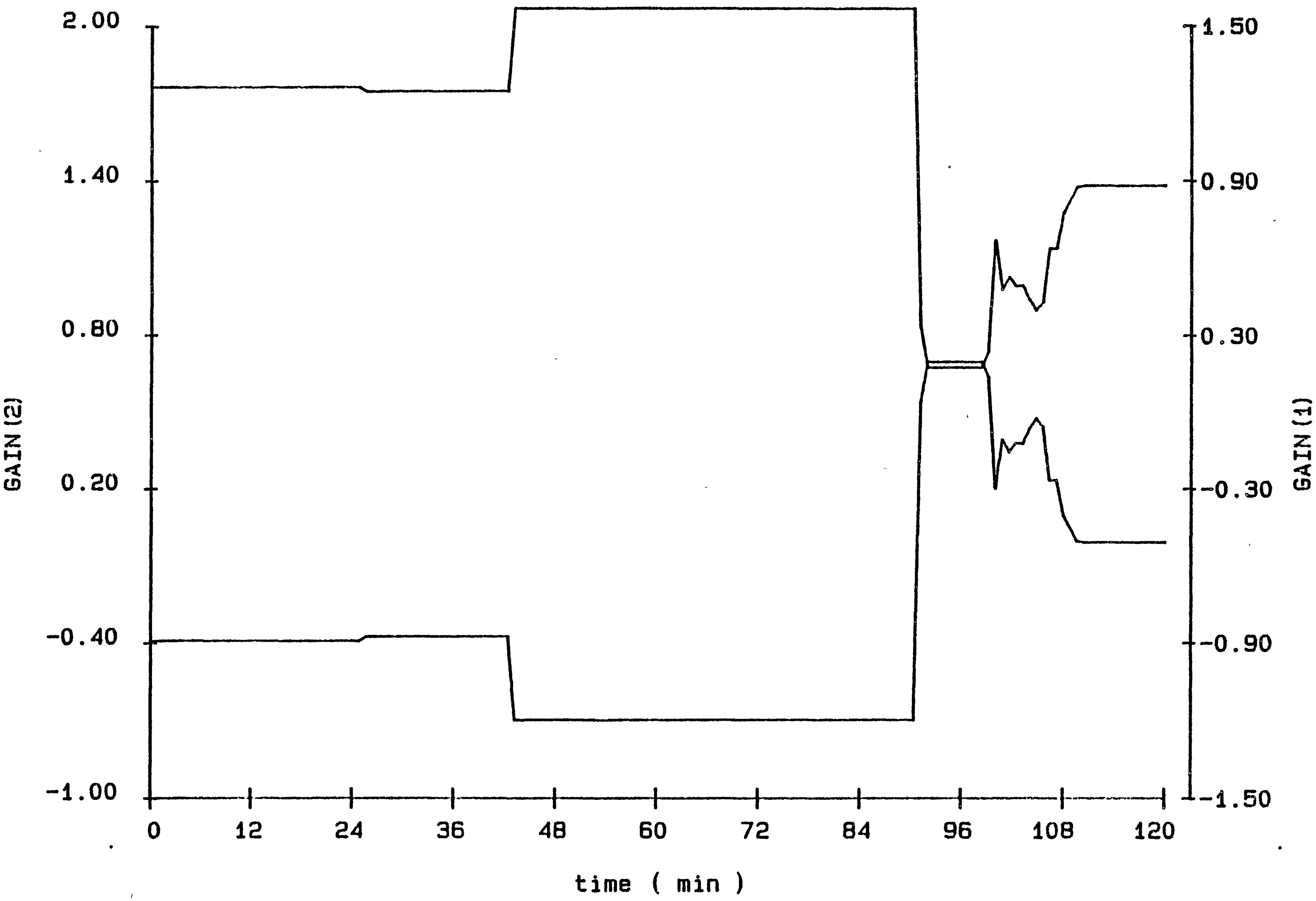


Figure 45. Gain History of First Two Input Coefficients During Optimal Adaptive Test

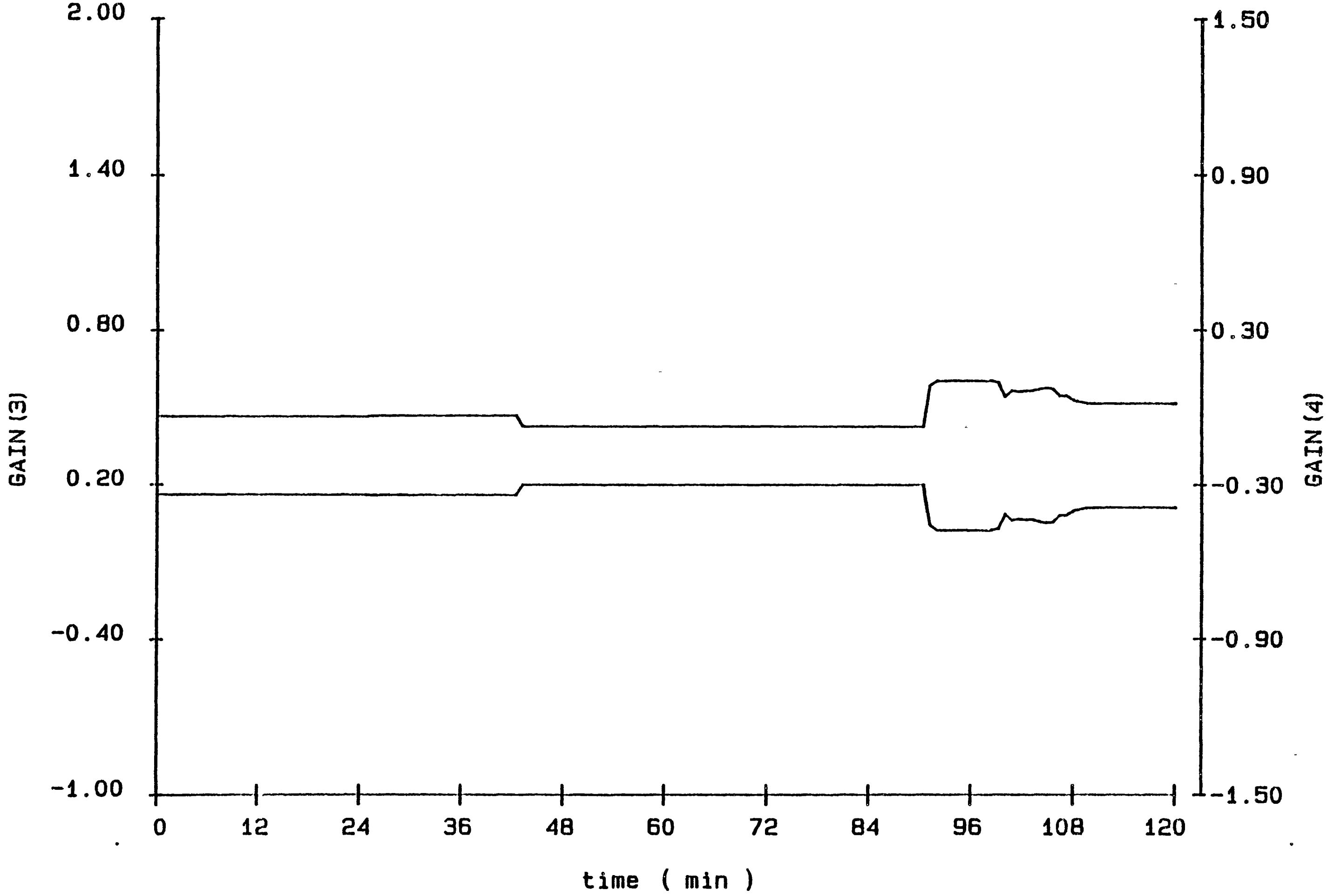


Figure 46. Gain History of Second Two Input Coefficients During Optimal Adaptive Test

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS

Due to problems associated with pesticides currently used on stored grain insects, knowledge concerning alternative methods of disinfestation is desirable. Heat has been successfully utilized in the past to disinfest several grains, including wheat. The general objective of this study was to analyze a thermal-based process that might be used to economically perform disinfestation of wheat.

The process developed and studied was counterflow particle-to-particle heat exchange, a means of heat transfer using small solid particles such as salt to conduct the heat energy while simultaneously allowing for significant heat recovery. A simplified mathematical model was derived and, along with laboratory test information, used in a simulation to predict potential heat exchanger characteristics. A prototype was designed, constructed, and tested to determine feasibilty and performance of a physical implementation. And lastly, several control schemes were investigated to optimize startup of the prototype heat exchanger.

Conclusions

The major conclusion of this investigation is that counterflow particle-to-particle heat exchange has good potential as an energy efficient heat transfer mechanism and can be implemented successfully in a real physical device.

Specific conclusions drawn from the investigation can be itemized as follows.

(1) A counterflow particle-to-particle heat exchanger has an excellent chance of successfully performing thermal disinfestation. The temperature profiles and transit times achieved in the prototype are compatible with previous work done on thermal disinfestation.

(2) Given a properly sized and insulated machine along with market forces dictating a need for high quality grain, thermal disinfestation performed on a counterflow particle-to-particle heat exchanger can be accomplished economically. This is in large part due to the energy efficient nature of the device, which is able to recover a significant amount of energy from the processed grain.

(3) The prototype counterflow particle-to-particle heat exchanger is not an ideal machine. The gravity-based flow design requires the volume and mass of the heat exchanger to be overly large in comparison to the amount of material processed.

(4) Using more than 5 cells/stage will be required to

obtain the most benefit out of the prototype design, up to a maximum of approximately 15 cells/stage.

(5) Problems exist with the possible heat transfer media. Common table salt was shown to be superior to Interprop with respect to heat transfer efficiency. However, salt absorbs moisture easily, and this property can cause flow and corrosion problems within a heat exchanger. Other materials need to be investigated for use as a heat transfer medium. Important medium properties include small particle size relative to grain, limited tendency to absorb moisture, a bulk density somewhat greater than the grain, minimal corrosion effects, and no harmful effects on man or animal.

(6) A properly designed and tuned control system can significantly increase heat exchanger performance during the startup period. A sophisticated adaptive controller, based on a digital control law, can minimize the amount of grain improperly processed during startup while also minimizing the energy requirement during this period.

(7) Provided that the grain feedrate is matched to the rotational speed of the heat exchanger, control of wheat temperature after the startup period is easily accomplished.

Recommendations

The following tasks are recommended based on the results and conclusions from this investigation:

(1) To design, construct, and test a new prototype counterflow particle-to-particle heat exchanger which has a lower (machine mass/processed material mass) ratio;

(2) To perform thermal disinfestation tests using the new prototype heat exchanger;

(3) To investigate the use of the new prototype as an energy efficient mechanism for processing grain sorghum for feed purposes;

(4) To investigate other potential heat transfer media;

(5) To further evaluate the adaptive controller to determine whether the controller structure or the adaptive nature of the controller is primarily responsible for the increased performance during the startup period.

BIBLIOGRAPHY

Anex, R. P. and M. Hubbard. 1984. Modelling and adaptive control of a mechanical manipulator. Journal of Dynamic Systems, Measurement, and Control 106(9): 211-217.

Aspegran, O. E. A. 1959. Heat treatment of piece-shaped material. U.S. Patent No. 2,872,386.

Astrom, K. J. and B. Wittenmark. 1973. On self-tuning regulators. Automatica 9:185-199.

Astrom, K. J. and B. Wittenmark. 1980. Self-tuning controllers based on pole-zero placement. Proceedings of the IEE 127D:120-126.

Baker, V. H., D. E. Wiant, and O. Taboada. 1956. Some effects of microwaves on certain insects which infest wheat and flour. Journal of Economic Entomology 49:33-37.

Bateson, R. H. and J. M. Harper. 1973. Apparatus and process for puffing food products. U.S. Patent No. 3,746,546.

Battu, G. S., S. S. Bains, and A. S. Atwal. 1975. The lethal effect of high temperature on the survival of the larvae of Trogoderma granarium. Indian Journal of Ecology 2:98-102.

Bell, C. H., B. D. Hole, and P. H. Evans. 1977. The occurrence of resistance to phosphine in adult and egg stages of strains of Rhyzopertha dominica (F.). Journal of Stored Product Research 13(2):91-94.

Bell, C. H. 1983. Effects of high temperature on larvae of Ephestia elutella in diapause. Journal of Stored Product Research 19(4):153-157.

Bell, K. J. 1983. Process Heat Transfer. Chemical engineering course notes at Oklahoma State University, Stillwater, OK.

Benson, J. O. 1966. Puffed food products. U.S. Patent No. 3,253,533.

Bhatia, S. K. 1978. Measures against the development of resistance to insecticides in insects infesting stored products. Bulletin of Grain Technology 16(2):114-117.

Bond, E. J. and E. Upitis. 1972. Persistance of tolerance to methyl bromide by Sitophilus granarius (L.) after cessation of selection. Journal of Stored Product Research 8(3):221-222.

Champ, B. R. and C. E. Dyte. 1976. Report of the FAO Global Survey of Pesticide Susceptibility of Stored Grain Pests. FAO Plant Production and Protection Service. FAO, Rome, Italy.

