LOS ALAMOS
NATIONAL LABORATORY

Advanced Control of Operations in the Blast Furnace Project

Phase One Effort Documentation:

Optimal Operation and Control of the Blast Furnace Stoves

Kenneth R. Muske
Department of Chemical Engineering
Villanova University, Villanova, PA 19085

James W. Howse, Glen A. Hansen, and Dominic J. Cagliostro
Computational Science Methods Group XCM, Mail Stop F645
Applied Theoretical and Computational Physics Division
Los Alamos National Laboratory, Los Alamos, NM 87545

LANL Technical Report: LA–UR–99–5051

Report Date: August 25, 1999

Abstract

This report documents the Phase One effort for the project “Advanced Control of Operations in the Blast Furnace” sponsored by the Department of Energy Office of Industrial Technology, Energy Efficiency and Renewable Energy, under Work Authorization ED/18019/AL04, and performed by Los Alamos National Laboratory and Ispat Inland Steel. The first phase of this project involved improving the thermal efficiency of blast furnace stoves using advanced control technology. This technology was developed at Los Alamos National Laboratory and implemented on the No. 7 blast furnace stoves at the Ispat Inland Steel facility in East Chicago, Indiana. A post-audit of the performance of the technology was carried out by Ispat Inland personnel in January, 1999. The result was a five percent reduction in the energy used by the stoves attributable to the advanced control technology. This reduction was achieved with the advanced control technology operating for approximately eighty percent of this test period. Similar results have been obtained for subsequent blast furnace stove operation with advanced control.

The advanced control technology developed for this phase of the project consists a model-based control strategy that uses a detailed heat transfer model of the hot blast stoves. The control strategy determines the minimum amount of fuel necessary to achieve the blast air energy requirements based on the blast furnace stove model predictions. State and parameter estimation is used to update the model predictions and adapt to changes in the blast furnace stove system and operation. The advanced control software consists of a FORTRAN 77 program that runs on a workstation computer. Implementation of the technology was accomplished by interfacing the workstation computer to the Ispat Inland process monitoring and control computer system network.

Contents

Acknowledgments

We would like to acknowledge financial support for this work from the Department of Energy Office of Industrial Technology, Energy Efficiency and Renewable Energy, under Work Authorization ED/18019/AL04 and the blast furnace department at Ispat Inland Inc. This project has been managed by the Department of Energy Office of Industrial Technologies Team Leader for Industrial Strategies in Steel, Scott Richlen, and the Process Lead for Industrial Processes in Steel, Gobind Jagtiani, and the Los Alamos National Laboratory Civilian and Industrial Technology Program Manager Melissa A. Miller.

We wish to acknowledge the technical contributions of Pinakin Chaubal, Paul Quisenberry, and Mike Washo of Ispat Inland during the development and demonstration of the advanced control technology presented in this document. Pinakin Chaubal served as the technical and project manager of the Phase One project effort for Ispat Inland Inc. Paul Quisenberry provided invaluable insights into the operation of the No. 7 blast furnace stoves and a number of useful suggestions concerning the advanced control technology implementation. Pinakin and Paul were also responsible for the post-audit of this technology. Mike Washo developed the interface software between the No. 7 blast furnace process monitoring and control computer system network and the workstation computer executing the software. He also provided the operating data required for the initial development and verification of the stove model and control algorithm to Los Alamos.

We also acknowledge the support of Madhu Ranade, manager of iron making operating technology, and Wendell Carter section manager of furnace operations for the No. 7 blast furnace, for access to the No. 7 blast furnace to perform plant tests and incorporate the technology. Finally, we wish to recognize the cooperation and suggestions from the No. 7 blast furnace operators and operating foreman during the implementation effort.

Chapter 1

Introduction

The Phase One effort for this project concerned the development of advanced control technology to improve the energy efficiency of blast furnace stoves. This technology was developed in cooperation with the Ispat Inland Steel Research and Blast Furnace departments in East Chicago, Indiana. The technology was implemented on the No. 7 blast furnace stoves at the Ispat Inland Steel facility in East Chicago. The results presented in this document reflect the operating experience with the technology on these blast furnace stoves. Although this technology was developed for the Ispat Inland No. 7 blast furnace stoves, we expect similar results for other blast furnace stove systems with similar operating conditions.

This document is organized as follows. An overview of the Phase One effort for this project is presented in Chapter 2. This chapter provides a brief introduction to the operation of blast furnace stoves, previous related work, and the advanced control technology developed for the Phase One effort. Chapter 3 discusses the Ispat Inland No. 7 blast furnace stove system. Chapter 4 outlines the Phase One effort objectives and Chapter 5 presents the results of the post-audit carried out on the advanced control technology by Ispat Inland.

The modeling and control technology developed for Phase One of this project is discussed in detail in Chapters 6 through 10. Chapter 6 discusses the hot blast stove process model development and modeling equations. Chapter 7 presents the numerical solution techniques used to solve the hot blast stove model equations. Chapter 8 discusses the model-based control and Chapter 9 discusses the state and parameter estimation methodology applied in the advanced control technology development. Chapter 10 presents the nonlinear programming approach used to solve the resulting control and estimation optimization problems.

Documentation of the software developed to implement the advanced control strategy on the No. 7 blast furnace stoves at the Ispat Inland facility is provided in Chapters 11 and 12. Chapter 11 documents the use and operation of the advanced control software. Chapter 12 contains the program documentation.

Chapter 2

Phase One Overview

This section provides a brief description of the hot blast stove along with the nomenclature used to describe its physical features and operation. An overview of related prior work concerning the modeling, control, and optimization of hot blast stoves is then presented. Finally, the advanced control technology developed as part of this project is summarized.

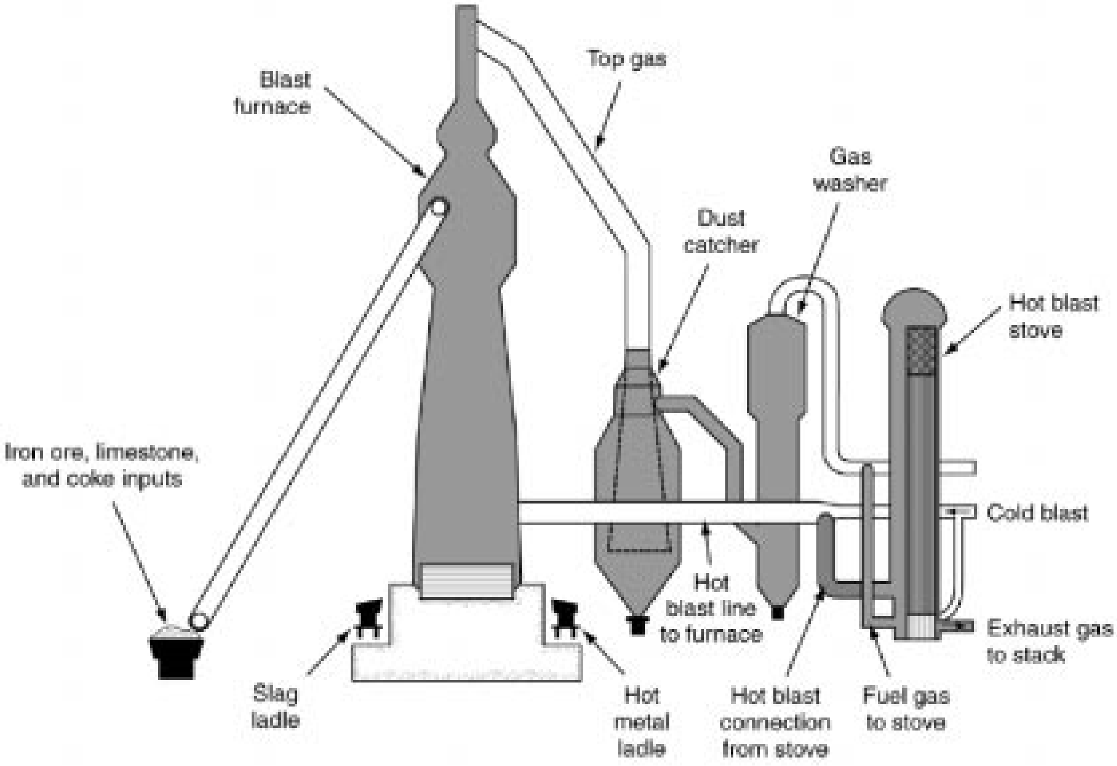


Figure 2.1: Blast furnace system.

2.1 Hot Blast Stove Description

A blast furnace is used to produce molten iron from iron oxides, coke, and flux. A diagram of a typical blast furnace system is shown in Figure 2.1. One of the major sources of energy for this process is the sensible heat coming from the preheated air, referred to as blast air, that is injected into the furnace. This air is preheated in tall, cylindrical, refractory-filled thermal regenerators called hot blast stoves. Each hot blast stove goes through alternate cycles of heating and cooling referred to as "on-gas" and "on-blast" cycles respectively.

During the on-gas cycle, the stove is heated by the combustion of fuel gas in the combustion chamber of the stove. The combustion products, which are referred to as the waste gas, enter the dome, or top of the stove, and then descend down through a series of hollow refractory bricks referred to as the checkers. The energy in the waste gas is transferred to the checkers during this cycle. After the on-gas cycle is completed and prior to the on-blast cycle, the stove is pressurized with air in the pressurization cycle. The stove pressure is raised above the blast furnace pressure during this cycle so that the blast air can enter the blast furnace. For the on-blast cycle, the flow through the stove is reversed. Blast air passes up through the checkers, where it is heated, enters the dome, and then proceeds downward into the combustion chamber before exiting the stove. The temperature of the blast air is controlled by diverting a fraction of the inlet air to the stove directly into the combustion chamber to mix with the heated air. This fraction of the blast air is diverted using a valve in the blast air line referred to as the mixer valve. When the mixer valve is closed, all of the blast air passes through the stove. After an on-blast cycle is complete, the stove is returned to atmospheric pressure during the blow-off cycle. In this cycle, the pressurized air remaining in the stove is vented to the atmosphere.

The principal fuel for the hot blast stoves is the carbon monoxide and hydrogen contained in the top gas coming from the blast furnace. In order to achieve the required blast air temperature, however, the top gas is typically enriched with a higher heating value fuel such as coke oven gas or natural gas. The resulting hot blast stove fuel gas is referred to as mixed fuel gas. The key to reducing the operating cost of the blast furnace stoves is to minimize the enrichment necessary to achieve the hot blast air energy requirements for the blast furnace.

2.2 Previous Related Work

The initial modeling studies of hot blast stoves were used as design tools. These models assumed constant gas flow rates, constant specific heat, no axial or radial conduction in the checkers, and dynamic equilibrium between the heating and cooling cycles, and they neglected the effect of the transitions between cycles [38] and [12]. Non-constant heat capacities and heat transfer coefficients were considered by Butterfield, *et al.* [5]. Varying gas flow rates were considered by Willmott [43] and Kwakernaak, *et al.* [20]. In these works, the effect of radial conduction in the checker material and radiation heat transfer from the gas to the checkers during the heating cycle were approximated by modification of the convective heat transfer coefficient as discussed by Hausen and Binder [12]. The effect of radial conduction was explicitly considered by Willmott [44].

Previous model-based control studies on hot blast stoves concerned the optimization of the stove cycles and the improvement in the thermal efficiency of the stove. The optimal operation of a single stove is considered by Kwakernaak, *et al.* [21] based on a simplified blast stove model. A detailed study of the optimal enrichment policy is presented by Walsh and Mitterer [42]. Optimization of multi-stove operation is considered by Zuidema [45] and Labossiere and Lee [23]. Feedforward control based on a dynamic equilibrium stove model is discussed by Jeffreson [18]. A hot blast stove model-based controller using a linear model is presented by Matoba, *et al.* [28]. A generic model controller implemented on a laboratory scale stove is presented by Labossiere and Lee [22]. A model-based controller using empirical models is discussed by Monkern, *et al.* [29].

2.3 Advanced Control Technology Description

A detailed, partial differential equation based, dynamic heat transfer model for the Ispat Inland No. 7 blast furnace hot blast stoves in the East Chicago, Indiana facility was developed and verified using plant data. The model is capable of accurately predicting the temperature and energy content of the stoves during the thermal regenerative cycles. Variation in the physical and operating properties, radiative heat transfer, radial and axial conduction in the checkers, and the transitions between cycles were explicitly considered in the development of this model. Since the model-based control technology requires a large number of successive model runs on-line, the computational requirement was the most important consideration for implementation. The assumptions made to reduce the physical model, and the computational requirement, were selected to minimize the effect on the control-relevant model results. A computationally fast and robust model solution algorithm was selected to solve the model equations on-line.

The advanced control technology consists of a nonlinear model-based control and estimation technique that uses the detailed heat transfer model to improve the energy efficiency of the blast furnace stoves. Batch nonlinear least squares estimation is used to update the predicted stove temperature profile and heat transfer coefficients after each on-blast cycle. These updates are necessary to correct the calculated values due to model error and adapt the controller for changes in the blast furnace stove system and operation. Nonlinear model predictive control is used to determine the minimum amount of fuel necessary to achieve the energy requirements from the stove for the subsequent regenerative heating cycle. The estimation and control techniques both require the on-line solution to a nonlinear optimization problem prior to each on-gas cycle. A commercially available nonlinear programming algorithm was selected to perform these optimizations.

Chapter 3

Ispat Inland Blast Furnace Stove System

Figure 3.1 presents a schematic of the hot blast stoves used with the Ispat Inland Steel No. 7 blast furnace in East Chicago, Indiana. The combustion chamber is the chamber on the left in Figure 3.1. The chamber on the right contains the refractory brick, or checkers. The insert in Brick Zone #2 shows the shape of one of the checkers.

3.1 Stove System Operation

The No. 7 blast furnace uses a three-stove system operation with constant cycle times. The typical on-gas cycle time is fifty minutes and the typical on-blast cycle time is thirty minutes. The transition time between the on-gas and on-blast cycle is typically five minutes and is referred to as the pressurization cycle. During this cycle, the pressure of the blast furnace stove is increased from atmospheric to blast furnace pressure. The transition between the on-blast and the on-gas cycle is referred to as the blow-off cycle and is also typically five minutes. During this cycle, the blast furnace stove is returned to atmospheric pressure by venting the contents.

For the No. 7 blast furnace, two stoves will be in an on-gas cycle operation and one stove will be in an on-blast cycle operation at any given time. The on-blast cycle stove must produce all of the blast air at the temperature and flow rate required for the blast furnace during its on-blast cycle. The temperature of the blast air is controlled by a temperature controller that adjusts the mixer valve position in the blast air line. After the mixer valve closes, all of the blast air is sent through the stove and the stove can no longer produce blast air at the required temperature. The nominal blast air operating temperature target is 1250 °C.

Natural gas enrichment is presently being used for the Ispat Inland No. 7 blast furnace

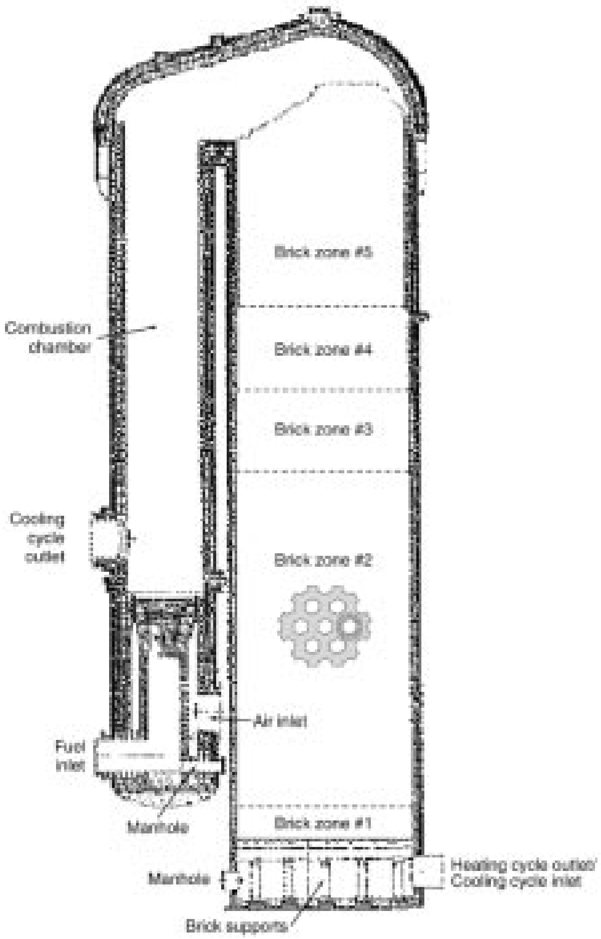


Figure 3.1: Blast furnace stove.

stoves. Since the No. 7 blast furnace hot blast air volume requirement is larger than that produced by most other blast furnace stove systems, higher stove temperatures must be obtained during the on-gas cycle in order to supply the required heat duty. Attaining these temperatures requires significant enrichment of the fuel gas with natural gas. The typical natural gas fraction of the mixed fuel gas is on the order of 7 vol%. The mixed fuel gas flow rate to the on-gas cycle stoves is under flow control. Typical mixed gas fuel flow rates are on the order of 20–30 m^3/sec. The natural gas enrichment is set by a mixed fuel gas heating value controller. Typical mixed gas heating values range from 140–160 BTU/ft^3. The combustion air flow rate to the burner in the combustion chamber of the stove is set by a waste gas excess oxygen controller. The typical excess oxygen target is 1.2–1.5 vol %.

3.2 Process Measurements

There are three temperature measurements in the stove. The dome temperature is measured by an optical pyrometer aimed at a target on the top of the checkerwork in the stove. It is assumed to be representative of the top checker temperature, but can exhibit unpredictable behavior. The interface temperature is measured by thermocouples contained within thermowells inside the checkers located eleven meters from the top of the checkerwork. Since the thermowell has contact to both solid and gas in the stove, the measured interface temperature represents some combination of both temperatures. Two interface temperatures that appear to be more representative of the solid temperature are averaged to obtain the value used in this work. The grid temperature is measured by thermocouples placed on the cast iron supports directly below the bottom of the checkerwork. It is assumed that this measurement is an indication of the bottom checker temperature.

The blast air inlet and exit temperatures are reliably measured. The temperature of the heated blast air exiting the stove before mixing with the by-passed blast air in the combustion chamber is not measured. The temperature of the combustion gases entering the checkerwork in the stove during the on-gas cycle is also not measured. This temperature is computed from an energy balance over the combustion chamber based on the top gas and natural gas compositions and flow rates. Since the burners were designed for coke oven gas enrichment, they perform poorly when using natural gas enrichment. The result is incomplete combustion of the natural gas and uncertainty in the calculated combustion temperature. The temperature of the waste gas exiting the stove during the on-gas cycle is not available.

Flow rates are measured by the pressure drop through a venturi tube. The total blast air flow rate and the blast air flow through the stove during the on-blast cycle is determined from the pressure drop compensated for the blast air temperature and pressure. The moisture and oxygen injected into the blast air are reliably measured. The total top gas and natural gas flow rates to the stove system are compensated flow rates. The mixed gas flow rate to each stove is not compensated. The fraction of total fuel gas to a stove is determined by the ratio of its uncompensated flow to the total uncompensated flow rates. The top gas composition is available from a process analyzer. The natural gas composition is taken as the nominal value from the supplier.

3.3 Computer System

The model-based control technology software executes on a Digital Equipment Corporation Alpha 500 AU workstation operating under the VMS operating system. The only function of this computer is the model-based control technology. Each of the process measurement values required by the model-based control is available in real-time from the Ispat Inland process computer system and also stored in an historical process data base. The interface between the Ispat Inland process control and monitoring computer system and the computer system used to execute the model-based control algorithm is through a series of data files transferred between the computers. The data files containing the process measurements required by the model-based controller are generated from software developed by Ispat Inland personnel. Implementation of the optimal fuel flow rate profile for each stove is also provided by software developed by Ispat Inland personnel.

Although this method is not the most efficient for transferring information between the model-based control computer and the Ispat Inland process monitoring and control computer system, the intent was to quickly develop an interface for demonstration purposes. This method was easily configured and sufficient for demonstration of the technology. The final configuration and implementation of the interface depends on the target process monitoring and control computer system. This development is not part of the Phase One advanced control technology effort documented in this report.

Chapter 4

Phase One Objectives

The objective of the Phase One effort was the development and demonstration of advanced control technology to improve the energy efficiency of the No. 7 blast furnace stoves at the Ispat Inland steel facility in East Chicago, Indiana. Improvement in the energy efficiency is related directly to reduction in the natural gas required by the blast furnace stove system. Reduction in the use of natural gas is also the key to reducing the operating cost of the hot blast stoves.

The Phase One effort concentrated on minimizing the natural gas used by the hot blast stove system. Since the No. 7 blast furnace is a three-stove system operated with constant cycle times, the control technology was not designed to manipulate the hot blast stove sequence and cycle times to reduce natural gas consumption. Since the natural gas flow rate cannot be independently set for each stove at the East Chicago facility, the control technology is also not designed to manipulate the natural gas enrichment. Although it is possible to adjust the natural gas enrichment from the process monitoring and control computer system, implementation of an optimal enrichment policy when two stoves are at a different point in their on-gas cycles at any time is a difficult problem with little benefit. Therefore, we consider the optimal operation of each individual blast furnace stove that minimizes the enriched fuel gas flow rate.

The optimal operation of a blast furnace stove is that which uses the minimum amount of fuel necessary during the on-gas cycle to achieve the blast air flow rate and temperature requirements for the next on-blast cycle. The minimum amount of fuel represents the point where no blast air is being by-passed to the combustion chamber to mix with the heated air exactly when the end of the on-blast cycle is reached. In practice, a small amount of additional heat is put into the stove to ensure that the on-blast cycle requirements are met. A measure of this additional heat is the mixer valve position at the end of the on-blast cycle.

The advanced control technology was designed to determine the minimum mixed fuel gas flow rate policy during the on-gas cycles that consistently achieve the desired final mixer valve position at the end of the following on-blast cycles. Additional heat added to the stove during the on-gas cycle increases the stove temperature at the end of the on-blast cycle. The result is a decrease in the energy efficiency of the stove because the increased stove temperatures at the end of the on-blast cycle reduce the temperature driving force for heat transfer during the next on-gas cycle causing an increase in the energy leaving the stove through the waste gas.

Chapter 5

Phase One Results

The blast furnace stove model and model-based control technology was developed at Los Alamos National Laboratory and implemented at the Ispat Inland steel East Chicago, Indiana facility in August, 1998. After approximately one month of operation, the controller was taken off-line due to operational problems with the software. Based on this initial operating experience, adjustments to the model and control algorithm were performed and implemented on the No. 7 blast furnace stoves in November, 1998. The control technology has been continuously operational since this time. The final version of the software, in which a number of minor adjustments and tuning issues were addressed, was delivered to Ispat Inland in July 1999.

A post-audit of the control technology was carried out in January, 1999. The results of this post-audit, documented in [31], are summarized in this section. Table 5.1 presents a comparison between the performance of the stoves with the controller in operation during December, 1998, referred to as the test period, and previous stove performance at similar sensible heat requirements without control during October, 1998, referred to as the base period. The natural gas used per Gcal of energy transferred to the blast air was reduced by 5 % during the test period. This reduction was achieved at a higher blast air energy requirement with an 8.58 % increase in the average hot blast flow rate and an 0.45 % increase in the average hot blast temperature between the test and base periods. The controller was estimated to be on-line for approximately 80 % of the test period. Similar results have been obtained for subsequent operation.

This reduction in fuel gas usage is obtained by more consistently achieving the desired final mixing valve position. Any excess heat left in the stove, as measured by the final mixer valve position, reduces the energy efficiency because increased stove temperatures at the end of the on-blast cycle reduce the temperature driving force for heat transfer during the on-gas cycle. The result is an increase in the energy leaving the stove through the waste gas. Table 5.2 compares the average and standard deviation of the final mixer valve position at the end of the on-blast cycle with and without control. This comparison was made at a blast air flow of 108 m^3/sec and a final mixer valve position target of 7 %

Time Period	*Avg. Hot Blast Flow* m^3/sec	*Avg. Hot Blast Temperature* °C	*Avg. Mixed Gas Heating Value* BTU/ft^3	*Avg. Natural Gas Usage* m^3/Gcal
Base Period	107.27	1256.53	145.75	65.72
Test Period	116.47	1262.20	142.39	62.45
Change	+8.58 %	+0.45 %	-2.30 %	-4.98 %

Table 5.1: Controller performance comparison based on volume of natural gas used per gigacalorie of heat transferred to the blast air.

for Stove #1. This comparison was performed at similar average blast air temperature targets, blast air temperature target changes, and mixed fuel gas heating values.

	Final Mixer Valve Position	
Operation	*Average*	*Deviation*
No control	13.2	6.2
Control	8.7	2.7
Change	-34 %	-56 %

Table 5.2: Final mixer valve comparison.

The performance of the controller can also be judged by comparing the amount of total blast heat supplied by natural gas, and the amount of heat supplied by natural gas to the blast air per natural ton of hot metal produced by the blast furnace. This comparison is shown in Table 5.3. In this table, the base period consisted of the full months of August and October and the first half of the month of November in 1998. The first test period was the full month of December 1998, the second test period was the month of February 1999, and the third test period was March 1999. The average test period is the average over the three test months. Note that the controller achieved a 5.2% reduction in the amount of total blast heat supplied by natural gas, and a 6.7% reduction in the heat supplied by natural gas per ton of hot metal produced, with a production rate increase of 6.4% and a top gas heating value decrease of 1.6%.

Operating Parameters	*Base Period*	*First Test Period*	*Second Test Period*	*Third Test Period*	*Average Test Period*
Operating Conditions					
Production Rate (NT/day)	9925	10659	10659	10360	10559
Top Gas Heating Value (BTU/ft^3)	86.6	87	88	89	88
Top Gas Flow Rate (m^3/day)	3178028	3649055	3505018	3481465	3545179
Natural Gas Heating Value (BTU/ft^3)	1017	1021	1022	1023	1022
Natural Gas Flow Rate (m^3/day)	216061	219427	214990	205213	213210
Mixed Gas Heating Value (BTU/ft^3)	145.8	140	142	141	141
Mixed Gas Flow Rate (m^3/day)	3394089	386482	3720008	3686678	3758389
Stove Performance					
Blast Heat (Kcal/D)	3.29e+9	3.51e+9	3.49e+9	3.34e+9	3.44e+9
Heat supplied from Natural Gas (Kcal/D)	1.95e+9	1.99e+9	1.96e+9	1.87e+9	1.94e+9
Natural Gas Heat per Blast Heat (%)	59.4	56.9	56.1	56.0	56.3
Natural Gas Vol per Blast Heat (m^3/Gcal)	65.7	62.6	61.7	61.5	61.9
Reduction in NG Heat per Blast Heat		**4.3%**	**5.6%**	**5.8%**	**5.2%**
Reduction in NG Heat (Gcal/yr)		5.1e+4	6.7e+4	6.9e+4	6.3e+4
Reduction in NG Vol per Blast Heat		**4.7%**	**6.1%**	**6.4%**	**5.7%**
Blast Furnace Performance					
Heat supplied from Mixed Gas (BTU/NT)	1.76e+6	1.79e+6	1.75e+6	1.77e+6	1.77e+6
Heat supplied from Mixed Gas (BTU/yr)	6.38e+12	6.98e+12	6.81e+12	6.70e+12	6.83e+12
Heat supplied from Natural Gas (BTU/NT)	7.81e+5	7.42e+5	7.28e+5	7.15e+5	7.29e+5
Heat supplied from Natural Gas (BTU/yr)	2.83e+12	2.89e+12	2.83e+12	2.71e+12	2.81e+12
Reduction in Heat Supplied from NG/NT		**5.0%**	**6.8%**	**8.4%**	**6.7%**

Table 5.3: Controller performance comparison based on amount of heat generated by natural gas both per quantity of total blast heat and per ton of hot metal produced.

Chapter 6

Blast Furnace Stove Model

A blast furnace stove is modeled by assuming that the gas channels in the checkers can be represented as thick walled tubes in which the gas flows through the center of the tubes heating or cooling the wall material. The outside walls of the tubes are assumed to be perfectly insulated. The tubes are divided into five zones each containing the wall material that corresponds to the checkers in the zone. The top of the stove is comprised of silica checkers. The lower zones are comprised of mullite, super duty, high duty, and high alumina checkers respectively. The stove model geometry is shown in Figure 6.1.

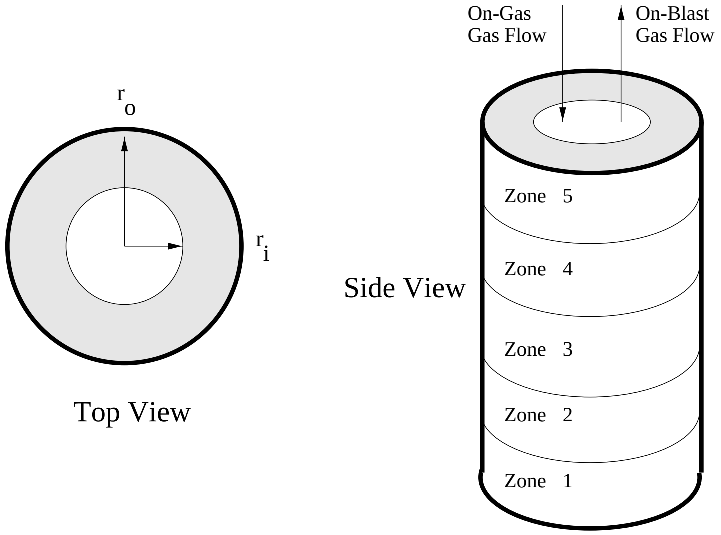


Figure 6.1: Stove model geometry.

The number of tubes used to represent the stove is the number of gas channels in the checkers. This value, denoted by N_c, is specified by the stove manufacturer and is the same for each zone. The radius of the gas channel in the tube, r_i, is one half of the average hydraulic diameter of the gas channels in the corresponding checker. The outside radius of the tube, r_o, is determined from the total number of gas channels, N_c, the hydraulic diameter of the gas channel, D_h, the total mass and density of the checkers, m_n and ρ_n, and the length of the zone, L_n, in which the subscript n refers to the zone. These values are specified by the stove manufacturer for each zone.

$$r_o = \sqrt{\frac{m_n}{\pi \rho_n N_c L_n} + \frac{D_h^2}{4}} \tag{6.1}$$

Table 6.1 presents the tube geometry used to represent each zone.

Zone	r_i cm	r_o cm	L_n m	ρ_n g/cm^3
1	3.68	4.28	1.52	2.48
2	3.68	4.29	15.85	2.17
3	3.68	4.27	3.96	2.34
4	3.15	3.95	3.96	2.48
5	3.15	3.95	10.82	1.83

Table 6.1: Tube geometry for each zone.

6.1 Gas Model

The blast air and waste gas are modeled by an energy balance over the gas flowing through a single tube. Assuming the blast air and waste gas are ideal gases, no radial variation of the gas temperature, and no heat conduction in the gas in the axial direction results in the following partial differential equation [4]

$$\rho_g C_{p,g} \left[\frac{\partial T_g}{\partial t} + v_g \frac{\partial T_g}{\partial z} \right] + v_g \frac{\partial P}{\partial z} = \frac{4h}{D_h} \left(T_w - T_g \right) \tag{6.2}$$

in which T_g is the gas temperature, v_g is the gas velocity, ρ_g is the gas density, $C_{p,g}$ is the gas heat capacity, P is the pressure, h is the gas-solid heat transfer coefficient, D_h is the hydraulic diameter of the gas channel, and T_w is the solid wall temperature.

The heat capacity of each of the components in the blast air and waste gas is determined by interpolating functions of the heat capacity vs. temperature data contained in the National Bureau of Standards Publication #564 on the thermal properties of gases [13]. The interpolating function for each component is shown in Table 6.2. The density of the gas is determined assuming an ideal gas

$$\rho_g = \frac{M_g P}{R T_g} \tag{6.3}$$

in which M_g is the average molecular weight. For the on-blast cycle, the average molecular weight of air, corrected for the moisture and oxygen injected into the blast air, is used to compute the gas density. For the on-gas cycle, the average molecular weight is determined from the computed waste gas composition.

Gas	$C_{p,g}(T_g)$ cal/gm-K
CO_2	$0.3628 - 77.0/T_g + 8623.0/T_g^2$
H_2O	$1.267 - 34.65/\sqrt{T_g} + 378.4/T_g$
N_2	$0.2164 + 7.299e–5T_g - 1.361e–8T_g^2 - 947.2/T_g^2$
O_2	$0.3003 - 46.64/T_g + 6541.0/T_g^2$
Air	$0.2566 - 1.211e–4T_g + 2.778e–7T_g^2 - 1.765e–10T_g^3 + 3.773e–14T_g^4$

Table 6.2: Gas heat capacity (T_g in deg K).

The gas velocity in the tubes is determined from the stove inlet mass flow rate, $\dot{m}_{\text{in}}$, the gas density, and the cross-sectional area of the tube assuming a uniform gas flow distribution through the channels in the checkers.

$$v_g = \frac{4\dot{m}_{\text{in}}}{\pi \rho_g N_c D_h^2} \tag{6.4}$$

For the on-gas cycle, the inlet mass flow rate is determined from the combustion air and fuel gas flow rates. For the on-blast cycle, the blast air mass flow rate through the stove varies as the cycle progresses depending on the amount of the total blast air flow that is diverted to achieve the desired blast air temperature. The mass flow rate of blast air through the stove can be determined from an energy balance in which T_g^{targ} is the desired blast air temperature, T_g^{in} is the blast air inlet temperature, T_g^{out} is the temperature of the blast air exiting the stove before mixing, and $\dot{m}_{\text{total}}$ is the total blast air mass flow rate.

$$\dot{m}_{\text{in}} = \left(\frac{\int_{T_g^{\text{in}}}^{T_g^{\text{targ}}} C_{p,g}\, dT}{\int_{T_g^{\text{in}}}^{T_g^{\text{out}}} C_{p,g}\, dT} \right) \dot{m}_{\text{total}} \tag{6.5}$$

The inlet pressure during the on-gas and on-blast cycles is measured. Assuming that the pressure drop across the stove is due to frictional losses that can be modeled in the same manner as friction losses in a pipe results in the following relationship [4]

$$\frac{\Delta P}{\Delta z} = \frac{f}{D_h} \frac{\rho_g v_g^2}{2}, \qquad f = f_w + f_c \tag{6.6}$$

in which the wall friction factor, f_w, is determined from the following correlation [16].

$$\frac{1}{\sqrt{f_w}} = 3.4841 - 1.7372 \ln \left[\frac{2\epsilon}{D_h} + \frac{42.5}{\mathrm{Re}^{0.9}} \right] \tag{6.7}$$

The equivalent sand roughness of the checker material, ϵ, is taken to be similar to that of concrete. A value of $\epsilon = 0.02$ was determined from a plot of equivalent sand roughness for commercial pipe surfaces [3].

There is also a contribution to the friction factor from expansion/contraction losses due to the slight taper of the gas channels in each checker. The top diameter of the gas channel is 4 mm less than the bottom diameter. This design is intended to increase the heat transfer in the stove. The pressure drop due to this loss at each of the checker interfaces is determined by

$$\Delta P = K \frac{\rho_g v_g^2}{2} \tag{6.8}$$

in which $K = 0.07$ is the expansion loss coefficient for the on-gas cycle and $K = 0.06$ is the contraction loss coefficient for the on-blast cycle. These values were determined from the area ratio between the gas channels at the top and bottom of the checkers [10]. The equivalent friction factor due to expansion/contraction losses at the checker interfaces is

$$f_c = \frac{K D_h}{L_c} \tag{6.9}$$

in which $L_c = 15.3$ cm is the average checker length.

6.2 Heat Transfer Coefficient Model

The heat transfer coefficient is comprised of a convective contribution for the on-blast cycle and both a convective and radiation contribution for the on-gas cycle.

$$\text{on-blast} \;\; h = h_c, \qquad \text{on-gas:} \;\; h = h_c + h_r \tag{6.10}$$

The convective heat transfer contribution, h_c, is determined from a correlation for rough pipes by Bhatti and Shah [3]

$$\mathrm{Nu} = \frac{(f/2)(\mathrm{Re} - 1000)\mathrm{Pr}}{1 + \sqrt{f/2}\left[(17.42 - 13.77\mathrm{Pr}_t^{0.8})\sqrt{\mathrm{Re}_\epsilon} - 8.48\right]} \tag{6.11}$$

with the Nusselt, Reynolds, and Prandtl numbers defined as follows

$$\mathrm{Nu} \;\; = \;\; h_c D_h / k_g$$

$$\begin{aligned}\mathrm{Re} &= D_h v_g \rho_g / \mu_g \\ \mathrm{Pr} &= C_{p,g} \mu_g / k_g\end{aligned}$$

in which μ_g is the gas viscosity and k_g is the gas thermal conductivity and the values of the turbulent Prandtl number, Pr_t, and roughness Reynolds number, Re_ϵ computed from the following relationships:

$$\mathrm{Pr}_t = 1.01 - 0.99\mathrm{Pr}^{0.36} \tag{6.12}$$

$$\mathrm{Re}_\epsilon = \frac{\epsilon \rho_g v_g}{\mu_g}\sqrt{\frac{f}{2}}. \tag{6.13}$$

This correlation is valid for Reynolds numbers greater than 2300 and Prandtl numbers greater than 0.5. The Reynolds number ranges from 2500 to 5500 during the on-gas cycle and from 4500 to 9500 during the on-blast cycle, and the Prandtl number ranges from 0.6 to 0.8 for typical stove operation.

The viscosity and thermal conductivity of the blast air and waste gas are required to determine the Nusselt, Reynolds, and Prandtl numbers for the correlation in Equation 6.11. These properties as a function of temperature at a pressure of 1 atm for air and each of the waste gas components are determined by interpolating functions of the data contained in the National Bureau of Standards Publication #564 on the thermal properties of gases [13]. The viscosity interpolating functions are presented in Table 6.3. The thermal conductivity interpolating functions are presented in Table 6.4.

Gas	$\mu_g(T_g)$ cm/gm-sec
CO_2	$-1.225e{-}4 + 3.107e{-}8T_g + 1.518e{-}5\sqrt{T_g}$
H_2O	$-7.962e{-}4 + 2.963e{-}5\sqrt{T_g} + 6.767e{-}3/\sqrt{T_g}$
N_2	$-1.044e{-}4 - 2.656e{-}8T_g + 1.679e{-}5\sqrt{T_g}$
O_2	$-8.908e{-}5 + 5.972e{-}8T_g + 1.602e{-}5\sqrt{T_g}$
Air	$-5.769e{-}5 + 1.534e{-}5\sqrt{T_g} - 4.031e{-}4/\sqrt{T_g}$

Table 6.3: Gas viscosity (T_g in deg K).

The oxygen and moisture injected into the blast air are each less than 5 mol%. Therefore, these components are ignored in the viscosity and thermal conductivity determination for the blast air. The pressure correction from 1 atm to the blast pressure

Gas	$k_g(T_g)$ cal/cm-sec-K
CO_2	$3.749e\text{–}5 + 3.559e\text{–}7 * T_g -$ $6.050e\text{–}6\sqrt{T_g}$
H_2O	$-3.977e\text{–}4 + 1.514e\text{–}5\sqrt{T_g} +$ $3.123e\text{–}3/\sqrt{T_g}$
N_2	$-5.330e\text{–}5 - 8.804e\text{–}9T_g +$ $6.842e\text{–}6\sqrt{T_g}$
O_2	$-3.683e\text{–}5 + 5.559e\text{–}8T_g +$ $4.863e\text{–}6\sqrt{T_g}$
Air	$-8.278e\text{–}5 + 7.445e\text{–}6\sqrt{T_g} +$ $2.882e\text{–}4/\sqrt{T_g}$

Table 6.4: Gas thermal conductivity (T_g in deg K).

of approximately 5 atm is estimated to be less than 1 % for the blast air viscosity and less than 4 % for the blast air thermal conductivity [35] and are also ignored.

The viscosity of the waste gas is determined from the pure component viscosities using the method of Wilke [35]

$$\mu_m = \sum_{i=1}^{n} \frac{y_i \mu_i}{\sum_{j=1}^{n} y_j \phi_{i,j}} \tag{6.14}$$

in which μ_m is the viscosity of the mixture, y_i is the mole fraction of component i, μ_i is the viscosity of pure component i, and the interaction parameter $\phi_{i,j}$ is computed by

$$\phi_{i,j} = \frac{\left(1 + (\mu_i/\mu_j)^{1/2}(M_j/M_i)^{1/4}\right)^2}{\sqrt{8(1 + M_i/M_j)}} \tag{6.15}$$

with M_i the molecular weight of component i. The thermal conductivity of the waste gas is determined from the pure component thermal conductivities using the method of Mason and Saxena [35]

$$k_m = \sum_{i=1}^{n} \frac{y_i k_i}{\sum_{j=1}^{n} 0.85 y_j \phi_{i,j}} \tag{6.16}$$

in which k_m is the mixture thermal conductivity, y_i is the mole fraction of component i, k_i is the thermal conductivity of pure component i, and $\phi_{i,j}$ is the interaction parameter shown in Equation 6.15.

Radiation from carbon dioxide and water vapor in the waste gas to the tube walls can contribute up to twenty percent of the heat transfered for the on-gas cycle. Since the concentration of these components in the blast air is small, and they are the only components that radiate appreciable energy in the operating temperature range of the stove, radiation is neglected for the on-blast cycle. Radiation from the walls down the length of the tube is not significant due to the unfavorable view factor, and it is also neglected.

The radiative heat transfer coefficient, h_r, can be determined based on the analysis presented by Hottell [14]

$$h_r = \sigma \epsilon_s \frac{(\alpha_h + \alpha_c)T_w^4 - (\epsilon_h + \epsilon_c)T_g^4}{T_w - T_g} \tag{6.17}$$

in which σ is the Stephan-Boltzmann constant, ϵ_s is the emissivity of the tube wall, T_w is the tube wall temperature, T_g is the gas temperature, ϵ_h is the water vapor emissivity at T_g, ϵ_c is the carbon dioxide emissivity at T_g, α_h is the water vapor absorptivity at T_w, and α_c is the carbon dioxide absorptivity at T_w. The emissivity and absorptivity values as a function of temperature and pressure for water vapor and carbon dioxide are determined from the following interpolating functions of the chart data presented in [14]

$$
\begin{aligned}
P_c L_\epsilon &= 0.059 M_{\mathrm{CO}_2} r_i \\
\epsilon_c &= \begin{cases} f_c(T_g) - \exp\left(a_c(T_g)(0.04 - P_c L_\epsilon)\right), & P_c L_\epsilon < 0.04 \\ f_c(T_g) + \exp\left(a_c(T_g)(P_c L_\epsilon - 0.04)\right), & P_c L_\epsilon \geq 0.04 \end{cases} \\
P_c L_\alpha &= 0.059 M_{\mathrm{CO}_2} r_i \left(\frac{T_w}{T_g}\right) \\
\alpha_c &= \begin{cases} \left[f_c(T_w) - \exp\left(a_c(T_w)(0.04 - P_c L_\alpha)\right)\right] \left(\frac{T_g}{T_w}\right)^{0.65}, & P_c L_\alpha < 0.04 \\ \left[f_c(T_w) + \exp\left(a_c(T_w)(P_c L_\alpha - 0.04)\right)\right] \left(\frac{T_g}{T_w}\right)^{0.65}, & P_c L_\alpha \geq 0.04 \end{cases} \\
f_c(T) &= 0.614 - 3.606 \times 10^{-4} T + 7.081 \times 10^{-8} T^2 - 334.9/T + 6.713 \times 10^4/T^2 \\
a_c(T) &= 468.6 - 0.5737T + 9.526 \times 10^{-5} T^2 - 6.274 \times 10^5/T + 1.294 \times 10^8/T^2 \\
P_h L_\epsilon &= 0.059 M_{\mathrm{H}_2} r_i \\
\epsilon_h &= \begin{cases} f_h(T_g) - \exp\left(a_h(T_g)(0.02 - P_h L_\epsilon)\right), & P_h L_\epsilon < 0.02 \\ f_h(T_g) + \exp\left(a_h(T_g)(P_h L_\epsilon - 0.02)\right), & P_c L_\epsilon \geq 0.02 \end{cases} \\
P_h L_\alpha &= 0.059 M_{\mathrm{H}_2} r_i \left(\frac{T_w}{T_g}\right) \\
\alpha_h &= \begin{cases} \left[f_h(T_w) - \exp\left(a_h(T_w)(0.02 - P_h L_\alpha)\right)\right] \left(\frac{T_g}{T_w}\right)^{0.45}, & P_h L_\alpha < 0.02 \\ \left[f_h(T_w) + \exp\left(a_h(T_w)(P_h L_\alpha - 0.02)\right)\right] \left(\frac{T_g}{T_w}\right)^{0.45}, & P_h L_\alpha \geq 0.02 \end{cases}
\end{aligned}
$$

$$\begin{aligned} f_h(T) &= 5.965 \times 10^{-2} - 5.701 \times 10^{-5} T + 1.486 \times 10^{-8} T^2 \\ a_h(T) &= -693.2 - 33.75T - 5.033 \times 10^{7}/T^2 \end{aligned}$$

in which M_{CO_2} and M_{H_2} are the mole fractions of carbon dioxide and water vapor, respectively, in the waste gas, T_g is the gas temperature, and T_w is the solid temperature at the wall.

For the operating range of the stove, the chart values were determined from experimental measurements of total emission and have been confirmed by further theoretical and experimental studies ([41], [40]). The tube wall emissivity is taken as that for refractory brick and is estimated to be 0.8 for each of the checker materials ([39], [40]).

A simpler radiative heat transfer coefficient calculation can be obtained from an interpolating function of the gas temperature plots presented by Butterfield *et al.* [38] based on the approximation by Hausen and Binder [12]. These plots assume a fixed waste gas composition and operating temperature range of the stove that vary significantly from the Ispat Inland operation. Extension of the operating temperature range and correction for the waste gas composition can be obtained using Equation 6.17; however, the slight decrease in the computational requirements does not justify the restrictions of this method.

6.3 Combustion Temperature Model

The temperature of the incoming blast air during the on-blast cycle is available from a process measurement. This value is used as the inlet temperature boundary condition for Equation 6.2 during the on-blast cycle. The temperature of the combustion gases entering the top of the stove during the on-gas cycle is not measured. This value is determined by an energy balance over the fuel and combustion air and is used as the inlet temperature boundary condition during the on-gas cycle.

The calculation of the combustion temperature for the on-gas cycle is performed by an adiabatic energy balance over the combustion zone. The fuel gas used in the stove is a mixture of top gas from the blast furnace and natural gas. The composition of the natural gas is taken as the nominal composition specified by the natural gas supplier. The top gas composition is available on a dry basis from a process analyzer. Since the top gas is scrubbed to remove particulates before being sent to the stoves, it is assumed to be saturated with water. The vapor pressure of water in the top gas is determined using the Antoine equation

$$\log(P_{\mathrm{H}_2\mathrm{O}}) = 0.154 - 32.258/(T - 45.150) \tag{6.18}$$

in which $P_{\mathrm{H}_2\mathrm{O}}$ is in psi and the top gas temperature T is in K. The water content of the combustion air is determined from an on-line relative humidity measurement. Representative natural gas, top gas, and combustion air compositions are shown in Table 6.5.

Stream	*Component*	*Mole fraction*
Natural Gas	CH_4	0.805
	C_2H_6	0.182
	N_2	0.013
Top Gas	CO	0.240
	H_2	0.034
	CO_2	0.226
	N_2	0.459
	O_2	0.002
	H_2O	0.039
Air	O_2	0.206
	N_2	0.774
	H_2O	0.020

Table 6.5: On-gas inlet stream compositions.

The ratio of natural gas to top gas in the mixed fuel gas stream sent to the burners is set by a mixed gas heating value controller. The heating value of the top gas typically ranges from 80–90 BTU/cuft. The typical target heating value of the mixed gas ranges from 145–160 BTU/cuft resulting in a natural gas fraction of the mixed fuel gas on the order of 7 vol%. Since the natural gas fraction cannot be independently set for each stove, the mixed gas heating value is not varied during on-gas cycles. The combustion air flow rate to the burners is set by an excess oxygen controller that maintains the excess oxygen in the waste gas at a nominal target of 1.2 vol%.

The natural gas fraction of the mixed gas in the model is computed from the compositions and the heating value target. The combustion air to mixed gas ratio in the model is computed using the waste gas excess oxygen target. Stove operating experience indicates that as the mixed gas flow rate and/or heating value increase, a small amount of combustible components appear in the waste gas. This incomplete combustion occurs because the burners were designed for coke oven gas enrichment and produce less efficient natural gas combustion at high fuel gas flow rates and/or enrichments. An adjustable parameter that specifies the fraction of natural gas that undergoes incomplete combustion is included in the model to account for this effect. The value of this parameter is based on the measurement of the combustible components from an analyzer in the waste gas stream.

The combustion temperature is obtained from the energy balance

$$\int_{T_{ref}}^{T_g^{\text{in}}} \left(\sum_{i=1}^{n_{\text{in}}} x_i C_{p,g,i} \, dT \right) + \sum_{i=1}^{n_{\text{in}}} x_i \Delta H_{v,i} \;=\; \int_{T_{ref}}^{T_g^{\text{comb}}} \left(\sum_{i=1}^{n_{\text{out}}} x_i C_{p,g,i} \, dT \right) \qquad (6.19)$$

in which T_g^{comb} is the combustion temperature, T_g^{in} is the inlet gas temperature, $T_{ref} = 25$ °C is the reference temperature, x_i is the mass fraction for component i corrected for the incomplete combustion fraction, $C_{p,g,i}$ is the gas heat capacity for component i, $\Delta H_{v,i}$ is the lower heating value for component i, $n_{\text{in}} = 8$ is the number of components in the fuel gas/combustion air inlet stream, and $n_{\text{out}} = 4$ is the number of components in the waste gas outlet stream. The heat capacities as a function of temperature and the lower heating values are taken from [8]. Since the combustible components in the waste gas are present at low levels, they are not considered in the energy balance. Brents Method [34] is used to determine the combustion temperature from the equality in Equation 6.19.

6.4 Solid Model

An energy balance over a single tube results in the following partial differential equation for the tube wall material [4]

$$\rho_s C_{p,s} \frac{\partial T_s}{\partial t} - \frac{1}{r}\frac{\partial}{\partial r}\left(r k_s \frac{\partial T_s}{\partial r} \right) - k_s \frac{\partial^2 T_s}{\partial z^2} = 0 \tag{6.20}$$

in which T_s is the temperature, ρ_s is the density, $C_{p,s}$ is the heat capacity, and k_s is the thermal conductivity of the solid. The density for each of the five checker materials is specified by the stove manufacturer. The heat capacity and thermal conductivity for each of the five materials are specified as polynomial functions of temperature by the stove manufacturer. Heat losses to the environment are neglected.

Assuming heat conduction between adjacent zones, the boundary conditions for Equation 6.20 in the axial direction are

$$k_n \left. \frac{dT_s^n}{dz} \right|_{z=L_n} = k_{n-1} \left. \frac{dT_s^{n-1}}{dz} \right|_{z=0} \tag{6.21}$$

in which k_n is the thermal conductivity of the checker material, T_s^n is the solid temperature, and L_n is the length of zone n. Assuming no heat conduction at the top of zone 5 and at the bottom of zone 1 results in the following axial boundary conditions for those zones.

$$\left. \frac{dT_s^5}{dz} \right|_{z=0} = \left. \frac{dT_s^1}{dz} \right|_{z=L_1} = 0 \tag{6.22}$$

Assuming the outside tube wall is perfectly insulated, the boundary conditions in the radial position are the following.

$$\left. \frac{dT_s}{dr} \right|_{r=r_o} = 0 \tag{6.23}$$

$$\left.\frac{dT_s}{dr}\right|_{r=r_i} = \frac{2h}{k_s(r_o^2/r_i^2-1)}\left(T_g - T_s|_{r=r_i}\right)$$

A simplification of Equation 6.20 is to assume that the checkers behave as a lumped parameter thermal system in the radial direction. In this case, there is no radial variation of the solid temperature, and the single tube energy balance is expressed by the following two-dimensional partial differential equation [4]

$$\rho_s C_{p,s}\frac{\partial T_s}{\partial t} - k_s\frac{\partial^2 T_s}{\partial z^2} = \frac{2h}{r_i(r_o^2/r_i^2-1)}(T_g - T_s) \tag{6.24}$$

in which the axial boundary conditions are given in Equations 6.21 and 6.22.

The characteristic time for heat conduction in the checkers is determined by the following ratio

$$\tau = \frac{(r_o - r_i)^2}{\alpha} \tag{6.25}$$

in which α is the thermal diffusivity of the checker material

$$\alpha = \frac{k_s}{\rho_s C_{p,s}}. \tag{6.26}$$

This value is on the order of 3 minutes for the checker materials in zones 1, 2, 3, and 5 at the normal operating temperature range of the stove. It is on the order of 5 minutes for checker material in zone 4. Since these values are an order of magnitude less than the stove cycle times, the lumped parameter model in the radial direction in Equation 6.24 is expected to adequately represent the thermal state of the stove.

Figures 6.2 and 6.3 present the radial temperature profiles for the tube shown in Figure 6.1 determined from the solution of Equation 6.20. Figure 6.2 shows the temperature variation in the radial direction from the insulated wall temperature at the end of a typical on-gas cycle. Figure 6.3 shows the temperature variation in the radial direction from the gas-solid wall temperature at the end of a typical on-blast cycle.

The profiles in these figures represent the radial temperature variation at each axial node location along the length of the stove. As shown in these figures, the maximum temperature variation in the radial direction is on the order of 10 °C for a small fraction of the stove length. The majority of the stove length shows less than a 5 °C variation for the on-gas cycle and less than a 7 °C variation for the on-blast cycle. The simulated axial temperature profile obtained from the simplified model in Equation 6.24 is essentially identical to the radial averaged axial temperature profile obtained from Equation 6.20. Comparison of the simulated energy transferred and removed from the stove for both solid models shows no significant differences.

These results confirm the conclusion based on the characteristic heat conduction times for the checker materials that a lumped radial model is adequate. The advantage

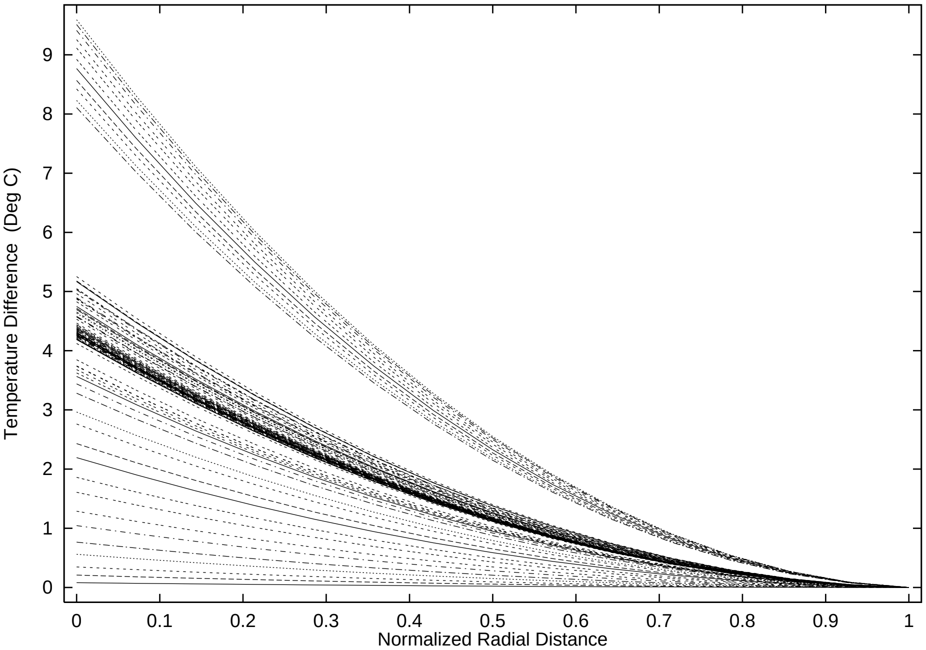


Figure 6.2: On-gas radial temperature variation.

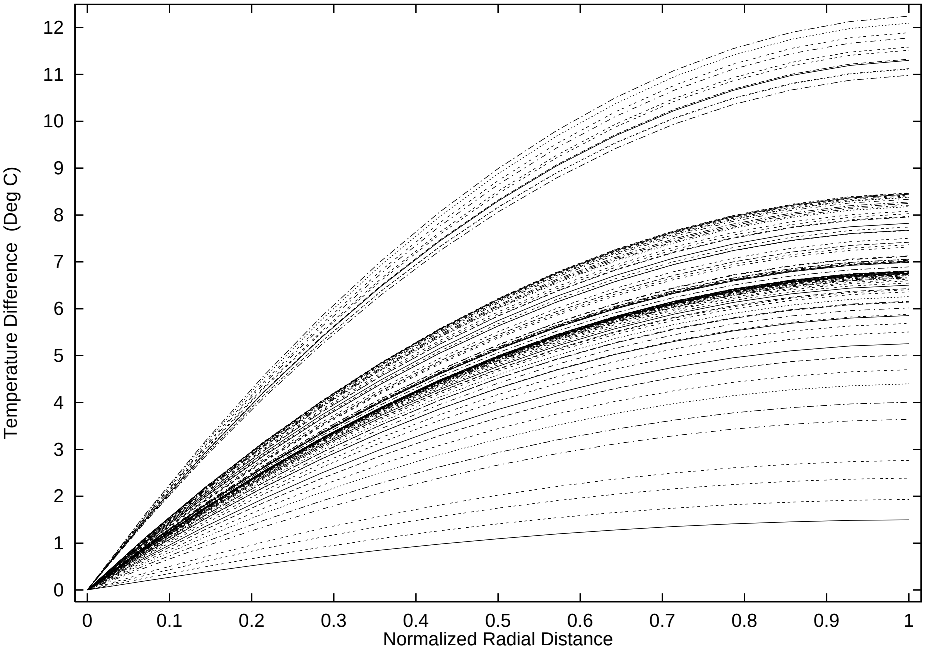


Figure 6.3: On-blast radial temperature variation.

of the lumped radial model is approximately a 75 % reduction in the computational time required using the two-dimensional partial differential equation in Equation 6.24 as opposed to the three-dimensional partial differential equation in Equation 6.20. For these reasons, the simplified solid model is used for the model-based control.

It should also be noted that the effect of axial conduction in the solid is not significant under the conditions studied in this work. Simulations without axial conduction result in essentially identical temperature profiles and trajectories as those in which axial conduction is considered. Based on these observations, the *a priori* neglect of axial conduction in previous work appears to be justified.

6.5 Pressurization and Blow-off Cycles

For blast air to enter the blast furnace, it must be above the blast furnace pressure. In order to achieve the necessary blast air pressure, each stove is pressurized with air prior to the on-blast cycle. This is referred to as the pressurization cycle and is on the order of five minutes for the No. 7 blast furnace stoves. The energy removed from the solid during the pressurization cycle is modeled by assuming the solid and gas temperatures are in local thermal equilibrium along the tube length after pressurization.

The equilibrium temperature profile, $T_{eq}(z)$, is determined by equating the energy removed from the solid tube wall to the energy necessary to heat the pressurization air. An energy balance over each axial node along the tube length results in the following equations

$$r_i^2 \int_{T_g^{\text{in}}}^{T_{eq}} \rho_g C_{p,g}\, dT = (r_o^2 - r_i^2)\rho_s \int_{T_{eq}}^{T_s^{\text{init}}} C_{p,s}\, dT \tag{6.27}$$

in which T_{eq} is the equilibrium solid and gas temperature at the axial node, T_g^{in} is the inlet pressurization air temperature, and T_s^{init} is the initial solid temperature determined from the previous on-gas cycle. The value of the equilibrium temperature is that which satisfies the equality in Equation 6.27. It is determined using Brents Method [34], a single-variable search technique, performed at each axial node.

After an on-blast cycle is complete, the stove is returned to atmospheric pressure during the blow-off cycle. This cycle is also on the order of five minutes for the No. 7 blast furnace stoves. In this cycle, the pressurized air remaining in the stove is vented to atmosphere. The vented air leaves the combustion chamber of the stove by flowing through the checkerwork. The energy lost from the stove during the blow-off cycle is modeled by considering the heat transfered from the tube wall to the air vented from the combustion chamber.

Assuming an ideal gas, the mass flow rate of the air leaving the stove is [26]

$$\dot{m}_{\text{vent}} \;=\; K_v m_i \sqrt{\frac{2}{\gamma - 1}} \left(\frac{P^*}{P}\right)^{\frac{1}{\gamma}} \left(\frac{m}{m_i}\right)^{\frac{\gamma+1}{2}} \sqrt{1 - \left(\frac{P^*}{P}\right)^{\frac{\gamma-1}{\gamma}}} \tag{6.28}$$

in which γ is the heat capacity ratio C_p/C_v, P is the stove pressure, P^* is the discharge pressure, m is the mass of gas in the stove, m_i is the initial mass of gas in the stove, and $K_v = 0.5$ is the discharge coefficient through the vent valve. The discharge coefficient is determined by matching the computed and actual time to vent the stove. The minimum ratio of P^*/P is 0.538 due to sonic flow through the vent valve. The temperature of the gas leaving the combustion zone is determined by assuming isentropic expansion of an ideal gas and is computed by the following differential:

$$\frac{dP}{P} = \frac{\gamma}{\gamma - 1}\frac{dT}{T}. \tag{6.29}$$

Since the the thermal mass of the checkers is so much greater than that of the blast air involved in the pressurization and blow-off cycles, the effect on the solid temperature is expected to be small. This conclusion is confirmed by the results of the preceding calculations. Although there are transient effects during these cycles, examination of the stove temperature data also confirms this conclusion. Therefore, the pressurization and blow-off cycles are not considered in the stove model in order to reduce the computational requirement.

6.6 Model Reduction

A detailed heat transfer model for model-based control of the hot blast stoves is presented in this section. Variation in the physical and operating properties, radiative heat transfer, radial and axial conduction in the checkers, and the transitions between cycles are explicitly considered in the development of this model. Since model-based control requires a large number of successive model runs on-line, the computational requirement is the most important consideration for implementation. Therefore, a number of simplifications to the blast furnace stove model have been made to reduce the computational requirements such that the model can be used as part of an implementable nonlinear, model-based control algorithm.

Model reduction includes modeling the system as a series of thick-walled tubes as opposed to considering the actual checker geometry, neglecting radial and axial conduction in the solid, neglecting the effects of the pressurization and blow-off cycles, neglecting the effect of the injected oxygen and moisture on the convective heat transfer coefficient, and neglecting radiation heat transfer during the on-blast cycle. These assumptions made to reduce the physical model, and the computational requirement, were selected to minimize the effect on the control-relevant model results such as the predicted temperature profile, blast air stove mass flow rate, and energy content. The justification for each assumption and simplification is based on a comparison between these model results.

Chapter 7

Stove Model Numerical Solution Technique

For computational purposes, Equations 6.2 and 6.24 are scaled to make all the quantities dimensionless and then are discretely approximated. Equation 6.2 represents the change in energy over time for the gas, and Equation 6.24 represents the energy change of the solid. The terms in these equations describe the convection of heat between the gas and the solid in the direction perpendicular to the gas flow in the channels and convection in the gas parallel to the gas flow. Other effects are neglected. A diagram depicting the mass and energy flow in the stove is shown in Figure 7.1. Note that the directions of both heat and mass flow during the cooling cycle are opposite to those occurring during the heating cycle and that a cylindrical coordinate system, centered at the middle of the gas channel in the brick, is used in this diagram.

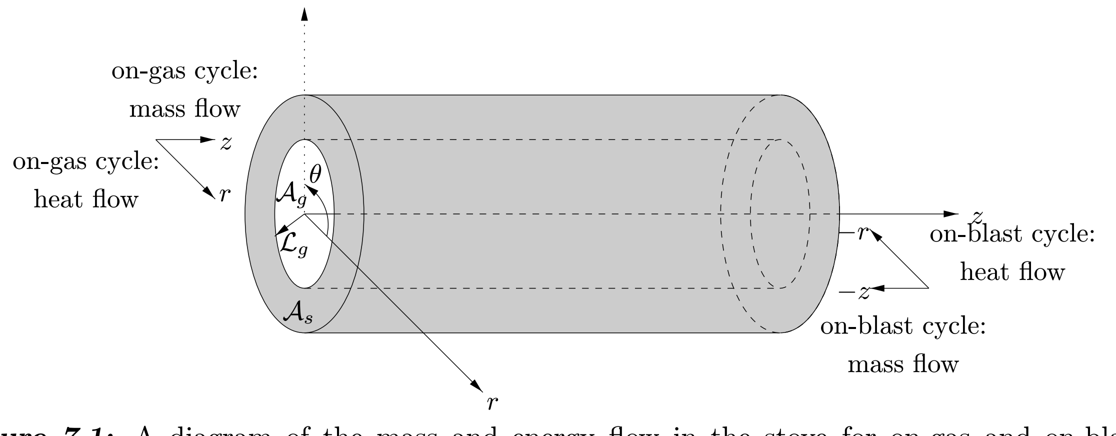


Figure 7.1: A diagram of the mass and energy flow in the stove for on-gas and on-blast operation.

7.1 Dimensionless Equations

Scaling is performed to prevent the choice of units from artificially altering the relative sizes of each term in the differential equations. Define the dimensionless scaling variables

$$\begin{array}{lllll}
\hat{T}_g = \dfrac{T_g - T_i}{T_0}, & \hat{T}_s = \dfrac{T_s - T_i}{T_0}, & \hat{P} = \dfrac{P}{P_{g0}}, & \hat{t} = \dfrac{t}{\tau_{g0}}, & \hat{h} = \dfrac{h}{h_0}, \\
\tau_{g0} = \dfrac{\rho_{g0}\,\mathcal{C}_{g0}\,\mathcal{A}_{g0}}{h_0\,\mathcal{L}_{g0}}, & \tau_{s0} = \dfrac{\rho_{s0}\,\mathcal{C}_{s0}\,\mathcal{A}_{s0}}{h_0\,\mathcal{L}_{g0}}, & \hat{\rho}_g = \dfrac{\rho_g}{\rho_{g0}}, & \hat{\rho}_s = \dfrac{\rho_s}{\rho_{s0}}, & \\
P_{g0} = \rho_{g0}\,\mathcal{C}_{g0}\,T_0, & \dot{\hat{m}}_g(\hat{t}) = \dfrac{\dot{m}_g(t)}{\rho_{g0}\,v_{g0}\,\mathcal{A}_{g0}}, & \hat{\mathcal{A}}_g = \dfrac{\mathcal{A}_g}{\mathcal{A}_{g0}}, & \hat{\mathcal{A}}_s = \dfrac{\mathcal{A}_s}{\mathcal{A}_{s0}}, & \\
\hat{z} = \dfrac{z}{\tau_{g0}\,v_{g0}}, & \hat{\mathcal{C}}_{p,g} = \dfrac{\mathcal{C}_{p,g}}{\mathcal{C}_{g0}}, & \hat{\mathcal{C}}_{p,s} = \dfrac{\mathcal{C}_{p,s}}{\mathcal{C}_{s0}}, & \hat{\mathcal{L}}_g = \dfrac{\mathcal{L}_g}{\mathcal{L}_{g0}}, &
\end{array} \tag{7.1}$$

in which v_0 is the reference gas velocity, τ_{g0} is the characteristic time for heat conduction in the gas, T_0 is the reference temperature for both the gas and solid, and P_{g0} is the reference gas pressure.