Chancellor, W. J. 1968. A simple grain dryer using conduction heat. Transactions of the ASAE 11:857-862.

Chowdhury, J. 1985. Process controllers don "expert" guises. Chemical Engineering 92(12):14-17.

Clarke, D. W. and P. J. Gawthrop. 1975. Self-tuning controller. Proceedings of the IEE 122:929-934.

Clarke, D. W. 1981. Implemetation of self-tuning controllers. p 144-165. In: Self-Tuning and Adaptive Control: Theory and Applications. Peter Peregrinus Ltd, Stevenage, U.K.

Cotton, R. T. 1963. Pests of Stored Grain and Grain Products. Burgess Publishing Company, Minneanapolis, Mn. p 306.

Davidson, R. H. and W. F. Lyon. 1979. Insect Pests of Farms, Garden, and Orchard. John Wiley and Sons, New York. 596 p.

Davidson, S. 1983. Heat as an insecticide. Rural Research #120. Science Communication Unit of CSIRO Bureau of Scientific Studies, Dickson, Australia.

Dean, G. A. 1912. Heat as a means of controlling mill insects. Journal of Economic Entomology 4(2):142-158.

Dermott, T. and D. E. Evans. 1978(a). Elevated temperature disinfestation of wheat. Conference on Agricultural Engineering held Aug 29-31, 1978 in Toowoomba, Australia.

Dermott, T. and D. E. Evans. 1978(b). An evaluation of fluidized bed heating as a means of disinfesting wheat. Journal of Stored Product Research 14(1):1-12.

Downs, H. W., J. D. Kellerby, J. M. Harper, R. D. Haberstorh, and W. P. Marlatt. 1977. Heat transfer by contact between agitated particles. Final Report to National Science Foundation. NSF Grant No. ENG-7418150.

Edwards, C. A. (Ed.) 1973. Environmental Pollution by Pesticides. Plenum Press, New York. 542 p.

Edwards, D. K. 1958. Effects of acclimation and sex on respiration and thermal resistance in Tribolium (Coloeptera and Tenebrionidae). Canadian Journal of Zoology 36:363-382.

Egardt, B. 1980. Unification of some discrete-time adaptive control schemes. IEEE Transactions on Automatic Control 25:693-698.

Ernst, S. H. 1968. Dispersal of three species of grain beetles as a function of thermal acclimation, storage temperature, and larval size. Unpublished Master's thesis. Iowa State University of Science and Technology, Ames, Iowa.

Evans, D. E. 1981(a). The influence of some biological and physical factors in the heat tolerance relationships for Rhyzopertha dominica (F.) and Sitophilus oryzae (L.). Journal of Stored Product Research 17(2):65-72.

Evans, D. E. 1981(b). Thermal acclimation in several species of stored-grain beetles. Australian Journal of Zoology 29(4):483-492.

Evans, D. E. and T. Dermott. 1981. Dosage-mortality relationships for Rhyzopertha dominica (F.) exposed to heat in a fluidized bed. Journal of Stored Product Research 17(2):53-64.

Evans, D. E., G. R. Thorpe and T. Dermott. 1983. The disinfestation of wheat in a continuous-flow fluidized bed. Journal of Stored Products Research 19(3):125-137.

Franklin, G. F. and J. D. Powell. 1980. Digital Control of Dynamic Systems. Addison and Wesley, Reading, Mass. 335 p.

Geissbuhler, H. (Ed.) 1978. World Food Production Environment - Pesticides. International IUPAC Congress of Pesticide Chemistry held July 24-28 in Zurich, Switzerland.

Ghaly, T. F. and P. A. Taylor. 1982. Quality effects of heat treatment of two wheat varieties. Journal of Agricultural Engineering Research 27:227-234.

Ghaly, T. F. and J. W. Van Der Touw. 1982. Heat damage studies in relation to high temperature disinfestation of wheat. Journal of Agricultural Engineering Research 27:329-336.

Gillenwater, H. B., L. M. Redlinger, J. L. Zettler, R. Davis, L. L. McDonald, and J. M. Zehner. 1981. Phosphine fumigation of corn in transit on a bulk-dry cargo ship. Journal of Georgia Entomological Society 16(4):457-462.

Giddings, G. G. and M. A. Welt. 1982. Radiation preservation of food. General Food World 27(1):17-20.

Gillet, J. W. (Ed.) 1970. Biological Impact of Pesticides in the Environment. p 124-150. Symposium held Aug 18-20, 1969 at Oregon State University. Oregon State University Publishing, Corvalis, Ore.

Goodship, G., K. A. Scudmore, and J. Hann. 1982. Disappearance of 1,1,1 Trichloroethane from fumigated cereal grains and other foodstuffs during storage and processing. Pesticide Science 13:139-148.

Goodwin, G. C. and E. K. Teoh. 1983. Adaptive control of a class of linear time-varying systems. IFAC Workshop on Adaptive Systems in Control and Signal Processing held Mar, 1983 in San Francisco, Ca.

Goodwin, G. C. and K. S. Sin. 1984. Adaptive Filtering Prediction and Control. Prentice Hall, Englewood Cliffs, N.J. 540 p.

Grierson, W. 1978. Scared of cancer? Don't blame agriculture. Farm Chemicals 141(9):69-76.

Grossman, E. F. 1931. Heat treatment for controlling insect pests of stored corn. Florida Agricultural Experiment Station Bulletin #239. University of Florida, Gainsville, Fla.

Harrell, R. C. 1984. Adaptive control of the bilinear heat exchange process. Unpublished PhD thesis. Washington State University, Pullman, Wash.

Hayes, W. J. 1982. Pesticides studied in man. Williams and Williams, Baltimore, Maryland. 246 p.

Holt, A. D. 1967. Heating and cooling of solids. Chemical Engineering 74(10):145-166.

Howe, R. W. 1965. A summary of estimates of optimal and minimal conditions for population increase of some stored-product insects. Journal of Stored Product

Research 1:177-184.

Huber, L. J. 1955. Process of preparing a puffed cereal product and the resulting product. U.S. Patent No. 2,701,200.

Johnson, D. W. and L. H. Townsend. 1981. Controlling insects in stored grain. Cooperative Extension Service Bulletin Ent-19, University of Kentucky.

Kalman, R. E. 1958. Design of a self-optimizing control system. Transactions of the American Society of Mechanical Engineers 80:468-478.

Katatkin, A. G., V. M. Lekae, and L. N. Elkin. 1964. Investigation of the operation of screw conveyor heat exchanger reactors. International Chemical Engineering 4(1):85-88.

Kenagen, E. E. and F. W. Fletcher. 1947. Effects of high temperature on several household and storage grain pests. Journal of Economic Entomology 35(6):944.

Khan, A. U., A. Amilhussin, J. R. Arboleda, A. S. Manalo, and W. J. Chancellor. 1974. Accelerated drying of rice using heat conduction media. Transactions of the ASAE 17:949-955.

Kirkpatrick, R. L., J. H. Brower, and E. W. Tilton. 1972. A comparison of microwave and infrared radiation to control rice weevils in wheat. Journal of the Kansas Entomological Society 45(4):434-438.

Kirkpatrick, R. L. and E. W. Tilton. 1972. Infrared radiation to control adult stored-product Coleoptera. Journal of the Georgia Entomological Society 7(1):67-73.

Kirkpatrick, R. L. and E. W. Tilton. 1973. Elevated temperatures to control insect infestations in wheat. Journal of the Georgia Entomological Society 8(4):264-268.

Kumar, A., G. P. Pandey, R. B. Doharey, and B. K. Varma. 1981. A note on the effective dosages of some fumigants against stored grain insect pests in commercial storage facilities. Bulletin of Grain Technology 19(2):137-139.

Landau, I. D. 1979. Adaptive Control: The Model Reference Approach. Marcel Dekker Inc, New York. 406 p.

Lapp, H. M. and L. R. Manchur. 1974 . Drying oilseeds with a solid heat transfer medium. Canadian Agricultural

Engineering 16(2):57-59.

Lapp, H. M., P. S. K. Leung, and J. S. Townsend. 1975. Solid heat transfer mediums for processing grains and oilseeds. Paper No. 75-302-NCR of the ASAE. St. Joseph, Mi.

Lapp, H. M., G. S. Mittal, and J. S. Townsend. 1976(a). Cereal grain drying and processing with solid heat transfer media. Paper No. 76-3524 of the ASAE. St. Joseph, Mi.

Lapp, H. M., G. S. Mittal, and J. S. Townsend. 1976(b). Drying and processing of small grains using solid heat transfer media. Paper No. 76-108 of the Canadian Society of Agricultural Engineering.

Lapp, H. M., G. S. Mittal, and J. S. Townsend. 1977. Drying wheat with heated sand in a continuous flow dryer. Paper No. 77-105 of the Canadian Society of Agricultural Engineering.

Lauhoff Grain Company. 1978. A Guide to Good Manufacturing Practices for the Food Industry. Danville, Ill.

Ljung, L. and I. D. Landau. 1978. Model reference adaptive systems and self-tuning regulators - some connections. Proceedings of the IFAC Congress 3:1973-1975.

McCabe, W. L. and J. C. Smith. 1967. Unit Operations of Chemical Engineering. McGraw Hill Inc, New York. 1007 p.

Mittal, G. S., L. B. Smith, and H. M. Lapp. 1981. Thermal disinfestation of wheat with solid heat transfer medium. Agricultural Mechanization in Asia 12(3):53-56.

Mittal, G. S., H. M. Lapp, and J. S. Townsend. 1983. Drying wheat with solid heat transfer medium. Journal of Scientific Food and Agriculture 34:773-779.

Mohsenin, N. N. 1970. Physical Properties of Plant and Animal Materials. Vol I. Structure, Physical Characteristics, and Mechanical Properties. Gordon and Breach Science Publishing, New York. 734 p.

Moulden, J. 1979. Fumigating a farm to control grain insects. Journal of Agriculture (Australia) 20(3):70-71.

Munro, J. W. 1966. Pests of Stored Products. Hutchinson and Company Ltd, London. 266.

Nakakita, H. and R. G. Winks. 1981. Phosphine resistance in immature stages of a laboratory strain of Tribolium castaneum (Herbst). Journal of Stored Product Research 17(2):43-52.

Nelson, S. O., L. E. Stetson, and J. J. Rhine. 1966. Factors influencing effectiveness of radio-frequency electric fields for stored-grain insect control. Transactions of the ASAE 9:809-815.

Nelson, S. O. and L. F. Charity. 1972. Frequency dependence of energy absorption by insects and grain in electric fields. Transactions of the ASAE 15:1099-1102.

Nelson, S. O. 1973. Insect control studies with microwaves and other radiofrequency energy. Bulletin of the Entomological Society of America 19:157-163.

Nielson, J. M. (Ed.) 1980. Material Safety Data Sheets. MSDS #92, #311, #410. Technology Marketing Operation of General Electric Company, New York.

Nielson, N. E. 1981. Food irradiation - FDA approval will signal a technological revolution. Food Development 15(8):46-53.

Nuttal, R. M. 1970. The effect of acclimation upon survival of Ptinus tectus and Tenebrio molitor when exposed to low temperature. Entomological Experimental Applications 13:217-28.

Otake, T. and S. Tone. 1960. Mechanism of heat transfer in solid particle beds. Chemical Engineering (Japan) 24:156-159.

Paster, N., R. Barkai-Golan, and M. Calderon. 1979. Methyl bromide fumigation for the control of Aspergilli and Penicilli in stored grains. Annals of Applied Biology 92:313-321.

Raghavan, G. S. V., J. M. Harper, and R. D. Haberstroh. 1974. Heat transfer study using granular media. Transactions of the ASAE 17:589-592.

Richard, P. and G. S. V. Raghavan. 1980. Heat transfer between flowing granular materials and immersed objects. Transactions of the ASAE 23:1564-1572.

Rowland, J. R. 1983. Linear Control Systems. Text Manuscript unpublished to date. John Wiley and Sons Inc, New York.

Rovira, A. A., P. W. Murrill, and C. L. Smith. 1969.

Tuning controllers for setpoint changes. Instruments and Control Systems 42(12):67-69.

Schroeder, H. W. and E. W. Tilton. 1961. Infrared radiation for the control of immature insects in kernals of rough rice. USDA AMS-445.

Seborg, D. E., S. L. Shah, and T. F. Edgar. 1983. Adaptive control strategies for process control: a survey. Paper presented at the American Institute of Chemical Engineers Diamond Jubilee Meeting held Nov, 1983 in Washington, D.C.

Shirasu, Y. 1977. Mutagenicity screening on pesticides and modification products: A basis of carcinogenic evaluation. p 267. In: Hiatt, Watson, and Winstein (Eds). Origins of Human Cancer. Cold Springs Harbor Laboratory, Cold Springs, Maryland.

Singh, K. N. and B. P. Srivastava. 1980. Studies on the efficacy and extent of residues of phosphine in stored pulses. Pesticides 14(2):32.

Sittag, M. (Ed). 1981. Handbook of Toxic and Hazardous Chemicals. Noyes Publications, Park Ridge, N.J. 729 p.

Soderman, J. V. (Ed.) 1982. CRC Handbook of Identified Carcinogens and Noncarcinogens, Vol I. CRC Press Inc, Boca Raton, Fla. 302 p.

Sontag, J. M. (Ed.) 1981. Carcinogens in Industry and the Environment. Marcel Dekker Inc, New York. 761 p.

South Dakota State Cooperative Extension Service. 1980. Insect control in farm-stored grain. Bulletin EC-738. Brookings, S.D.

Storey, C. L. and L. E. Bulla, Jr. 1978. Controlling insect pests in stored grain. Farm Chemicals 141(9):86-92.

Sullivan, W. N. and R. H. Sabersky. 1975. Heat transfer to flowing granular media. International Journal of Heat and Mass Transfer 18:97-107.

Tilton, E. W. and H. W. Schroeder. 1963. Some effects of infrared radiation on the mortality of immature insects in kernals of rough rice. Journal of Economic Entomology 56(6):727-730.

Tilton, E. W. and A. K. Burditt. 1983. Insect disinfestation of grain and fruit. In: E. S. Josephson and M S. Peterson (Eds). Preservation of food by ionizing radiation. CRC Press, Boca Raton, Fla.

Torkelson, T. R., H. R. Hoyle, and V. K. Rowe. 1966. Toxilogical Hazards and Properties of Fumigants. Pest Control reprint from July, 1966. p 1-8.

Uhl, V. W. and W. L. Root III. 1967. Heat transfer to granular solids in agitated units. Chemical Engineering Progess 67(7):81-92.

United States Department of Agriculture. 1969. Leaflet No. 553. Market Quality Research Division. U.S. Government Printing Office, Washington, D.C.

Upitis, E., H. A. U. Muonro, and E. J. Bond. 1973. Some aspects of inheritance of tolerance to methyl bromide by Sitophilus granarius (L.). Journal of Stored Product Research 9(1):13-17.

Van Amerongen, J. 1984. Adaptive steering of ships - a model reference approach. Automatica 20(1):3-14.

Vardell, H. H. and E. W. Tilton. 1981. Control of the lesser grain borer and the rice weevil in wheat with a heated fluidized bed. Journal of the Kansas Entomological Society 54(3):481-485.

Vogel, E. F. 1982. Adaptive control of chemical processes with variable deadtime. Unpublished PhD thesis. University of Texas at Austin.

Ware, G. H. 1980. Complete Guide to Pest Control. Thomson Publications, Fresno, Ca. 212 p.

Wycoff, G. H. and R. D. Anderson. 1971. Fumigation or famine. Journal of the American Veterinary Medicine Association 157(11):1828-1834.

Young, P. C. 1969. Applying parameter estimation to dynamic systems - Part I. Control Engineering 16(10): 119-125.

APPENDIX

SOURCE LISTINGS FOR SOFTWARE

CFLOW-1

```
**************************  CFLOW  ****************************
*                                                             *
*       The following program generates a steady-state        *
*    temperature profile of wheat and heat transfer media     *
*    within a discrete counterflow particle-to-particle       *
*    heat exchanger.                                          *
*                                                             *
***************************************************************


program cflow;


const
   eff=0.80;   mass=1.0;   cw=0.365;   delta=30.0;
   twinp=70.0;   tminp=150.0;

type
   cell=array[1..30] of real;

var
   datamout: text;
   datawout: text;
   m, d: char;
   pass, i, n, r, plimit: byte;
   cm, lag, ua, h, khat: real;
   k, tau, c1, c2, mt, wt: real;
   tmin: cell;
   tmout: cell;
   tw: cell;


begin
           {Obtain input data}

write('Enter number of cells:  ');
readln(n);

repeat
   write('Enter salt (s) or prop (p):  ');
   readln(m);
until (m='s') or (m='p');

repeat
   write('Enter lbs of media per lb of wheat:   ');
   readln(r);
until (r=3) or (r=4) or (r=5);

writeln(' '); writeln(' ');
write('Printout desired? ');  readln(d);
```

CFLOW-2

```
if (d<>'y') then
   begin
      assign(datamout,'S3S5.DAT'); assign(datawout,'S3W5.DAT');
      rewrite(datamout);  rewrite(datawout);
   end;

 {Initialize variables}

pass:=1;   plimit:=8*n;

if (m='s') and (r=3) then
   begin cm:=0.20; ua:=0.306; tau:=18.0; end;
if (m='s') and (r=4) then
   begin cm:=0.20; ua:=0.415; tau:=18.0; end;
if (m='s') and (r=5) then
   begin cm:=0.20; ua:=0.547; tau:=18.0; end;
if (m='p')                then
   begin cm:=0.24; ua:=0.332; tau:=13.0; end;

k:=1/tau;  lag:=exp(-k*delta);
h:=-(ua)*(1/(r*mass*cm) + 1/(mass*cw));
khat:=exp(h);
c1:=r*mass*cm*eff*(1-khat)/(r*mass*cm*eff + mass*cw);
c2:=mass*cw*(1-khat)/(r*mass*cm*eff + mass*cw);



     {Initialize wheat & media temperatures}

for i:=1 to n do
   begin
   if i=n then tmin[i]:=tminp else tmin[i]:=twinp;
   tw[i]:=twinp;
   end;

              {Print data header}
if (d='y') then
   begin
   writeln(lst,
 'MEDIA  MASS RATIO   HT PERIOD   TIME CONSTANT   MEDIA TEMP');
   writeln(lst,
        '(m:w)        (sec)        (sec)            (F)    ');
   writeln(lst,' ');

   writeln(lst,
      m:3,r:9,delta:15:1,tau:13:1,tminp:16:1);
   writeln(lst,' '); writeln(lst,' '); writeln(lst,' ');
   end

          else

   begin
   writeln(
     'REVOLUTION   WHEAT OUTLET TEMP   MEDIA OUTLET TEMP');
```

CFLOW-3

```
   writeln(  ');
   end;

          {Perform simulation}

while pass<plimit do
   begin


   for i:=n downto 1 do
      begin
      mt:=tmin[i];
      if i=1 then wt:=twinp else wt:=tw[i-1];
      tw[i]:=wt + c1*(mt-wt)*(1-lag);
      tmout[i]:=mt - c2*(mt-wt)*(1-lag);
      end;

   for i:=1 to n do
      if i=n then tmin[i]:=tminp else tmin[i]:=tmout[i+1];

   pass:=pass+1;
   if (d<>'y') then writeln(pass:7,tw[n]:17:1,tmout[1]:20:1);

   end;


                {Print final states}

if (d='y') then
   begin
   writeln(lst,
'     TEMPERATURE PROFILE IN CELLS AFTER REVOLUTIONS= ',PASS:3);
   writeln(lst,' ');
   writeln(lst,
'         CELL        WHEAT TEMP        MEDIA TEMP');
   writeln(lst,
'       -------------------------------------------');
   writeln(lst,' ');
   for i:=1 to n do
      writeln(lst,i:11,tw[i]:17:1,tmout[i]:17:1);
   end

           else

   begin
   for i:=1 to n do
      begin
      writeln(datamout,i:3,tmout[i]:7:2);
      writeln(datawout,i:3,tw[i]:7:2);
      end;
   close(datamout);  close(datawout);
   end;
end.
```

SUPER-1

```
REM        THIS PROGRAM PROVIDES SUPERVISORY MANUAL OPERATION OF
REM        THE PARTICLE-TO-PARTICLE COUNTERFLOW HEAT EXCHANGE
REM        MACHINE.
REM
REM     SET BASE ADDRESSES & CONSTANTS:
REM
        PIA$="FFCC" : PIA=HEX(PIA$) : ADB$="EC40" : ADB=HEX(ADB$)
        AHT$="EC28" : AHT=HEX(AHT$) : FDT$="EC20" : FDT=HEX(FDT$)
        MHT$="EC30" : MHT=HEX(MHT$) : PHT$="EC38" : PHT=HEX(PHT$)
        SDT$="EC18" : SDT=HEX(SDT$) : EMT$="EC00" : EMT=HEX(EMT$)
        HCT$="EC08" : HCT=HEX(HCT$)
REM
        MAINHI=HEX("A100")  :  MAINLO=HEX("A102")
         PREHI=HEX("A104")  :   PRELO=HEX("A106")
         AUXHI=HEX("A108")  :   AUXLO=HEX("A10A")
        INTVECT=HEX("E713")
REM
        POKE MAINHI,HEX("03")  :  POKE (MAINHI+1),HEX("E8")
        POKE MAINLO,HEX("00")  :  POKE (MAINLO+1),HEX("00")
        POKE  PREHI,HEX("03")  :  POKE  (PREHI+1),HEX("E8")
        POKE  PRELO,HEX("00")  :  POKE  (PRELO+1),HEX("00")
        POKE  AUXHI,HEX("01")  :  POKE  (AUXHI+1),HEX("F4")
        POKE  AUXLO,HEX("01")  :  POKE  (AUXLO+1),HEX("F4")
REM
        MSB=HEX("CC2B") : MEME=256*PEEK(MSB) + PEEK(MSB+1)
REM
        MSBSUB1$="A0" : MSBSUB1=HEX(MSBSUB1$)
        LSBSUB1$="00" : LSBSUB1=HEX(LSBSUB1$)
        MSBSUB2$="A0" : MSBSUB2=HEX(MSBSUB2$)
        LSBSUB2$="50" : LSBSUB2=HEX(LSBSUB2$)
REM
        POKE (PIA+1),04              :        POKE (PIA+3),04
        POKE (PIA),HEX("FF")         :        POKE (PIA+2),HEX("FF")
        POKE (PIA+1),00              :        POKE (PIA+3),00
        POKE (PIA),HEX("0F")         :        POKE (PIA+2),HEX("FF")
        POKE (PIA+1),04              :        POKE (PIA+3),04
REM
        POKE (INTVECT),HEX("A2")
        POKE (INTVECT+1),00
        POKE (HCT+1),HEX("A1")
        POKE (HCT+2),HEX("FF")
        POKE (HCT+3),HEX("FF")
        POKE (HCT+4),HEX("01")
        POKE (HCT+5),HEX("B3")
        HTISON=0
REM
        ON ERROR GOTO FIXIT
REM
REM     PRINT MENU & GET INPUT:
REM
MENU    FOR I=1 TO 10 : PRINT : NEXT I
        PRINT "    ************ FUNCTIONS AVAILABLE *********** "
        PRINT
        PRINT "            (00) SHUT DOWN ENTIRE MACHINE" : PRINT
        PRINT "(01) TURN OFF MAIN DRIVE      (02) TURN ON MAIN DRIVE"
        PRINT "(03) TURN OFF FEEDER DRIVE    (04) TURN ON FEEDER DRIVE"
        PRINT "(05) TURN OFF MAIN HEATER     (06) TURN ON MAIN HEATER"
```

SUPER-2

```
        PRINT "(07) TURN OFF PREHEATER       (08) TURN ON PREHEATER"
        PRINT "(09) TURN OFF AUX HEATER      (10) TURN ON AUX HEATER"
        PRINT "(11) TURN OFF SALT AUGER      (12) TURN ON SALT AUGER"
        PRINT
        PRINT "              (13) ADJUST MAIN DRIVE SPEED"
        PRINT "              (14) ADJUST FEEDER DRIVE SPEED"
        PRINT "              (15) ADJUST MAIN HEATER POWER"
        PRINT "              (16) ADJUST PREHEATER POWER"
        PRINT "              (17) ADJUST AUXILIARY HEATER POWER"
        PRINT "              (18) RETURN TO BASIC"
        PRINT
        PRINT "              (19) DISPLAY ENDS' TEMPS & ODD WHEAT TEMPS"
        PRINT "              (20) DISPLAY EVEN WHEAT TEMPS"
        PRINT "              (21) INITIALIZE ENERGY MEASUREMENT CIRCUIT"
        PRINT "              (22) DISPLAY CURRENT ENERGY MEASUREMENT"
        PRINT
        INPUT "ENTER DESIRED FUNCTION:";TASK
        IF (TASK<0 OR TASK>22)  GOTO MENU
        IF TASK=0 THEN GOSUB SHUT : GOTO MENU
        IF TASK=1 THEN GOSUB MDOF : GOTO MENU
        IF TASK=3 THEN GOSUB FDOF : GOTO MENU
        IF TASK=5 THEN GOSUB MHOF : GOTO MENU
        IF TASK=7 THEN GOSUB PHOF : GOTO MENU
        IF TASK=9 THEN GOSUB AHOF : GOTO MENU
        IF TASK=2 THEN GOSUB MDON : GOTO MENU
        IF TASK=4 THEN GOSUB FDON : GOTO MENU
        IF TASK=6 THEN GOSUB MHON : GOTO MENU
        IF TASK=8 THEN GOSUB PHON : GOTO MENU
        IF TASK=10 THEN GOSUB AHON : GOTO MENU
        IF TASK=13 THEN GOSUB MDCH : GOTO MENU
        IF TASK=14 THEN GOSUB FDCH : GOTO MENU
        IF TASK=15 THEN GOSUB MHCH : GOTO MENU
        IF TASK=16 THEN GOSUB PHCH : GOTO MENU
        IF TASK=17 THEN GOSUB AHCH : GOTO MENU
        IF TASK=18 THEN GOTO  GOOS
        IF TASK=11 THEN GOSUB SDOF : GOTO MENU
        IF TASK=12 THEN GOSUB SDON : GOTO MENU
        IF TASK=19 THEN GOSUB DISO : GOTO MENU
        IF TASK=20 THEN GOSUB DISE : GOTO MENU
        IF TASK=21 THEN GOSUB ENIT : GOTO MENU
        IF TASK=22 THEN GOSUB ENRD : GOTO MENU
FIXIT   PRINT : PRINT "Enter a number dummy ....."
        FOR I=1 TO 5000 : NEXT I
        RESUME MENU
        END
REM
REM     SHUT DOWN MACHINE:
REM
SHUT    POKE FDT,01 : POKE PIA,04 : POKE MHT,01 : POKE PHT,01
        POKE (PIA+2),127 : POKE AHT,01 : POKE SDT,01
        POKE (PIA+1),04  : POKE (PIA+3),04  :  POKE HCT,01
        HTISON=0
        POKE (HCT+4),01 : POKE (HCT+5),HEX("B3")
        RETURN : END
REM
REM     TURN OFF MAIN DRIVE:
REM
```