Substituting these scaled variables into the original partial differential equations results in the following dimensionless system of partial differential equations:

$$\frac{\partial \hat{T}_g}{\partial \hat{t}} = \frac{\hat{h}\,\hat{\mathcal{L}}_g}{\hat{\rho}_g\,\hat{\mathcal{C}}_{p,g}\,\hat{\mathcal{A}}_g}\left(\hat{T}_s - \hat{T}_g\right) - \left(\frac{1}{\hat{\rho}_g\,\hat{\mathcal{A}}_g}\frac{\partial \hat{T}_g}{\partial \hat{z}} + \frac{1}{\hat{\rho}_g^2\,\hat{\mathcal{C}}_{p,g}\,\hat{\mathcal{A}}_g}\frac{\partial \hat{P}}{\partial \hat{z}}\right)\dot{\hat{m}}_g(\hat{t}) \tag{7.2a}$$

$$\frac{\partial \hat{T}_s}{\partial \hat{t}} = \frac{\tau_{g0}}{\tau_{s0}}\,\frac{\hat{h}\,\hat{\mathcal{L}}_g}{\hat{\rho}_s\,\hat{\mathcal{C}}_{p,s}\,\hat{\mathcal{A}}_s}\left(\hat{T}_g - \hat{T}_s\right) \tag{7.2b}$$

in which τ_{s0} is the characteristic time for heat conduction in the solid and $\dot{\hat{m}}_g(\hat{t})$ is the dimensionless mass flow rate for the gas. Note that the dimensionless energy equations are expressed using the gas mass flow rate.

7.2 Discrete Equation Formulation

The finite-volume formulation is used to discretize Equation 7.2 in both space and time [33]. In this technique, the computational domain is divided into some number of non-overlapping control volumes such that only one grid point lies inside each control volume. This set of control volumes must completely cover the original domain. The differential equations are then integrated over each control volume. For discretizing the system, denote the states of the system by the vector $\boldsymbol{x}^\dagger = [T_g\ T_s]$. These integrals are evaluated by approximating the variation of $\boldsymbol{x}$ between each grid point using piecewise linear profiles. This procedure results in a set of discrete equations containing values of

$\boldsymbol{x}$ for each grid point. Intuitively these discrete equations define a conservation principle for $\boldsymbol{x}$ over the finite volume of each cell, just as the original differential equations express it for an infinitesimal volume. The algebraic equations are assumed to be constant over each control volume, so they can be removed from the volume integrals. As an example consider discretizing an equation of the form

$$C_{p,g} \frac{\partial T_g}{\partial z} \dot{m}_g(t) = 0.$$

Applying Gauss' theorem to the volume integral gives

$$\int_{\mathcal{V}} C_{p,g}\, \dot{m}_g(t) \frac{\partial T_g}{\partial z}\, d\mathcal{V} = C_{p,g} \int_{\mathcal{V}} \nabla \cdot (\dot{m}_g\, T_g)\, d\mathcal{V} = C_{p,g} \oint_{\mathcal{S}} (\dot{m}_g\, T_g) \cdot \hat{n}\; d\mathcal{S} = 0.$$

In this problem, the dimensions of each control volume are identical, hence the discretized form of this equation for the ith space node at the jth time step is

$$C_{p,g}(j,i)\, \dot{m}_g(j)\, \frac{T_g(j,i+1) - T_g(j,i)}{\Delta z} = 0.$$

In this particular example the result is identical to that obtained with the finite-difference method, but in general this is not necessarily the case. Since this is a convection problem, the upwind-difference scheme is used to discretize the convective term in the gas equation. This approach approximates the derivative of the temperature at a node by taking the difference between the temperature at that node and the temperature at the preceding node relative to the direction of the flow. Therefore if the direction of flow changes, then the node defined as preceding also changes. This occurs in the stove since the direction of gas flow through the stove reverses between the on-gas and on-blast cycles.

Using this approach, the discretized form of the scaled differential equations in Equation 7.2 is

$$\begin{aligned} \frac{\hat{T}_g(j,i) - \hat{T}_g(j-1,j)}{\Delta t} &= \frac{\hat{h}(j,i)\, \hat{\mathcal{L}}_g(j,i)}{\hat{\rho}_g(j,i)\, \hat{\mathcal{C}}_{p,g}(j,i)\, \hat{\mathcal{A}}_g(j,i)} \left(\hat{T}_s(j,i) - \hat{T}_g(j,i)\right) \\ &\quad - \frac{1}{\hat{\rho}_g(j,i)\, \hat{\mathcal{A}}_g(j,i)} \left(\frac{\hat{T}_g(j,i) - \hat{T}_g(j,i-1)}{\Delta z}\right) \dot{\hat{m}}_g(j) \\ &\quad - \frac{1}{\hat{\rho}_g^2(j,i)\, \hat{\mathcal{C}}_{p,g}(j,i)\, \hat{\mathcal{A}}_g(j,i)} \left(\frac{\hat{P}(j,i+1) - \hat{P}(j,i-1)}{2\,\Delta z}\right) \dot{\hat{m}}_g(j), \end{aligned} \tag{7.3a}$$

$$\begin{aligned}\frac{\hat{T}_g(j,i)-\hat{T}_g(j-1,i)}{\Delta t} &= \frac{\hat{h}(j,i)\,\hat{\mathcal{L}}_g(j,i)}{\hat{\rho}_g(j,i)\,\hat{\mathcal{C}}_{p,g}(j,i)\,\hat{\mathcal{A}}_g(j,i)}\big(\hat{T}_s(j,i)-\hat{T}_g(j,i)\big)\\ &- \frac{1}{\hat{\rho}_g(j,i)\,\hat{\mathcal{A}}_g(j,i)}\left(\frac{\hat{T}_g(j,i+1)-\hat{T}_g(j,i)}{\Delta z}\right)\hat{\dot{m}}_g(j)\\ &- \frac{1}{\hat{\rho}_g^2(j,i)\,\hat{\mathcal{C}}_{p,g}(j,i)\,\hat{\mathcal{A}}_g(j,i)}\left(\frac{\hat{P}(j,i+1)-\hat{P}(j,i-1)}{2\,\Delta z}\right)\hat{\dot{m}}_g(j),\end{aligned} \tag{7.3b}$$

$$\frac{\hat{T}_s(j,i)-\hat{T}_s(j-1,i)}{\Delta t} = \frac{\tau_{g0}}{\tau_{s0}}\,\frac{\hat{h}(j,i)\,\hat{\mathcal{L}}_g(j,i)}{\hat{\rho}_s(j,i)\,\hat{\mathcal{C}}_{p,s}(j,i)\,\hat{\mathcal{A}}_s(j,i)}\big(\hat{T}_g(j,i)-\hat{T}_s(j,i)\big), \tag{7.3c}$$

in which i is the space node index and j is the time step index. Equation 7.3a is the discrete energy equation for the gas during the on-gas cycle, Equation 7.3b is the discrete energy equation for the gas during the on-blast cycle, and Equation 7.3c is the discrete energy equation for the solid for both cycles. A time grid is used because the mass flow rate $\dot{m}_g$ during the on-blast cycle varies over time due to the blast air temperature controller. Since the same solution approach is used for both the on-gas and on-blast cycles, a time grid is also used during the on-gas cycle. The conditions at the initial time become conditions along a boundary of the time dimension of the two-dimensional grid. This treatment of time requires the temporal derivatives to be incorporated into the Jacobian of the discrete system in the same manner that spatial derivatives are normally incorporated.

7.3 Newton–Krylov Technique

An implicit Newton-Krylov technique is used to solve the discrete model representation in Equation 7.3 because the solution technique must be robust for systems having disparate eigenvalues in the linear approximation and it must provide rapid convergence without using tuning parameters. The disparity in eigenvalues is created by the different time scales for convection in the gas and conduction in the brick. Rapid convergence is required in order to allow the controller to compute the optimal fuel gas flow rate for the on-gas cycle during the five-minute time period between the end of an on-blast cycle and the beginning of an on-gas cycle. A parameter-free method also allows the use of the technique by operating personnel with limited experience in nonlinear solution techniques. The algorithm is an inexact version of Newton's method, in which the update to the current solution at each stage is computed by approximately solving a linear system. This linear system results from linearizing the discrete approximation to the partial differential equations using a numerical approximation for the Jacobian of the discretized system. The resulting linear system is solved iteratively for the update using a preconditioned Krylov subspace projection method. Various methods of this type are discussed by Kelley [19].

The blast furnace stove model solution separates naturally into two parts. The first part consists of searching for a nonlinear update to the current solution. Conceptually, Equation 7.3 can be rewritten as the vector equation $\boldsymbol{f}(T_g(j,i), T_s(j,i)) = \mathbf{0}$. An approximate solution to this differential algebraic system is given by a set of states $T_g(k,i)$, and $T_s(k,i)$ that make the value of $\boldsymbol{f}(\cdot)$ close to zero for each space node i and time step j. Intuitively, this is a root-finding problem in which the roots are functions of the distance i and the time j. The function $\boldsymbol{f}(\cdot)$ is called the nonlinear residual. Collectively the states are denoted by the vector $\boldsymbol{x}^\dagger = [T_g(j,i)\, T_s(j,i)]$. The root-finding problem is to find the state $\boldsymbol{x}$ that minimizes the nonlinear residual $\boldsymbol{f}(\boldsymbol{x})$. One way to solve this problem is to compute the second order Taylor series expansion of $\boldsymbol{f}(\boldsymbol{x})$ about the point $\boldsymbol{x}$

$$f_i(\boldsymbol{x} + \delta\boldsymbol{x}) = f_i(\boldsymbol{x}) + \sum_{m=1}^{n} \frac{\partial f_m}{\partial x_m} \delta x_m + O(\delta\boldsymbol{x}^2), \tag{7.4}$$

in which $n = n_v\, n_s\, n_t$ is the product of the number of state variables (*i.e.*, $n_v = 2$ in this case), the number of grid nodes in space n_s, and the number of grid nodes in time n_t . Neglecting terms of order $\delta\boldsymbol{x}^2$ and higher and setting $\boldsymbol{f}(\boldsymbol{x} + \delta\boldsymbol{x}) = \mathbf{0}$, results in a set of linear equations for the corrections $\delta\boldsymbol{x}$ that move each residual toward zero simultaneously. For the kth iteration of the algorithm, the vector form of these equations is

$$\boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{x}_k)\, \delta\boldsymbol{x}_k = -\boldsymbol{f}(\boldsymbol{x}_k), \tag{7.5}$$

in which $\boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{x}_k)$ is the Jacobian matrix of the discrete system. The Jacobian matrix is computed numerically by perturbing each component in the unknown temperature vector $\boldsymbol{x}_k$ and computing the nonlinear residual using the perturbed state vector. The perturbed residual vector is subtracted from the unperturbed residual vector and divided by the perturbation to obtain a numerical approximation of the derivative. These derivatives are then used to form the Jacobian matrix as follows

$$\frac{\partial \boldsymbol{f}^j}{\partial \boldsymbol{x}^i} \approx \frac{\boldsymbol{f}^j(\boldsymbol{x} + \delta x^i) - \boldsymbol{f}^j(\boldsymbol{x})}{\delta x^i}, \qquad \boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{x}) \approx \begin{bmatrix} \frac{\partial \boldsymbol{f}^1}{\partial \boldsymbol{x}^1} & \frac{\partial \boldsymbol{f}^1}{\partial \boldsymbol{x}^2} & \cdots & \frac{\partial \boldsymbol{f}^1}{\partial \boldsymbol{x}^n} \\ \frac{\partial \boldsymbol{f}^2}{\partial \boldsymbol{x}^1} & \frac{\partial \boldsymbol{f}^2}{\partial \boldsymbol{x}^2} & \cdots & \frac{\partial \boldsymbol{f}^2}{\partial \boldsymbol{x}^n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{f}^n}{\partial \boldsymbol{x}^1} & \frac{\partial \boldsymbol{f}^n}{\partial \boldsymbol{x}^2} & \cdots & \frac{\partial \boldsymbol{f}^n}{\partial \boldsymbol{x}^n} \end{bmatrix} \tag{7.6}$$

in which $\boldsymbol{f}^j$ is the jth component of the nonlinear residual, $\boldsymbol{x}^i$ is the ith component of the unknown temperature vector, and δx^i is the perturbation to the ith component of $\boldsymbol{x}$.

The corrections are added to the solution vector giving the update rule

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k\, \delta\boldsymbol{x}_k, \tag{7.7}$$

in which $\alpha_k \in (0, 1]$ is a weighting factor to keep the algorithm from overshooting the solution. This algorithm for root solving is commonly known as the Newton-Raphson or Newton's method [9]. Methods for selecting α_k are discussed in [34]. The algebraic equations of state are recomputed using the new temperature estimate $\boldsymbol{x}_{k+1}$ after each Newton iteration. The Newton iterations are stopped when the criteria

$$\|\boldsymbol{f}(\boldsymbol{x}_k)\|_2 < \tau_r \, \|\boldsymbol{f}(\boldsymbol{x}_0)\|_2 + \tau_a \tag{7.8}$$

is satisfied. In this equation $\tau_r \in (0, 1)$ is the relative error tolerance, $\tau_a \in (0, 1)$ is the absolute error tolerance, and $\|\cdot\|_2$ is the Euclidean norm. Intuitively this criteria means that the Newton iterations are stopped when either the current residual $\|\boldsymbol{f}(\boldsymbol{x}_k)\|_2$ becomes less than τ_r of its initial value $\|\boldsymbol{f}(\boldsymbol{x}_0)\|_2$, or when it becomes less than τ_a.

The second part of the algorithm consists of finding the solution for the linear system in Equation 7.5. This equation is of the general form $\boldsymbol{A}\,\boldsymbol{y} = \boldsymbol{b}$, where $\boldsymbol{A}$ is an $(n \times n)$ matrix. The method used in this work is a conjugate-gradient-like polynomial-based iterative scheme. The general solution update is

$$\boldsymbol{y}_l = \boldsymbol{y}_0 + (\gamma_{l0}\,\boldsymbol{r}_0 + \gamma_{l1}\,\boldsymbol{A}\,\boldsymbol{r}_0 + \gamma_{l2}\,\boldsymbol{A}^2\,\boldsymbol{r}_0 + \cdots + \gamma_{l(l-1)}\,\boldsymbol{A}^{l-1}\,\boldsymbol{r}_0), \tag{7.9}$$

in which $\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{A}\,\boldsymbol{y}_0$, $\boldsymbol{r}_l$ is the linear residual at step l, and $\boldsymbol{y}_0$ is the initial guess for the solution of the linear system. The solution $\boldsymbol{y}_l$ at step l is the initial solution $\boldsymbol{y}_0$ plus a linear combination of vectors in the set $\{\boldsymbol{r}_0,\ \boldsymbol{A}\,\boldsymbol{r}_0,\ \boldsymbol{A}^2\,\boldsymbol{r}_0,\ \ldots,\ \boldsymbol{A}^{l-1}\,\boldsymbol{r}_0\}$. The space spanned by this set of vectors is the Krylov subspace, which is denoted by $\mathcal{K}_l(\boldsymbol{r}_0,\ \boldsymbol{A})$. Since new solution approximations are computed by projecting the linear residual $\boldsymbol{r}_l$ onto a Krylov subspace, these algorithms are collectively known as Krylov subspace projection methods [36].

Equation 7.9 can be written in the simpler form $\boldsymbol{y}_l = \boldsymbol{y}_0 + \sum_{j=0}^{l} \gamma_{lj}\,\boldsymbol{p}_j$. The manner in which $\boldsymbol{p}_l$ is computed defines a particular Krylov subspace method. In general, two criteria can be used to compute the $\boldsymbol{p}_l$ vectors. The first criterion is to pick $\boldsymbol{p}_l$ to minimize some norm of the current linear residual $\boldsymbol{r}_l$. The second criterion is to choose $\boldsymbol{p}_l$ so that the the current linear residual $\boldsymbol{r}_l$ is orthogonal to some set of vectors $\mathcal{L}_l$, where $\mathcal{L}_l$ may be different from $\mathcal{K}_l$. Mathematically, these two criteria are

$$\min_{p_l \in \mathcal{K}_l} \|\boldsymbol{r}_l\|_{\mathcal{N}} = \min_{p_l \in \mathcal{K}_l} \left\| \boldsymbol{r}_0 - \boldsymbol{A} \sum_{j=0}^{l} \gamma_{lj}\,\boldsymbol{p}_j \right\|_{\mathcal{N}}, \tag{7.10}$$

$$\boldsymbol{r}_l = \left(\boldsymbol{r}_0 - \boldsymbol{A} \sum_{j=0}^{l} \gamma_{lj}\,\boldsymbol{p}_j \right) \perp \mathcal{L}_l, \tag{7.11}$$

in which $\|\cdot\|_{\mathcal{N}}$ represents an arbitrary norm. By satisfying the first criterion, the algorithm is guaranteed to converge to a solution which minimizes some measure of the error between the exact and approximate solutions. By satisfying the second criterion, the algorithm is guaranteed to converge in a finite number of iterations. The conjugate gradient algorithm is derived assuming that $\boldsymbol{A}$ is symmetric positive definite, in which case both of these criteria can be satisfied simultaneously with $\mathcal{L}_l \equiv \mathcal{K}_l$.

In most cases the Jacobian is not symmetric positive definite, hence both of the above criteria cannot be satisfied simultaneously. There are numerous algorithms based on different implementations of one of these two criteria. The technique use here is the Generalized Minimal Residual (GMRES) algorithm [37]. This algorithm has three distinguishing features. First, it is guaranteed to minimize the 2-norm of the linear residual

$$\|\boldsymbol{r}_l\|_2 = \left\| \boldsymbol{r}_0 - \boldsymbol{A} \sum_{j=0}^{l} \gamma_{lj}\, \boldsymbol{p}_j \right\|_2 = \|\boldsymbol{b} - \boldsymbol{A}\, \boldsymbol{y}_l\|_2$$

Second, the search directions $\boldsymbol{p}_l$ are $\boldsymbol{I}$-orthonormal, meaning that $\boldsymbol{p}_i^\dagger \boldsymbol{p}_j = 0$ for all $i \neq j$ and $\|\boldsymbol{p}_i\|_2 = 1$. Third, the linear residual at any iteration is $\boldsymbol{A}$-orthogonal to all previous search directions, which means $\boldsymbol{r}_i^\dagger \boldsymbol{A}\, \boldsymbol{p}_j = 0$ for all $i > j$. Another way to state the last condition is that the linear residual $\boldsymbol{r}_l$ is orthogonal to the Krylov subspace $\mathcal{L}_l = \boldsymbol{A}\, \mathcal{K}_l(\boldsymbol{r}_0,\, \boldsymbol{A})$. The stopping criteria for the GMRES iterations is

$$\|\boldsymbol{r}_l\|_2 < \epsilon_r\, \|\boldsymbol{r}_0\|_2 + \epsilon_a, \tag{7.12}$$

in which $\epsilon_r \in (0,1)$ is the relative error tolerance and $\epsilon_a \in (0,1)$ is the absolute error tolerance. The explanation of this criteria is the same as that for the Newton stopping criteria.

The speed of convergence for finding the solution $\boldsymbol{y}$ of the linear system $\boldsymbol{A}\,\boldsymbol{y} = \boldsymbol{b}$ depends on the ratio of the maximum to the minimum eigenvalues of the matrix $\boldsymbol{A}$. If this ratio is large, then $\boldsymbol{A}$ is said to be poorly conditioned. In many practical cases $\boldsymbol{A}$ is so poorly conditioned that Krylov methods, such as GMRES, do not converge at all. Preconditioning makes a linear system easier to solve by improving its condition number. Preconditioning is accomplished by multiplying both sides of the linear system by a matrix $\boldsymbol{P}$ which resembles $\boldsymbol{A}^{-1}$ in some sense. The matrix $\boldsymbol{P}$ can be multiplied on either the left or right hand sides of the original system, giving rise to one of two new systems

$$\boldsymbol{P}\,\boldsymbol{A}\,\boldsymbol{y} = \boldsymbol{P}\,\boldsymbol{b}, \tag{7.13a}$$

$$\boldsymbol{A}\,\boldsymbol{P}\,\boldsymbol{u} = \boldsymbol{b}, \quad \boldsymbol{u} = \boldsymbol{P}^{-1}\,\boldsymbol{y}. \tag{7.13b}$$

Equation 7.13a is the new system under left preconditioning, and Equation 7.13b is the new system under right preconditioning. In this application a right preconditioner is applied to Equation 7.5 prior to solving for the linear correction $\delta \boldsymbol{x}_k$ using GMRES.

The preconditioner used in this application is MILUT [36]. It is an incomplete LU factorization of the Jacobian matrix $\boldsymbol{J_f}(\boldsymbol{x}_k)$ with threshold dropping and diagonal compensation. Incomplete LU factorization consists of performing Gaussian elimination on the matrix $\boldsymbol{A}$ and dropping some elements from the intermediate matrix LU_m at each step m of elimination. This procedure guarantees that the matrix obtained at the final step of this incomplete factorization LU_n is sparse, which would generally not be the case for the matrix obtained by Gaussian elimination. At the mth elimination step, the element $LU_m(i,j)$ is dropped if

$$|LU_m(i,j)| < \eta \, \frac{\|LU_m(i,\cdot)\|_\infty}{|LU_m(i,i)|}$$

in which $LU_m(i,\cdot)$ is the ith row of the matrix LU_m, $LU_m(i,i)$ is the diagonal element of the ith row, and $\eta \in (0,\infty)$ is the drop tolerance parameter. The diagonal entries $LU_m(i,i)$ at each iteration are compensated for the dropped terms by subtracting a weighted sum of the dropped terms from the diagonal entry

$$LU_m(i,i) = LU_m^*(i,i) - \mu \sum_{j=\mathrm{drop}} LU_m(i,j)$$

in which $LU_m^*(i,i)$ is the diagonal entry computed by elimination and $\mu \in [0,1]$ is the compensation parameter. The matrix obtained after the last step of incomplete elimination LU_n is used as the preconditioner matrix $\boldsymbol{P}$ such that $\boldsymbol{P} = LU_n$.

This solution technique has numerous positive features. It has been proven in [19] that the upper bound on the convergence rate of inexact Newton's method is superlinear and the bound for GMRES is also superlinear. Since the algorithm is implicit, any time or space scale in the problem can be followed rather than being forced to follow the fastest time scale or smallest space scale, as in explicit methods. This algorithm directly minimizes both the absolute and relative error of the solution. Because this method is based on root finding, the resulting solution is one for which $\boldsymbol{f}(\boldsymbol{x}_k) \approx \boldsymbol{0}$, and $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\|_{\mathcal{N}} \approx 0$ for some iteration k. Also this algorithm has modest memory requirements and very few parameters.

This solution algorithm also has some negative features. In practice Newton's method often diverges unless it is started fairly close to a root. Furthermore, for roots with order greater than one, the upper bound on the convergence rate is linear. In this application, although both of these difficulties are still possible, there is one feature of the problem which simplifies matters. For a well-posed system of differential equations, there is a unique real-valued solution which depends continuously on the initial and boundary conditions. Assuming that this is also true for the discretized system, there is only one real root for $\boldsymbol{f}(\boldsymbol{x}_k) = \boldsymbol{0}$. The fact that there is only one real root may simplify the task of computing it. Newton's method can be made more robust to the initial guess by adjusting the size of the Newton step taken in each iteration k using the parameter α_k in Equation 7.7. A number of line search methods for accomplishing this are discussed in [6]. Another potential difficulty is that GMRES is not guaranteed to converge in a finite number of iterations. This difficulty is dealt with by preconditioning the linear system in Equation 7.5. The goal of preconditioning is to make this equation much easier to solve without expending much computational effort constructing the preconditioner.

7.4 Numerical Grid Selection

The selection of the spatial and temporal grid is based on the relative energy balance error in the model with respect to the number of time and space nodes for the discrete approximation in Equation 7.3. Intuitively, as the the number of nodes is increased, the cell volume in the discretization is decreased and at some point the changes in the simulation results should become very small. One appropriate metric for evaluating the simulation results is the error between the energy change in the gas and the energy change in the solid for the on-gas and on-blast cycles. The energy changes in the gas and solid for the discrete system, and the relative error between these two quantities, are given by

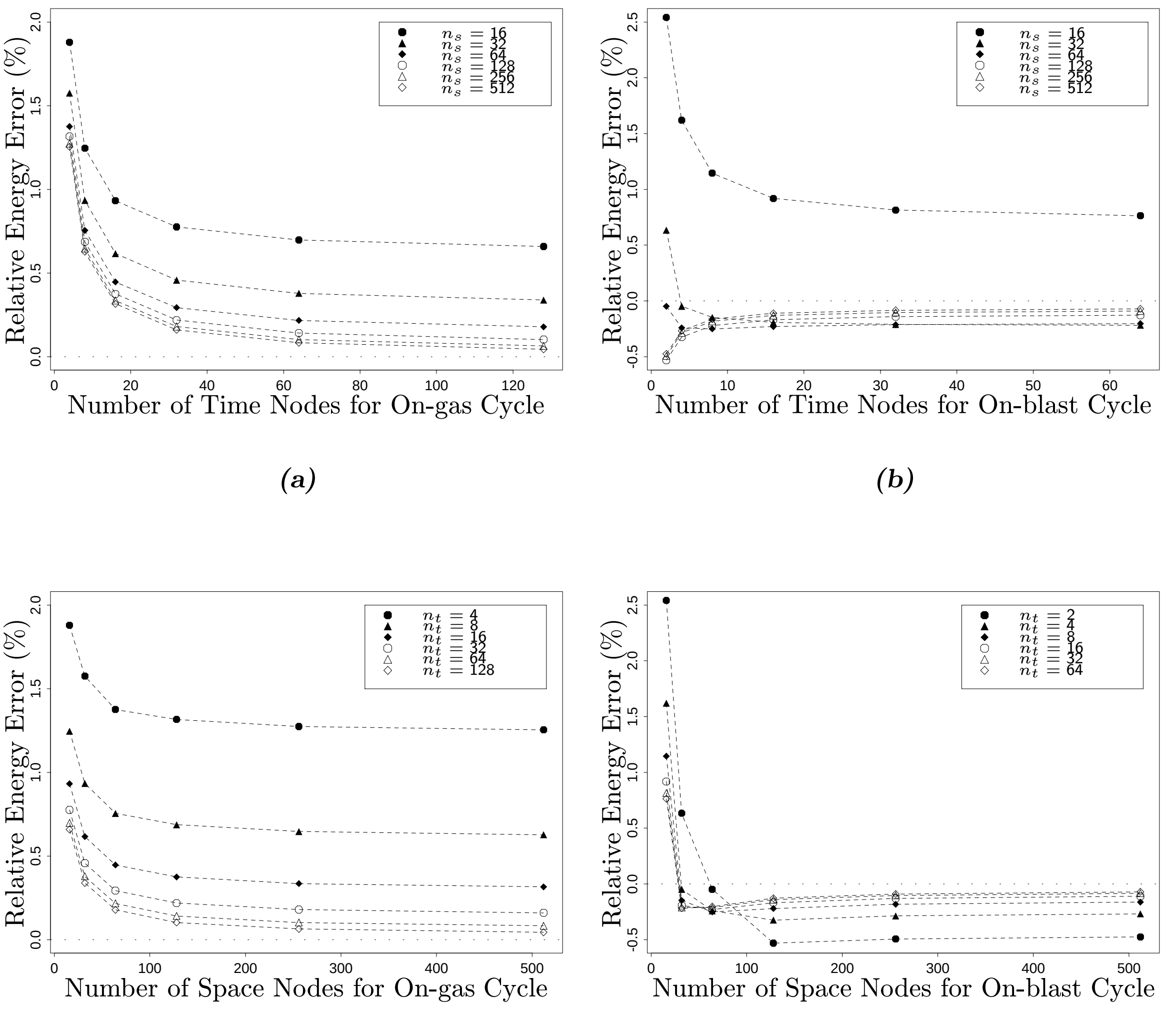
$$\Delta E_g = \sum_{j=1}^{n_t} \dot{m}_g(j) \left(\int_{T_g^{\text{in}}}^{T_g^{\text{out}}} C_{p,g}(T_g)\, dT_g \right) \Delta t, \tag{7.14a}$$

$$\Delta E_s = \sum_{i=1}^{n_s} \rho_s(i) \mathcal{A}_s \left(\int_{T_s^{\text{init}}}^{T_s^{\text{fin}}} C_{p,s}(T_s)\, dT_s \right) \Delta z, \tag{7.14b}$$

$$\Delta E_{\text{error}} = 100\, \frac{\Delta E_g - \Delta E_s}{\Delta E_s}, \tag{7.14c}$$

in which T_g^{in} and T_g^{out} are the inlet and outlet gas temperatures respectively, T_s^{init} and T_s^{fin} are the initial and final solid temperatures, n_t is the number of time nodes, n_s is the number of space nodes, Δt is the time step size, and Δz is spatial step size. As the number of time nodes goes to infinity, $n_t \to \infty$, the sum in Equation 7.14a becomes an integral over time. Likewise, as the number of space nodes goes to infinity, $n_s \to \infty$, the sum in Equation 7.14b becomes an integral over space.

A grid-convergence study is performed by computing the relative energy error ΔE_{error} as a function of both the number of time nodes, n_t, and space nodes, n_s. In this study, the number of time nodes for the on-blast cycle is chosen as one half of the number of time nodes for the on-gas cycle because the on-blast cycle is roughly half as long as the on-gas cycle. The results of these computations are shown in Figure 7.2. Figures 7.2(a) and 7.2(b) present the relative energy error versus the number of time nodes n_t with various space node values for the on-gas and on-blast cycles respectively. In these two figures, each set of similar points connected by a dotted line corresponds to a different number of space nodes. Similarly, Figures 7.2(c) and 7.2(d) present the relative energy error versus the number of space nodes n_s with various time node values for the on-gas and on-blast cycles respectively. In these two figures, each set of connected points also represents a different number of time nodes. It is clear from these figures that the relative energy error ΔE_{error} approaches zero for a sufficiently large number of time and space nodes during both the on-gas and on-blast cycles. There are errors associated with the discrete approximation of the differential equations in Equation 7.3 and the discrete



(a) ***(b)***

(c) ***(d)***

Figure 7.2: (a) Relative energy error versus the number of time nodes n_t for the on-gas cycle with various space nodes n_s.
(b) Relative energy error versus the number of time nodes n_t for the on-blast cycle with various space nodes n_s.
(c) Relative energy error versus the number of space nodes n_s for the on-gas cycle with various time nodes n_t.
(d) Relative energy error versus the number of space nodes n_s for the on-blast cycle with various time nodes n_t.

approximation of the integrals in Equation 7.14. The effect of these errors is most clearly seen in Figures 7.2(b) and 7.2(d) where ΔE_{error} is positive for a small number of nodes and then becomes negative as the number of nodes increases.

The computation time required by the solution algorithm is a important consideration for on-line implementation of the model-based controller. The goal is to reduce the computation time as much as possible while retaining the desired accuracy of the numerical solution. Based on Figure 7.2, the coarsest grid for which the solution becomes invariant to the number of grid nodes is selected. Specifically, 100 spatial nodes over the 36 meter length of the stove, 30 temporal nodes over the 50 minute on-gas cycle, and 20 temporal nodes over the 30 minute on-blast cycle are used. For this spatial and temporal grid spacing, the relative energy balance error is on the order of 0.25%. A reduction in the error can be obtained with additional grid points, as shown in Figure 7.2, at the expense of additional computation time. For the purposes of this model, a relative energy balance error on the order of 0.25% is acceptable.

The result of this grid spacing is $n = 6000$ nonlinear equations $\boldsymbol{f}(\cdot)$ of the form given by Equations 7.3a and 7.3c for the on-gas cycle and $n = 4000$ nonlinear equations $\boldsymbol{f}(\cdot)$ as specified in Equations 7.3b and 7.3c for the on-blast cycle. It is clear from Equations 7.3a and 7.3b that the portion of the Jacobian $\boldsymbol{J_f}(\cdot)$ associated with the discrete gas energy equations has four bands. One band is the diagonal, and the other three are off-diagonal. The off-diagonal bands are associated with the temporal derivatives, the spatial derivatives, and the coupling between the gas and the solid, with one band for each of these terms. Likewise from Equation 7.3c, the portion of the Jacobian $\boldsymbol{J_f}(\cdot)$ for the discrete solid energy equations has only three bands because there are no spatial derivatives in this equation.

7.5 Solution Algorithm Performance

The computation time necessary to solve the model equations is directly related to the number of Newton iterations in Equation 7.5 and the number of linear system iterations in Equation 7.9 for each Newton iteration that are required. The convergence properties of the Newton-Krylov algorithm with respect to the number of iterations for the stove model are presented in this section. As discussed by Luenberger [27], the rate of convergence of a sequence $\{\boldsymbol{u}_n\}_{n=0}^{\infty}$ that converges to a limit $\boldsymbol{u}^*$ can be assessed by computing

$$\beta = \lim_{n\to\infty} \frac{\|\boldsymbol{u}_{n+1} - \boldsymbol{u}^*\|}{\|\boldsymbol{u}_n - \boldsymbol{u}^*\|^p}$$

in which p is a positive integer. The order of convergence is the largest number p for which $0 \leq \beta < \infty$. If $p = 1$ and $0 < \beta < 1$, then the convergence rate is said to be linear. If $p = 1$ and $\beta = 0$, then the convergence rate is superlinear. If $p = 2$, then the convergence rate is quadratic. For example, given a real number a such that $0 < a < 1$, the sequence $u_n = a^n$ converges linearly, the sequence $u_n = a^{n^2}$ converges superlinearly, and $u_n = a^{2^n}$ converges quadratically. The convergence properties of the Newton-Krylov algorithm for this application are illustrated in Figure 7.3. Figures 7.3(a) and 7.3(b) present the nonlinear residual $\|\boldsymbol{f}(\boldsymbol{x}_k)\|_2$ versus the nonlinear iteration k during an on-gas cycle and an

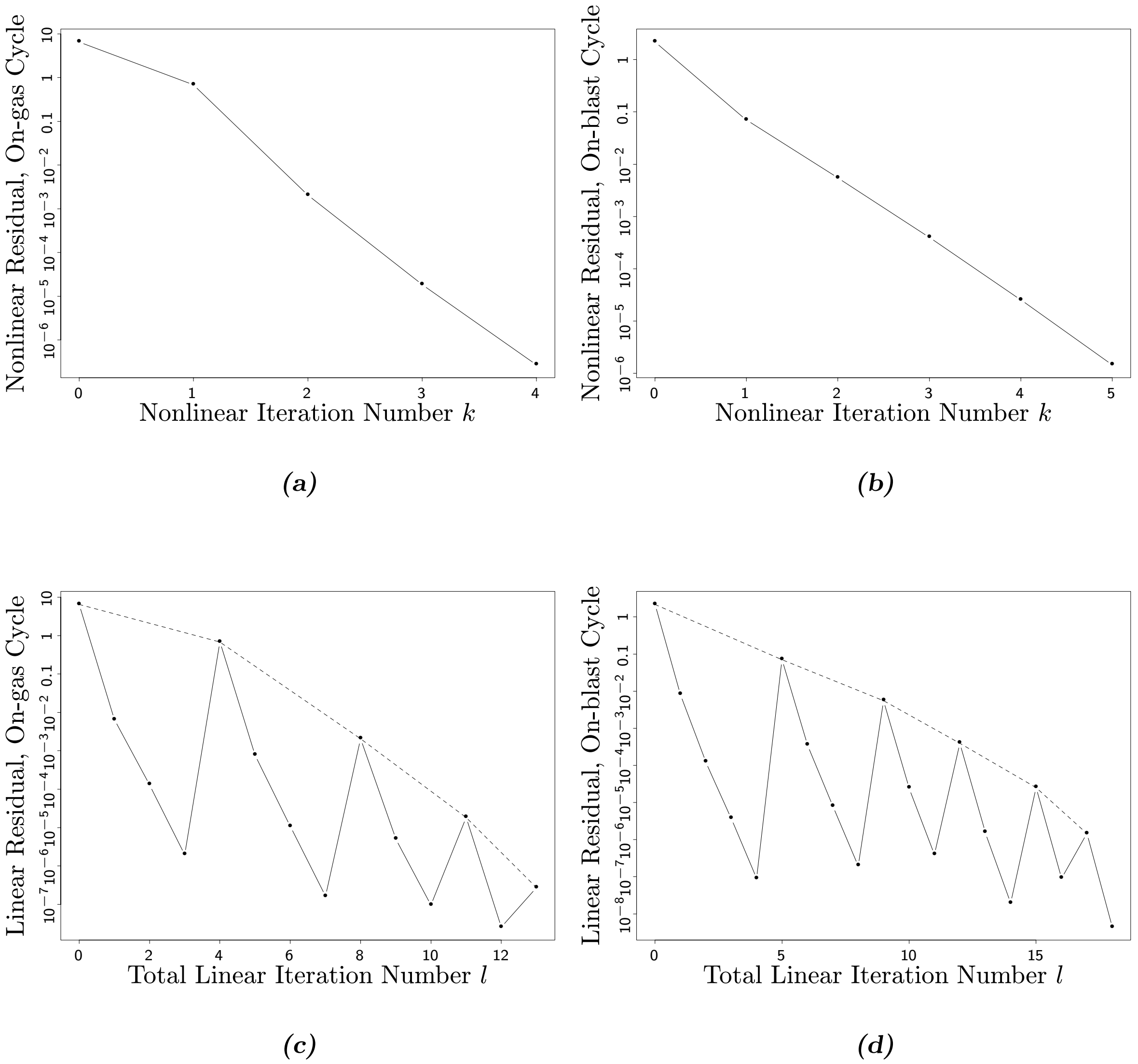


Figure 7.3: (a) The convergence of the residual $\|\boldsymbol{f}(\boldsymbol{x}_k)\|_2$ for Newton's method while solving for $\boldsymbol{x}_{\text{fin}}$ using Equation 7.7 during an on-gas cycle.
(b) The convergence of the residual for Newton's method during an on-blast cycle.
(c) The convergence of the residual $\|\boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{x}_k)\,\delta\boldsymbol{x}_l\|_2$ for MILUT preconditioned GMRES while solving for $\delta\boldsymbol{x}_k$ in Equation 7.5 during an on-gas cycle.
(d) The convergence of the residual for MILUT preconditioned GMRES during an on-blast cycle.

on-blast cycle respectively. These two figures show the convergence of Newton's method while solving for the final solution $\boldsymbol{x}_{\text{fin}}$ using Equation 7.7. Similarly, Figures 7.3(c) and 7.3(d) present the linear residual $\|\boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{J}_{\boldsymbol{f}}(\boldsymbol{x}_k)\,\delta\boldsymbol{x}_l\|_2$ versus the total number of linear iterations l during an on-gas cycle and an on-blast cycle respectively. These two figures show the convergence of MILUT preconditioned GMRES while solving for $\delta\boldsymbol{x}_k$

in Equation 7.5 during each nonlinear iteration of the cycles. Note that Figures 7.3(c) and 7.3(d) show all the linear iterations taken during an on-gas cycle and an on-blast cycle respectively. The points connected by the dashed lines in these two figures are the nonlinear residual values shown in Figures 7.3(a) and 7.3(b). The number of linear iterations for each nonlinear iteration can be deduced by counting the number of points between adjacent points on the dashed line. Notice that the vertical axes of all these plots are logarithmic. On a semi-logarithmic plot, a sequence which converges linearly will appear as a straight line. A sequence converging faster than linearly will have a negative curvature (*i.e.*, curving downward), and one converging slower than linearly will have positive curvature (*i.e.*, curving upward). Therefore Figures 7.3(c) and 7.3(d) indicate that MILUT preconditioned GMRES is converging superlinearly. Figures 7.3(a) and 7.3(b) indicate that Newton's method is converging linearly. The large number of nonlinear algebraic equations in the model is limiting the nonlinear convergence rate. Another way to assess convergence rate is by computing the number of iterations required to reduce the the residual by a factor of 10. This quantity is computed by evaluating the expression

$$\lambda = \frac{k_f - k_i}{\log\left(\|\boldsymbol{f}(\boldsymbol{x}_{k_i})\|_2\right) - \log\left(\left\|\boldsymbol{f}(\boldsymbol{x}_{k_f})\right\|_2\right)},$$

in which k_i is the initial and k_f is the final nonlinear iteration number. For the on-gas cycle shown in Figure 7.3(a), $\lambda_{\text{heat}} = 0.542$ nonlinear iterations are required to reduce the residual $\|\boldsymbol{f}(\boldsymbol{x}_k)\|_2$ by a factor of 10, and for the on-blast cycle in Figure 7.3(b) $\lambda_{\text{cool}} = 0.809$ iterations are needed.

In spite of the fact that the Newton convergence rate is linear, the rate of decrease for the residual is acceptably fast. A typical model run requires on the order of five Newton iterations each for the on-gas and on-blast cycle, fifteen total linear system solution iterations for the on-gas cycle, and twenty total linear iterations for the on-blast cycle in order to reduce the Euclidean norm of the dimensionless residual vector below 1×10^{-6}. Decreasing this convergence tolerance has essentially no effect on the solution. Execution time for a single on-gas and on-blast cycle is approximately one second on the DEC Alpha 500 AU at the Ispat Inland facility. Details on this solution technique for the blast furnace stoves are also documented in [15].

Chapter 8

Model-Based Blast Furance Stove Control

The model-based controller determines the minimum mixed gas flow rate profile during the on-gas cycle that achieves the desired blast air requirements while respecting the temperature and final mixer valve position constraints. The minimum amount of fuel represents the point where the final mixer valve position reaches its minimum limit exactly when the end of the on-blast cycle is reached. The on-gas cycle model calculations determine the effect of the fuel rate profile on the final temperature profile in the stove and the maximum temperature constraints. The on-blast cycle model calculations determine the effect of the fuel flow rate profile on the minimum temperature and mixer valve position constraints.

The maximum and minimum temperature constraints are the normal operating limits necessary to prevent thermal damage to the stoves. The exception is the minimum grid temperature constraint. This value is set by the requirement of a minimum waste gas temperature for use in pulverized coal drying at the Ispat Inland PCI facility. The minimum final mixer valve position constraint is specified by the operator and represents the additional energy to be added to the stove. The controller also considers the mixed gas flow rate constraints. The maximum mixed gas flow rate constraint represents a mechanical vibration limit on the stoves and the minimum represents a burner limitation. These constraints are summarized in Table 8.1.

8.1 Control Algorithm

The model-based controller is implemented as an optimization algorithm with the fuel rate profile for each on-gas cycle determined from the solution to the following nonlinear

Constraint	*Maximum*	*Minimum*
Dome Temperature	1500 °C	900 °C
Interface Temperature	1200 °C	800 °C
Grid Temperature	350 °C	200 °C
Final Mixer Valve		5–10 %
Mixed Fuel Gas Flow	30 m^3/sec	18 m^3/sec

Table 8.1: Controller constraints.

optimization problem

$$\min_{F_j} \sum_{j=1}^{4} F_j \tag{8.1}$$

$$\text{Subject to:} \quad \begin{array}{c} F^{\text{min}} < F_j < F^{\text{max}},\ i = 1, \ldots, N \\ T_d^{\text{gas}} \leq T_d^{\text{max}},\ \ T_d^{\text{blast}} \geq T_d^{\text{min}} \\ T_i^{\text{gas}} \leq T_i^{\text{max}},\ \ T_i^{\text{blast}} \geq T_i^{\text{min}} \\ T_g^{\text{gas}} \leq T_g^{\text{max}},\ \ T_g^{\text{blast}} \geq T_g^{\text{min}} \\ V_m^{\text{final}} \geq V_m^{\text{targ}} \end{array} \tag{8.2}$$

in which F_1 through F_4 represent the fuel flow rate profile during the on-gas cycle, T_x^{gas} are the model predicted temperatures at the end of the on-gas cycle and T_x^{blast} are the model predicted temperatures at the end of the on-blast cycle where x indicates the temperature measurement (d = dome; i = interface; g = grid), V_m^{final} is the model predicted final mixer valve position at the end of the on-blast cycle, V_m^{targ} is the target final mixer valve position, and the maximum and minimum constraints are as previously specified in Table 8.1.

Note that the final mixer valve position constraint will always be achieved since reducing the fuel used during the on-gas cycle will decrease the stove temperature resulting in a decrease in the amount of blast air that is diverted during the on-blast cycle to control the blast air temperature. Therefore, the model predicted final mixer valve position achieved by the controller will always be the minimum mixer valve position constraint value since the controller always attempts to minimize the amount of fuel used by the stove.

The predicted final mixer valve position is determined from the calculated final blast air flow rate through the stove and the following correlation between the stove flow and the mixer valve position

$$V_m = -560 \left(\frac{\dot{m}_{\text{stove}}}{\dot{m}_{\text{total}}} \right)^2 + 848 \left(\frac{\dot{m}_{\text{stove}}}{\dot{m}_{\text{total}}} \right) - 290 \tag{8.3}$$

in which V_m is the mixer valve position in percent, $\dot{m}_{\text{stove}}$ is the blast air mass flow rate through the stove, and $\dot{m}_{\text{total}}$ is the total blast air mass flow rate. This correlation was based on operating data from the No. 7 blast furnace. The final mixer valve position target is determined from the final mixer valve position at the end of the preceding on-blast cycle, V_m^{prev}, and the minimum final mixer valve position, V_m^{min}, using a first order target filter.

$$V_m^{\text{targ}} = V_m^{\text{prev}} + \alpha\left(V_m^{\text{min}} - V_m^{\text{prev}}\right), \qquad \alpha = \begin{cases} \alpha_+, & V_m^{\text{min}} > V_m^{\text{prev}} \\ \alpha_-, & V_m^{\text{min}} < V_m^{\text{prev}} \end{cases} \tag{8.4}$$

The use of a filter provides a smooth transition to the desired minimum final mixer valve position. Different filter factors allow tuning for a more aggressive control response when the previous final mixer valve position is below the minimum value. In this case, more control action is appropriate to prevent losing control range on the mixer valve during succeeding on-blast cycles. The filter also provides a tuning parameter to compensate for mismatch between the model and the process. In practice, the modeled predicted final mixer valve position tends to be more sensitive to changes in the fuel rate profile than the actual final mixer valve position.

The decision variables for this optimization problem are the fuel flow rates F_1 through F_4. Each represents the mixed gas flow rate for an equal fraction of the on-gas cycle. With $N = 4$, the fuel flow rate profile is comprised of four flow rates each with a duration of approximately twelve minutes. The choice for the parameterization of the fuel rate is based on a compromise between the range of fuel flow rate profiles that can be considered by the controller and the solution time required by the optimization problem. The solution time, and the number of times the optimization fails, both increase considerably with the number of decision variables. Since this optimization problem must be solved on-line, the profile was limited to four decision variables in order to obtain a reliable solution within the five minute period between the previous on-blast cycle and the next on-gas cycle. In addition, a smaller number of fuel rate changes during the on-gas cycle enhances the initial operator acceptance of the control algorithm.

The blast air temperature target is typically changed every hour based on the CO/CO$_2$ ratio in the top gas. As this ratio increases, there is an increase in the reduction of iron oxides toward the bottom of the furnace that can result in a decrease in the hearth temperature. In order to compensate for this decrease, the blast air temperature target is increased. Feedforward control based on blast air temperature target changes is implemented by resolving the control algorithm in Equations 8.1 and 8.2 for the remaining fuel flow rates in the current on-gas cycle. Feedforward control is initiated when the temperature target change is greater than a user specified value and at least two fuel flow rates in the current profile remain. The feedforward control capabilities of the algorithm have been programmed into the control algorithm, but they have not been implemented at Ispat Inland.

8.2 Control Results

Figures 8.1 and 8.2 show the final mixer valve position at the end of the on-blast cycle for Stove #1 with and without the controller in operation. This comparison was made at a blast air flow of 108 m^3/sec and a final mixer valve position targets of 7 % and 5 % for Stove #1. The comparison was performed at similar average blast air temperature targets, blast air temperature target changes, and mixed fuel gas heating values. The blast air temperature target changes are also shown in these figures. Feedforward control for blast air temperature target changes was not used during the closed-loop operation. As shown in these figures, there is a significant reduction in the average and the variation of the final mixer valve position with the controller in operation. Similar results for the final mixer valve position were obtained with the other two stoves.

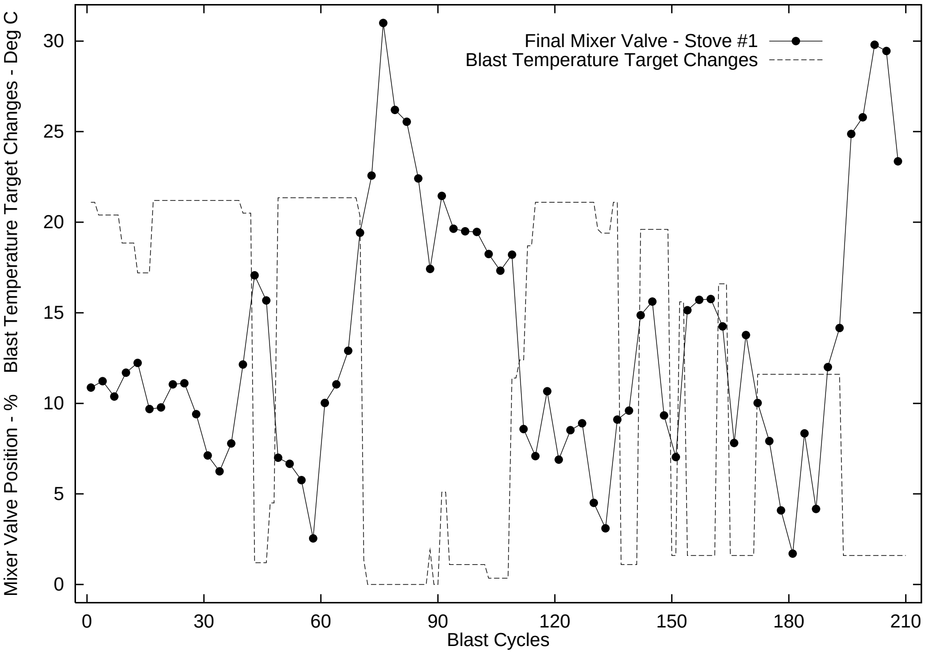


Figure 8.1: Without control final mixer valve position history.

Figure 8.3 presents a comparison between a constant mixed gas flow rate profile and the optimal profile determined by the controller for a blast air requirement of 108 m^3/sec at 1270 °C and a mixed gas heating value of 150 BTU/cuft on Stove #1. Each of these profiles resulted in a final mixer valve position of four percent at the end of the succeeding on-blast cycle. As shown in Figure 8.3, the model-predicted optimal fuel flow rate profile increases the amount of fuel used at the beginning of the on-gas cycle and then decreases the fuel usage as the cycle progresses. By taking advantage of the interaction between the temperature driving force, waste gas residence time, and waste gas velocity on the heat transferred to the stove, the reduction in mixed gas usage obtained on Stove #1 with the optimal fuel rate profile as compared to a constant profile is 1.3 vol% at the

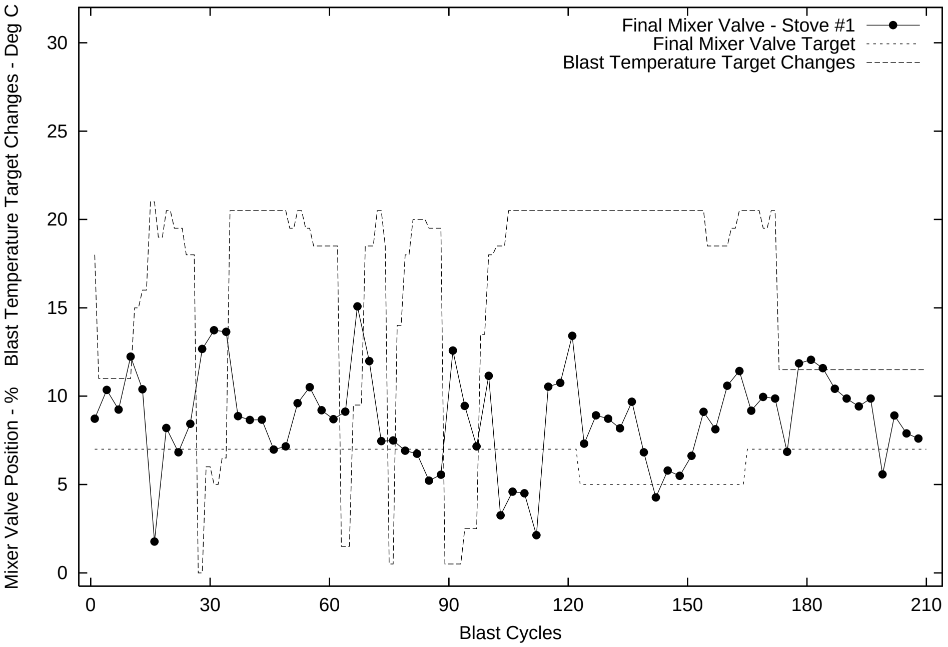


Figure 8.2: With control final mixer valve position history.

same final mixer valve position. This reduction is in agreement with estimates based on predictions from the stove model.

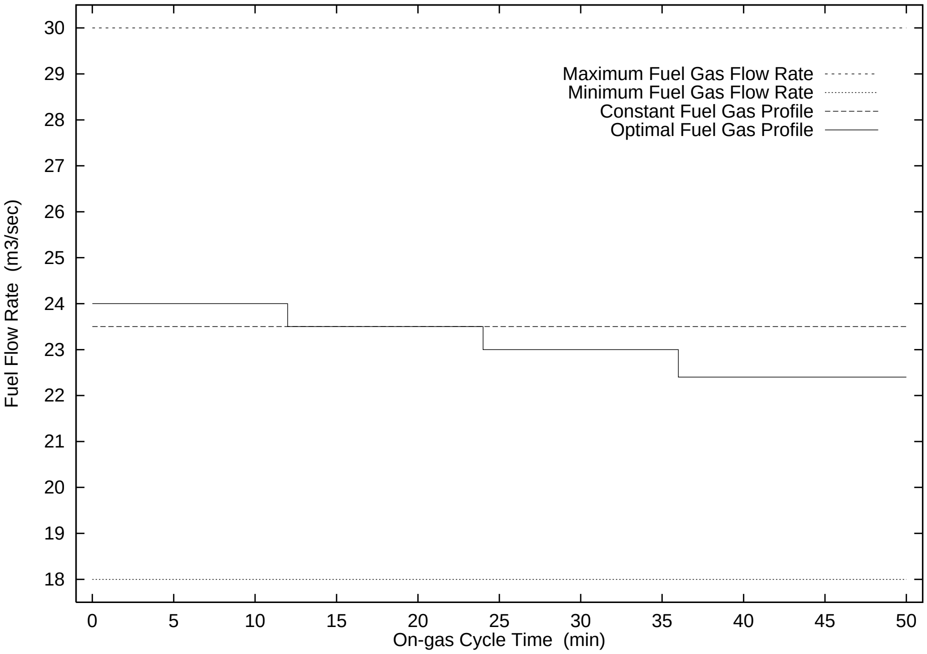


Figure 8.3: Fuel flow rate comparison.

This further reduction in the fuel gas usage is obtained by allowing the fuel gas profile to vary during the on-gas cycle. Typical operation without control is to maintain a constant fuel flow rate profile during the entire on-gas cycle. The fuel flow rate for each stove would typically be adjusted by the operator before the next on-gas cycle based on the final mixer valve position at the end of the preceding on-blast cycle. Previous analysis, assuming constant heat capacities and a heat transfer coefficient independent of temperature, indicates that the optimal fuel flow rate is constant [20] and that an on–off control policy for the natural gas enrichment of each stove should be used [42]. Since the natural gas enrichment cannot be independently set for each stove at the East Chicago facility, the enrichment is set by a heating value controller and typically remains constant. When the temperature and flow rate dependence of the physical and thermal properties are taken into account, the optimal fuel profile is no longer constant.

Chapter 9

Blast Furnace Stove State and Parameter Estimation

The temperature profile of the stove after the preceding on-blast cycle is required in order to determine the optimal fuel flow rate for the next on-gas cycle. Simulation studies in which the model was used to predict the temperature profiles without updating the states resulted in significant deviation between the model predicted and measured temperatures after several cycles. Since the checkers are approximated by tubes and the calculated heat transfer coefficients are not exact, this behavior is expected and indicates that the model must be updated based on the available measurements.

Corrections are made to the on-gas and on-blast heat transfer coefficients, the computed combustion temperature, and the stove temperature profile based on the operating data from the previous on-gas and on-blast cycles. The on-gas and on-blast heat transfer coefficients are adjusted by multiplicative scaling factors that are constrained between maximum and minimum limits. The computed combustion temperature is adjusted by a bias that is also constrained between maximum and minimum limits. The temperature profile is adjusted by a linear bias.

Figures 9.1 and 9.2 present the model predicted temperature profiles along the length of the stove at the end of a typical on-gas and on-blast cycle. The measured dome temperature, interface temperature, and grid temperature are also included. As shown in the figures, the model predicts an almost linear temperature profile along the length of the stove for both cycles that is in accord with the available measurements. Because the temperature profile in the stove is nearly linear, the stove temperature profile is adjusted by a linear correction such that the dome, interface, and grid temperature changes are constrained within maximum and minimum limits.

The choice of the model parameters to estimate is based on the uncertainty in the calculated values. The convective heat transfer coefficient correlation is only accurate to within 20 %. Additional error in the calculated heat transfer coefficients arises from the

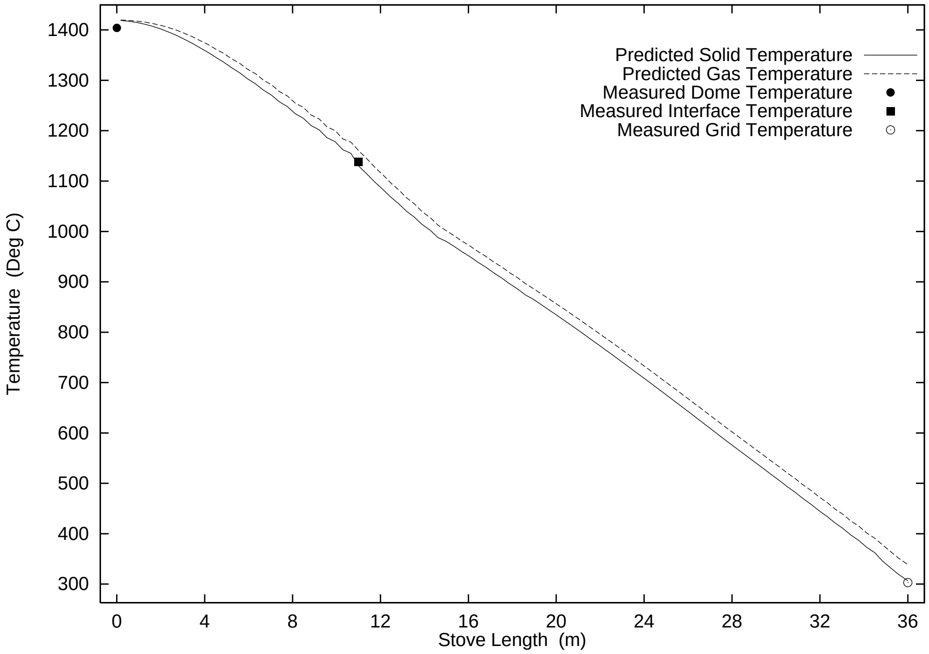


Figure 9.1: On-gas cycle temperature profile.

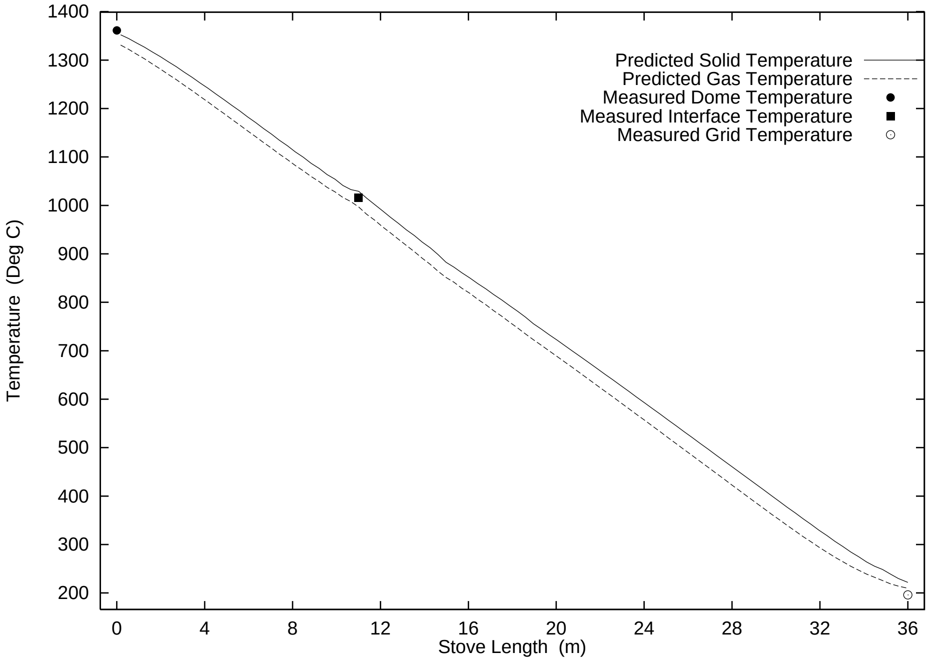


Figure 9.2: On-blast cycle temperature profile.

4 mm taper of the gas channels in each checker intended to increase the heat transfer and the assumed checker wall emissivity values. The calculated combustion temperature is uncertain due to incomplete combustion with natural gas enrichment. The calculated

temperature profiles are effected by the assumptions and omissions made during the development of the model.

9.1 Estimation Algorithm

The heat transfer coefficient scaling factors, combustion temperature bias, and linear temperature profile correction are determined using a nonlinear batch least squares estimation procedure as discussed by Jazwinski [17]. A nonlinear batch implementation is presented by Muske and Rawlings [32]. The least squares estimator attempts to minimize the deviation between the model predicted and measured dome, interface, and grid temperatures and the deviation between the model predicted and measured blast air stove flow for the preceding on-gas and on-blast cycles. The least squares problem is stated as the following nonlinear optimization problem

$$\begin{gathered}\min_{Sh^g, Sh^b, \Delta T^c, \Delta T^d, \Delta T^i, \Delta T^g} \sum_{j=1}^{N_g} \left(\alpha_d(\Delta T_{d,j}^{\text{gas}})^2 + \alpha_i(\Delta T_{i,j}^{\text{gas}})^2 + \alpha_g(\Delta T_{g,j}^{\text{gas}})^2\right) + \\ \sum_{j=1}^{N_b} \left(\alpha_d(\Delta T_{d,j}^{\text{blast}})^2 + \alpha_i(\Delta T_{i,j}^{\text{blast}})^2 + \alpha_g(\Delta T_{g,j}^{\text{blast}})^2 + \alpha_b(\Delta B_j^{\text{blast}})^2\right) + \\ \alpha_h\left((1-Sh^g)^2 + (1-Sh^b)^2\right) + \alpha_c(\Delta T^c)^2 + \alpha_p\left((\Delta T^d)^2 + (\Delta T^i)^2 + (\Delta T^g)^2\right)\end{gathered} \tag{9.1}$$

$$\text{Subject To:} \quad \begin{array}{rcl} 0.55 & \leq Sh^g \leq & 2.0 \\ 0.55 & \leq Sh^b \leq & 2.0 \\ -200 & \leq \Delta T^c \leq & 200 \\ -100 & \leq \Delta T^d \leq & 100 \\ -100 & \leq \Delta T^i \leq & 100 \\ -35 & \leq \Delta T^g \leq & 35 \end{array} \tag{9.2}$$

$$\alpha_d = 0.3, \ \alpha_i = \alpha_g = 3.0, \ \alpha_b = 1.0 \tag{9.3}$$

$$\alpha_h = 0.15, \ \alpha_c = 0.01, \ \alpha_p = 0.1 \tag{9.4}$$

in which $\Delta T_{x,j}^{\text{gas}}$ is the difference between the predicted and measured temperatures during the previous on-gas cycle and $\Delta T_{x,j}^{\text{blast}}$ is the difference between the predicted and measured temperatures during the previous on-blast cycle at sample period j where x refers to the stove temperature (d = dome; i = interface; g = grid), $\Delta B_j^{\text{blast}}$ is the difference between the predicted and measured blast air stove flow at sample period j, $N_g \approx 30$ is the number of sample periods for the on-gas cycle, $N_b \approx 20$ is the number of sample periods for the on-blast cycle, Sh^g is the on-gas cycle heat transfer coefficient scaling factor, Sh^b is the on-blast cycle heat transfer coefficient scaling factor, ΔT^c is the combustion temperature bias, ΔT^d, ΔT^i, ΔT^g are the changes to the dome, interface, and grid temperatures due to the linear adjustment of the stove temperature profile, and α_n are the weighting parameters.

The weighting parameter values in Equation 9.3 for the prediction errors are chosen as the inverse of the relative uncertainty in each of the process measurements after scaling. The weighting parameter values in Equation 9.4 for the decision variables are chosen based on the relative accuracy of the calculated values. An additional consideration for the decision variable weights is physically reasonable variation in the decision variable values between cycles. The constraints in Equation 9.2 specify the maximum allowable decision variable range.

The model parameter values that minimize the least squares objective in Equation 9.1 for the previous regenerative cycle are used in the determination of the fuel flow rate profile for the subsequent on-gas cycle. The initial temperature profile used for model-based control is the estimated profile from the end of the previous on-blast cycle. The temperature of the combustion gas entering the stove is the computed combustion temperature for the next on-gas cycle corrected by the estimated combustion temperature bias. The calculated heat transfer coefficients used in the next on-gas and on-blast cycles are scaled by the estimated heat transfer coefficient scaling factors.