SUPER-3

```
MDOF    POKE (ADB+10),01 : POKE (ADB+11),01
CONA    EOC=PEEK(ADB+12)
        IF EOC=0 THEN GOTO CONA
        MTACH=256*PEEK(ADB+13) + PEEK(ADB+14)
        IF MTACH)400 THEN POKE (PIA+2),127 \
           ELSE PRINT "MAIN DRIVE IS NOT PRESENTLY RUNNING" : \
                FOR I=1 TO 5000 : NEXT I
        RETURN : END
 REM
 REM    TURN OFF FEEDER DRIVE:
 REM
FDOF    POKE (ADB+10),02 : POKE (ADB+11),01
CONB    EOC=PEEK(ADB+12)
        IF EOC=0 GOTO CONB
        FTACH=256*PEEK(ADB+13) + PEEK(ADB+14)
        IF FTACH)100 THEN POKE FDT,01 \
           ELSE PRINT "FEEDER DRIVE IS NOT PRESENTLY RUNNING" : \
                FOR I=1 TO 5000 : NEXT I
        RETURN : END
 REM
 REM    TURN OFF MAIN HEATERS:
 REM
MHOF    POKE MHT,01
        HTISON=HTISON-1
        IF HTISON=0 THEN                                                \
           POKE (PIA+1),04 : POKE (PIA+3),04 : POKE HCT,01
        RETURN : END
 REM
 REM    TURN OFF PREHEATERS:
 REM
PHOF    POKE PHT,01
        HTISON=HTISON-1
        IF HTISON=0 THEN                                                \
           POKE (PIA+1),04 : POKE (PIA+3),04 : POKE HCT,01
        RETURN : END
 REM
 REM    TURN OFF AUX HEATERS:
 REM
AHOF    POKE AHT,01
        HTISON=HTISON-1
        IF HTISON=0 THEN                                                \
           POKE (PIA+1),04 : POKE (PIA+3),04 : POKE HCT,01
        RETURN : END
 REM
 REM    TURN OFF SALT AUGER
 REM
SDOF    POKE SDT,01
        RETURN : END
 REM
 REM    TURN ON MAIN DRIVE:
 REM
MDON    POKE (ADB+10),01 : POKE (ADB+11),01
CONC    EOC=PEEK(ADB+12)
        IF EOC=0  GOTO CONC
        MDSP=1.0
        MTACH=256*PEEK(ADB+13) + PEEK(ADB+14)
        IF MTACH(400 THEN POKE (PIA+2),104                      : \
```

SUPER-4

```
                             POKE (PIA),00                        \
                        ELSE PRINT "Drive already running"      : \
                             FOR I=1 TO 5000 : NEXT I
        RETURN : END
 REM
 REM    TURN ON FEEDER DRIVE:
 REM
FDON    POKE (ADB+10),02 : POKE (ADB+11),01
COND    EOC=PEEK(ADB+12)
        IF EOC=0 GOTO COND
        FTACH=256*PEEK(ADB+13) + PEEK(ADB+14)
        IF FTACH(100 THEN                                         \
                    POKE (MEME-2),MSBSUB1                       : \
                    POKE (MEME-1),LSBSUB1                       : \
                    POKE PIA,00                                 : \
                    DUMMY=USR(1)                                : \
                    POKE (FDT+4),HEX("17")                      : \
                    POKE (FDT+5),HEX("80")
        IF FTACH)=100 THEN PRINT "Feeder already running"       : \
                              FOR I=1 TO 5000 : NEXT I
        RETURN : END
 REM
 REM    TURN ON SALT AUGER:
 REM
SDON    PRINT : INPUT "ENTER FEEDRATE (1.00-4.25 #/MIN): ";RATE
        IF (RATE(1.0 OR RATE)4.25) GOTO SDON
        SASP=0.151 + 3.346*RATE
        DUTY=1/(0.04718 - 0.00182*SASP)
        YAL=(DUTY/100)*20000 : MS=INT(YAL/256) : LS=YAL - MS*256
           POKE (MEME-2),MSBSUB1 : POKE(MEME-1),LSBSUB1
           POKE PIA,00           : DUMMY=USR(0)
        POKE (SDT+4),MS : POKE (SDT+5),LS
        RETURN : END
 REM
 REM    TURN ON MAIN HEATERS:
 REM
MHON    POKE (MEME-2),MSBSUB2 : POKE (MEME-1),LSBSUB2 : DUMMY=USR(1)
        IF HTISON=0 THEN                                                \
           DUMMY=PEEK(PIA) : DUMMY=PEEK(PIA+2)                         :\
           POKE (PIA+1),07 : POKE (PIA+3),07
        HTISON=HTISON+1
        RETURN : END
 REM
 REM    TURN ON PREHEATERS:
 REM
PHON    POKE (MEME-2),MSBSUB2 : POKE (MEME-1),LSBSUB2 : DUMMY=USR(2)
        IF HTISON=0 THEN                                                \
           DUMMY=PEEK(PIA) : DUMMY=PEEK(PIA+2)                         :\
           POKE (PIA+1),07 : POKE (PIA+3),07
        HTISON=HTISON+1
        RETURN : END
 REM
 REM    TURN ON AUXILIARY HEATER:
 REM
AHON    POKE (MEME-2),MSBSUB2 : POKE (MEME-1),LSBSUB2 : DUMMY=USR(0)
        IF HTISON=0 THEN                                                \
           DUMMY=PEEK(PIA) : DUMMY=PEEK(PIA+2)                         :\
```

SUPER-5

```
           POKE (PIA+1),07 : POKE (PIA+3),07
        HTISON=HTISON+1
        RETURN : END
 REM
 REM    PRINT CT TEMPS & ODD SCREEN TEMPS
 REM
DISO    FOR I=1 TO 15 : PRINT : NEXT I
        OPEN "0.PRINT.SYS" AS 0
        STORE=(HEX("FC") AND PEEK(PIA)) :  POKE PIA,STORE
        K=1 : TACC=0
        POKE (ADB+10),0
        FOR I=1 TO 9
           POKE PIA,(STORE+1) : POKE (ADB+11),01
CON4       EOC=PEEK(ADB+12)
           IF EOC=0 GOTO CON4
           SIGNAL=(5/4095)*(256*PEEK(ADB+13) + PEEK(ADB+14))
           IF (I=1 OR I=2) THEN                                    \
              TEMP=232.34*SIGNAL + 45.36                          :\
              TACC=TACC+TEMP                                      :\
              PRINT "SALT  TEMP IN HEATED END: ";TEMP             :\
              PRINT #0,"SALT  TEMP IN HEATED END: ";TEMP
           IF I=2 THEN                                             \
                 TAVE=TACC/2                                      :\
                 PRINT "                   AVERAGE: ";TAVE        :\
                 PRINT #0,"                   AVERAGE: ";TAVE     :\
                 TACC=0
           IF (I=3 OR I=5) THEN PRINT : PRINT #0
           IF (I=3 OR I=4) THEN                                    \
              TEMP=(SIGNAL - 0.041122)/0.0029994                  :\
              TACC=TACC+TEMP                                      :\
              PRINT "WHEAT TEMP IN HEATED END: ";TEMP             :\
              PRINT #0,"WHEAT TEMP IN HEATED END: ";TEMP
              IF I=4 THEN                                          \
                 TAVE=TACC/2                                      :\
                 PRINT "                   AVERAGE: ";TAVE        :\
                 PRINT #0,"                  AVERAGE: ";TAVE      :\
                 TACC=0
           IF (I>4) THEN                                           \
              TEMP=(SIGNAL - 0.041122)/0.0029994                  :\
              PRINT "WHEAT TEMP ON SCREEN #";K;TEMP               :\
              PRINT #0,"WHEAT TEMP ON SCREEN #";K;TEMP            :\
              K=K+2
           POKE PIA,(STORE+3)
        NEXT I
 REM
        FOR I=1 TO 5
           POKE PIA,(STORE+1)
           POKE PIA,(STORE+3)
        NEXT I
           PRINT : PRINT #0
        FOR I=1 TO 2
           POKE PIA,(STORE+1) : POKE (ADB+11),01
CON6       EOC=PEEK(ADB+12)
           IF EOC=0 THEN CON6
           SIGNAL=(5/4095)*(256*PEEK(ADB+13) + PEEK(ADB+14))
           TEMP=(SIGNAL - 0.041122)/0.0029994
           TACC=TACC+TEMP
```

SUPER-6

```
           PRINT "SALT TEMP IN INLET END:    ";TEMP
           PRINT #0,"SALT TEMP IN INLET END:    ";TEMP
           IF I=2 THEN                                                 \
              TAVE=TACC/2                                             :\
              PRINT "                AVERAGE:    ";TAVE               :\
              PRINT #0,"                AVERAGE:    ";TAVE
           POKE PIA,(STORE+3)
        NEXT I
        PRINT : PRINT : PRINT #0 : PRINT #0 : CLOSE 0
        INPUT "ENTER 'M' TO GO BACK TO MENU:";DUM$
        IF (DUM$<>'M' AND DUM$<>'m') GOTO DISO
        RETURN : END
 REM
 REM    PRINT WHEAT TEMPS ON EVEN NUMBERED SCREENS
 REM
DISE    FOR I=1 TO 20 : PRINT : NEXT I
        OPEN "0.PRINT.SYS" AS 0
        STORE=(HEX("FC") AND PEEK(PIA)) :  POKE PIA,STORE
        K=10
        POKE (ADB+10),0
        FOR I=1 TO 9
           POKE PIA,(STORE+1)
           POKE PIA,(STORE+3)
        NEXT I
        FOR I=1 TO 5
           POKE PIA,(STORE+1) : POKE (ADB+11),01
CON5       EOC=PEEK(ADB+12)
           IF EOC=0 GOTO CON5
           SIGNAL=(5/4095)*(256*PEEK(ADB+13) + PEEK(ADB+14))
           TEMP=(SIGNAL - 0.041122)/0.0029994
           PRINT "WHEAT TEMP ON SCREEN";K;TEMP
           PRINT #0,"WHEAT TEMP ON SCREEN";K;TEMP
           K=K-2
           POKE PIA,(STORE+3)
        NEXT I
        PRINT : PRINT
        PRINT #0,"-------------------------------------"
        PRINT #0 : PRINT #0 : CLOSE 0
        INPUT "ENTER 'M' TO GO BACK TO MENU:";DUM$
        IF (DUM$<>'M' AND DUM$<>'m') THEN DISE
        RETURN : END
 REM
 REM    INITIALIZE ENERGY MEASUREMENT CIRCUITRY:
 REM
ENIT    PRINT : INPUT "ENTER '0' TO HALT OR '1' TO INITIALIZE: ";ENX
        IF (ENX<>0 AND ENX<>1) THEN ENIT
        IF ENX=0 THEN                                                  \
              POKE EMT,01                                             :\
              ENVAL=256*PEEK(EMT+6) + PEEK(EMT+7)                     :\
              PRINT "FINAL ENERGY VALUE = ";ENVAL                     :\
              INPUT "ENTER garbage TO RETURN TO MENU: ";DUM$
        IF ENX=1 THEN                                                  \
              POKE (EMT+1),01                                         :\
              POKE (EMT+2),HEX("FF") : POKE (EMT+3),HEX("FF")         :\
              POKE (EMT),00
        RETURN : END
 REM
```