9.2 Estimation Results

Figure 9.3 presents the estimated bias to the calculated combustion temperature. The bias is approximately 0.75 % of the combustion temperature on average. The major effect of changes to the calculated combustion temperature is in the dome temperature prediction. Since the dome temperature prediction error weight is an order of magnitude less than the other temperatures due to the uncertainty of this measurement, the estimated bias remains small. Figure 9.4 presents the multiplicative scaling factors for the on-gas and on-blast cycle heat transfer coefficients for one hundred Stove #1 cycles. As shown in this figure, there is almost no correction to the on-gas heat transfer coefficient whereas the on-blast coefficient is increased by 8 % on average. Figure 9.5 presents the changes to the dome, interface, and grid temperatures due to the linear profile correction. The opposite change to the dome and interface temperatures as opposed to the grid temperature during the transients in the first thirty cycles changes the distribution of heat in the stove in order to match both the temperature and stove blast air flow rate profiles. Note the similar large changes to the on-blast heat transfer coefficient scaling factor during these cycles. The variation in the estimated parameters decreases significantly for the next seventy cycles after these transients.

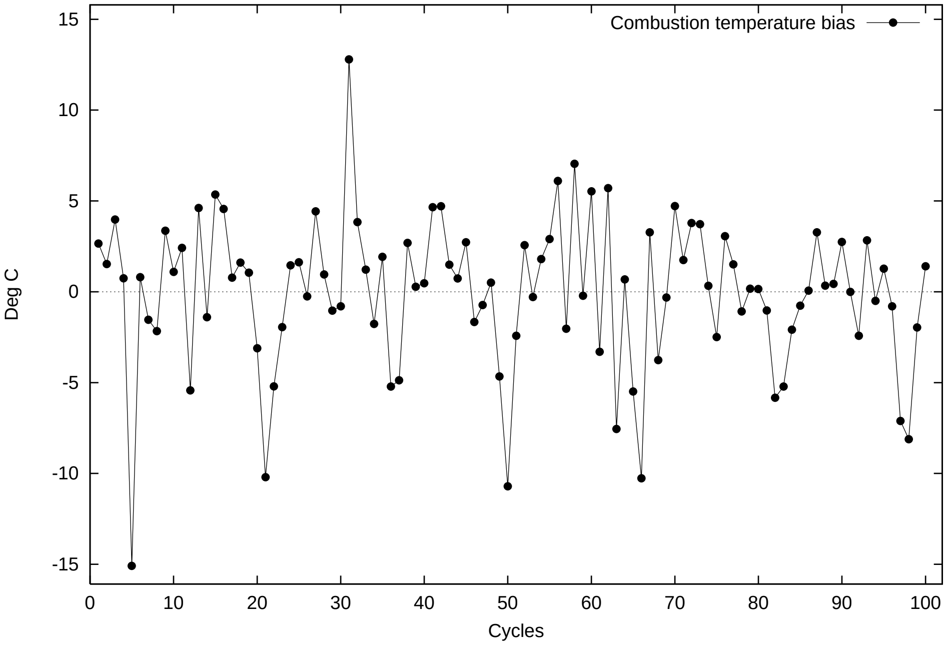


Figure 9.3: Estimated combustion temperature bias.

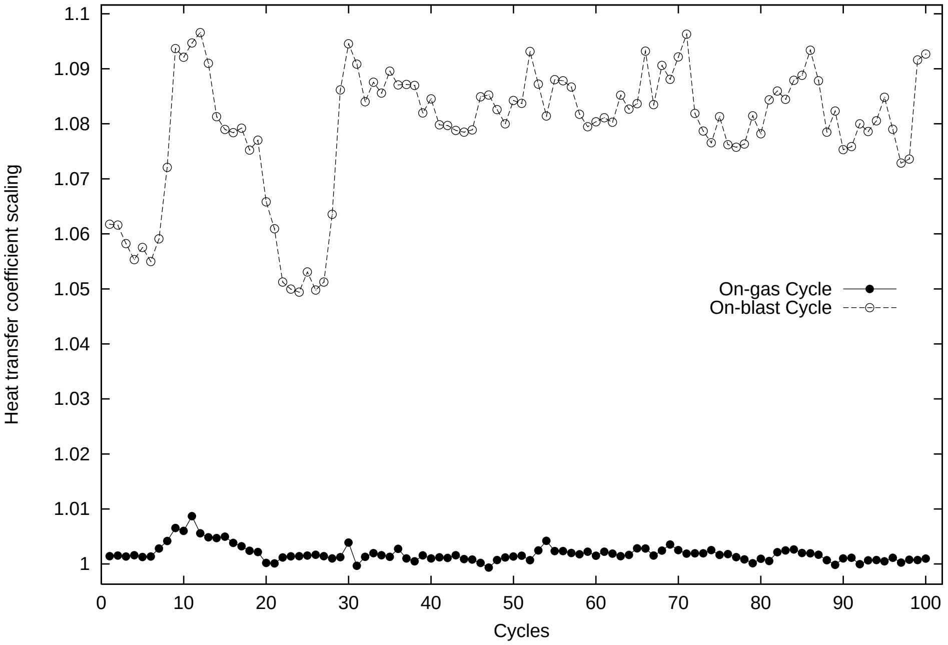


Figure 9.4: Estimated heat transfer coefficient scaling factors.

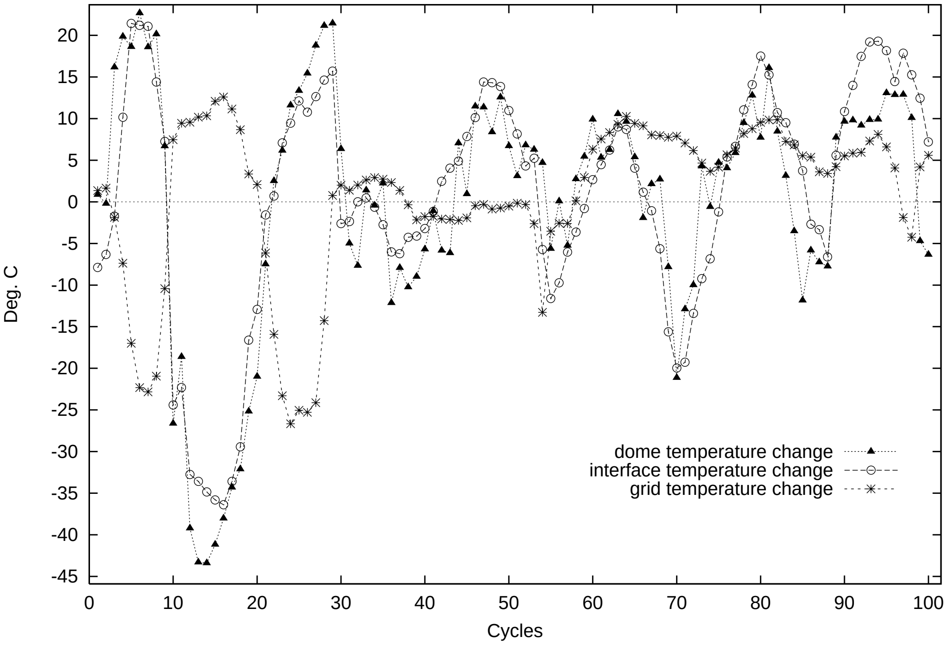


Figure 9.5: Estimated stove temperature changes.

Figures 9.6 and 9.7 compare the model predicted and estimated interface and grid temperatures to the measured values for a representative on-gas and on-blast cycle for Stove #1. Figure 9.8 compares the model predicted and estimated blast air stove flows to the measured flow for the on-blast cycle. Figure 9.9 compares the model predicted and estimated dome temperature to the measured values for the on-gas and on-blast cycles. The model predicted values are determined from the nominal parameter values computed by the model. The estimated values are determined after estimating the heat transfer coefficient scaling factors, combustion temperature bias, and temperature profile. The optimal values for these decision variables are presented in Equation 9.5.

$$\begin{array}{c} \Delta T_c = -4.22, \;\; Sh^g = 1.01, \;\; Sh^b = 1.10 \\ \Delta T_d = 15.19, \;\; \Delta T_i = 14.76, \;\; \Delta T_g = 4.30 \end{array} \tag{9.5}$$

The predicted and estimated interface and grid temperatures along with the blast air stove flow exhibit the same general trends as the measurements. The exception is the dome temperature. As presented in Figure 9.9, the measured temperature shows almost no increase during the on-gas cycle, a slight decrease during the on-blast cycle, and then a significant transient at the end of the on-blast cycle and during the blow-off cycle. This behavior is typical for some of the regenerative cycles while for other cycles the dome temperature variation is similar to that of the interface temperature. For all the cycles observed during this study, the dome temperature measurement is a noisy signal. For these reasons, the weighting factor for the dome temperature prediction error in Equation 9.3 is an order of magnitude less than the other two temperature measurements.

The estimator increased the temperature of the stove and the heat transfer coefficients resulting in an average relative over-prediction error of approximately 3 % for the dome temperature, 2 % for the interface temperature, 3 % for the grid temperature, and 2 % for the blast air stove flow. These trajectories present the optimal distribution of the prediction errors based on the estimation weights. In order to decrease the predicted temperatures, the predicted stove flow must increase since there would be less energy in the stove. Similarly, the predicted temperatures must increase in order to decrease the predicted stove flow.

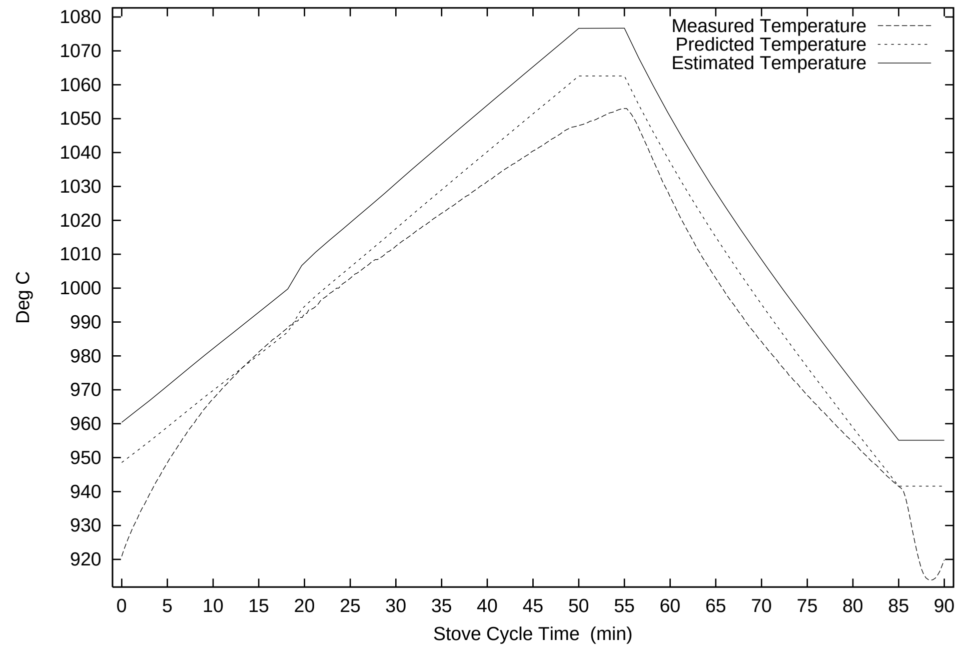


Figure 9.6: Estimated interface temperature comparison.

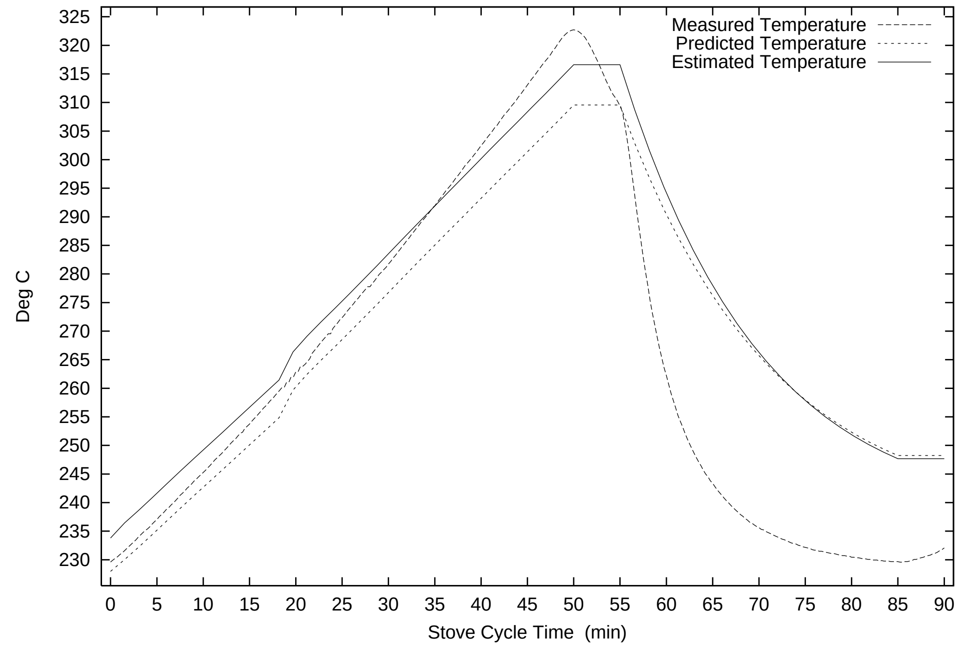


Figure 9.7: Estimated grid temperature comparison.

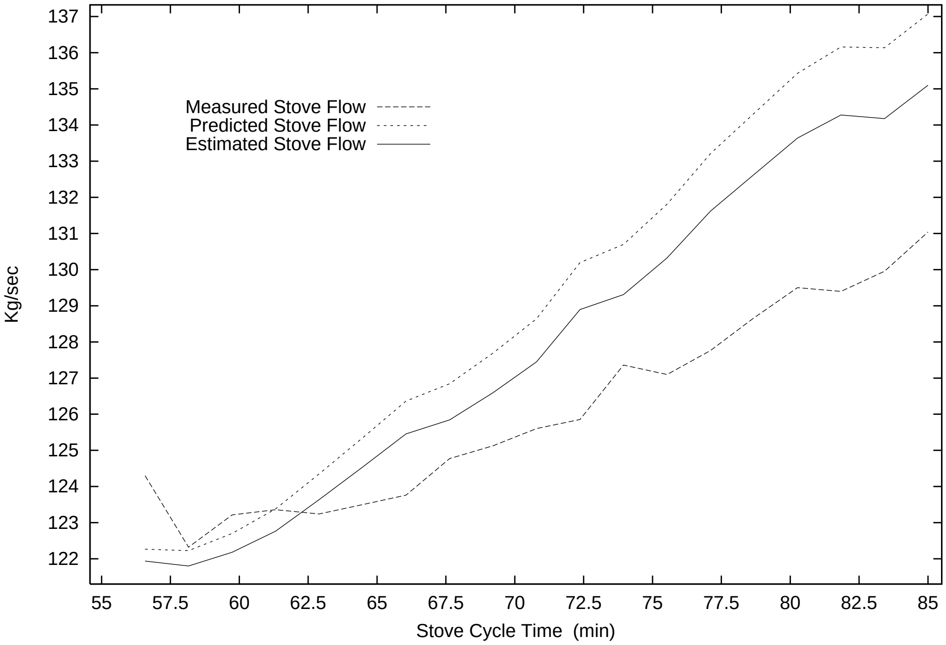


Figure 9.8: Estimated on-blast cycle stove flow comparison.

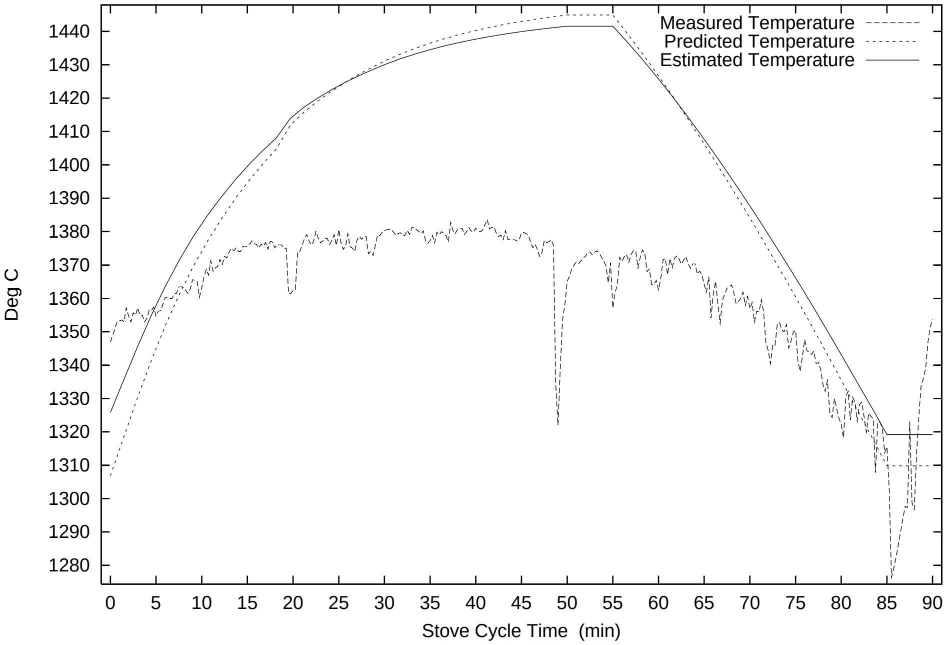


Figure 9.9: Estimated dome temperature comparison.

Chapter 10

Numerical Optimization Technique for Estimation and Control

The model predictive control and estimation algorithms are each stated as nonlinear optimization problems that must be solved on-line prior to each on-gas cycle in order to determine the minimal fuel rate profile. Luenberger [27], Fletcher [9], and Bertsekas [2] discuss nonlinear optimization, which is also referred to as nonlinear programming. The control and estimation optimization problems in this work are solved using GRG2 [25]. GRG2 is a feasible path, gradient based, generalized reduced gradient algorithm. Additional information on this class of algorithms is presented by Gill, *et al.* [11].

Determination of the minimal fuel rate profile proceeds as follows. After an on-blast cycle is complete, the estimation optimization is first solved using the stove data from the previous on-gas and on-blast cycles to determine the heat transfer coefficient scaling factors, combustion temperature bias, and stove temperature profile that will be used for the control calculation. After the estimation is complete, the control optimization problem is solved. The computation time required to solve these nonlinear optimization problems on-line is approximately three minutes on the DEC Alpha 500 AU at the Ispat Inland facility. This computational requirement is well within the five minute transition period between the preceding on-blast cycle and the subsequent on-gas cycle.

The estimation optimization problem in Equations 9.1 and 9.2 is solved sequentially using GRG2. Sequential estimation algorithms are discussed by Muske and Edgar [30]. The initial estimate of the heat transfer scaling factors is unity. The initial estimate of the combustion and stove profile temperature biases is zero. These estimates correspond to no correction of the values calculated by the model. The temperature and stove flow rate prediction errors are computed from the stove model and the previous cycle data based on the decision variables determined by the optimization algorithm at each function evaluation. The gradients of the objective function in Equation 9.1, and each of the constraints in Equation 9.2 are determined numerically.

The control nonlinear optimization problem in Equations 8.1 and 8.2 is also solved sequentially using GRG2 in which the initial estimate of the fuel profile is constant at the average value of the fuel rate for the preceding on-gas cycle. The predicted temperatures and final mixer valve position are computed from the stove model based on the fuel gas flow rate profile determined by the optimization algorithm at each function evaluation. The gradients of the objective function in Equation 8.1 and each of the constraints in Equation 8.2 are determined numerically.

Chapter 11

Software Operation

The model-based control software is configured to use a series of parameter files, input data files, and output data files to provide the necessary process information for the estimation/control algorithms and to present the results of the calculation. When the software is executed, either automatically as an on-line controller or manually as an off-line calculation, the parameter and input data files are read by the program. The results of the calculation are then written to the output files.

The physical parameters necessary to describe the blast furnace stoves and the tuning parameters for the estimation and control calculations are contained in parameter files that are input by the software prior to each program execution. The format and content for each of these parameter files are documented in Section 11.1. The process data required by the estimation and control algorithms is contained in a series of data files that would typically be generated by the process monitoring and control computer system. The control results are contained in a series of output data files generated by the software. Sections 11.2 and 11.3 describe the format and content for each of these files. The result of the estimation/control calculation is the optimal fuel flow rate profile for the subsequent on-gas cycle. This information is contained in the output file control.out. Detailed information concerning these calculations is contained in the output file dump.out. Section 11.4 discusses the content and interpretation of these files. The tuning parameters used by the software are documented in Section 11.5. The estimation/control calculation and software error conditions are discussed in Section 11.6.

11.1 Parameter Files

A series of three parameter files are used by the software to determine the numerical values for physical parameters describing the blast furnace stoves and tuning parameters for the numerical solution algorithms used by the estimation/control. The physical parameters for the checkers in each zone of the stove are contained in the file brick.in. The average physical parameters used for scaling the differential equations and the parameters required for the numerical solution of the model equations are contained in the file stove.in. The contents of these files are determined during the initial set-up of estimation/control program and should not normally be modified. The tuning parameters and constraints for the estimation/control optimization are contained in the parameter file param.in. The content of this file would typically be changed to reflect changes in the operation of the blast furnace stoves.

11.1.1 brick.in

The parameter file brick.in contains the physical dimensions and properties of each type of brick contained in the different zones of the stove. The number of zones used in the stove model is also specified by this file. An example file is presented in Figure 11.1. The first column is the zone number, the second column is the height of the zone in cm, the third column is the average hydraulic diameter of the gas channel in the checker in cm, the fourth column is the density of the checker material in g/cm^3, and the fifth column is the total checker mass contained in the zone in grams. Each row corresponds to a different zone.

```
5  1082.0  3.150d0  1.83  6.733d8
4   396.0  3.150d0  2.48  3.339d8
3   396.0  3.675d0  2.34  2.673d8
2  1585.0  3.675d0  2.17  9.919d8
1   152.0  3.675d0  2.48  1.084d8
```

Figure 11.1: Contents of the file brick.in

11.1.2 stove.in

The parameter file stove.in contains the information necessary for the scaling, discretization, and solution of the blast furnace stove model. The parameter values were determined by tuning the Newton-Krylov solution algorithm as discussed in Chapter 7. The parameter values in this file should normally not be changed unless the Newton-Krylov solution fails consistently. Each of these parameters will be discussed briefly in this section. The first column contains the numerical value, the second column contains the mnemonic used in the code, and the third column contains a brief description for each entry in this file.

The first four entries specify the number of time steps to be used in the numerical solution grid for each of the four stove cycles. The selection of these values is discussed in Section 7.4. Note that a single time step is specified for the pressurization and blow-off cycles. At each time step for these cycles, a thermal equilibrium calculation between the solid and the gas is performed. The number of time steps for these cycles should not be increased from one without modifying this calculation. For the on-gas and on-blast cycles, increasing the number of time steps will increase the computational time required for the model. The number of time steps for each cycle **must not** be zero.

```
c Input file for stove simulation
c
c Simulation parameters
c
 30        ntimg   (max time steps, on-gas cycle)
 20        ntimb   (max time steps, on-blast cycle)
 1         ntimpres(max time steps, pressure-up cycle)
 1         ntimblow(max time steps, blow-down cycle)
```

The next four entries are the cycle times for each of the four stove cycles specified in seconds. The time step used in the model solution for each cycle is determined from the cycle time divided by the number of time steps. The computational time step for the on-blast and on-gas cycles must correspond to the time interval of the data collected in the files onblast.in and ongas.in, discussed in Section 11.2, for the estimation to work properly. The computational time step is also the time interval for the data output to the files est_blst.out and est_gas.out discussed in Section 11.3.

```
 3.00d+3   dtgas   (on-gas cycle time (s))
 1.80d+3   dtbls   (on-blast cycle time (s))
 3.00d+2   dtpres  (pressure-up cycle time (s))
 3.00d+2   dtblow  (blow-down cycle time (s))
```

The following entry specifies the number of complete regenerative cycles (on-gas; pressurization; on-blast; blow-off) to run for each execution of the model solution subroutine. This value **must** be 1 when the model is used for estimation/control calculations.

```
 1         ncyc    (number of complete stove cycles, gas, blast, etc.)
```

The next five parameters specify the spatial grid. The mesh size in the axial direction is determined from the number of nodes in the z direction and the length of the stove. The selection of this value is discussed in Section 7.4. The length of the stove is specified in cm using the largest z coordinate value entry. The largest and smallest z coordinate entry values are also used to scale the axial length dimension as discussed in Section 7.1. Note that the stove length specified in this file should be consistent with the sum of the zone lengths specified in the parameter file brick.in. Also note that the number of nodes in the axial direction should be greater than the number of stove zones specified in the parameter file brick.in. The mesh size in the radial direction is determined by the number of nodes in the r direction entry and the tube wall thickness computed by Equation 6.1. When this entry is 1, radial conduction in the solid is not considered. Note that this entry **must** be 1 for estimation/control calculations. Also note that increasing the number of spatial nodes in either direction will significantly increase the computational time required for the model. The number of spatial nodes for each direction **must not** be zero. Axial heat transfer is not used in the model since it has almost no effect on the computed temperature profile.

```
100       nx      (number of nodes in mesh axial direction)
1         nr      (number of nodes in mesh radial direction, 1 = lumped)
0         axflag  (use axial heat transfer, 1 = use, 0 = do not use)
3.611d+3  xmax    (largest axial coordinate value (cm))
0.0d0     xmin    (smallest axial coordinate value (cm))
```

The next two entries specify the location of the dome and interface temperature measurements in the stove specified in cm from the top of the stove. The grid temperature measurement is assumed to be at the bottom of the stove and is not specified in this parameter file.

```
0.0d0     domloc  (location of dome temperature reading (cm))
1.10d+3   intloc  (location of interface temperature reading (cm))
```

The next entry is the maximum stove temperature, T_o in Equation 7.1, and the following entry is the reference temperature, T_i in Equation 7.1, used to scale the partial differential equations. The temperature tmax is used as the combustion temperature if data drop-outs occur. In this case, tmax is the Dirichlet boundary condition at the top of the stove during the on-gas cycle.

```
1.45d+3   tmax    (maximum stove temperature (C))
2.0d+1    tinit   (reference temperature (C))
```

The following seven entries specify the boundary conditions to use for the on-gas and on-blast cycles respectively. These entries **must not** be changed for proper operation of the stove model. The temperature bbls is used as the cold blast temperature if data drop-outs occur.

```
0         tgasneu (Neumann at top on-gas, 1 = Neumann, 0 = Dirichlet)
1         bgasneu (Neumann at bottom on-gas, 1 = Neumann, 0 = Dirichlet)
3.0d+2    bgas    (the boundary condition, Dirichlet, bottom, on-gas (C))
1         tblsneu (Neumann at top on-blast, 1 = Neumann, 0 = Dirichlet)
1.3d+3    tbls    (the boundary condition, Dirichlet, top, on-blast (C))
0         bblsneu (Neumann at bottom on-blast, 1 = Neumann, 0 = Dirichlet)
2.30d+2   bbls    (the boundary condition, Dirichlet, bottom, on-blast (C))
```

The next thirteen entries specify the average physical property values used to scale the partial differential equations as discussed in Section 7.1. Unless the stove operation changes significantly so that these values are no longer representative, they should not be changed. The flow rates mdotg and mdotb are used as the on-gas mixed gas mass flow rate and on-blast cold blast flow rate respectively if data drop-outs occur. The pressures ping and pinb are used as the on-gas top gas pressure and on-blast cold blast pressure respectively if data drop-outs occur.

```
6.20d+4   mdotg   (average mass flow rate, on-gas (g/s))
1.40d+5   mdotb   (average mass flow rate, on-blast (g/s))
1.25d+3   ugb(1)  (average gas velocity, on-gas (cm/s))
3.65d+2   ugb(2)  (average gas velocity, on-blast (cm/s))
9.43d-4   hgb(1)  (average heat transfer coefficient, on-gas (cal/cm^2 s K))
1.41d-3   hgb(2)  (average heat transfer coefficient, on-blast (cal/cm^2 s K))
4.24d-4   rgb(1)  (average gas density, on-gas (g/cm^3))
2.22d-3   rgb(2)  (average gas density, on-blast (g/cm^3))
2.87d-1   cgb(1)  (average gas heat capacity, on-gas (cal/g K))
2.70d-1   cgb(2)  (average gas heat capacity, on-blast (cal/g K))
1.15d0    ping    (inlet pressure, on-gas (atm, absolute))
4.75d0    pinb    (inlet pressure, on-blast (atm, absolute))
1.5d0     ri      (inside gas channel (flue) radius (cm))
```

The following two entries specify the total number of gas flues in the stove, N_c in Section 6.1, and the relative roughness of the tube wall, ϵ in Section 6.1.

```
1.898d+4  nc      (total number of gas channels (flues))
2.0d-2    relrgh  (relative gas channel (flue) roughness ratio)
```

The next series of entries are the solver parameters for the Newton-Krylov solution technique used on the discretized model equations. Each of these solver parameters is discussed in Section 7.3. These parameters have been optimized for the solution of the discretized model equations and should not need to be changed. The first entry is the perturbation used to compute the Jacobian as shown in Equation 7.6. The next entry is the α parameter used for the nonlinear Newton updates in Equation 7.7. The next four entries are the relative and absolute convergence tolerance for the nonlinear and linear iterations in Equations 7.8 and 7.12. The next entry is the maximum number of iterations. If the solution does not converge within this number of iterations, the value of the last iterate is used as the solution. The next four entries specify the linear solution and preconditioning technique contained in the software package SPARSEKIT that is used. These values should not be changed unless one is familiar with this package and these solution techniques. The last five entries are the tuning parameters for the preconditioner.

```
c
c Solver parameters
c
 1.0d-6    bb       (Numerical Jacobian perturbation parameter)
 1.0d0     damp     (Newton update damping, 0 < damp <= 1)
 1.0d-6    nlrerr   (Relative nonlinear convergence tolerance)
 1.0d-6    nlaerr   (Absolute nonlinear convergence tolerance)
 1.0d-6    lrerr    (Relative linear convergence tolerance)
 1.0d-6    laerr    (Absolute linear convergence tolerance)
 500       maxits   (Maximum number of linear solver iterations)
 2         prconpos (Preconditioner status, 0=none, 1=left, 2=right)
 1         stpcrt   (Stopping criteria for linear solver)
c                   -2 == || dx(i) || <= rtol * || rhs || + atol
c                   -1 == || dx(i) || <= rtol * || dx(1) || + atol
c                    0 == solver will choose test 1 (next)
c                    1 == || residual || <= rtol * || initial residual ||
c                    2 == || residual || <= rtol * || rhs || + atol
c                 where dx(i) denote the change in the solution at the ith update.
c                 ||.|| denotes 2-norm. rtol = fpar(1) and atol = fpar(2).
 9         linsol   (Linear solution algorithm)
c                   1 = LINPACK
c                   2 = Conjugate Gradient Method
c                   3 = Conjugate Gradient Method with Normal Residual
c                   4 = Bi-Conjugate Gradient Method
c                   5 = Bi-Conjugate Gradient Method with Partial Pivoting
c                   6 = Stabilized Bi-Conjugate Gradient Method (Bi-CGSTAB)
c                   7 = Transpose-Free Quasi-Minimum Residual Method (TFQMR)
c                   8 = Full Orthogonalization Method (FOM)
c                   9 = Generalized Minimum Residual Method (GMRES)
c                   10 = Flexible version of Generalized Minimum Residual
c                   11 = Direct Quasi Generalized Minimum Residual Method
c                   12 = Preconditioned Generalized Minimum Residual Method
 4         prconalg (Preconditioning algorithm)
c                   0 = Incomplete LU Factorization with no fill-in (ILU0)
c                   1 = Incomplete LU with Diagonal Compensation (MILU0)
c                   2 = Incomplete LU Factorization with K-level fill-in (ILUK)
c                   3 = Incomplete LU Factorization with Truncation (ILUT)
c                   4 = Incomplete LU with Diagonal Compensation and Dropping (ILUD)
c                   5 = ILUT with pivoting (ILUTP)
c                   6 = ILUD with pivoting (ILUDP)
 12        levfil   (Level of fill-in for ILUT and ILUK)
 1.d-6     droptol  (Drop tolerance for ILUT)
 0.75      diagcomp (Diagonal compensation for ILUD and ILUDP, 0 <= dc <= 1)
 1.d-3     thrstol  (Threshold for eliminating terms for ILUD and ILUDP)
 0.5       permtol  (Threshold for permuting columns for ILUTP and ILUDP)
```

The last two parameters are used to generate output files containing information about the model solution. These files are documented in Section 11.3. Note that these flags **must** be set to zero for estimation/control calculations because the execution time of the model is significantly increased when these files are generated.

```
c
c Printing parameters
c
 0         grf      (1=write iteration diagnostic file)
 0         spout    (1=write temperature and property diagnostic file)
```

11.1.3 param.in

The parameter file param.in contains the tuning parameters and constraint values for the estimation and control optimization. These parameters would be changed when the stove and/or estimation/control algorithm operation dictates. Each of these parameters will be discussed briefly in this section. The first column contains the numerical value, the second column contains the mnemonic used in the code, and the third column contains a brief description for each entry in this file.

The first six entries specify the maximum and minimum heat transfer scaling and combustion temperature constraint values for the estimation optimization in Equation 9.2. These values should represent a physically reasonable range for the parameter values that allow a sufficient degree of freedom for the estimation.

```
 2.d+0     maxgsht    (maximum on-gas heat transfer scaling)
 5.5d-1    mingsht    (minimum on-gas heat transfer scaling)
 2.0d+0    maxblht    (maximum on-blast heat transfer scaling)
 5.5d-1    minblht    (minimum on-blast heat transfer scaling)
 1600.0    maxcomb    (maximum combustion temperature)
 1200.0    mincomb    (minimum combustion temperature)
```

The next three entries specify the absolute maximum increase or decrease in the dome, interface, and grid temperatures in °C due to the linear correction to the stove temperature profile. Note that values between -1 and 1 will be estimated by the software. The actual change in the temperature is determined by multiplying the estimated value by these scaling factors.

```
 100.0     tmpscl(1)  (dome temperature change scaling)
 100.0     tmpscl(2)  (interface temperature change scaling)
 35.0      tmpscl(3)  (grid temperature change scaling)
```

The next eleven entries specify the convergence tolerance for the estimation optimization problem in Equations 9.1 and 9.2 and the weighting factors used to determine the objective function values. The weighting factors are the α_n values in Equation 9.3.

```
5.d-5     estconv    (estimation convergence tolerance)
0.3       domwght    (estimation objective dome temperature weight)
3.0       intwght    (estimation objective interface temperature weight)
3.0       grdwght    (estimation objective grid temperature weight)
1.0       flowght    (estimation objective stove flow weight)
5         initbltm   (first point for matching on-blast flow rate history)
100.0     mindidif   (minimum temperature difference; dome and interface)
0.15      htcwght(1) (change in on-gas heat transfer scale weight)
0.15      htcwght(2) (change in on-blast heat transfer scale weight)
0.01      combwght   (change in combustion temperature weight)
0.1       tmpwght    (change in temperature change weight)
```

The following six entries are the stove temperature constraint values for the control optimization specified in °C.

```
1500.0    mxdomlim   (maximum dome temperature constraint)
900.0     mndomlim   (minimum dome temperature constraint)
1350.0    mxintlim   (maximum interface temperature constraint)
800.0     mnintlim   (minimum interface temperature constraint)
360.0     mxgrdlim   (maximum grid temperature constraint)
100.0     mngrdlim   (minimum grid temperature constraint)
```

The next three entries are used to determine the final mixer valve position constraint. The final mixer valve target bias is added to the operator target before executing the control calculations. The target filter entries correspond to those shown in Equation 8.4. These values set the response speed of the controller and compensate for differences between the model and the actual stove in the sensitivity of the final mixer valve position to the on-gas cycle fuel profile.

```
2.0       mxr_bias   (final mixer valve target bias)
2.5d-1    byp_fup    (final mixer valve target filter, upward moves)
1.0d-1    byp_fdn    (final mixer valve target filter, downward moves)
```

The following entry is the maximum waste gas flow rate constraint specified in m^3/sec. It is used to determine the maximum mixed fuel gas flow rate constraint for the control optimization.

```
72.0      maxigas    (maximum waste gas flow rate constraint)
```

The next entry is the mixed gas fuel flow rate bias specified in m^3/sec. It is added to each of the fuel flow rates in the profile determined by the control optimization.

```
 0.40      fuelbias   (mixed gas fuel flow rate bias)
```

The final entry is the control optimization convergence tolerance.

```
 1.d-4     cntconv    (control convergence tolerance)
```

11.2 Input Files

The input files required by the estimation/control software are presented in this section. The information contained in these files is read prior to the execution of the estimation/control algorithm. These files must be updated with the correct values before the software is executed.

11.2.1 control.out

The file control.out contains the results from the control calculation. It also contains the biased constraint values for the controller that are used by the feedforward control algorithm. This file is input prior to a feedforward control calculation and is then overwritten. It is not an input file for the initial estimation/control calculation performed prior to each on-gas cycle. A detailed description of the contents of this file is presented in Section 11.4.1.

11.2.2 target.in

The file target.in contains the desired operating targets for the subsequent regenerative cycle. An example file is presented in Figure 11.2. The first column is the number of fuel rates in the profile to consider in the control optimization. If this value is zero, an initial estimation/control calculation is specified in which estimation is first performed and then the fuel flow rate profile is determined by the controller. This is the normal mode for a stove that just finished an on-blast cycle. A value other than zero indicates a feedforward control calculation in which estimation is not performed, and the entry specifies the number of fuel rates remaining in the profile. The fuel rates in the profile that have already been implemented are not considered in the optimization. Feedforward control is normally implemented when the blast temperature and/or flow rate target has changed significantly sometime during on on-gas cycle. The second column specifies the total blast air flow rate target in m^3/sec. The third column specifies the waste gas excess oxygen target in mol%. The fourth column specifies the desired final mixer valve position at the end of the subsequent on-blast cycle in %open. The fifth column specifies the blast air oxygen injection rate in m^3/sec. The sixth column specifies the blast air temperature target in °C. The seventh column specifies the blast air moisture injection in g/m^3. The eighth column specifies the minimum mixed fuel gas flow rate constraint in m^3/sec. The ninth column specifies the mixed fuel gas heating value target in BTU/ft^3.

```
 0 106.00     1.50     5.00     5.50 1269.25   20.08   20.00  150.00
```

Figure 11.2: Contents of the file target.in

11.2.3 temp.in

The file temp.in contains the estimated stove temperatures, combustion temperature, and heat transfer scaling factors. This file is read by the estimator prior to an estimation/control calculation and then overwritten with the current estimated values. When a feedforward control calculation is specified, the values in the file are read by the software, but not overwritten since estimation is not performed. Note that multiple estimation/control runs of the software for the same cycle cannot be made without resetting the file to its original values between the runs. An example file is presented in Figure 11.3. The first entry is always zero. The next three entries are the estimated dome, interface, and grid temperatures respectively. The fifth entry is the estimated combustion temperature. The sixth and seventh entries are the heat transfer coefficient scaling factors for the on-gas and on-blast cycles respectively.

```
 0
   1325.9582427239
   969.67596959594
   234.72294313120
   1430.0573399585
   1.0050449237208
   1.1063923486482
```

Figure 11.3: Contents of the file temp.in

11.2.4 onblast.in

The file onblast.in contains the process measurements from the previous on-blast cycle. This file is read by the estimator. The data is used to determine the temperature profile, combustion temperature bias, and heat transfer scaling factors that minimize the model prediction errors for the previous cycle. These values are then used for the optimal fuel flow rate profile determination for the subsequent cycle . An example file is presented in Figure 11.4. The first row contains the time stamp for the end of the previous on-gas cycle and the stove number. The time stamp is in the first three columns in which the entries are hours, minutes, and seconds. The fourth column is the stove number. The second row contains the time stamp and stove number of the initial data set for the on-blast cycle. The third and fourth rows contain the corresponding data. For the third row, the first column is the blast air flow rate in m^3/sec, the second column is the oxygen injection flow rate in m^3/sec, the third column is the blast air moisture in g/m^3, the fourth column is the inlet blast air temperature in °C, the fifth column is the inlet blast air pressure in kPa, the sixth column is the outlet blast air temperature in °C, and the seventh column is the outlet blast air pressure in kPa. For the fourth row, the first three columns are the blast air flow through stoves one through three in m^3/sec respectively, the fourth column is the mixer valve position in %open, the next three columns are the dome, interface, and grid temperatures in °C respectively. The fifth row contains the time stamp and stove number of the second data set for the on-blast cycle. The next two rows contain the corresponding data. This pattern is repeated for each data set. Data must be collected into the file at the same time interval as the time step for the on-blast cycle model calculations.

```
13 35 35  1
13 38 50  1
 106.3195    5.8393   18.9280  236.7001  377.2585 1276.6816  366.1960
 101.5375   13.5262   12.5310   16.9844 1372.1028 1047.4902  294.7698
13 40 21  1
 106.2680    5.8094   19.0670  236.8062  378.2386 1271.0339  367.4914
  99.5608   13.3919   12.6917   15.2281 1372.2152 1038.6304  278.0809
```

Figure 11.4: Contents of the file onblast.in

11.2.5 ongas.in

The file ongas.in contains the process measurements from the previous on-gas cycle. This file is read by the estimator and is used to determine the temperature profile, combustion temperature bias, and heat transfer scaling factors that minimize the model prediction errors for the previous cycle. An example file is presented in Figure 11.5. The first row contains the time stamp for the end of the previous on-blast cycle, the stove number, and the number of the other stove that was on-gas. The time stamp is in the first three columns in which the entries are hours, minutes, and seconds. The fourth column is the stove number and the fifth column is the number of the other stove that was on-gas. The other stove on-gas is required in order to determine the mixed fuel flow rate to the stove. The fuel rate is determined as the product of the total mixed fuel gas flow rate, which is a compensated value, and the ratio of the uncompensated mixed fuel stove flow rate to the sum of the uncompensated mixed fuel flow rates of each stove on-gas. The second row contains the time stamp and stove numbers of the initial data set for the on-gas cycle. The third through sixth rows contain the corresponding data. For the third row, the first three columns contain the uncompensated mixed fuel gas flow rates to each stove in m^3/sec respectively, the fourth column contains the top gas flow rate to mixed fuel gas in m^3/sec, the fifth column contains the top gas temperature in °C, the sixth column contains the top gas pressure in kPa, the seventh column contains the top gas CO in mol%. For the fourth row, The first column contains the top gas H_2 in mol%, the second column contains the top gas CO_2 in mol%, the third column contains the top gas N_2 in mol%, the fourth column contains the natural gas flow rate in m^3/sec, and the fifth through seventh columns are not used. For the fifth row, the first column contains the combustion air temperature in °C, the second column contains the combustion air moisture in g/m^3, the third column contains the waste gas excess oxygen in mol%, the fourth through sixth columns contain the dome, interface, and grid temperatures in °C respectively. For the sixth row, the first column is the waste gas combustible components and the second column is the waste gas methane in ppm. The seventh row contains the time stamp and stove numbers of the second data set for the on-gas cycle. The next four rows contain the corresponding data. This pattern is repeated for each data set. Data must be collected into the file at the same time interval as the time step for the on-gas cycle model calculations.

```
12 38 57  1  3
12 41 59  1  3
   21.2393    0.7268   28.1687   42.5907   31.6846   17.7021   23.2301
    4.3659   23.1960   48.8818    2.5727   26.4898   -0.0392   28.6594
   26.7365   13.4643    4.9810 1341.4543  920.1892  229.4793
    0.0452    0.0095
12 43 29  1  3
   23.9625    0.5795   28.0420   45.4339   31.9745   17.8734   23.4063
    4.2591   23.3082   48.6977    2.6983   34.9946   -0.0372   29.5011
   26.7702   13.5090    0.9858 1353.3950  929.3666  231.2478
    0.0453   -0.0261
```

Figure 11.5: Contents of the file ongas.in

11.3 Output Files

The output files generated by the estimation/control software are presented in this section. The information contained in these files is the result of the estimation/control software execution. These files are updated with new values after the software is executed.

11.3.1 temp.in

The file temp.in contains the estimated stove temperatures, combustion temperature, and heat transfer scaling factors. The previously estimated stove temperatures are read by the the estimation/control software before the values are overwritten by the estimator after each estimation/control calculation. A detailed description of this file is contained in Section 11.2.3.

11.3.2 control.out

The file control.out contains the results from the control calculation. It is output after the calculations have been successfully completed. If the file is not output by the software, the calculation failed. A detailed description of the contents of this file is presented in Section 11.4.1.

11.3.3 dump.out

The file dump.out contains detailed information about the estimation and control calculations. It is output as the calculations are being performed. If the software fails during a calculation, the source of the failure may be identified from examining this file. A detailed description of the contents of this file is presented in Section 11.4.2.

11.3.4 est_blst.out

The file est_blst.out contains the measured and predicted dome, interface, and grid temperatures, in °C, and blast air stove mass flow rates, in g/sec, respectively for the previous on-blast cycle. It is output after an estimation/control calculation by the estimator. The predicted values are those determined after estimating the stove temperature profile, combustion temperature bias, and heat transfer coefficient scaling factors. It is output for estimator performance information only and is not used by the estimation/control software for any other purpose. The file is in **octave** text file format. An example file is presented in Figure 11.6.

```
# name: blast
# type: matrix
# rows:   19
# columns: 8
 1372.10  1407.79  1047.49  1061.88   294.77   310.14  124644.38  123274.02
 1372.22  1403.45  1038.63  1053.88   278.08   302.32  122322.01  123063.78
```

Figure 11.6: Contents of the file est_blst.out

11.3.5 est_gas.out

The file est_gas.out contains the measured and predicted dome, interface, and grid temperatures, in °C, respectively for the previous on-gas cycle. It is output after an estimation/control calculation by the estimator. The predicted values are those determined after estimating the stove temperature profile, combustion temperature bias, and heat transfer coefficient scaling factors. It is output for estimator performance information only and is not used by the estimation/control software for any other purpose. The file is in **octave** text file format. An example file is presented in Figure 11.7.

```
# name: gas
# type: matrix
# rows:   33
# columns: 6
 1341.45  1334.41   920.19   941.41   229.48   236.53
 1353.39  1342.68   929.37   947.45   231.25   238.77
```

Figure 11.7: Contents of the file est_gas.out

11.3.6 diter.out

The file diter.out contains information concerning the performance of the numerical technique used to solve the stove model. This file is only output when flag grf is set to 1 in the file stove.in and will be output for every model calculation. This file should not be output during normal operation. It should only be used to debug the model solution technique when the stove model is run in stand-alone mode. The file contains information on the Newton and linear system solver steps of the Newton-Krylov solution technique. A description of the values presented is contained in the file.

11.3.7 dtemp.out & dprop.out

The files dtemp.out and dprop.out contain a detailed output of all the physical parameter values during a model calculation. These files are only output when flag spout is set to 1 in the file stove.in and will be output for every model calculation. These files should not be output during normal operation. They should only be used to examine the model solution when the stove model is run in stand-alone mode. The file dtemp.out contains the model predicted data for each measured value at each time step in the stove cycles. The file dprop.out contains the calculated physical properties of the gas and solid along the length of the stove at each time step in the stove cycles. A description of the values presented is contained in each file.

11.4 Calculation Results

The fuel flow rate profile resulting from an estimation/control calculation and a feedforward control calculation is output to the file control.out. Information concerning the estimation/control calculations is contained in the file dump.out.

```
4
0    25.753644102923
12    25.511400888698
24    25.259639148923
36    24.748645574722
  25.280353305210
  1.2955015667576
1
  1223.2290086545
  1549.2302826889      896.23079935204
  1399.0953169059      830.40019681809
  364.08132109351      118.60210224616
  2.5156737913003D-02
19
```

Figure 11.8: Contents of the file control.out

11.4.1 control.out

The file control.out contains the results from the control calculation. It is output after the calculations have been successfully completed. If the file is not output by the software, the calculation failed. This file also contains the biased constraint values for the controller. These values are determined during the initial estimation/control calculation and output to this file in order to avoid recalculation when a feedforward control calculation is specified. In this case, the file is input for a feedforward control calculation before being overwritten. An example file is presented in Figure 11.8. The first row contains the number of fuel flow rates in the profile. The next n rows contain the fuel flow rates in the profile in m^3/sec in which n is the number of fuel flow rates entry in the first row. The second column of these rows is the fuel flow rate and the first column is the starting time of that flow rate specified in minutes from the start of the on-gas cycle. The following four rows are the average fuel flow rate for the on-gas cycle in m^3/sec, the calculated air/fuel ratio required to achieve the waste gas excess oxygen target, the final status of the control optimization from GRG2, and the computed mixed fuel gas density in g/m^3, respectively. The following three rows contain the biased maximum and minimum temperature constraints for the dome, interface, and grid temperatures in °C respectively. The following column is the biased minimum blast air by-pass fraction constraint in mole fraction. The last column is the number of data sets for the previous on-blast cycle.

11.4.2 dump.out

The file dump.out contains detailed information about the estimation and control calculations. An example dump.out file is presented in the sequel.

The first item in the file is the run status flag. It is the first entry in the file target.in that specifies whether an estimation/control or a feedforward control calculation is to be performed. In this example, an estimation/control calculation is performed.

```
Run status =   0
```

The next several items describe the estimation optimization. The values for the optimization parameters are first displayed. The optimization parameters are specified in the file param.in and in the subroutine main_estim.f. They are described in the GRG2 reference manual [24] and will not be discussed further here.

```
NUMBER OF VARIABLES IS     6
NUMBER OF FUNCTIONS IS     2
SPACE RESERVED FOR HESSIAN HAS DIMENSION     6
LIMIT ON BINDING CONSTRAINTS IS     1
ACTUAL LENGTH OF Z ARRAY IS    170
  0.1000000E+01  0.1000000E+01  0.1000000E+01  0.0000000E+00  0.0000000E+00
  0.0000000E+00
EPNEWT =  0.1000E-04  EPINIT =  0.1000E-04  EPSTOP =  0.5000E-04
EPPIV =  0.1000E-02  PH1EPS = 0.00000E+00
NSTOP =    2  ITLIM =    8  LIMSER =      1000
IPR =    1  PN4 =    0  PN5 =    0  PN6 =    0  PER =    0
QUADRATIC EXTRAPOLATION FOR INITIAL ESTIMATES OF BASIC VARIABLES WILL BE USED.
THE FINITE DIFFERENCE PARSH USING FORWARD DIFFERENCING
WITH STEP SIZE =  5.00000E-06 WILL BE USED
OBJECTIVE FUNCTION WILL BE MINIMIZED.
LIMIT ON HESSIAN IS     6
```

The initial decision variable values for the estimation optimization are then displayed. Note that the objective to be minimized is that presented in Equation 9.1, and the constraints are those shown in Equation 9.2. The first two decision variables shown in Section 2 are on-gas and on-blast heat transfer coefficient scaling factors, the next is the estimated combustion temperature, and the last three are the dome, interface, and grid temperature profile biases. The constraint limit for the combustion temperature is scaled based on the maximum and minimum combustion temperature limits specified in the file param.in. The constraint limits for the stove temperatures are scaled to be between −1 and 1 in which the values are unscaled in the subroutine main_estim.f. The initial values for the decision variables are the values calculated by the stove model.

```
                              OUTPUT OF INITIAL VALUES

Estimation Optimization Cycle

SECTION 1 -- FUNCTIONS
     FUNCTION                    INITIAL        LOWER         UPPER
NO.   NAME     STATUS  TYPE       VALUE         LIMIT         LIMIT
  1  OBJECTIV          OBJ   0.2962271
  2  DELTA T           RNGE   376.4114       100.0000      1.0000000E+31


SECTION 2 -- VARIABLES
    VARIABLE            INITIAL         LOWER          UPPER
NO.   NAME     STATUS     VALUE          LIMIT          LIMIT
  1  GAS RE            1.0000000E+00  5.5000000E-01  2.0000000E+00
  2  BLAST RE          1.0000000E+00  5.5000000E-01  2.0000000E+00
  3  COMB TEM          1.0000000E+00  8.2761547E-01  1.1081631E+00
  4  DOME TEM          0.0000000E+00 -1.0000000E+00  1.0000000E+00
  5  INTR TEM          0.0000000E+00 -1.0000000E+00  1.0000000E+00
  6  GRID TEM          0.0000000E+00 -1.0000000E+00  1.0000000E+00
```

The next section displays the iterations of the GRG2 algorithm. The information displayed is presented in the GRG2 reference manual [24] and will not be further discussed here.

```
ITN   OBJECTIVE   BINDING  SUPER  INFEAS  NORM RED  HESSIAN  UP   STEP
NO.   FUNCTION    CONSTRS  BASIC  CONSTR  GRADIENT  COND NO  DATE SIZE
  0  2.962271E-01     0       6      0   4.334E-01 1.000E+00  F 0.000E+00
  0  2.962271E-01     0       6      0   4.334E-01 1.000E+00  F 0.000E+00
  1  2.942429E-01     0       6      0   1.070E-01 6.073E+01  T 1.891E-02
  1  2.942429E-01     0       6      0   1.070E-01 6.073E+01  T 1.891E-02
  2  2.719544E-01     0       6      0   8.500E-02 8.841E+01  T 1.643E+00
  2  2.719544E-01     0       6      0   8.500E-02 8.841E+01  T 1.643E+00
  3  2.704855E-01     0       6      0   7.880E-02 1.017E+02  T 1.647E+00
  3  2.704855E-01     0       6      0   7.880E-02 1.017E+02  T 1.647E+00
  4  2.704029E-01     0       6      0   7.682E-03 1.087E+02  T 1.124E+00
  4  2.704029E-01     0       6      0   7.682E-03 1.087E+02  T 1.124E+00
```

```
COULD NOT DROP ANY CONSTRAINT. TRY -VE GRADIENT DIRECTION.
  5  2.703985E-01     0       6      0   4.386E-04 1.000E+00  F 1.533E+00
  5  2.703985E-01     0       6      0   4.386E-04 1.000E+00  F 1.533E+00
TOTAL FRACTIONAL CHANGE IN OBJECTIVE LESS THAN  5.00000E-05
          FOR   2 CONSECUTIVE ITERATIONS
TOTAL FRACTIONAL CHANGE IN OBJECTIVE LESS THAN  5.00000E-05
          FOR   2 CONSECUTIVE ITERATIONS
  6  2.703985E-01     0       6      0   4.592E-04 1.000E+00  F 3.194E-02
  6  2.703985E-01     0       6      0   4.592E-04 1.000E+00  F 3.194E-02
```

The next section is the final results of the estimation optimization problem. This optimization terminated because the fractional change in the value of the objective function in Equation 9.1 was less than the convergence tolerance value specified in the file param.in for two consecutive iterations. Essentially every estimation optimization should terminate for this reason since a rather large convergence tolerance is specified due to execution time limitations for the estimator. There should not be a failure in the estimation optimization. If this does occur, the most likely cause is unreasonable data and/or estimation parameters being used in the optimization. All of these values should be carefully checked if the estimation fails. The initial and final values of the objective, in Section 1, and decision variables, in Section 2, are then displayed. Note that the estimation increased both heat transfer scaling coefficients, decreased the calculated combustion temperature, and decreased the temperature profile in the stove for the example presented.

```
               FINAL RESULTS

Estimation Optimization Cycle

  REASON FOR TERMINATION: FRACTIONAL CHANGE IN OBJECTIVE

SECTION 1 -- FUNCTIONS
                                                    DISTANCE
               INITIAL      FINAL                     FROM      LAGRANGE
NO.   NAME      VALUE       VALUE     STATUS    NEAREST    MULTIPLIER
                                                     BOUND
  1 OBJECTIV 2.96227E-01 2.70398E-01   OBJ
  2 DELTA T  3.76411E+02 3.56720E+02   FREE   2.567E+02:U

SECTION 2 -- VARIABLES
                                                    DISTANCE
               INITIAL      FINAL                     FROM      REDUCED
NO.   NAME      VALUE       VALUE     STATUS    NEAREST     GRADIENT
```

```
                                                 BOUND
  1 GAS RE   1.00000E+00 1.00504E+00 SUPBASIC 5.050E-01:L-1.84201E-05
  2 BLAST RE 1.00000E+00 1.19639E+00 SUPBASIC 6.264E-01:L-1.05508E-04
  3 COMB TEM 1.00000E+00 9.88971E-01 SUPBASIC 1.192E-01:U-4.59224E-04
  4 DOME TEM 0.00000E+00 4.62924E-02 SUPBASIC 9.537E-01:U 3.85997E-04
  5 INTR TEM 0.00000E+00 2.43208E-01 SUPBASIC 7.568E-01:U 4.94862E-05
  6 GRID TEM 0.00000E+00 1.40873E-01 SUPBASIC 8.591E-01:U-3.29829E-04

NUMBER OF DEGREES OF FREEDOM (NVARS-# ACTIVE CONSTRAINTS) =    6

RUN STATISTICS
Estimation Optimization Cycle
NUMBER OF ONE-DIMENSIONAL SEARCHES =      6
NEWTON CALLS =    0  NEWTON ITERATIONS =    0  AVERAGE =    0.00
FUNCTION CALLS =   28  GRADIENT CALLS =     8
ACTUAL FUNCTION CALLS (INC. FOR GRADIENT) =     76
NUMBER OF TIMES BASIC VARIABLE VIOLATED A BOUND =     0
NUMBER OF TIMES NEWTON FAILED TO CONVERGE =     0
TIMES STEPSIZE CUT BACK DUE TO NEWTON FAILURE =     0
```

The next section displays the maximum and minimum heat transfer coefficient values after scaling for the on-gas and on-blast cycles respectively, the initial dome, interface, and grid temperatures, and the estimated combustion temperature used for the control calculation.

```
Estimation heat transfer limits:
  1    2.9910588157835D-03
  2    8.2432470172412D-04
  3    2.8140035376136D-03
  4    1.3237588947849D-03

Estimation initial temperatures:
    1341.1570990539
    984.43720403745
    239.02231646597

Estimation combustion temperature:
    1430.0573399585
```

The next section displays the results of the constraint calculations for the controller. Each temperature constraint for the stove is biased by the difference between the model predicted value after estimation and the measured value. For the minimum constraint, the difference is taken at the end of the previous on-blast cycle. For the maximum constraint, the difference is taken at the end of the previous on-gas cycle. Biasing is used to take into account the difference between the model predictions and the actual values. The constraints are displayed in the order of dome, interface, and grid. The mixing valve constraint is actually a blast air stove by-pass flow fraction constraint in the control software. It is computed by comparing the difference between the final blast air stove by-pass flow fraction computed from the final mixer valve position and that determined from the model using the previous on-gas cycle conditions. The constraint is then computed using a target filter as shown in Equation 8.4.

```
Minimum Temperature Constraints:
    896.23079935204    830.40019681809    118.60210224616
Maximum Temperature Constraints:
    1549.2302826889    1399.0953169059    364.08132109351
Previous mixing valve =    1.45310000000000
Previous by-pass flow =     7.9156411362017D-02
Target by-pass flow:     9.5950708865301D-02
Initial By-pass Difference =     1.6794297503284D-02
Average mixed fuel =     24.002224242424
Simulation by-pass flow:     2.0958163537182D-02
By-pass constraint =     2.5156737913003D-02
```

The next several items describe the control optimization. The values for the optimization parameters are first displayed. The optimization parameters are specified in the file param.in and in the subroutine main_cntrl.f. They are described in the GRG2 reference manual [24] and will not be discussed further here except to note that a Davidon-Fletcher-Powell conjugate gradient approach is used for the control optimization.

```
NUMBER OF VARIABLES IS     4
NUMBER OF FUNCTIONS IS     8
SPACE RESERVED FOR HESSIAN HAS DIMENSION     4
LIMIT ON BINDING CONSTRAINTS IS     7
ACTUAL LENGTH OF Z ARRAY IS    328
  0.2400222E+02  0.2400222E+02  0.2400222E+02  0.2400222E+02
EPNEWT =  0.1000E-03  EPINIT =  0.1000E-03  EPSTOP =  0.1000E-03
EPPIV =  0.1000E-02  PH1EPS = 0.00000E+00
NSTOP =    2  ITLIM =    8  LIMSER =      1000
IPR =    1  PN4 =    0  PN5 =    0  PN6 =    0  PER =    0
QUADRATIC EXTRAPOLATION FOR INITIAL ESTIMATES OF BASIC VARIABLES WILL BE USED.
THE FINITE DIFFERENCE PARSH USING FORWARD DIFFERENCING
WITH STEP SIZE =  1.00000E-05 WILL BE USED
OBJECTIVE FUNCTION WILL BE MINIMIZED.
LIMIT ON HESSIAN IS     0
```

The initial decision variable values for the control optimization are then displayed. Note that the objective to be minimized is that presented in Equation 8.1, and the constraints are those shown in Equation 8.2. The decision variables shown in Section 2 are the fuel flow rate profile for the on-gas cycle. The constraint limit for the stove temperatures are biased as discussed previously based on the maximum and minimum temperature limits specified in the file param.in. The minimum fuel gas flow rate constraint is specified in the file target.in. The maximum fuel gas flow rate constraint is computed from the maximum waste gas flow rate constraint specified in the file param.in and the calculated air/fuel ratio. The initial value for each of the fuel flow rates is the average fuel flow rate from the previous cycle. The initial value of the stove temperature and by-pass flow fraction constraints is determined from a model run using the initial fuel flow rate profile. The status "***" for the by-pass indicates that it violates the constraint.

```
                              OUTPUT OF INITIAL VALUES

Control Optimization Cycle

SECTION 1 -- FUNCTIONS
     FUNCTION                    INITIAL        LOWER         UPPER
NO.   NAME     STATUS  TYPE        VALUE         LIMIT         LIMIT
  1  OBJECTIV          OBJ    24.00222
  2  MAX DOME          RNGE   1426.714     -1.0000000E+31   1549.230
  3  MIN DOME          RNGE   1315.645       896.2308      1.0000000E+31
  4  MAX INTR          RNGE   1087.823     -1.0000000E+31   1399.095
  5  MIN INTR          RNGE   968.2154       830.4002      1.0000000E+31
  6  MAX GRID          RNGE   324.3777     -1.0000000E+31   364.0813
  7  MIN GRID          RNGE   248.2655       118.6021      1.0000000E+31
  8  BY-PASS    ***    RNGE  2.0958164E-02  2.5156738E-02  1.0000000E+31

SECTION 2 -- VARIABLES
    VARIABLE             INITIAL         LOWER           UPPER
NO.   NAME     STATUS     VALUE          LIMIT           LIMIT
  1  FUEL #1           2.4002224E+01  2.0000000E+01  3.0965694E+01
  2  FUEL #2           2.4002224E+01  2.0000000E+01  3.0965694E+01
  3  FUEL #3           2.4002224E+01  2.0000000E+01  3.0965694E+01
  4  FUEL #4           2.4002224E+01  2.0000000E+01  3.0965694E+01

HESSIAN IS TOO LARGE FOR VARIABLE METRIC--SWITCH TO CONJUGATE GRADIENTS
COMPLEMENTARY DFP DIRECTION WILL BE USED
```

The next section displays the iterations of the GRG2 algorithm. The information displayed is presented in the GRG2 reference manual [24] and will not be further discussed here.

```
ITN   OBJECTIVE   BINDING  SUPER  INFEAS  NORM RED  HESSIAN  UP   STEP
NO.   FUNCTION    CONSTRS  BASIC  CONSTR  GRADIENT  COND NO  DATE SIZE
  0  4.198574E-03     0       4      1   1.421E-03 0.000E+00  F 0.000E+00
  0  4.198574E-03     0       4      1   1.421E-03 0.000E+00  F 0.000E+00
  1  2.523434E+01     0       4      0   3.000E-01 0.000E+00  F 1.024E+03
  1  2.523434E+01     0       4      0   3.000E-01 0.000E+00  F 1.024E+03
  2  2.490810E+01     1       3      0   1.951E-03 0.000E+00  F 1.288E+00
  2  2.490810E+01     1       3      0   1.951E-03 0.000E+00  F 1.288E+00
  3  2.489246E+01     1       3      0   5.373E-03 0.000E+00  F 9.933E+02
  3  2.489246E+01     1       3      0   5.373E-03 0.000E+00  F 9.933E+02
  4  2.488447E+01     1       3      0   3.186E-03 0.000E+00  F 4.467E-01
  4  2.488447E+01     1       3      0   3.186E-03 0.000E+00  F 4.467E-01
  5  2.488187E+01     1       3      0   1.972E-03 0.000E+00  F 4.467E-01
  5  2.488187E+01     1       3      0   1.972E-03 0.000E+00  F 4.467E-01
COULD NOT DROP ANY CONSTRAINT. TRY -VE GRADIENT DIRECTION.
  6  2.488107E+01     1       3      0   1.301E-03 0.000E+00  F 4.467E-01
  6  2.488107E+01     1       3      0   1.301E-03 0.000E+00  F 4.467E-01
TOTAL FRACTIONAL CHANGE IN OBJECTIVE LESS THAN  1.00000E-04
          FOR   2 CONSECUTIVE ITERATIONS
TOTAL FRACTIONAL CHANGE IN OBJECTIVE LESS THAN  1.00000E-04
          FOR   2 CONSECUTIVE ITERATIONS
  7  2.488035E+01     1       3      0   2.709E-04 0.000E+00  F 2.287E+02
  7  2.488035E+01     1       3      0   2.709E-04 0.000E+00  F 2.287E+02
```

The following section is the final results of the control optimization problem. This optimization terminated because the fractional change in the value of the objective function in Equation 8.1 was less than the convergence tolerance value specified in the file param.in for two consecutive iterations. The initial and final values of the objective and constraints, in Section 1, and the fuel flow rates, in Section 2, are then displayed. Note that the stove by-pass constraint is at its lower bound and that the controller increased the fuel flow rate since the previous stove by-pass flow fraction was below the target.

```
               FINAL RESULTS
Control Optimization Cycle

  REASON FOR TERMINATION: FRACTIONAL CHANGE IN OBJECTIVE

SECTION 1 -- FUNCTIONS
                                                    DISTANCE
               INITIAL      FINAL                 FROM      LAGRANGE
NO.   NAME      VALUE       VALUE     STATUS    NEAREST    MULTIPLIER
                                                 BOUND
  1 OBJECTIV 2.40022E+01 2.48804E+01   OBJ
  2 MAX DOME 1.42671E+03 1.42703E+03   FREE   1.222E+02:L
  3 MIN DOME 1.31564E+03 1.31931E+03   FREE   4.231E+02:U
  4 MAX INTR 1.08782E+03 1.09216E+03   FREE   3.069E+02:L
  5 MIN INTR 9.68215E+02 9.72056E+02   FREE   1.417E+02:U
  6 MAX GRID 3.24378E+02 3.27772E+02   FREE   3.631E+01:L
  7 MIN GRID 2.48265E+02 2.48740E+02   FREE   1.301E+02:U
  8 BY-PASS  2.09582E-02 2.50572E-02 LOWERBND-9.956E-05:U 2.18366E+02

SECTION 2 -- VARIABLES
                                                DISTANCE
               INITIAL      FINAL                 FROM      REDUCED
NO.   NAME      VALUE       VALUE     STATUS    NEAREST     GRADIENT
                                                 BOUND
  1 FUEL #1  2.40022E+01 2.53536E+01 SUPBASIC 5.354E+00:L-2.70892E-04
  2 FUEL #2  2.40022E+01 2.51114E+01 SUPBASIC 5.111E+00:L-6.05710E-05
  3 FUEL #3  2.40022E+01 2.48596E+01 SUPBASIC 4.860E+00:L 1.36972E-04
  4 FUEL #4  2.40022E+01 2.43486E+01  BASIC   4.349E+00:L

NUMBER OF DEGREES OF FREEDOM (NVARS-# ACTIVE CONSTRAINTS) =     3

RUN STATISTICS
Control Optimization Cycle
NUMBER OF ONE-DIMENSIONAL SEARCHES =      7
NEWTON CALLS =   27  NEWTON ITERATIONS =    8  AVERAGE =    0.30
FUNCTION CALLS =   48  GRADIENT CALLS =     9
ACTUAL FUNCTION CALLS (INC. FOR GRADIENT) =     84
NUMBER OF TIMES BASIC VARIABLE VIOLATED A BOUND =      1
NUMBER OF TIMES NEWTON FAILED TO CONVERGE =     0
TIMES STEPSIZE CUT BACK DUE TO NEWTON FAILURE =     0
```

The final section summarizes the fuel flow rate profile for the on-gas cycle. Note that the fuel flow rates displayed in this section are higher than those shown in the optimization final results. Since pellet dumps to the blast furnace tend to cause pressure disturbances to the top gas, the average mixed fuel gas flow rate tends to be below the setpoint value during the on-gas cycle. A bias value, specified in the file param.in, is added to the profile determined by the optimization to compensate for this effect. The value of Inform is the final status of the estimation/control calculation. A value of zero indicates that the control optimization found an optimal point that satisfied the Kuhn–Tucker conditions. A value of one indicates that the control optimization terminated due to the change in the objective function being less than the convergence tolerance for two successive iterations. A value of ten indicates that the control optimization could not find a feasible solution the optimization problem. This condition is discussed in detail in Section 11.6.

```
fuel(  1) =   0    25.753644102923
fuel(  2) =   12    25.511400888698
fuel(  3) =   24    25.259639148923
fuel(  4) =   36    24.748645574722
avgfuel =     25.280353305210
air_fuel =     1.2955015667576
Inform =   1
```

11.5 Tuning Parameter Selection

The parameters that would normally be changed by users of the technology on a regular basis have been placed in the file param.in discussed in Section 11.1.3. This file contains the tuning parameters and constraint values for the estimation and control optimization. Selection of the values for these parameters is discussed in this section in the order the parameters appear in the file.