SUPER-7

```
 REM   DISPLAY CURRENT ENERGY MEASUREMENT:
 REM
ENRD   ENVAL=256*PEEK(EMT+2) + PEEK(EMT+3)
       PRINT "CURRENT ENERGY VALUE = ";ENVAL
       INPUT "ENTER 'M' TO RETURN TO MENU: ";DUM$
       IF (DUM$()'M' AND DUM$()'m') THEN ENRD
       RETURN : END
 REM
 REM   CHANGE MAIN DRIVE SPEED {AND ADJUST HEAT CYCLES}:
 REM
MDCH   PRINT "AVAILABLE SPEED RANGE= 0.25 TO 3.00 RPM"
       INPUT "ENTER DESIRED SPEED : ";MDSP
       IF (MDSP(0.25 OR MDSP)3.00) GOTO MDCH
          IF HTISON()0 THEN    POKE HCT,01                                 :\
             POKE (MHT+4),HEX("01") : POKE (MHT+5),HEX("90")               :\
             POKE (PHT+4),HEX("01") : POKE (PHT+5),HEX("90")
          HCYCLE=(0.94*15.259)/(MDSP/30.0)
          MS=INT(HCYCLE/256)   :   LS=HCYCLE - MS*256
          POKE HCT+4,MS        :   POKE HCT+5,LS
       X=2.625 + 14.893*MDSP
       YAL=127*(1 - (X/100)) : POKE (PIA+2),YAL
       RETURN : END
 REM
 REM   CHANGE FEEDER DRIVE SPEED:
 REM
FDCH   INPUT "ENTER FEEDRATE (0.10-2.25 #/MIN): ";RATE
       IF (RATE(0.1 OR RATE)2.25) GOTO FDCH
       MASP=0.506 + 10.01*RATE
       DUTY=0.9/(0.04718 - 0.00182*MASP)
       YAL=(DUTY/100)*20000 : MS=INT(YAL/256) : LS=YAL - MS*256
       POKE (FDT+4),MS : POKE (FDT+5),LS
       RETURN : END
 REM
 REM   CHANGE MAIN HEATER POWER:
 REM
MHCH   GOSUB POWR : POKE (MAINHI),MS : POKE (MAINHI+1),LS
       RETURN : END
 REM
 REM   CHANGE PREHEATER POWER:
 REM
PHCH   GOSUB POWR : POKE (PREHI),MS : POKE (PREHI+1),LS
       RETURN : END
 REM
 REM   CHANGE AUXILIARY HEATER POWER:
 REM
AHCH   GOSUB POWR : POKE (AUXHI),MS : POKE (AUXHI+1),LS
       RETURN : END
 REM
 REM   DETERMINE POWER CYCLE CHANGE BINIS:
 REM
POWR   PRINT : PRINT "AVAILABLE POWER CYCLE RANGE = 0 TO 100%"
       INPUT "ENTER DESIRED POWER CYCLE (IN PERCENT):";X
       IF (X(0 OR X)100) GOTO POWR
       YAL=(X/100)*2000 : MS=INT(YAL/256) : LS=YAL - MS*256
       RETURN : END
 REM
 REM   RETURN TO BASIC:
```

ADAPT-1

```
{************************ ADAPT **************************
 *   The following program provides adaptive control      *
 *   using second-order (numerator and denominator)       *
 *   discrete recursive parameter estimation and a        *
 *   second-order pole placement algorithm for the        *
 *   particle-to-particle heat exchange system.  The      *
 *   setpoints for main drive speed and temperatures      *
 *   are user inputs.  The feedrate setpoint is cascaded  *
 *   from the main drive speed such that 1 lb/cell is     *
 *   insured at all times.  Update: 11-18-85.             *
 *                                                        *
 **********************************************************
```

```
                          PROCEDURE LISTING


menu --------------- Prints available functions to screen.
timeupdate --------- Reads time marker in RAM; marker is refreshed once
                     per revolution by the interrupt routine scanning
                     the real-time clock.
set_print ---------- Sets the FLEX output default to the printer.
set_term ----------- Sets the FLEX output default to the terminal.
shut_down ---------- Performs immediate/uncontrolled shutdown of system.
"xx"off ------------ Controlled shutdown of device "xx": "md" = main drive;
                     "fd" = feeder drive; "sd" = make-up media drive; "mh"
                     = main heaters; "ph" = preheaters.
"xx"onn ------------ Controlled start-up of device "xx".
display ------------ Displays all temperature and energy measurements.
oddplus_temp ------- Prints all odd-numbered cell (wheat) temperatures
                     and temperatures measured in the heated and inlet
                     end sections.
even_temp ---------- Prints all even-numbered cell (wheat) temperatures.
mdset -------------- Sets main drive speed setpoint, adjusts the heater
                     "half-cycle" duration accordingly, and adjusts the
                     feeder drive setpoint.
Twmaxset ----------- Sets wheat temperature setpoint.
parinit ------------ Initializes the parameter estimation algorithm
                     variables and the controller variables.
Pupdate ------------ Recursively updates elements of the parameter
                     estimation gain (covariance) matrix.
Preset ------------- Resets the parameter estimation gain matrix to an
                     integer multiple of the identity matrix.


                          CONSTANT LISTING

"xx"tim ------------ Starting address of a MC6840 timer used for I/O; "em"
                     = energy measurment; "uu" = unused; "ah" = auxiliary
                     heater (also unused); others as shown above.
pia ---------------- Starting address of MC6821 PIA for digital I/O and
                     interrupts.
adb ---------------- Starting address of RTI analog-to-digital board.
kbstatus ----------- Starting address of MC6850 ACIA for terminal I/O.
T ------------------ approximate sampling time for drive loops.
"xx"kc ------------- PID algorithm proportional gain.
"xx"taui ----------- PID algorithm integral gain.
```

ADAPT-2

```
"yyyy"hi ----------- Shared memory locations between interrupt routine and
                     main program; serves as paramter passing of heater
                     control requirements;"hi" indicates high cycle output.
"yyyy"lo ----------- As above, where "lo" indicates low cycle output; in
                     majority of tests, yyyylo is zero for energy
                     efficiency reasons.
intvector ---------- FLEX interrupt vector address.
enval1,2 ----------- Shared memory between interrupt routine and main
                     program; contains current and previous reading of the
                     energy measurement counter.
intflag ------------ Shared memory also; serves as an indicator of what
                     caused the previous interrupt (i.e. limit switch or
                     timer); cleared by main program.
outswitch ---------- FLEX output switch for terminal/printer I/O.
outvector ---------- FLEX "second" serial I/O port address; in this case,
                     ACIA for printer.


                          VARIABLE LISTING

"xx"stpt ----------- Setpoint, in English engineering units, for device
                     "xx"; units normalized to align with device feedback.
"xx"otpt ----------- Output, generally of type integer, required to control
                     device "xx".
"xx"tach ----------- Tachometer feedback on device "xx".
"xx"err,o ---------- Current, or previous (o=old), error.
"xx"filt ----------- Filtered version of the feedback variable.
Iae ---------------- Performance measure #1: Approximation of the integral
                     of the absolute error (Euler-type integration).
Iein --------------- Performance measure #2: Approximation of the integral
                     of the input energy (Euler-type integration).
enval -------------- Energy measurement (Btu/min).
temp --------------- Temporary (dummy) variable.
beta --------------- Discrete-time pole of PMI motor tachometer digital
                     filter.
feed --------------- Wheat feedrate setpoint (lb/min).
salt --------------- Make-up media setpoint (lb/min).
delta -------------- Time required per revolution (min).
alpha -------------- Forgetting factor used in parameter estimation.
denp --------------- Temporary storage used in parameter estimation.
gain --------------- Final value (per revolution) of gain of the parameter
                     estimation calculation.
eoff --------------- Error between predicted and filtered version of actual
                     wheat temperature in heated end section.
gamma -------------- Discrete-time pole of digital temperature filter.
Twact -------------- Wheat temperature measured in heated end section.
Twest -------------- Wheat temperature predicted by estimator.
Twfilt ------------- Filtered Twact.
traceP ------------- Trace of the matrix P.
Progain ------------ Gain of the estimated process transfer function.
sum ---------------- Summation of absolute value of "eoff" for previous
                     six iterations; an indicator of the performance of
                     the estimator.
poorjob ------------ Desired limit on the value of "sum".
pole1,2 ------------ Poles of desired closed-loop transfer function.
K ------------------ Controller gain acting on measured error history.
Kmin --------------- Minimum clamp on the above gain K.
```

ADAPT-3

```
c1,2 --------------- Gains in controller; equivalent to coefficients
                     predicted by the estimator during the last adaptive
                     update of controller gains.
eoc ---------------- End-of-conversion flag for a/d request.
i,j,store ---------- Temporary integer variables.
hour,min,sec ------- Time stamps; used to mark each revolution when the
                     heaters are on.
test --------------- Keyboard status word for terminal ACIA.
hcycle ------------- Heater control cycle (period) of approximately
                     one-half of one revolution.
pass --------------- Marker for heater timer interrupt.
limswitch ---------- Marker for limit switch interrupt.
punch -------------- Current byte stored in ACIA for terminal I/O.
n ------------------ Dimension of parameter estimation algorithm.


  The following arrays concern only the adaptive calculations required.

P ------------------ Parameter estimation gain (covariance) matrix.
PHI ---------------- Vector of input/output history used in estimator.
EST ---------------- Vector of estimated coefficients of the difference
                     equation model of the process.
NUM ---------------- Intermediate calculation vector.
CONTROL ------------ Vector of clamped control history (0-100%).
UCON --------------- Vector of actual control history (0-250%).
GAIN --------------- Adaptive controller gain vector.
MEAERR ------------- Vector of measured error history.
PARERR ------------- Vector of the absolute value of predicted error
                     history.
TW ----------------- Vector of temperature measurement history of the wheat
                     in the heated end section.}



const
                                                      {Hex locations for I/O}
   pia=$FFCC;   adb=$EC40;   kbstatus=$FFD5;
   emtim=$EC00; hctim=$EC08; uutim=$EC10; sdtim=$EC18;
   fdtim=$EC20; ahtim=$EC28; mhtim=$EC30; phtim=$EC38;
                                                        {PI gain constants}
   T=0.075;     kf=8.33;     sdkc=10.0;  sdtaui=3.0;
   mdkc=0.012;  mdtaui=1.0;  fdkc=5.0;   fdtaui=5.0;
   mhkc=60.0;   mhtaui=4.6;
                                              {Shared memory and FLEX vectors}

   mainhi=$E400;     mainlo=$E402;       outvector=$CD10;
   prehi=$E404;      prelo=$E406;        outswitch=$CC22;
   auxhi=$E408;      auxlo=$E40A;        printer=$CCE4;
   loctim=$E410;     intvector=$E713;    goprint=0;
   enval1=$E420;     enval2=$E422;       goterm=1;
   intflag=$E425;

type

   parmatrix = array[1..4, 1..4] of real;
   parvector = array[1..4] of real;
   meavector = array[1..2] of real;
```

ADAPT-4

```
   chkvector = array[1..6] of real;


var

   mdstpt,mderr,mderro,u:                                  real;
   fdstpt,fderr,fderro,fdfilt:                             real;
   sdstpt,sderr,sderro,sdfilt:                             real;
   mherr,mherro,Iae,Iein,enval:                            real;
   temp,beta,feed,salt,delta:                              real;
   alpha,denp,gain,eoff,gamma:                             real;
   Twmax,Twact,Twest,Twfilt:                               real;
   traceP,Progain,sum,poorjob:                             real;
   pole1,pole2,K,c1,c2,Kmin:                               real;

   P:                                                      parmatrix;
   PHI,EST,NUM,CONTROL:                                    parvector;
   GAIN,MEAERR,UCON:                                       parvector;
   PARERR:                                                 chkvector;
   TW:                                                     meavector;

   mdotpt,mdout,mdtach:                                    integer;
   fdotpt,fdtach,sdotpt,sdtach:                            integer;
   eoc,i,j,test,muxsig,store:                              integer;
   hour,min,sec,hcycle,mhout:                              integer;
   pass,limswitch,punch,n:                                 integer;

   mdon,fdon,tb,mhon,phon,paron:                           boolean;
   par,htison,scan,sdon:                                   boolean;

procedure menu;
   begin

   for i:=0 to 7 do  writeln;

   writeln('   *********** FUNCTIONS AVAILABLE ***********');
   writeln; writeln;
   writeln('             (0) Shut down entire machine '); writeln;
   writeln('(m) Turn on main drive        (n) Turn off main drive');
   writeln('(f) Turn on feeder            (g) Turn off feeder');
   writeln('(h) Turn on main heater       (i) Turn off main heater');
   writeln('(p) Turn on pre-heater        (q) Turn off pre-heater');
   writeln('(s) Turn on salt auger        (t) Turn off salt auger');
   writeln;
   writeln('(1) Adjust main drive setpt   (3) Adjust max wheat temp');
   writeln('(2) Adaptive controller: on   (4) Adaptive controller: off');
   writeln;
   writeln('          (d) Display temperatures and energy meas');
   writeln('          (o) Print odd/pl screen temperatures');
   writeln('          (e) Print  even  screen temperatures');
   writeln('          (c) Continuous temperature record');
   writeln('          (x) Return to FLEX          '); writeln;
   write('Enter desired function:  ');
   end;

procedure timeupdate(var hour,min,sec: integer);
   begin
```