The maximum and minimum heat transfer scaling and combustion temperature constraint values for the estimation optimization in Equation 9.2 should represent a physically reasonable range for the parameter values that allow a sufficient degree of freedom for the estimation. If the range is too large, unreasonable estimates of the parameters may occur. If the range is too small, the model may not be able to adapt to changes in the operation of the stoves due to operational changes, ambient weather conditions, and other factors. Care should be exercised when changing these values to ensure that these conditions are met.

```
 2.d+0     maxgsht    (maximum on-gas heat transfer scaling)
 5.5d-1    mingsht    (minimum on-gas heat transfer scaling)
 2.0d+0    maxblht    (maximum on-blast heat transfer scaling)
 5.5d-1    minblht    (minimum on-blast heat transfer scaling)
 1600.0    maxcomb    (maximum combustion temperature)
 1200.0    mincomb    (minimum combustion temperature)
```

The absolute maximum increase or decrease in the dome, interface, and grid temperatures due to the linear correction to the stove temperature profile should represent a reasonable error bound on the temperature measurements. Increasing these values can lead to physically unrealistic stove temperature profiles. Decreasing these values will reduce the ability of the estimation to match the temperature and stove blast air flow rate profiles.

```
100.0     tmpscl(1)  (dome temperature change scaling)
100.0     tmpscl(2)  (interface temperature change scaling)
35.0      tmpscl(3)  (grid temperature change scaling)
```

A large estimation convergence tolerance is used to reduce the computation time required by the estimation optimization. Decreasing this tolerance can result in a significant increase in the computation time with only a marginal decrease in the estimation objective. Verification that the estimation optimization completes execution within the blow-off cycle should be performed every time this value is changed. It should not be changed by more than an order of magnitude in either direction.

```
5.d-5     estconv    (estimation convergence tolerance)
```

The weighting factors for the estimation optimization are selected based on the discussion presented in Chapter 9. These weighting factors are the α_n values in Equation 9.3. Significant changes to these weights can result in physically unrealistic parameter values. The magnitude of the weights were initially selected to obtain an objective function value on the order of one. Tuning is accomplished by changing the relative ratio of the weights. Adjusting all of the weights by the same fraction will have no effect on the solution of the estimation optimization.

```
0.3       domwght    (estimation objective dome temperature weight)
3.0       intwght    (estimation objective interface temperature weight)
3.0       grdwght    (estimation objective grid temperature weight)
1.0       flowght    (estimation objective stove flow weight)
5         initbltm   (first point for matching on-blast flow rate history)
100.0     mindidif   (minimum temperature difference; dome and interface)
0.15      htcwght(1) (change in on-gas heat transfer scale weight)
0.15      htcwght(2) (change in on-blast heat transfer scale weight)
0.01      combwght   (change in combustion temperature weight)
0.1       tmpwght    (change in temperature change weight)
```

The estimation does not attempt to match the first part of the stove blast air flow rate profile during the on-blast cycle. The model does not account for the energy contained in the combustion zone after the on-gas cycle, which appears to cause the stove blast air flow rate to flatten at the beginning of the on-blast cycle. In order to prevent the initial portion of this profile from affecting the estimated model parameters, it is ignored. The data point in the file onblast.in, discussed in Section 11.2.4, that the estimation uses as the starting point is specified after the stove flow weighting parameter.

In order to prevent the estimation from skewing the stove temperature profile too much, a constraint is specified on the minimum difference between the estimated dome and interface temperatures. This constraint will prevent the estimation from either reducing the dome temperature, increasing the interface temperature, or both, such that the difference is less than the value specified in this entry. The value should not be greater than the nominal difference between the dome and interface temperatures during normal operation. It is specified in the entry above the heat transfer coefficient, combustion temperature, and temperature profile weight entries.

The stove temperature maximum and minimum constraint entries should reflect the operating limits of the stove. Note that the controller is not designed to trade-off constraint violations. For example, the controller will not increase one constraint violation in order to decrease another. This point is discussed in more detail in Section 11.6.2.

```
1500.0    mxdomlim   (maximum dome temperature constraint)
900.0     mndomlim   (minimum dome temperature constraint)
1350.0    mxintlim   (maximum interface temperature constraint)
800.0     mnintlim   (minimum interface temperature constraint)
360.0     mxgrdlim   (maximum grid temperature constraint)
100.0     mngrdlim   (minimum grid temperature constraint)
```

The final mixer valve target bias is used to compensate for any mismatch between the model predicted and actual changes in the final mixer valve position. It is selected by observing the average difference between the actual final mixer valve and the target value over a number of cycles. It can also be used to specify a minimum final mixer valve target value for the control calculations.

```
2.0       mxr_bias   (final mixer valve target bias)
```

The target filter, shown in Equation 8.4, determines the response speed of the controller and compensates for differences between the model and the actual stove in the sensitivity of the final mixer valve position to the on-gas cycle fuel profile. Care must be exercised when changing these values. If they are increased too much, the fuel rate profile may oscillate between cycles causing controller-induced variation in the final mixer valve position. If they are decreased too much, the controller will be too sluggish and not be able to respond to disturbances to the stove system. Note that these parameters should be selected such that the upward move filter value is larger than the downward move filter value. With this tuning, the controller will be more aggressive when the final mixer valve position is below the target and the fuel flow rate profile must be increased, than when the final mixer valve position is above its target. This tuning allows for a faster control response when the fuel flow rate profile must be increased.

```
2.5d-1    byp_fup    (final mixer valve target filter, upward moves)
1.0d-1    byp_fdn    (final mixer valve target filter, downward moves)
```

The maximum waste gas flow rate constraint is used to determine the maximum mixed fuel gas flow rate constraint for the control optimization. The fuel constraint is calculated by dividing the waste gas constraint by the current air/fuel ratio computed by the model. This value should reflect the actual limit.

```
 72.0      maxigas    (maximum waste gas flow rate constraint)
```

The mixed gas fuel flow rate bias is added to each of the mixed gas fuel flow rates in the profile determined by the control optimization. This bias is used to compensate for the top gas flow rate disturbances caused by pellet dumps that cause the average mixed gas fuel flow rate to be consistently below the target value. This value should be set based on the average difference between the mixed gas flow rate target and the actual flow rate value. This difference can change significantly over time depending on the blast furnace operation.

```
 0.40      fuelbias   (mixed gas fuel flow rate bias)
```

The control optimization convergence tolerance is selected as a compromise between the computation time required by the control optimization and the resulting optimal fuel flow rate profile. Decreasing this tolerance will result in a marginal decrease in the objective, which is the average fuel flow rate for the on-gas cycle, but will increase the computation time required for the optimization. Decreasing this tolerance can result in a significant increase in the computation time with only a marginal decrease in the average fuel flow rate. Increasing this tolerance will increase the fuel flow rate profile. Verification that the control optimization completes its execution within the blow-off cycle should be performed every time this value is changed. It should not be changed by more than an order of magnitude in either direction.

```
 1.d-4     cntconv    (control convergence tolerance)
```

11.6 Software Error Conditions

This section discusses the error conditions that may occur with the software. Estimation and control optimization errors are discussed in Section 11.6.1. The most common error is a control final status of 10 indicating that the control optimization could not find a feasible solution. This condition is discussed in Section 11.6.2. Model calculation errors are discussed in the final section.

11.6.1 Estimation and Control Errors

Essentially every estimation optimization should terminate due to a fractional change in the objective, status = 1. There should not be a failure in the estimation optimization. If this does occur, the most likely cause is unreasonable data and/or estimation parameters being used in the optimization. All of these values should be carefully checked if the estimation fails.

The controller optimization will terminate due to satisfaction of the Kuhn–Tucker conditions, status = 0, or due to a fractional change in the objective, status = 1, if it was successful. If the optimization could not find a feasible starting point, status = 10. The most likely cause is that a maximum temperature constraint occurs when the final mixer valve is below the minimum limit or a maximum fuel constraint. In either case, the controller can do nothing to satisfy the constraints and uses the average fuel rate from the previous on-gas cycle. In these situations, it is probably advantageous for the operator to take control of the stove.

11.6.2 Control Error Example

An example `dump.out` file is presented to illustrate determining the cause of a final error status of 10 produced by the controller. Starting with the estimation results for this example,

```
               FINAL RESULTS
Estimation Optimization Cycle

   REASON FOR TERMINATION: FRACTIONAL CHANGE IN OBJECTIVE
```

note that the estimation was successful and terminated due to a fractional change in the objective, status=1. The next step is to look at the control results.

```
               FINAL RESULTS
Control Optimization Cycle

  REASON FOR TERMINATION: PH1-FEAS  XXX       XXX
```

The termination condition PH1-FEAS is status 10. This status means that the optimization could not find a feasible solution to the optimization problem. There are two possibilities:

1. The fuel flow rate profile required to meet the final mixer valve minimum constraint will cause the stove to overheat and violate a maximum grid, interface, or dome temperature constraint.

2. The fuel flow rate profile required to meet the final mixer valve constraint is greater than the maximum fuel flow rate constraint.

In order to determine which of these problems is present, examine the “FUNCTIONS” section of the control optimization output for the temperature and mixer valve constraints

```
SECTION 1 -- FUNCTIONS
                                                        DISTANCE
               INITIAL      FINAL                        FROM      LAGRANGE
NO.   NAME      VALUE       VALUE     STATUS           NEAREST    MULTIPLIER
                                                        BOUND
  1 OBJECTIV 3.01613E+01 3.03523E+01   OBJ
  2 MAX DOME 1.37455E+03 1.37456E+03   FREE   1.444E+02:L
  3 MIN DOME 1.31803E+03 1.31848E+03   FREE   4.145E+02:U
  4 MAX INTR 1.07297E+03 1.07405E+03   FREE   2.484E+02:L
  5 MIN INTR 9.58504E+02 9.59455E+02   FREE   1.644E+02:U
  6 MAX GRID 3.11730E+02 3.12446E+02   FREE   7.745E+01:L
  7 MIN GRID 2.20624E+02 2.20763E+02   FREE   9.796E+01:U
  8 BY-PASS  2.91129E-02 2.96435E-02 VIOLATED-5.550E-03:U
```

and note in the “STATUS” field that the BY-PASS constraint is the only violation. The final by-pass flow fraction is used as the constraint instead of the final mixer valve position for the control optimization. The two are related by the correlation between the mixer valve position and the total blast air flow fraction through the stove shown in Equation 8.3. Therefore, there are no temperatures at their constraints and the maximum fuel flow rate constraints must be the problem. To verify this assumption, examine the “VARIABLES” section

```
SECTION 2 -- VARIABLES
                                                        DISTANCE
               INITIAL      FINAL                        FROM      REDUCED
NO.   NAME      VALUE       VALUE     STATUS           NEAREST    GRADIENT
                                                        BOUND
  1 FUEL #1  3.01613E+01 3.03523E+01 NONBASIC  UPPERBND  -6.51617E-04
  2 FUEL #2  3.01613E+01 3.03523E+01 NONBASIC  UPPERBND  -6.48210E-04
  3 FUEL #3  3.01613E+01 3.03523E+01 NONBASIC  UPPERBND  -6.44173E-04
  4 FUEL #4  3.01613E+01 3.03523E+01 NONBASIC  UPPERBND  -8.21532E-04
```

and note that all four fuel flow rates are at their maximum constraint.

If the infeasibility was caused by a temperature constraint violation, one of the temperatures will be violating a maximum constraint in addition to the minimum by-pass constraint violation. In this case, note the violation of the maximum interface temperature and the minimum by-pass constraint in the “FUNCTIONS” section and also note that none of the four fuel flow rates are at their maximum constraint in the “VARIABLES” section.

```
SECTION 1 -- FUNCTIONS
                                                     DISTANCE
              INITIAL      FINAL                       FROM      LAGRANGE
NO.   NAME     VALUE       VALUE     STATUS    NEAREST   MULTIPLIER
                                                      BOUND
  1 OBJECTIV 3.01613E+01 3.03523E+01   OBJ
  2 MAX DOME 1.37455E+03 1.37456E+03   FREE   1.444E+02:L
  3 MIN DOME 1.31803E+03 1.31848E+03   FREE   4.145E+02:U
  4 MAX INTR 1.07297E+03 1.07405E+03 VIOLATED 2.000E+00:L
  5 MIN INTR 9.58504E+02 9.59455E+02   FREE   1.644E+02:U
  6 MAX GRID 3.11730E+02 3.12446E+02   FREE   7.745E+01:L
  7 MIN GRID 2.20624E+02 2.20763E+02   FREE   9.796E+01:U
  8 BY-PASS  2.91129E-02 2.96435E-02 VIOLATED-5.550E-03:U

SECTION 2 -- VARIABLES
                                                     DISTANCE
              INITIAL      FINAL                       FROM      REDUCED
NO.   NAME     VALUE       VALUE     STATUS    NEAREST     GRADIENT
                                                      BOUND
  1 FUEL #1  3.01613E+01 3.03523E+01 NONBASIC -1.00E+00:U -6.51617E-04
  2 FUEL #2  3.01613E+01 3.03523E+01 NONBASIC -1.00E+00:U -6.48210E-04
  3 FUEL #3  3.01613E+01 3.03523E+01 NONBASIC -1.00E+00:U -6.44173E-04
  4 FUEL #4  3.01613E+01 3.03523E+01 NONBASIC -1.00E+00:U -8.21532E-04
```

The maximum interface temperature is violated by 2°C and the minimum stove by-pass flow fraction is violated by -0.00555 in this example. In this case, the controller is not able to resolve the conflicting constraints. If the fuel flow rate profile is increased, the final by-pass constraint will be relieved, but the interface temperature constraint violation will increase. The converse is true if the fuel flow rate profile is reduced.

11.6.3 Model Calculation Errors

The most likely cause of a calculation error in the software is an invalid parameter or data value used in the calculations. The software development did not include exhaustive parameter and data validation as part of the project. Therefore, the user must either provide this functionality in the code or carefully check the inputs to the software when a failure does occur. The first place to examine is the file dump.out. It is output as the calculations progress, so it should be an indication of where in the calculations the failure occurred.

Chapter 12

Software Documentation

This section provides the software documentation for the advanced control technology developed during the Phase One effort of this project. This software was developed at Los Alamos National Laboratory by the University of California (University) under Contract W-7405-ENG-36 with the United States Department of Energy (DOE). All rights are reserved by DOE on behalf of the Government and the University pursuant to the contract. Neither the Government nor the University makes any warranty, express or implied, or assumes any liability or responsibility for use of this software. Any copies made of this software should carry this notice.

The software developed for the Phase One effort of this project should be considered as research code intended to demonstrate the technology. It was not developed as, or ever intended to be, a commercial software product. A number of issues that are important for a commercial software product have not been addressed in the development of this technology. The issues include an interactive user interface, exhaustive parameter and data validation, and a general process control system interface.

This software was also developed for implementation on the No. 7 blast furnace stoves at the Ispat Inland East Chicago facility for demonstration of the technology. Although we expect similar results for other blast furnace stove systems, some modification to the software is most likely required to implement this technology on other blast furnace stove systems. The interface between this software and the process monitoring and control system computer network was designed for demonstration purposes only and is specific to the Ispat Inland process monitoring and control computer system. Modification to the interface is also most likely required for implementation on other blast furnace stove systems. A development effort would be required to generalize and commercialize this software.

The advanced control technology is implemented as a FORTRAN 77 program consisting of one main program and a number of subroutines. The software was developed at Los Alamos on a Digital Equipment Corp. Alpha 500 AU workstation under the

DEC Unix operating system and on a Sun UltraSparc 10 workstation under the Solaris operating system. The software is running at the Ispat Inland East Chicago facility on a Digital Equipment Corp. Alpha 500 AU workstation under the DEC VMS operating system. The only software modification required to switch between operating systems was the FORTRAN OPEN statements used to access the data files. A version of the correct OPEN statement for each operating system is contained in the code in which the statement for the version not being used is changed to a comment.

This section is organized as follows. Auxiliary software packages that are used to implement the technology but not developed at Los Alamos are discussed in Section 12.1.1. Detailed software documentation is not provided for these packages in this report. The variables used in the software are documented in Section 12.2. The main program is discussed in Section 12.3. Each of the FORTRAN subroutines comprising the software package are outlined in Section 12.4

12.1 Auxiliary Software Packages

A number of auxiliary software packages are used for the implementation of the advanced control technology. These packages are discussed in the following sections.

12.1.1 Interface Software

The interface between the Ispat Inland process control and monitoring computer system and the DEC Alpha 500 AU computer used to execute the model-based control algorithm is through a series of data files transferred between the computers. The data files presented in Section 11.2 that contain the process measurements required by the model-based controller are generated by the Ispat Inland process monitoring and control computer system using software developed by Ispat Inland personnel. Execution of the advanced control software program at the end of each on-blast cycle and implementation of the resulting optimal fuel flow rate profile for each stove contained in the data files presented in Section 11.3 are also performed with software developed by Ispat Inland personnel. Since this software was not developed at Los Alamos and is specific to the Ispat Inland No. 7 blast furnace process monitoring and control computer system, this interface software is not documented in this report.

12.1.2 GRG2 Software

The GRG2 optimization software used for the control and estimation calculations is commercial technology that has been licensed from Optimal Methods Inc. by Ispat Inland Steel for this application. It is not public domain software and cannot be freely distributed. Detailed documentation of the GRG2 optimization software is available in the GRG2 reference manual [24] distributed with the software. Due to licensing restrictions, documentation of the software and the information provided in the reference manual is not contained in this report. The tuning parameters required for implementing GRG2 are discussed in Section 11.5.

12.1.3 SPARSEKIT software

The linear solver, GMRES, the preconditioner, ILUD, and the supporting sparse matrix utility function implementations used in this project are part of SPARSEKIT. This software package is freely available from the following web site maintained by the package author, Yousef Saad.

```
http://www-users.cs.umn.edu/~saad/
```

Details concerning the specific application of GMRES and ILUD to this project are discussed in Section 7.3. Documentation of the SPARSEKIT software package is provided in [36] and is not contained in this report. The selection and tuning parameters required for GMRES and ILUD are discussed in Section 11.1.2.

12.2 Variable Description

The variables used by the stove model, estimation procedure, and the controller are defined in a series of named FORTRAN common variable blocks contained in the FORTRAN include file stove.h and defined in the main program main_stove.f. The include file also specifies the maximum size of the variable arrays used to store the computation grid and calculation results through a series of FORTRAN parameter statements. These parameters and the contents of each named FORTRAN common block are documented in this section.

12.2.1 Parameter Values

- `parameter (kdim = 50)`
- `parameter (maxtim = 61)`
- `parameter (nxd = 111, nrd = 3)`
- `parameter (nxdj = maxtim*nxd*(nrd + 1))`
- `parameter (nxdjp1 = nxdj + 1)`

-
  ```
  parameter (nnxd = maxtim*(nrd + 1)*((nxd-2)*3 + 4) +
              maxtim*2*nrd*nxd + (maxtim - 1) * (nrd + 1) * nxd)
  ```
- `parameter (maxcntrl = 10)`
- `parameter (maxestim = 20)`
- `parameter (maxcyc = 6)`

12.2.2 Common Block iounit

This FORTRAN Common block contains the logical unit assignments for the data files.

- `ioin`

  This entry is the logical unit used to input data from files.

- `ioout`

  This entry is the logical unit used to output data to files.

- `iodump`

  This entry is the logical unit used to output to the file dump.out.

- `ioterm`

  This entry is the logical unit used to output error messages.

12.2.3 Common Block iinput

This FORTRAN Common block contains the integer valued parameters for the software.

- `ntimg`

  The number of time steps used to represent the on-gas cycle.

- `ntimb`

  The number of time steps used to represent the on-blast cycle.

- `ntimpres`

  The number of time steps used to represent the pressurization cycle.

- `ntimblow`

  The number of time steps used to represent the blow-off cycle.

- `ncyc`

  Number of complete (i.e., on-gas, pressure-up, on-blast, blow-down) stove cycles to simulate.

- `nx`

  The number of axial node points used to represent the stove length.

- `tgasneu`

  Flag for whether the boundary condition at the top of the stove during the on-gas cycle is Neumann or Dirichlet. 1 = Neumann, 0 = Dirichlet

- `bgasneu`

  Flag for whether the boundary condition at the bottom of the stove during the on-gas cycle is Neumann or Dirichlet. 1 = Neumann, 0 = Dirichlet

- `ntim`

  The number of time steps used to represent the current cycle. Note that this quantity changes depending on the cycle being simulated.

- `tblsneu`

  Flag for whether the boundary condition at the top of the stove during the on-blast cycle is Neumann or Dirichlet. 1 = Neumann, 0 = Dirichlet

- `bblsneu`

  Flag for whether the boundary condition at the bottom of the stove during the on-blast cycle is Neumann or Dirichlet. 1 = Neumann, 0 = Dirichlet

- `spout`

  Flag for whether to print the temperature and property diagnostic files `dtemp.out` and `dprop.out`. 1 = print, 0 = do not print

- `grf`

  Flag for whether to print the iteration diagnostic file `diter.out`. 1 = print, 0 = do not print

- `nr`

  The number of radial node points used to represent the checker surrounding each gas channel diameter.

- `axflag`

  Flag for whether to consider the effect of axial heat conduction in the brick during the simulation. 1 = consider axial conduction, 0 = do not consider axial conduction

- `linsol`

  Flag to select the algorithm for solving the linear system.
  1 = LINPACK
  2 = Conjugate Gradient Method
  3 = Conjugate Gradient Method with Normal Residual equation
  4 = Bi-Conjugate Gradient Method
  5 = Bi-Conjugate Gradient Method with Partial Pivoting
  6 = Stabilized Bi-Conjugate Gradient Method
  7 = Transpose-Free Quasi-Minimum Residual Method
  8 = Full Orthogonalization Method
  9 = Generalized Minimum Residual Method
  10 = Flexible version of Generalized Minimum Residual Method
  11 = Direct Quasi-Generalized Minimum Residual Method
  12 = Preconditioned Generalized Minimum Residual Method

- `prconalg`

  Flag to select the algorithm for preconditioning the linear system.
  0 = Incomplete LU Factorization with no fill-in (ILU0)
  1 = Incomplete LU with Diagonal Compensation (MILU0)
  2 = Incomplete LU Factorization with k-level fill-in (ILUK)
  3 = Incomplete LU Factorization with Dual Truncation (ILUT)
  4 = Incomplete LU with Diagonal Compensation and Dropping (ILUD)
  5 = ILUT with partial pivoting (ILUTP)
  6 = ILUD with partial pivoting (ILUDP)

- `levfil`

  Level of fill-in for the ILUT and ILUK preconditioners.

- `maxits`

  Maximum number of iterations allowed for the linear solver.

- `stpcrt`

  Flag to select the stopping criteria for the linear solver.
  -2 = $\|\boldsymbol{x}_l - \boldsymbol{x}_{l-1}\|_2 \leq \epsilon_r \; \|\boldsymbol{b}\|_2 + \epsilon_a$
  -1 = $\|\boldsymbol{x}_l - \boldsymbol{x}_{l-1}\|_2 \leq \epsilon_r \; \|\boldsymbol{x}_0 - \boldsymbol{x}_1\|_2 + \epsilon_a$
  0 = solver automatically chooses criteria number 1
  1 = $\|\boldsymbol{r}_l\|_2 \leq \epsilon_r \; \|\boldsymbol{r}_0\|_2 + \epsilon_a$
  2 = $\|\boldsymbol{r}_l\|_2 \leq \epsilon_r \; \|\boldsymbol{b}\|_2 + \epsilon_a$

- `prconpos`

  Flag to select the configuration of the preconditioner for the linear solver.
  0 = no preconditioner

1 = use left preconditioning
2 = use right preconditioning

- `zone(nxd)`

  The zone number corresponding to each axial node along the length of the stove.

- `lint`

  The number of the axial node that is associated with the simulated interface temperature.

- `ldom`

  The number of the axial node that is associated with the simulated dome temperature.

12.2.4 Common Block rinput

This FORTRAN Common block contains the real valued parameters for the software.

- `bb`

  The amount that the current state values are perturbed when numerically approximating the derivatives in the Jacobian matrix.

- `dtgas`

  The time step size for the on-gas cycle in seconds.

- `dtbls`

  The time step size for the on-blast cycle in seconds.

- `dtpres`

  The time step size for the pressurization cycle in seconds.

- `dtblow`

  The time step size for the blow-off cycle in seconds.

- `xmax`

  The distance from some origin at the top of the stove to the bottom of the brick stack.

- `xmin`

  The distance from some origin at the top of the stove to the top of the brick stack. Note that currently the origin is defined at the top of the brick stack.

- `bgas`

  If the boundary condition at the bottom of the stove during the on-gas cycle is Dirichlet, this value is that fixed gas temperature in °C.

- `tbls`

  If the boundary condition at the top of the stove during the on-blast cycle is Dirichlet, this value is that fixed gas temperature in °C.

- `bbls`

  If the boundary condition at the bottom of the stove during the on-blast cycle is Dirichlet, this value is that fixed gas temperature in °C. Note that this value is *not* used unless the reported cold blast temperatures drop below zero, which we assume represents data drop out.

- `damp`

  The damping coefficient for the Newton update shown in Equation 7.7.

- `tmax`

  If the boundary condition at the top of the stove during the on-gas cycle is Dirichlet, this value is that fixed gas temperature in °C. Note that this value is *not* used unless the computed combustion temperatures drop below zero, which we assume represents data drop out. If data drop out occurs this quantity is also used to scale the simulation temperatures.

- `tinit`

  A reference temperature used to scale the temperatures in the simulation.

- `mdotg`

  If the computed mass flow rate in gm/sec during an on-gas cycle drops below zero, data drop out is assumed to have occurred, and this value is used in place of the computed flow rate.

- `mdotb`

  If the computed mass flow rate in gm/sec during an on-blast cycle drops below zero, data drop out is assumed to have occurred, and this value is used in place of the computed flow rate.

- `rga`

  The average gas density in gm/cm^3. This value is the average of rgb(1) and rgb(2) specified in the parameter file stove.in. It is used to make the differential equations dimensionless.

- `cga`

  The average gas heat capacity in cal/gm-°K. This value is the average of cgb(1) and cgb(2) specified in the parameter file stove.in. It is used to make the differential equations dimensionless.

- `uga`

  The average gas velocity in cm/sec. This value is the average of ugb(1) and ugb(2) specified in the parameter file stove.in. It is used to make the differential equations dimensionless.

- `hga`

  The average heat transfer coefficient in cal/cm^2-sec-°K. This value is the average of hgb(1) and hgb(2) specified in the parameter file stove.in. It is used to make the differential equations dimensionless.

- `ri`

  The average flue diameter in cm. This value is specified in the parameter file stove.in. It is used to make the differential equations dimensionless.

- `nc`

  The number of gas channels, or flues, in the stove.

- `ping`

  If the computed top gas pressure in atm during an on-gas cycle drops below zero, data drop out is assumed to have occurred, and this value is used in place of the computed fuel pressure.

- `pinb`

  If the computed cold blast pressure in atm during an on-blast cycle drops below zero, data drop out is assumed to have occurred, and this value is used in place of the computed blast pressure.

- `relrgh`

  The relative roughness of the checker material.

- `droptol`

  Drop tolerance for the ILUT preconditioner.

- `nlrerr`

  Relative convergence tolerance for the nonlinear system solver.

- `nlaerr`

  Absolute convergence tolerance for the nonlinear system solver.

- `lrerr`

  Relative convergence tolerance for the linear system solver.

- `laerr`

  Absolute convergence tolerance for the linear system solver.

- `thrstol`

  Threshold for dropping terms in the ILUD and ILUDP preconditioners.

- `domloc`

  The distance from xmin to the location of the dome temperature measurement.

- `pi`

  The constant π.

- `delt`

  The *scaled* length of the time step in the current cycle.

- `intloc`

  The distance from xmin to the location of the interface temperature measurement.

- `xp`

  The spatial scaling factor used in the simulation.

- `tgimti`

  The temperature scaling factor used in the simulation.

- `maxtemp`

  The maximum computed combustion temperature during an on-gas cycle. This quantity is used to make the differential equations dimensionless.

- `taugp`

  The temporal scaling factor used in the simulation.

- `diagcomp`

  Diagonal compensation parameter for the ILUD and ILUDP preconditioners. $0 \leq \mu \leq 1$

- `fntmpdom`

  The final measured dome temperature at the end of a cycle.

- `fntmpint`

  The final measured interface temperature at the end of a cycle.

- `fntmpgrd`

  The final measured grid temperature at the end of a cycle.

- `fnmixpos`

  The measured final mixer valve position at the end of the on-blast cycle.

- `bfimti`

  The maximum computed mass flow rate through the stove an on-blast cycle. This quantity is used to make the differential equations dimensionless.

- `storfbyp`

  The simulated final bypass value.

- `permtol`

  Threshold for permuting columns in the ILUTP and ILUDP preconditioners.

12.2.5 Common Block arrays

This FORTRAN Common block contains the arrays used in the software.

- `hg(0:nxd+1,0:maxtim,4)`

  The computed heat transfer coefficient along the length of the stove at each time step for each of the four cycles.

- `rhog(0:nxd+1,0:maxtim,4)`

  The computed gas density along the length of the stove at each time step for each of the four cycles.

- `cpg(0:nxd+1,0:maxtim,4)`

  The computed gas heat capacity along the length of the stove at each time step for each of the four cycles.

- `pg(0:nxd+1,0:maxtim,4)`

  The computed gas pressure along the length of the stove at each time step for each of the four cycles.

- `mub(0:nxd+1,0:maxtim,4)`

  The computed gas viscosity along the length of the stove at each time step for each of the four cycles.

- `kb(0:nxd+1,0:maxtim,4)`

  The computed gas thermal conductivity along the length of the stove at each time step for each of the four cycles.

- `bval(0:maxtim)`

  The percentage of the total cold blast flow that the simulation predicts must be bypassed at each time step to maintain the given hot blast temperature.

- `velg(0:nxd+1,0:maxtim,4)`

  The computed gas velocity along the length of the stove at each time step for each of the four cycles.

- `rhoc(nxd)`

  The brick density at each axial node along the length of the stove.

- `cc(nxd,0:maxtim)`

  The brick heat capacity along the length of the stove at each time step.

- `ro(nxd)`

  The gas channel outside radius at each axial node along the length of the stove.

- `kbrk(nxd,0:maxtim)`

  The brick thermal conductivity along the length of the stove at each time step.