ADAPT-5

```
   hour:=10*(PEEK(loctim+7)) + PEEK(loctim+8);
   min:=10*(PEEK(loctim+9)) + PEEK(loctim+10);
   sec:=10*(PEEK(loctim+11)) + PEEK(loctim+12);
   end;

procedure set_print;
   begin
   POKE(outswitch,goprint);
   end;

procedure set_term;
   begin
   POKE(outswitch,goterm);
   end;

procedure shut_down(var mdon,fdon,mhon,phon,sdon,htison: boolean;
                    var mdotpt,mhout:                   integer;
                    var Iae,Iein,mherr:                 real);
   begin
   POKE(fdtim,01); POKE(pia,04);   POKE(pia+2,$7F);
   POKE(mhtim,01); POKE(phtim,01); POKE(sdtim,01);
   POKE(pia+1,04); POKE(pia+3,04); POKE(hctim,01);
   POKE(emtim,01); mdotpt:=0;  mherr:=0.0;  mhout:=0;
   mdon:=false; fdon:=false; mhon:=false;   par:=false;
   sdon:=false; phon:=false; htison:=false;
   POKEW(enval1,00); POKEW(enval2,00);
   Iae:=0.0;  Iein:=0.0;
   end;


procedure mdoff(var  mdon: boolean;var mdotpt: integer);
   begin
   POKE(pia+2,$7F);  mdon:=false;
   mdotpt:=0;
   end;

procedure fdoff(var  fdon: boolean;var fdotpt: integer);
   begin
   POKE(fdtim,01);  fdon:=false;
   fdotpt:=0;
   end;

procedure sdoff(var  sdon: boolean;var sdotpt: integer);
   begin
   POKE(sdtim,01);  sdon:=false;
   sdotpt:=0;
   end;

procedure mhoff(var  mhout,hour,min,sec:          integer;
               var  mhon,phon,htison:            boolean;
               var  mherr,Iae,Iein,mdstpt:       real);
   var
      i,store:                                   integer;
      mdspeed,enval:                             real;
   begin
   POKE(mhtim,01);  mhon:=false;
   if phon then htison:=true else htison:=false;
```

ADAPT-6

```
    if not(htison) then
       begin

       timeupdate(hour,min,sec);
       set_print;
       writeln;   writeln;
       writeln('     ********** ',hour:2,':',min:2,':',sec:2,' **********');
       writeln;
       writeln('   I(ae)= ',Iae:10:1,'   I(ein)= ',Iein:10:1); writeln;
       store:=ABS(PEEKW(enval1) - PEEKW(enval2));
       mdspeed:=0.0011646*mdstpt;
       enval:=(478.0*(mdspeed*CONV(store) - 1.5))/1055.0;
       if enval<0.0 then enval:=0.0;
       writeln('Speed= ',mdspeed:5:2,' rpm',' Energy input rate= ',
               enval:5:2,' btu/min');
       for i:=1 to 5 do writeln;
       set_term;

       POKE(pia+1,04);
       POKE(pia+3,04);
       POKE(hctim,01);
       POKE(emtim,01);
       POKEW(enval1,00);
       POKEW(enval2,00);
       mhout:=0; mherr:=0.0;
       Iae:=0.0; Iein:=0.0;
       POKEW(mainhi,0); POKEW(prehi,0);
       end;
   end;

procedure phoff(var  mhout: integer;var  phon,mhon,htison: boolean;
                var  mherr: real);
   begin
   POKE(phtim,01);  phon:=false;
   if mhon then htison:=true else htison:=false;
   if not(htison) then
      begin
      POKE(pia+1,04);
      POKE(pia+3,04);
      POKE(hctim,01);
      POKE(emtim,01);
      POKEW(enval1,00);
      POKEW(enval2,00);
      mhout:=0; mherr:=0.0;
      end;
   end;


procedure mdonn(var mdon: boolean);
   begin
   mdon:=true;
   POKE(pia,00);
   end;


procedure fdonn(var fdon: boolean);
   begin
```

ADAPT-7

```
   fdon:=true;
   POKE(fdtim,$82);
   POKE(pia,00);
   end;


procedure sdonn(var sdotpt: integer; var sdon: boolean; var sdstpt: real);
   var
   sdtach,eoc,i:        integer;

   begin
   sdon:=true;
   writeln; writeln;
   repeat
      write('Enter desired salt feedrate (1.00-4.25 #/min): ');
      readln(salt);
   until ((salt>=1.0) and (salt<=4.25));

   sdstpt:=0.151 + 3.346*salt;
   sdstpt:=(sdstpt/6.052)*(4095.0/5.0);
   POKE(sdtim,$82);
   POKE(pia,00);

   menu;
   end;


procedure mhonn(var  mhon,phon,htison: boolean);
   var
   store:                                              integer;

   begin
   mhon:=true;
   if not(htison) then
      begin
      store:=PEEK(pia);     store:=PEEK(pia+2);
      POKE(pia+1,07);       POKE(pia+3,07);
      POKE(emtim,00);
      end;
   htison:=true;
   POKEW(mhtim+4,00);    POKE(mhtim,$82);
   end;

procedure phonn(var  phon,mhon,htison: boolean);
   var
   store:                                              integer;

   begin
   phon:=true;
   if not(htison) then
      begin
      store:=PEEK(pia);     store:=PEEK(pia+2);
      POKE(pia+1,07);       POKE(pia+3,07);
      POKE(emtim,00);
      end;
   htison:=true;
   POKEW(phtim+4,00);    POKE(phtim,$82);
```

ADAPT-8

```
   end;

procedure display(var  mdstpt: real);
   var
   eoc,i,muxsig,store:    integer;
   temp,mdspeed,enval:    real;

   begin
   for i:=1 to 5 do writeln;
   writeln('  Channel    Temperature (F)');
   writeln(' -----------------------------');
   store:=(PEEK(pia) div 4)*4;
   POKE(pia,store);
   POKE(adb+10,$00);

   for i:=1 to 16 do
      begin
      POKE(pia,store+1);
      POKE(adb+11,$01);
      repeat
         eoc:=PEEK(adb+12);
      until (eoc<>0);
      muxsig:=256*PEEK(adb+13) + PEEK(adb+14);

      if ((i=1) or (i=2)) then
         temp:=232.3*(5.0/4095.0)*CONV(muxsig) + 45.36
                             else
         temp:=((5.0/4095.0)*CONV(muxsig) - 0.041122)/0.0029994;

      writeln('   ',i:3,'           ',temp:5:1);
      POKE(pia,store+3);
      end; {for loop}

  store:=ABS(PEEKW(enval1) - PEEKW(enval2));
  mdspeed:=0.0011646*mdstpt;
  enval:=(478.0*(mdspeed*CONV(store) - 1.5))/1055.0;
  if enval<0.0 then enval:=0.0;
  writeln;
  writeln('Current input energy rate = ',enval:5:2,' btu/min');
  end; {display}


procedure oddplus_temp(var  hour,min,sec: integer);
   var
   eoc,i,muxsig,store:    integer;
   temp:                  real;

   begin
   store:=(PEEK(pia) div 4)*4;
   POKE(pia,store);
   POKE(adb+10,00);
   set_print;  timeupdate(hour,min,sec);

   for i:=1 to 4 do writeln;
   writeln('     ',hour:2,':',min:2,':',sec:2);

   for i:=1 to 9 do
```

```
      begin
      POKE(pia,store+1);
      POKE(adb+11,$01);
      repeat
         eoc:=PEEK(adb+12);
      until (eoc<>0);
      muxsig:=256*PEEK(adb+13) + PEEK(adb+14);

      if ((i=1) or (i=2)) then
         temp:=232.3*(5.0/4095.0)*CONV(muxsig) + 45.36
                          else
         temp:=((5.0/4095.0)*CONV(muxsig) - 0.041122)/0.0029994;

      writeln('   ',i:3,'   ',temp:5:1);
      POKE(pia,store+3);
      end; {for loop}

   for i:=1 to 5 do
      begin
      POKE(pia,store+1); POKE(pia,store+3);
      end;

   for i:=15 to 16 do
      begin
      POKE(pia,store+1);
      POKE(adb+11,$01);
      repeat
         eoc:=PEEK(adb+12);
      until (eoc<>0);
      muxsig:=256*PEEK(adb+13) + PEEK(adb+14);
      temp:=((5.0/4095.0)*CONV(muxsig) - 0.041122)/0.0029994;
      writeln('   ',i:3,'   ',temp:5:1);
      POKE(pia,store+3);
      end; {for loop}

   set_term;
   end; {oddplus_temp}


procedure even_temp;
   var
   eoc,i,muxsig,store:      integer;
   temp:                    real;

   begin
   store:=(PEEK(pia) div 4)*4;
   POKE(pia,store);
   POKE(adb+10,$00);
   set_print;

   for i:=1 to 9 do
      begin
      POKE(pia,store+1); POKE(pia,store+3);
      end;

   for i:=10 to 14 do
      begin
```

```
      POKE(pia,store+1);
      POKE(adb+11,$01);
      repeat
         eoc:=PEEK(adb+12);
      until (eoc<>0);
      muxsig:=256*PEEK(adb+13) + PEEK(adb+14);
      temp:=((5.0/4095.0)*CONV(muxsig) - 0.041122)/0.0029994;

      writeln('   ',i:3,'   ',temp:5:1);
      POKE(pia,store+3);
      end; {for loop}

   set_term;
   end;  {even_temp}


procedure mdset(var mdstpt,fdstpt,delta: real;var  hcycle: integer);
   var
      store:                                    integer;
   begin
   POKE(pia+3,04);
   POKE(hctim,01);
   POKEW(mhtim+4,400);
   POKEW(phtim+4,400);

   repeat
      writeln; writeln; write('Enter speed setpoint (0.5-2.00 rpm): ');
      readln(mdstpt)
   until ((mdstpt>=0.5) and (mdstpt<=2.0));

   hcycle:=ROUND((0.90*15.259)/(mdstpt/30.0));
   POKEW(hctim+4,hcycle);
   store:=PEEK(pia+2);
   POKE(pia+3,07);

   delta:=1.0/mdstpt;
   fdstpt:=0.506 + 10.01*mdstpt;
   fdstpt:=(fdstpt/4.80)*(4095.0/5.0);
   mdstpt:=mdstpt*262.1*0.004*(4095.0/5.0);

   menu;
   end;

procedure Twmaxset(var  Twmax: real);
   begin
   writeln; writeln;
   repeat
      write('Enter desired maximum wheat temp (90.0-150.0 F): ');
      readln(Twmax)
   until ((Twmax>=90.0) and (Twmax<=150.0));
   menu;
   end;

procedure parinit(var  P: parmatrix; var  TW:        meavector;
                  var  CONTROL,NUM,PHI,EST,GAIN:   parvector;
                  var  alpha,traceP,K,pole1,pole2: real;
                  var  par:   boolean; var  n:     integer);
```

ADAPT-11

```
  var
     i:          integer;
  begin

  writeln;
  n:=4;

  write('Enter value of first pole: ');
  readln(pole1);
  write('Enter value of second pole: ');
  readln(pole2);

  for i:=1 to n do
     begin
     NUM[i]:=0.0;
{**** write('Enter coefficient #',i:1,' of EST: ');
     readln(EST[i]); ****}
     end;

   for i:=1 to 2 do
      begin
      PHI[i]:=TW[i];
      end;

   for i:=3 to n do
      begin
      PHI[i]:=CONTROL[i-1];
      end;


   alpha:=1.000;  par:=true;

   end;


 procedure Pupdate(var  P: parmatrix; var  PHI,NUM: parvector;
                   var  alpha,denp,traceP: real;
                   var  n:   integer);

   var
      R:          parvector;
      i,j:        integer;
      den:        real;
   begin

 {**alpha:=0.99*alpha + 0.01;**}      den:=denp + alpha;

    for i:=1 to n do
       begin
       R[i]:=0.0;
       for j:=1 to n do
          begin
          R[i]:=R[i] + PHI[j]*P[j,i];
          end;
       end;

    for i:=1 to n do
       begin
```

ADAPT-12

```
      for j:=1 to n do
         begin
         P[i,j]:=(P[i,j] - NUM[i]*R[j]/den)/alpha;
         end;
      end;

   traceP:=0.0;

   for i:=1 to n do
      begin
      for j:=1 to n do
         begin
         P[i,j]:=0.5*(P[i,j] + P[j,i]);
         P[j,i]:=P[i,j];
         if i=j then traceP:=traceP + P[i,j];
         end;
      end;

   POKE(intflag+1,01);                          {Clear limit switch marker}
   end;


procedure Preset(var  P: parmatrix; var  NUM: parvector;
                 var  sum,traceP:  real);
   var
      i,j:        integer;
   begin

   sum:=0.0;
   traceP:=0.0;

   for i:=1 to n do
      begin
      NUM[i]:=0.0;
      for j:=1 to n do
         begin
         if (i=j) then
            begin
            P[i,j]:=5.0;
            traceP:=traceP + P[i,j];
            end
                else
            P[i,j]:=0.0;
         end;
      end;

   POKE(intflag+1,01);                          {Clear limit switch marker}

   end;



begin                   {MAIN BODY OF PROGRAM BEGINS HERE}

   repeat
```

ADAPT-13

```
   write('Enter desired main drive speed (0.5-2.0 rpm): ');
   readln(mdstpt)
until ((mdstpt)>=0.5) and (mdstpt<=2.0));
repeat
   write('Enter desired max wheat temperature (90.0-150.0 F): ');
   readln(Twmax)
until ((Twmax)>=90.0) and (Twmax<=150.0));
repeat
   write('Enter desired closed-loop pole1 (0.0-1.0): ');
   readln(pole1);
   write('Enter desired closed-loop pole2 (0.0-1.0): ');
   readln(pole2);
until ((pole1)>=-1.0) and (pole1<=1.0)  and
      (pole2)>=-1.0) and (pole2<=1.0));
repeat
   write('Enter minimum proportional error gain (1.0-3.0): ');
   readln(Kmin);
until ((Kmin)>=1.0) and (Kmin<=3.0));


POKE(pia+1,$04);                    POKE(pia+3,$04);
POKE(pia,$FF);                      POKE(pia+2,$FF);
POKE(pia+1,00);                     POKE(pia+3,00);
POKE(pia,$0F);                      POKE(pia+2,$FF);
POKE(pia+1,$04);                    POKE(pia+3,$04);

POKE(mhtim+1,00);                   POKE(phtim+1,00);
POKE(mhtim,$80);                    POKE(phtim,$80);
POKE(mhtim+1,$A1);                  POKE(phtim+1,$A1);
POKEW(mhtim+6,$03E8);               POKEW(phtim+6,$03E8);
POKEW(mhtim+2,$01F4);               POKEW(phtim+2,$01F4);

POKE(fdtim+1,$A3);                  POKEW(mainhi,$0000);
POKEW(fdtim+2,$2710);               POKEW(mainlo,0000);
POKE(sdtim+1,$A3);                  POKEW(prehi,$0000);
POKEW(sdtim+2,$2710);               POKEW(prelo,0000);
                                    POKEW(auxhi,0000);
sdon:=false;  tb:=false;            POKEW(auxlo,0000);
mhon:=false;  mdon:=false;
fdon:=false;  phon:=false;
scan:=true;   htison:=false;
par:=false;   paron:=false;

hcycle:=ROUND((0.90*15.259)/(mdstpt/30.0));
POKE(hctim+1,$A1);
POKEW(hctim+2,$FFFF);
POKEW(hctim+4,hcycle);

POKE(emtim+1,01);
POKEW(emtim+2,$FFFF);
POKEW(enval1,0);
POKEW(enval2,0);

POKE(intflag,01);
POKE(intflag+1,01);
POKEW(intvector,$E000);
POKEW(outvector,printer);
```

ADAPT-14

```
POKE(outswitch,goterm);

gamma:=EXP(-1.400);
beta:=EXP(-kf*T);
delta:=1.0/mdstpt;
feed:=mdstpt;
mdstpt:=mdstpt*262.1*0.004*(4095.0/5.0);
fdstpt:=0.506 + 10.01*feed;
fdstpt:=(fdstpt/4.80)*(4095.0/5.0);

mdotpt:=0;     fdotpt:=0;       sdotpt:=0;             mhout:=0;
mderr:=0.0;    fderr:=0.0;      sderr:=0.0;            mherr:=0.0;
Iein:=0.0;     fdfilt:=0.0;     sdfilt:=0.0;           Iae:=0.0;
denp:=0.0;     traceP:=0.0;     Twfilt:=72.0;          n:=1;
sum:=0.0;      poorjob:=6.0;    POKE(intflag+1,01);
u:=0.0;        Progain:=0.0;    Twest:=0.0;

for i:=1 to 6 do
   begin
   PARERR[i]:=0.0;
   if i<=4 then
      begin
      PHI[i]:=0.0;
      CONTROL[i]:=0.0;
      UCON[i]:=0.0;
      MEAERR[i]:=0.0;
      end;
   if i<=2 then TW[i]:=72.0;
   end;

EST[1]:=0.803;   EST[2]:=0.171;
EST[3]:=0.0584;  EST[4]:=-0.0154;

K:=(1.0-pole1)*(1.0-pole2)/EST[3];
   if K<Kmin then K:=Kmin;
   if K>8.0  then K:=8.0;
GAIN[1]:=-(EST[4] - EST[3]*(pole1+pole2))/EST[3];
GAIN[2]:=-(EST[3]*pole1*pole2 - EST[4]*(pole1+pole2))/EST[3];
GAIN[3]:=-(EST[4]*pole1*pole2 - EST[3]*(1.0-pole1)*(1.0-pole2))/EST[3];
GAIN[4]:=EST[4]*(1.0-pole1)*(1.0-pole2)/EST[3];
c1:=EST[1];
c2:=EST[2];

menu;


              {The following contains all loops}

while scan do
begin

if mdon then                                    {Service main drive loop}
   begin
   POKE (adb+10,01);
   POKE (adb+11,01);
   repeat
```

ADAPT-15

```
      eoc:=PEEK(adb+12);
   until (eoc<>0);

   mdtach:=PEEKW(adb+13);
   mderro:=mderr;
   mderr:=mdstpt-CONV(mdtach);
   mdotpt:=mdotpt + ROUND(mdkc*(mderr-mderro + (T/mdtaui)*mderro));

   if (mdotpt<0)   then mdotpt:=0;
   if (mdotpt>127) then mdotpt:=127;
   mdout:=127 - mdotpt;

   POKE(pia+2,mdout);
   end;


if fdon then                                      {Service feed drive loop}
   begin
   POKE (adb+10,02);
   for i:=1 to 50 do store:=i;
   POKE (adb+11,01);
   repeat
      eoc:=PEEK(adb+12);
   until (eoc<>0);

   fdtach:=PEEKW(adb+13);
   fdfilt:=beta*fdfilt + (1.0-beta)*CONV(fdtach);
   fderro:=fderr;
   fderr:=fdstpt-fdfilt;
   fdotpt:=fdotpt + ROUND(fdkc*(fderr-fderro + (T/fdtaui)*fderro));

   if (fdotpt<0)   then fdotpt:=0;
   if (fdotpt>20000) then fdotpt:=20000;

   POKEW(fdtim+4,fdotpt);
   end;


if sdon then                                      {Service salt drive loop}
   begin
   POKE (adb+10,03);
   for i:=1 to 50 do store:=i;
   POKE (adb+11,01);
   repeat
      eoc:=PEEK(adb+12);
   until (eoc<>0);

   sdtach:=PEEKW(adb+13);
   sdfilt:=beta*sdfilt + (1.0-beta)*CONV(sdtach);
   sderro:=sderr;
   sderr:=sdstpt-sdfilt;
   sdotpt:=sdotpt + ROUND(sdkc*(sderr-sderro + (T/sdtaui)*sderro));

   if (sdotpt<0)   then sdotpt:=0;
   if (sdotpt>20000) then sdotpt:=20000;

   POKEW(sdtim+4,sdotpt);
```

ADAPT-16

```
    end;


pass:=PEEK(intflag);  limswitch:=PEEK(intflag+1);


if par then                                      {Update estimator condition}
   if limswitch=0 then
   begin

   if paron then                                      {Quit updating EST}
      if traceP<0.15 then
      begin
      paron:=false;
      traceP:=0.0;
      end;

   if paron then                              {Update covariance matrix P}
      begin
      Pupdate(P,PHI,NUM,alpha,denp,traceP,n);
      if traceP<0.5 then
         if sum>poorjob then
            Preset(P,NUM,sum,traceP);
      end;

   if not(paron) then                                {Begin updating EST}
      if sum>poorjob then
      begin
      Preset(P,NUM,sum,traceP);
      paron:=true;
      end;

   end; {if ... for est con}


                                       {**Perform heater loop control**}
if (mhon and phon and (pass=00)) then
   begin
   store:=(PEEK(pia) div 4)*4;
   POKE(pia,store);
   POKE(adb+10,00);
   for i:=1 to 2 do
      begin
      POKE(pia,store+1);
      POKE(pia,store+3)
      end;

   muxsig:=0;
   for i:=1 to 2 do
      begin

      POKE(pia,store+1);
      POKE(adb+11,01);

      repeat                                        {Get data for Twact}
         eoc:=PEEK(adb+12);
      until (eoc<>0);
```

ADAPT-17

```
   muxsig:=muxsig + PEEKW(adb+13);
   POKE(pia,store+3);
   end;

                                           {Calculate Tw and refresh}
temp:=0.5*CONV(muxsig);                       {measured error history}
Twact:=0.40708*temp - 13.71;
Twfilt:=gamma*Twfilt + (1.0-gamma)*Twact;

MEAERR[3]:=MEAERR[2];
MEAERR[2]:=MEAERR[1];
MEAERR[1]:=Twmax-Twact;
if MEAERR[2]<>0.0 then                            {Clamp error change}
   if (ABS(MEAERR[1])>2.0*ABS(MEAERR[2])) then    {to minimize effect}
      MEAERR[1]:=2.0*MEAERR[2];                        {of noise meas}


if par then
  begin

   for i:=1 to 2 do                     {Refresh temperature history}
      begin                                 {in the parameter vector}
      PHI[i]:=TW[i];
      end;

   for i:=3 to 4 do                         {Refresh control history}
      begin                                 {in the parameter vector}
      PHI[i]:=CONTROL[i-1];                    {i.e.   u1=u(k-3) ...}
      end;

   Twest:=0.0;

   for i:=1 to n do
      begin
      Twest:=Twest + PHI[i]*EST[i];             {Calculate Tw estimate}
      end;

   eoff:=Twfilt - Twest;                   {Calculate estimate error}

   sum:=ABS(eoff);                                {Update history of}
   for i:=1 to 5 do                                {estimation error}
      begin
      PARERR[7-i]:=PARERR[6-i];
      sum:=sum + PARERR[7-i];
      end;
   PARERR[1]:=ABS(eoff);


   if paron then                                {Update EST only if}
      begin                                     {conditions warrant}

      for i:=1 to n do
         begin
         NUM[i]:=0.0;
         for j:=1 to n do                           {Calculate P*PHI}
            begin
```

ADAPT-18

```
         NUM[i]:=NUM[i] + P[i,j]*PHI[j];
         end;
      end;

   denp:=0.0;

   for i:=1 to n do
      begin                                      {Calculate PHIt*P*PHI}
      denp:=denp + PHI[i]*NUM[i];
   end;

   if ((denp+alpha)<>0.0) then
      begin

      gain:=eoff/(denp + alpha);                      {Recursion gain}

      Progain:=1.0; temp:=0.0;

      for i:=1 to n do
         begin                               {New parameter estimate}
         EST[i]:=EST[i] + gain*NUM[i];       {and associated gain of}
         if (i<=2) then                           {discrete transfer}
            Progain:=Progain - EST[i]                      {function}
                  else
            temp:=temp + EST[i];
         end;
         Progain:=temp/Progain;
      end; {if ((den+ ...}


                                               {If conditions OK, }
   if EST[3]>0.01  then                   {update controller gains}
   if EST[3]<0.120 then
      if ABS(EST[3])>ABS(EST[4]) then
         if PARERR[1]<1.5 then
            if Progain<2.0 then
            if Progain>1.0 then
               begin
               K:=(1.0-pole1)*(1.0-pole2)/EST[3];
                 if K<Kmin then K:=Kmin;
                 if K>8.0  then K:=8.0;
               GAIN[1]:=-(EST[4] - EST[3]*(pole1+pole2))/EST[3];
               GAIN[2]:=-(EST[3]*pole1*pole2 -
                          EST[4]*(pole1+pole2))/EST[3];
               GAIN[3]:=-(EST[4]*pole1*pole2 -
                          EST[3]*(1.0-pole1)*(1.0-pole2))/EST[3];
               GAIN[4]:=EST[4]*(1.0-pole1)*(1.0-pole2)/EST[3];
               c1:=EST[1];
               c2:=EST[2];
               end;

   end; {if paron}
end; {if par}


                                      {Pole-placement control law}
```

ADAPT-19

```
u:=K*(MEAERR[1] - c1*MEAERR[2] - c2*MEAERR[3]);

for i:=1 to 4 do   u:=u + GAIN[i]*UCON[i];


for i:=1 to 3 do                                      {Shift control vectors}
   begin
   CONTROL[5-i]:=CONTROL[4-i];
   UCON[5-i]:=UCON[4-i];
   end;

if u<0.0   then u:=0.0;                              {Update both control}
if u>250.0 then u:=250.0;
UCON[1]:=u;                                                {vector leads}
if u>100.0 then u:=100.0;
CONTROL[1]:=u;

mhout:=ROUND(20.0*u);
POKEW(phtim+4,mhout);                           {Output values for control}
POKEW(mainhi,mhout);
POKEW(prehi,mhout);



                                                {Shift temperature vector}
TW[2]:=TW[1];
TW[1]:=Twfilt;


                                                         {Display results}
   set_print;
   writeln('Twact=',Twact:5:1,'  u=',UCON[1]:7:1,'  traceP=',
          traceP:6:2,'  Progain=',Progain:8:3,'  Sum=',sum:6:2);
   writeln('Twest=',Twest:5:1,'  EST1-n: ',EST[1]:8:4,EST[2]:8:4,
          EST[3]:8:4,EST[4]:8:4);
   writeln('Twfil=',Twfilt:5:1,'  PHI1-n: ',PHI[1]:8:1,PHI[2]:8:1,
          PHI[3]:8:1,PHI[4]:8:1);
   writeln('K=  ',K:6:3,'   GAINs: ',GAIN[1]:8:3,GAIN[2]:8:3,
          GAIN[3]:8:3,GAIN[4]:8:3);
   for i:=1 to 5 do writeln;
   set_term;

POKE(intflag,01);                             {Clear interrupt timer marker}

                                              {Update performance measures}
Iae:=Iae + delta*ABS(MEAERR[1]);
store:=ABS(PEEKW(enval1) - PEEKW(enval2));
enval:=0.0005277*mdstpt*CONV(store) - 0.6796;
if enval<0.0 then enval:=0.0;
Iein:=Iein + delta*enval;

end;{***** Major heater loop *****}
```

ADAPT-20

```
                                    {Check keyboard and service if necessary}
test:=PEEK(kbstatus) div 8;
tb:=ODD(test);


if tb then
   begin
   punch:=PEEK(kbstatus-1);
   case punch of

      $30:         shut_down(mdon, fdon, mhon, phon, sdon, htison, mdotpt,
                        mhout, Iae, Iein, mherr);
      $4D, $6D:    mdonn(mdon);
      $4E, $6E:    mdoff(mdon, mdotpt);
      $46, $66:    fdonn(fdon);
      $47, $67:    fdoff(fdon, fdotpt);
      $48, $68:    mhonn(mhon, phon, htison);
      $49, $69:    mhoff(mhout, hour, min, sec, mhon, phon, htison, mherr,
                        Iae, Iein, mdstpt);
      $50, $70:    phonn(phon, mhon, htison);
      $51, $71:    phoff(mhout, phon, mhon, htison, mherr);
      $53, $73:    sdonn(sdotpt, sdon, sdstpt);
      $54, $74:    sdoff(sdon, sdotpt);
      $31:         mdset(mdstpt, fdstpt, delta, hcycle);
      $32:         parinit(P, TW, CONTROL, NUM, PHI, EST, GAIN, alpha, traceP,
                          K, pole1, pole2, par, n);
      $33:         Twmaxset(Twmax);
      $34:         par:=false;
      $44, $64:    display(mdstpt);
      $4F, $6F:    oddplus_temp(hour, min, sec);
      $45, $65:    even_temp;
      $43, $63:              ;
      $58, $78:    scan:=false;
      $20:         menu;

      $4A, $6A, $4B, $6B, $4C, $6C, $52, $72, $0D:                   ;
      $55, $75, $56, $76, $57, $77, $59, $79:                        ;
      $5A, $7A, $35, $36, $37, $38, $39, $41, $61, $42, $62:
   end; {case}

 end; {if}

 end; {major while}
end.
```

ADAPT-21

```
************************* INTPLUS *************************
*             This routine  services the interrupts          *
*       generated on the solid particle heat exchanger      *
*       in order to cycle the heaters properly.  It         *
*       has been designed to function with the PASCAL       *
*       closed-loop program called "ADAPT."                 *
*                                                           *
*************************************************************
*
PIAA    EQU     $FFCC
PIAB    EQU     $FFCE
MHT     EQU     $EC30                   ;Main heater control timer
PHT     EQU     $EC38                   ;Preheater control timer
AHT     EQU     $EC28                   ;Auxheater control timer
HCT     EQU     $EC08                   ;Heater dual cycle timer
EMT     EQU     $EC00                   ;Energy measurement timer
*                                                       (counter)
MAINHI  EQU     $E400
MAINLO  EQU     $E402
PREHI   EQU     $E404
PRELO   EQU     $E406
AUXHI   EQU     $E408
AUXLO   EQU     $E40A
LOOP    EQU     $E40D
LOCTIM  EQU     $E410
ENVAL1  EQU     $E420
ENVAL2  EQU     $E422
LASTEN  EQU     $E424
INTFLAG EQU     $E425
*
*
        ORG     $E000
*
        LDA     PIAA+1          ;Interrupt from limit switch?
        LSLA
        BCC     CHKTIM          ;If so, begin cycle #2.
        LDA     #$82
        STA     HCT
        LDD     MAINHI
        STD     MHT+4
        LDD     PRELO
        STD     PHT+4
        LDD     AUXLO
        STD     AHT+4
        LDA     PIAA
        LDA     PIAB
        CLR     INTFLAG+1               ;Marker for limit switch.
        BRA     RPASCAL
*
CHKTIM  LDA     PIAB+1          ;Interrupt from timer?
        LSLA
        BCC     OOPS            ;If so, begin cycle #1,
        LDA     #01             ;   otherwise go on low cycles.
        STA     HCT
        LDD     MAINLO
        STD     MHT+4
        LDD     PREHI
```

ADAPT-22

```
        STD     PHT+4
        LDD     AUXLO
        STD     AHT+4
        CLR     INTFLAG         ;Marker for timer.
 *
        LDD     EMT+2           ;Update input energy measurements.
        PSHS    D
        LDA     LASTEN
        BNE     SECVAL
        PULS    D
        STD     ENVAL1
        LDA     #01
        STA     LASTEN
        BRA     CONT
SECVAL  PULS    D
        STD     ENVAL2
        CLR     LASTEN
*
CONT    JSR     GETIME          ;Update time stamp.
        LDA     PIAB
        LDA     PIAA
        BRA     RPASCAL
*
OOPS    LDA     PIAA            ;If neither, reset and go to
        LDA     PIAB            ;     a low power mode.
        LDD     MAINLO
        STD     MHT+4
        LDD     PRELO
        STD     PHT+4
        LDD     AUXLO
        STD     AHT+4
*
RPASCAL RTI                     ;Return to Pascal program.
*
*
***********************************************************
*                                                         *
*   The following is taken from the CMS 9619 Debug19      *
*   resident monitor.  Some modifications have been       *
*   made so that the routine can be called from           *
*   PASCAL.  The purpose of the routine is to read the    *
*   real-time on-board clock, and write the contents      *
*   of its registers in ram for later examination.        *
*                                                         *
***********************************************************
*
TIMDAT      EQU   $FFC4
ADATA       EQU   0
ACTRL       EQU   1
BDATA       EQU   2
BCTRL       EQU   3
WRITON      EQU   %00111100
WRITOFF     EQU   %00110100
READON      EQU   WRITOFF
YRADDR      EQU   %1100
DAYADDR     EQU   %1000
HRSADDR     EQU   %0101
```

ADAPT-23

```
FLEXYR     EQU    $CC10
FLEXMO     EQU    $CCOE
FLEXDA     EQU    $CCOF
WARMS      EQU    $CD03
*
*
GETIME     LDX    #LOCTIM
           BSR    LABO          ;Read real-time clock
           RTS
*
*
LABO       PSHS   X
           LDU    #TIMDAT
LAB1       LDB    #YRADDR
           LDX    ,S
LAB2       BSR    SUB1
           BCS    LAB1
           CMPB   #HRSADDR
           BNE    LAB3
           ANDA   #$07
LAB3       STA    ,X+
           DECB
           BPL    LAB2
           PULS   PC,X
*
SUB1       PSHS   B,A
           LBSR   SUB2
           LDB    #$38
           STB    BCTRL,U
           LDA    #$30
           STA    BDATA,U
           ORB    #$04
           STB    BCTRL,U
           LDA    #$20
           STA    BDATA,U
           LDB    #$34
           STB    BCTRL,U
           EXG    X,Y
           EXG    Y,X
           LDA    BDATA,U
           ANDA   #$0F
           STA    ,S
           LDB    #WRITON
           STB    BCTRL,U
           BSR    SUB3
           PULS   PC,B,A
*
SUB2       PSHS   B,A
           LDB    #$38
           STB    BCTRL,U
           LDA    #$3F
           STA    BDATA,U
           ORB    #04
           STB    BCTRL,U
           LDB    1,S
           STB    BDATA,U
           LDB    #$34
```

ADAPT-24

```
           STB    BCTRL,U
           LBRN   SUB2
           LDB    #WRITON
           STB    BCTRL,U
           PULS   PC,B,A
*
SUB3       PSHS   B,A
           LDB    #$0F
           LBSR   SUB2
           LDB    #$38
           STB    BCTRL,U
           LDA    #$30
           STA    BDATA,U
           ORB    #04
           STB    BCTRL,U
           LDA    #$20
           STA    BDATA,U
           LDB    #$34
           STB    BCTRL,U
           LDA    BDATA,U
           EORA   #$80
           ASLA
           PULS   PC,B,A
*
           END
```

TUNSECN-1

```
   (*********************** TUNESECN ************************
    *                                                      *
    *  The following program is designed to help tune free  *
    *  parameters in discrete pole-placement control laws   *
    *  for the solid particle heat exchanger.               *
    *                                                      *
    ********************************************************

program tunesecn;

type

   history=array[1..4] of real;
   errhist=array[1..3] of real;
var

   PHI,EST,CON,GAIN,UACT:                           history;
   ERR:                                             errhist;
   n,i,j,pass:                                      integer;
   Tact,pole,iae,y,K,r,u:                           real;
   alpha1,alpha2:                                   real;

begin

                         (Initialization)

   for i:=1 to 4 do
      begin
      PHI[i]:=0.0;
      CON[i]:=0.0;
      UACT[I]:=0.0;
      if i<4 then
         ERR[i]:=0.0;
      end;

   EST[1]:=0.803; EST[2]:=0.171; EST[3]:=0.0584; EST[4]:=-0.0400;
   n:=1; iae:=0.0; Tact:=72.0; pass:=1; r:=130.0;

   write('Enter value of first pole: '); readln(alpha1);
   write('Enter value of second pole: '); readln(alpha2);

   K:=(1.0-alpha1)*(1.0-alpha2)/EST[3];
   GAIN[1]:=-(EST[4] - EST[3]*(alpha1 + alpha2))/EST[3];
   GAIN[2]:=-(EST[3]*alpha1*alpha2 -
             EST[4]*(alpha1+alpha2))/EST[3];
   GAIN[3]:=-(EST[4]*alpha1*alpha2 -
             EST[3]*(1.0-alpha1)*(1.0-alpha2))/EST[3];
   GAIN[4]:=EST[4]*(1.0-alpha1)*(1.0-alpha2)/EST[3];

   writeln(alpha1:6:2,alpha2:6:2,K:6:2,GAIN[1]:8:3,
           GAIN[2]:8:3,GAIN[3]:8:3,GAIN[4]:8:3);
```

TUNSECN-2

```
                     (Begin simulation)

while (n<=150) do
   begin

   ERR[3]:=ERR[2];
   ERR[2]:=ERR[1];
   ERR[1]:=r - Tact;

   u:=K*(ERR[1] - EST[1]*ERR[2] - EST[2]*ERR[3]);
   for i:=1 to 4 do u:=u + GAIN[i]*CON[i];

   CON[1]:=u;
   if u<0.0   then u:=0.0;
   if u>100.0 then u:=100.0;
   UACT[1]:=u;

   y:=0.0;
   for i:=1 to 4 do y:=y + PHI[i]*EST[i];
   Tact:=y+72.0;

   if (((pass=1) and (n=1)) or (pass=10)) then
      begin
      writeln(n:4,Tact:6:1,UACT[1]:6:1,
      CON[1]:7:1,CON[2]:7:1,CON[3]:7:1,CON[4]:7:1,
      PHI[1]:7:1,PHI[2]:7:1,PHI[3]:7:1,PHI[4]:7:1);
      end;

   for i:=1 to 3 do
      begin                                 (Shift histories)
      CON[5-i]:=CON[4-i];
      UACT[5-i]:=UACT[4-i];
      end;

   PHI[4]:=PHI[3];
   PHI[3]:=UACT[3];
   PHI[2]:=PHI[1];
   PHI[1]:=y;
   n:=n+1;
   if (pass=10) then pass:=1 else pass:=pass+1;
   iae:=iae + ABS(ERR[1]);

   end; (while ...)
   writeln('IAE= ',iae:12:2);
end.
```

2

VITA

Hayward Ray Simonton, Jr

Candidate for the Degree of

Doctor of Philosophy

Thesis: PERFORMANCE EVALUATION OF A COUNTERFLOW PARTICLE-TO-PARTICLE HEAT EXCHANGER

Major Field: Agricultural Engineering

Biographical:

Personal Data: Born in Griffin, Georgia, July 21, 1957, the son of Hayward R. and Betty W. Simonton. Married to Regina Maria Re on August 1, 1981.

Education: Graduated from Griffin High School, Griffin, Georgia in June, 1975; received Bachelor of Science Degree in Agricultural Engineering from University of Georgia in December, 1979; received Master of Science Degree in Agricultural Engineering from University of Georgia in August, 1981; completed requirements for the Doctor of Philosophy Degree at Oklahoma State University in May, 1986.

Extracurricular training: Short Courses on Human Relations (1977), MC6800 Microprocessor (1980), Allen-Bradley PLC-III Programming (1981), Mathematical Modeling (1982), and Process Control (1984).

Professional Experience: Graduate research assistant, Agricultural Engineering, University of Georgia, January, 1980 to August, 1981; Engineer for the Advanced Technology Division of the Kellogg Company, Battle Creek, Michigan, August, 1981 to June, 1983; Graduate research assistant, Agricultural Engineering, Oklahoma State University, June, 1983 to January, 1986.

Professional Organization: American Society of Agricultural Engineers.