- `rad(nxd,nrd+1)`

  The scaled radial distance in the brick as a function of both axial and radial node number.

- `totmdot(0:maxtim)`

  The measured cold blast flow rate during the on-blast cycle converted to gm/sec.

- `measbbls(0:maxtim)`

  The measured cold blast temperature during the on-blast cycle.

- `measping(0:maxtim)`

  The measured top gas pressure during the on-gas cycle converted to atm absolute.

- `measpinb(0:maxtim)`

  The measured cold blast pressure during the on-blast cycle converted to atm absolute.

- `bmolfrac(3,0:maxtim)`

  The gas composition (mole fraction) for each component (1 = Air; 2 = oxygen; 3 = water vapor) for each time step during the on-blast cycle

- `gmolfrac(4,0:maxtim)`

  The gas composition (mole fraction) for each component (1 = carbon dioxide; 2 = nitrogen; 3 = water vapor; 4 = oxygen) for each time step during the on-gas cycle.

- `bltmptrg(0:maxtim)`

  The blast temperature target for each time step during the on-blast cycle.

- `combtmp(0:maxtim)`

  The computed combustion temperature for each time step during the on-gas cycle.

- `actstvfl(0:maxtim)`

  The actual stove blast air mass flow rate history for the previous on-blast cycle.

12.2.6 Common Block flows

This FORTRAN Common block contains the measured mass flow rates of the gas for the on-gas and on-blast cycles.

- `gasflow(0:maxtim)`

  The measured mass flow rate of the gas during the on-gas cycle.

- `blstflow(0:maxtim)`

  The measured mass flow rate of the gas during the on-blast cycle.

12.2.7 Common Block temps

This FORTRAN Common block contains the estimated temperature parameters for the on-gas and on-blast cycles.

- `x(0:nxd+1)`

  The scaled axial distance in the brick as a function of axial node number.

- `inbrktmp(0:3)`

  The biased dome, interface, and grid temperatures at the end of the previous on-blast cycle. These quantities are used as initial conditions for the simulation.

- `modstvfl(0:maxtim)`

  The computed mass flow rate through the stove for each time step during the on-blast cycle.

- `htscl(maxestim)`

  The heat transfer coefficient scale factors (on-gas; on-blast) and the combustion temperature and stove temperature estimated biases from the estimation.

12.2.8 Common Block cntrl

This FORTRAN Common block contains the variables used for the control calculation.

- `numphas`

  The number of stove sub-cycles to execute.

- `phase(4)`

  The identity of the stove sub-cycle to execute (1 = on-gas; 2 = pressure-up; 3 = on-blast; 4 = blow-down).

- `mix_den`

  The calculated mixed fuel gas density.

- `air_fuel`

  The calculated air/fuel ratio.

- `nvar`

  The number of fuel flow rates to compute for the profile.

- `istart`

  The starting time (min) for the optimal fuel flow rate profile. This value will be zero for estimation/control calculations. It may not be zero for feedforward control calculations.

- `itotal`

  The total number of minutes for the fuel flow rate profile.

- `icount(maxcntrl)`

  The number of minutes for each fuel flow rate in the profile.

- `fuelflow(5)`

  The optimal values of the fuel gas flow rates in the fuel flow rate profile.

12.2.9 Common Block estim

This FORTRAN Common block contains the variables used for the estimation calculation.

- `actgsdmtp(0:maxtim)`

  The actual dome temperature history for the previous on-gas cycle.

- `actbldmtp(0:maxtim)`

  The actual dome temperature history for the previous on-blast cycle.

- `actgsintp(0:maxtim)`

  The actual interface temperature history for the previous on-gas cycle.

- `actblintp(0:maxtim)`

  The actual interface temperature history for the previous on-blast cycle.

- `actgsgrtp(0:maxtim)`

  The actual grid temperature history for the previous on-gas cycle.

- `actblgrtp(0:maxtim)`

  The actual grid temperature history for the previous on-blast cycle.

- `stortemp(3)`

  The dome, interface, and grid temperatures from the end of the previous on-blast cycle read from the file `temp.in`.

12.2.10 Common Block targets

This FORTRAN common block contains the operating targets used for the control calculation.

- `blst_t_trg`

  The blast air temperature target for the on-blast cycle.

- `blst_f_trg`

  The blast air total flow rate target for the on-blast cycle.

- `blst_O2_trg`

  The blast air oxygen injection rate target for the on-blast cycle.

- `blst_m_trg`

  The blast air moisture density target for the on-blast cycle.

- `mxr_v_trg`

  The final mixer valve position target at the end of the on-blast cycle.

- `O2_trg`

  The waste gas excess oxygen target.

- `minifuel`

  The minimum mixed fuel gas flow rate target constraint.

- `BTU_trg`

  The mixed fuel gas heating value target.

12.2.11 Common Block estparam

This FORTRAN Common block contains the parameters used for the estimation optimization calculation.

- `maxgsht`

  The maximum on-gas heat transfer coefficient scaling factor.

- `mingsht`

  The minimum on-gas heat transfer coefficient scaling factor.

- `maxblht`

  The maximum on-blast heat transfer coefficient scaling factor.

- `minblht`

  The minimum on-blast heat transfer coefficient scaling factor.

- `maxcomb`

  The maximum estimated combustion temperature.

- `mincomb`

  The minimum estimated combustion temperature.

- `tmpscl(3)`

  The dome, interface, and grid temperature scaling factors.

- `estconv`

  The convergence tolerance for the estimation optimization.

- `domwght`

  The weighting factor for the dome temperature prediction error.

- `intwght`

  The weighting factor for the interface temperature prediction error.

- `grdwght`

  The weighting factor for the grid temperature prediction error.

- `flowght`

  The weighting factor for the stove blast air flow rate prediction error.

- `htcwght(2)`

  The weighting factor for the change in the on-gas and on-blast heat transfer coefficient scaling factors.

- `combwght`

  The weighting factor for the change in the computed combustion temperature.

- `tmpwght`

  The weighting factor for the sum of the changes in the scaled dome, interface, and grid temperatures.

- `mindidif`

  The minimum constraint for the difference between the estimated dome and interface temperatures.

- `intcomb`

  The previous estimate for the combustion temperature

- `initbltm`

  The first time step to use when trying to match the simulated and measured on-blast flow rate histories.

12.2.12 Common Block cntparam

This FORTRAN Common block contains the parameters used for the control optimization calculation.

- `mxdomlim`

  The maximum dome temperature constraint.

- `mndomlim`

  The minimum dome temperature constraint.

- `mxintlim`

  The maximum interface temperature constraint.

- `mnintlim`

  The minimum interface temperature constraint.

- `mxgrdlim`

  The maximum grid temperature constraint.

- `mngrdlim`

  The minimum grid temperature constraint.

- `mxr_bias`

  The final mixer valve target bias value.

- `byp_fup`

  The target filter factor for increasing the final mixer valve position.

- `byp_fdn`

  The target filter factor for decreasing the final mixer valve position.

- `fuelbias`

  The mixed fuel flow rate profile bias.

- `cntconv`

  The convergence tolerance for the control optimization.

- `maxigas`

  The maximum mixed fuel gas flow rate constraint.

12.2.13 Local Variables

The following local variables are used in a number of the subroutines.

- `u(0:maxtim,0:nxd+1)`

  Scaled gas temperature array.

- `uc(0:maxtim,nxd,nrd)`

  Scaled solid temperature array.

- `x(0:nxd+1)`

  The scaled axial distance in the brick as a function of axial node number.

- `mdot(0:maxtim)`

  Scaled gas mass flow rate.

12.3 Main Program Description

The main program operation is documented in this section. The FORTRAN common blocks used for the control and estimation are first defined. The program then inputs the operating targets from the file target.in, discussed in Section 11.2.2, and opens the output file dump.out, discussed in Section 11.4.2.

```
c
c     Read target data from target.in file and put in targets common
c
      call ontarget(stove_stat)
c
c     Logical unit numbers for optimization and output
c
      ioin = 5
      ioout = 33
      iodump = 33
      ioterm = 33

      open(33, file='dump.out', status='new')
C      open(33, file='dump.out')
      write(33, *) 'Run status = ', stove_stat
```

The model parameters are then read by the software and the default model simulation, estimation, and control parameters are set. The file stove.in, discussed in Section 11.1.2, contains the model simulation parameter and the file param.in, discussed in Section 11.1.3 and read by the subroutine inptparm, contains the optimization parameters.

```
c
c     Set up the simulation
c
      open(1, file='stove.in')
      call simsetup(x, inbrktmp, htscl, 1)
      close (1)

      itimb = ntimb
      itimg = ntimg
```

Note that the variables itimb and itimg are used as storage for the number of time steps for the on-blast and on-gas cycles, respectively, since the previous cycles may not have run for the default cycle lengths.

```
c
c     Default cycle settings
c
      numphas  = 4
      phase(1) = 1
      phase(2) = 2
      phase(3) = 3
      phase(4) = 4
c
c     Default controller settings
c
      istart = 1
      itotal = ntimg

      icount(1) = 7
      icount(2) = 7
      icount(3) = 7
      icount(4) = ntimg - 21
c
c     Read optimization parameters
c
      call inptparm()
```

If an estimation/control calculation is specified, stove_stat = 0, the estimation is first performed and then the control. The estimation is carried out using the operating data from the previous on-blast and on-gas cycles contained in the files onblast.in, discussed in Section 11.2.4, and ongas.in, discussed in Section 11.2.5, respectively. The number of time steps in each cycle along with the gas flow rates and the temperature scaling factor are determined from the data in these files.

```
c
c     Read previous on-blast data from onblast.in file to determine
c     on-blast operating conditions
c
         call onblstdat(blstflow, numblst, actbldmtp, actblintp,
     .        actblgrtp)
         ntimb = numblst
         blstflow(0) = blstflow(1)
         fnbltflo = blstflow(ntimb)
c
c     Read previous on-gas data from ongas.in file to determine
c     on-gas operating conditions
c
         T_comb = htscl(3)
         call ongasdat(gasflow, T_comb, numgas, actgsdmtp, actgsintp,
     .        actgsgrtp)
         ntimg = numgas
         gasflow(0) = gasflow(1)

         tgimti = maxtemp - tinit
c
c     Estimate state here
c
         call main_estim(htscl, T_comb, inform)
```

After the model parameters have been estimated by the subroutine main_estim, the previous cycle is simulated with these parameters. The difference between the simulated temperatures and the actual measured temperatures at the end of the on-gas and on-blast cycles is used to compute the bias for the maximum and minimum temperature constraints for the control calculation.

```
c
c     Run previous cycle
c
         call sim(gasflow, blstflow, x, inbrktmp, modstvfl, htscl,
     .        templims, htclims, fbyp, modgsdmtp, modbldmtp, modgsintp,
     .        modblintp, modgsgrtp, modblgrtp, byparea, phase, numphas)
c
c     Put the initial temperatures for current cycle
```

```
c
         inbrktmp(1) = actbldmtp(ntimb) + htscl(4) * tmpscl(1)
         inbrktmp(2) = actblintp(ntimb) + htscl(5) * tmpscl(2)
         inbrktmp(3) = actblgrtp(ntimb) + htscl(6) * tmpscl(3)
```

The operating conditions for the next on-blast and on-gas cycles are determined from the average operating conditions of the previous cycle.

```
c
c     Read previous on-blast targets from onblast.in file to determine
c     average on-blast operating conditions
c
         ntimb = itimb
         ntim  = ntimb
         call onblsttrg(blstflow, numblst)
         bl_fl_trg = blstflow(ntimb)
c
c     Determine minimum temperature limits
c
         dome_min = mndomlim + (templims(2) - fntmpdom)
         int_min  = mnintlim + (templims(4) - fntmpint)
         grid_min = mngrdlim + (templims(6) - fntmpgrd)
         write(ioout,*) ’Minimum Temperature Constraints:’
         write(ioout,*) dome_min, int_min, grid_min
c
c     Read previous on-gas data from ongas.in file to determine average
c     on-gas operating conditions
c
         ntimg = itimg
         ntim  = ntimg
         call ongastrg(mix_den, air_fuel, avgfuel, T_comb, numgas)
c
c     Determine maximum temperature limits
c
         dome_max = mxdomlim + (templims(1) - fntmpdom)
         int_max  = mxintlim + (templims(3) - fntmpint)
         grid_max = mxgrdlim + (templims(5) - fntmpgrd)
         write(ioout,*) ’Maximum Temperature Constraints:’
         write(ioout,*) dome_max, int_max, grid_max
```

The final mixer valve position constraint is then determined. The software uses the stove blast air by-pass mass flow fraction as the constraint. It is computed from the mixer valve target and previous final mixer valve position as discussed in Section 8.1.

The initial fuel profile used in the control optimization is the average fuel flow from the previous on-gas cycle.

```
c
c     Determine previous by-pass flow
c
         last_byp = ((( 9.103 + 2.212d-1 * fnmixpos +
     .        5.458d-3 * (fnmixpos ** 2.d0)) * 29.d0 * 42.2195d0) /
     .        fnbltflo)
         write(ioout,*) 'Previous mixing valve = ', fnmixpos
         write(ioout,*) 'Previous by-pass flow = ', last_byp
c
c     Determine by-pass target
c
         byp_trg = ((( 9.103 + 2.212d-1 * (mxr_v_trg + mxr_bias) +
     .        5.458d-3 * ((mxr_v_trg + mxr_bias) ** 2.d0)) *
     .        29.d0 * 42.2195d0) / bl_fl_trg)
         write(ioout,*) 'Target by-pass flow: ', byp_trg
c
c     Determine difference between previous by-pass & by-pass target
c
         byp_min = byp_trg - last_byp
         write(ioout,*) 'Initial By-pass Difference = ', byp_min
c
c     Determine initial fuel profile
c
         do i = 0, ntimg
            gasflow(i) = avgfuel * mix_den * (1.d0 + air_fuel)
         end do
         write(ioout,*) 'Average mixed fuel = ', avgfuel

         call sim(gasflow, blstflow, x, inbrktmp, modstvfl, htscl,
     .        templims, htclims, fbyp, modgsdmtp, modbldmtp, modgsintp,
     .        modblintp, modgsgrtp, modblgrtp, byparea, phase, numphas)
c
c     Determine by-pass target value
c
         if (byp_min .lt. 0.0) then
            byp_min = fbyp + byp_fdn*byp_min
         else
            byp_min = fbyp + byp_fup*byp_min
         end if
         write(ioout,*) 'Simulation by-pass flow: ', fbyp
         write(ioout,*) 'By-pass constraint = ', byp_min
c
c     Set up initial fuel profile
c
         nvar   = 4
         inform = nvar

         do i = 1, nvar
           fuelflow(i) = avgfuel
         end do
```

If a feedforward control calculation is specified, the biased temperature constraints and the stove blast air by-pass mass flow fraction constraint are read from the file control.out, discussed in Section 11.4.1, which contain the results of the calculation performed for the estimation/control.

```
         open(4, file='control.out')
         read(4,*) nvar
         do i = 1, nvar
            read(4,*) inform, fuelblow(i)
            fuelflow(i) = fuelblow(i) - fuelbias*min(
     .                    max(fuelblow(i)-minifuel,0.d0), 1.d0)
         end do
         read(4,*) avgfuel
         read(4,*) air_fuel
         read(4,*) inform
         read(4,*) mix_den
         read(4,*) dome_max, dome_min
         read(4,*) int_max, int_min
         read(4,*) grid_max, grid_min
         read(4,*) byp_min
         read(4,*) numblst
         close(4)
```

The estimated parameters are read from the file temp.in, discussed in Section 11.2.3, which contain the results of the calculation performed for the estimation.

```
         open(4, file='temp.in')
         read(4,*) inform
         read(4,*) inbrktmp(1)
         read(4,*) inbrktmp(2)
         read(4,*) inbrktmp(3)
         read(4,*) T_comb
         read(4,*) htscl(1)
         read(4,*) htscl(2)
         close(4)
```

The average on-blast operating conditions and combustion temperature are then determined.

```
c
c     Read previous on-blast targets from onblast.in file to determine
c     average on-blast operating conditions;  set combustion temp array
c
         ntim  = ntimb
         call onblsttrg(blstflow, numblst)

         do i = 0, ntimg
            combtmp(i) = T_comb
         end do

         tgimti = T_comb - tinit
```

The initial fuel flow rate profile, in which the fuel flow rates already implemented are included, are then specified.

```
c
c     Determine initial fuel profile
c
         nvar = 4
         k = 0

           do j = 1, icount(i)
             k = k + 1
             gasflow(k) = fuelflow(i) * mix_den * (1.d0 + air_fuel)
           end do
         end do
         gasflow(0) = gasflow(1)
```

The stove blast air by-pass mass flow fraction constraint is then computed.

```
         call sim(gasflow, blstflow, x, inbrktmp, modstvfl, htscl,
     .        templims, htclims, fbyp, modgsdmtp, modbldmtp, modgsintp,
     .        modblintp, modgsgrtp, modblgrtp, byparea, phase, numphas)
c
c     Determine by-pass target value
c
         write (ioout,*) 'Old By-pass constraint =', byp_min

         byp_min = byp_min - fbyp
         if (byp_min .lt. 0.0) then
            byp_min = fbyp + byp_fdn*byp_min
         else
            byp_min = fbyp + byp_fup*byp_min
         end if
```

The control calculation is then performed and the fuel flow rate profile determined from the results of the optimization and the fuel flow rate bias.

```
c
c     Control Calculation
c
      call main_cntrl(dome_max, dome_min, int_max, int_min, grid_max,
     >                grid_min, byp_min, inform)

      if (stove_stat .gt. 0) then
        do i = 1, nvar-stove_stat
          fuelflow(i) = fuelblow(i)
        end do
        if (inform .gt. 1) then
          do i = nvar-stove_stat+1, nvar
            fuelflow(i) = fuelblow(i)
          end do
        else
          do i = nvar-stove_stat+1, nvar
            fuelflow(i) = fuelflow(i) + fuelbias*min(
     .                    max(fuelflow(i)-minifuel,0.d0),1.d0)
          end do
        end if
      else
        if (inform .gt. 1  .and.  inform .ne. 11) then
          do i = 1, nvar
            fuelflow(i) = max(minifuel, avgfuel + fuelbias*min(
     .                    max(avgfuel-minifuel,0.d0),1.d0))
          end do
          if (inform .lt. 10  .or.  inform .gt. 11) then
            inform = -inform
          end if
        else
          do i = 1, nvar
            fuelflow(i) = fuelflow(i) + fuelbias*min(
     .                    max(fuelflow(i)-minifuel,0.d0),1.d0)
          end do
        end if
      end if
```

The results are then output to the file control.out, discussed in Section 11.4.1.

```
      strttim = 0.d0
      delt    = dtgas / (60.d0 * dble(ntimg))
      avgfuel = 0.d0

      open(4,file='control.out')
      write(4,*) nvar

      do i = 1, nvar
        write(4,*) nint(strttim), fuelflow(i)
        write(ioout,*) 'fuel(',i,') = ', nint(strttim), fuelflow(i)
        strttim = strttim + 0.25 + delt * dble(icount(i))
        avgfuel = avgfuel + fuelflow(i)*dble(icount(i))
      end do

      avgfuel = avgfuel / dble(itotal)
      write(4,*) avgfuel
      write(4,*) air_fuel
      write(4,*) inform
      write(4,*) mix_den
      write(4,*) dome_max, dome_min
      write(4,*) int_max, int_min
      write(4,*) grid_max, grid_min
      write(4,*) byp_min
      write(4,*) numblst
      close(4)
```

12.4 Subroutine Description

Each of the FORTRAN 77 subroutines and functions comprising the advanced control technology software are documented in this section. The subroutine files are presented in alphabetical order. Several subroutine files contain more than one subroutine and/or function. In these cases, each subroutine and function contained in the file is documented in the order found within the file.

12.4.1 bc.f

The file `bc.f` contains the following subroutine.

- `subroutine bc`

  This subroutine sets the spatial boundary conditions at each time step for the simulation.

12.4.2 blas1.f

The file `blas1.f` contains the level 1 BLAS (basic linear algebra subroutines) routines. These subroutines were taken from the LINPACK linear algebra subroutine library [7]. A detailed discussion of these subroutines is contained in the current version of the reference manual [1].

- `subroutine dcopy`

  Copies a vector into a vector.

- `subroutine daxpy`

  Performs vector–scalar multiplication.

- `subroutine dscal`

  Scales a vector by a scalar.

- `subroutine dswap`

  Interchanges two vectors.

- `subroutine drot`

  Performs a plane rotation on a vector.

- `subroutine drotg`

  Performs a Givens plane rotation on a vector.

- `subroutine ccopy`

  Copies a complex vector into a complex vector.

- `subroutine cscal`

  Scales a complex vector by a complex scalar.

- `subroutine csrot`

  Performs a plane rotation on a complex vector.

- `subroutine cswap`

  Interchanges two complex vectors.

- `subroutine csscal`

  Scales a complex vector by a real scalar.

12.4.3 blassm.f

The file blassm.f contains the SPARSKIT basic linear algebra for sparse matrices routines. These subroutines comprise the sparse matrix operations for the SPARSKIT software package discussed in Section 12.1.3.

- `subroutine amub`

  Performs the sparse matrix multiplication C = AB.

- `subroutine aplb`

  Performs the sparse matrix addition C = A+B.

- `subroutine aplb1`

  Performs the sparse matrix addition for matrices in sorted CSR format.

- `subroutine aplsb`

  Performs the operation C = A+sB for matrices in sorted CSR format.

- `subroutine aplsb1`

  Performs the operation C = A+sB for matrices in sorted CSR format.

- `subroutine apmbt`

  Performs the matrix sum C = A+transp(B) or C = A-transp(B).

- `subroutine aplsbt`

  Performs the matrix sum C = A+transp(B).

- `subroutine diamua`

  Performs the matrix by matrix product B = Diag*A (in place).

- `subroutine amudia`

  Performs the matrix by matrix product B = A*Diag (in place).

- `subroutine aplsca`

  Adds a scalar to the diagonal entries of a sparse matrix A = A+s.

- `subroutine apldia`

  Adds a diagonal matrix to a general sparse matrix B = A+Diag.

12.4.4 combusted.f

The file combusted.f contains the following subroutine.

- `subroutine combusted`

  This subroutine computes the combustion temperature during the on-gas cycle as discussed in Section 6.3.

12.4.5 dgbco.f

The file dgbco.f contains the following subroutine.

- `subroutine dgbco`

  This subroutine factors a double precision band matrix by Gaussian elimination and estimates the condition of the matrix.

12.4.6 dgbfa.f

The file dgbfa.f contains the following subroutine.

- `subroutine dgbfa`

  This subroutine factors a double precision band matrix by elimination. It is called by dgbco.

12.4.7 dgbsl.f

The file dgbsl.f contains the following subroutine.

- `subroutine dgbsl`

  This subroutine solves the double precision band system Ax = b or transp(A)x = b using the factors computed by dgbco or dgbfa.

12.4.8 energy.f

The file energy.f contains the calculations for the energy changes in the gas and solid for the discrete system, and the relative error between these two quantities, as shown in Equation 7.14.

12.4.9 eqn.f

The file eqn.f contains the following routines.

- `function eqnresg`

  This function computes the residual vector for the gas temperature discretized equations shown in Equation 7.3.

- `function eqnresc`

  This function computes the residuals for the solid temperature discretized equations shown in Equation 7.3.

- `subroutine updatestatevars`

  This subroutine computes the temperature dependent physical properties, heat transfer coefficient, gas velocity, pressure, and stove blast air flow for each of the stove cycles.

12.4.10 formats.f

The file formats.f contains the SPARSKIT sparse matrix format conversion routines. These subroutines comprise the sparse matrix conversion operations for the SPARSKIT software package discussed in Section 12.1.3.

12.4.11 gcomp_cntrl.f

The file gcomp_cntrl.f contains the following subroutine.

- `subroutine gcomp_cntrl`

  This subroutine calculates the objective function in Equation 8.1 for the control optimization discussed in Section 8.1. Additional information on the format of the objective function subroutine for GRG2 is contained in the GRG2 reference manual [24].

12.4.12 gcomp_estim.f

The file gcomp_estim.f contains the following subroutine.

- `subroutine gcomp_estim`

  This subroutine calculates the objective function in Equation 9.1 for the estimation optimization discussed in Section 9.1. Additional information on the format of the objective function subroutine for GRG2 is contained in the GRG2 reference manual [24].

12.4.13 grg2.f

The file grg2.f contains the following subroutine.

- `subroutine grgsub`

  This subroutine performs the control and estimation optimizations for the nonlinear optimization problems defined in Equations 8.1 and 8.2, discussed in Section 8.1, for the control and Equations 9.1 and 9.2, discussed in Section 9.1, for the estimation. Additional information on the on the GRG2 algorithm and subroutine interface is contained in the GRG2 reference manual [24].

12.4.14 ilut.f

The file ilut.f contains the SPARSKIT sparse matrix preconditioner routines. These subroutines comprise the sparse matrix linear system preconditioner algorithms for the SPARSKIT software package discussed in Section 12.1.3. Preconditioning is discussed in Section 7.3.

12.4.15 initmesh.f

The file initmesh.f contains the following subroutine.

- `subroutine initmesh`

  This subroutine initializes the simulation mesh, composed of the spatial and temporal nodes used to solve the discretized model equations shown in Equation 7.3. The determination of the mesh is discussed in Section 7.4. The size of the mesh is determined based on the parameters outlined in Section 11.1.2.

12.4.16 inittemp.f

The file inittemp.f contains the following subroutine.

- `subroutine inittemp`

  This subroutine determines an initial estimate for the gas and solid temperatures at each spatial and temporal node in the mesh. Linear interpolation between the previous dome, interface, and grid temperatures from the previous cycle is used to determine the initial temperature profile estimates in space and time for the current cycle.

12.4.17 inptparm.f

The file inptparm.f contains the following subroutine.

- `subroutine inptparm`

  This subroutine is used to read the parameter file param.in, discussed in Section 11.1.3. This file is input by the software prior to the execution of each estimation/control calculation. Changes to any of the tuning parameters in this file will be reflected the next time the software runs.

12.4.18 input.f

The file input.f contains the following subroutine.

- `subroutine input`

  This subroutine is used to read the parameter file stove.in, discussed in Section 11.1.2. This file is input by the software prior to the execution of each estimation/control calculation. Changes to any of the stove parameters in this file will be reflected the next time the software runs.

12.4.19 itaux.f

The file itaux.f contains the following routines.

- `subroutine runrc`

  This subroutine performs the iterative solution of the linear system in Equation 7.5 obtained from the nonlinear Newton step. The solution method is determined by the entries in the parameter file stove.in discussed in Section 11.1.2. The linear system that is solved is discussed in Section 7.3.

- `function distdot`

  This function computes the dot product of two vectors.

12.4.20 iters.f

The file iters.f contains the SPARSKIT sparse matrix iterative linear system solver routines. These subroutines comprise the sparse matrix linear system iterative solution algorithms for the SPARSKIT software package discussed in Section 12.1.3. Iterative linear system solution is discussed in Section 7.3.

12.4.21 jacobian.f

The file jacobian.f contains the following subroutine.

- `subroutine jacobian`

  This subroutine computes a numerical approximation of the Jacobian matrix in the residual equations shown in Equation 7.6. The Jacobian matrix is computed and stored in compressed sparse row format in order to use the SPARSKIT iterative solver GMRES, in the subroutine runrc, to solve the linear system in Equation 7.5.

12.4.22 main_cntrl.f

The file main_cntrl.f contains the following subroutine.

- `subroutine main_cntrl`

  This subroutine is the interface to the GRG2 optimization subroutine grgsub for the control optimization. The GRG2 optimization parameter and constraint values required for grgsub are set in this subroutine prior to executing the GRG2 optimization. Documentation on each of these parameters is provided in the GRG2 reference manual [24].

12.4.23 main_estim.f

The file main_estim.f contains the following subroutine.

- `subroutine main_estim`

  This subroutine is the interface to the GRG2 optimization subroutine grgsub for the estimation optimization. The GRG2 optimization parameter and constraint values required for grgsub are set in this subroutine prior to executing the GRG2 optimization. Documentation on each of these parameters is provided in the GRG2 reference manual [24]. This subroutine also generates the output files est_gas.out and est_blst.out discussed in Section 11.3 and the estimation results in the output file dump.out discussed in Section 11.4.2.

12.4.24 matvec.f

The file matvec.f contains the SPARSKIT sparse matrix matrix–vector multiplication and tri-diagonal solver routines. These subroutines comprise the sparse matrix multiplication and tri-diagonal system solution algorithms for the SPARSKIT software package discussed in Section 12.1.3.

12.4.25 mesh.f

The file mesh.f contains the following subroutine.

- `subroutine mesh`

  This subroutine creates the spatial and temporal grid for the numerical solution of the discretized model equations shown in Equation 7.3. The mesh size is determined by parameters in the file stove.in discussed in Section 11.1.2.

12.4.26 newt_conv.f

The file newt_conv.f contains the following subroutine.

- `subroutine newt_conv`

  This subroutine computes the Euclidean and Infinity norms for the nonlinear residual. This information is used by the model to assess when the simulation has converged.

12.4.27 onblstdat.f

The file onblstdat.f contains the following subroutine.

- `subroutine onblstdat`

  This subroutine computes the conditions for the previous on-blast cycle. It reads the data file onblast.in, discussed in Section 11.2, and based on this data determines the physical parameter values in addition to the temperature and stove blast air flow rate for the cycle. This subroutine is used by the estimation to initialize these values before estimating the parameters.

12.4.28 onblsttrg.f

The file onblsttrg.f contains the following subroutine.

- `subroutine onblsttrg`

  This subroutine determines the average operating conditions for the previous on-blast cycle, based on the data contained in the data file onblast.in discussed in Section 11.2, that are used for the on-blast cycle in the control calculation.

12.4.29 ongasdat.f

The file ongasdat.f contains the following subroutine.

- `subroutine ongasdat`

  This subroutine computes the conditions for the previous on-gas cycle. It reads the data file ongas.in, discussed in Section 11.2, and based on this data determines the physical parameter values, the temperature profile, and the mixed fuel gas flow rate for the cycle. This subroutine is used by the estimation to initialize these values before estimating the parameters.

12.4.30 ongastrg.f

The file ongastrg.f contains the following subroutine.

- `subroutine ongastrg`

  This subroutine determines the average operating conditions for the previous on-gas cycle, based on the data contained in the data file ongas.in discussed in Section 11.2, that are used for the on-gas cycle in the control calculation.

12.4.31 ontarget.f

The file ontarget.f contains the following subroutine.

- `subroutine ontarget`

  This subroutine is used to read the target input file target.in, discussed in Section 11.2.2. This file contains the desired operating targets for the next regenerative cycle and is input by the software prior to the execution of each estimation/control calculation. Changes to any of the operating targets in this file will be reflected the next time the software runs.

12.4.32 printdat.f

The file printdat.f contains the following subroutines.

- `subroutine prn_init`

  This subroutine prints headers and initial condition information to the file diter.out, which is discussed in Section 11.3.6.

- `subroutine prn_iter`

  This subroutine prints diagnostic information about each linear and nonlinear simulation iteration to the file diter.out, which is discussed in Section 11.3.6.

- `subroutine prn_temp`

  This subroutine prints diagnostic information about the model predicted data for each measured value at each time step in the stove cycle to the file dtemp.out, which is discussed in Section 11.3.7.

- `subroutine prn_prop`

  This subroutine prints diagnostic information about the model predicted physical properties of the gas and solid along the length of the stove at each time step in the stove cycle to the file dprop.out, which is discussed in Section 11.3.7.

12.4.33 property.f

The file property.f contains the following subroutine.

- `subroutine bricks`

  This subroutine computes the temperature-dependent heat capacity and thermal conductivity of the material in the bricks for each of the zones in the stove specified in the file brick.in, discussed in Section 11.1.1. There must be a physical property calculation in property.f for each stove zone specified in the parameter file brick.in. Note that there is not a separate physical property subroutine for the gas. These calculations are performed directly in the subroutine updatestatevars since the components in the blast air and waste gas are identical for all blast furnace stoves.

12.4.34 res.f

The file res.f contains the following subroutine.

- `subroutine res`

  This subroutine computes the nonlinear residual for the current simulation iteration by calling the functions eqnresg and eqnresc in the file eqn.f.

12.4.35 simsetup.f

The file simsetup.f contains the following subroutine.

- `subroutine simsetup`

  This subroutine sets up the stove model simulation. It calls the subroutine input, contained in the file input.f, to read the parameter file stove.in, mesh, contained in the file mesh.f, to compute the computational grid, and then initmesh, contained in the fileinitmesh.f, to initial the grid values.

12.4.36 solv.f

The file solv.f contains the following subroutine.

- `subroutine solv`

  This subroutine sets up the right hand side of the linear system in Equation 7.5. It also sets the initial guess for the state correction $\delta \boldsymbol{x}_k$ to zero.

12.4.37 spgmres.f

The file spgmres.f contains the following subroutine.

- `subroutine spgmres`

  This subroutine sets up and calls the preconditioner and linear solver selected by the parameters in the file stove.in.

12.4.38 stove.f

The file stove.f contains the following subroutines.

- `subroutine sim`

  This subroutine is the main portion of the stove simulation.

- `subroutine newt_kryl`

  This subroutine is the main portion of the Newton-Krylov differential equation solver.

- `subroutine ongas`

  This subroutine simulates the on-gas cycle

- `subroutine presup`

  This subroutine simulates the pressure-up cycle

- `subroutine onblast`

  This subroutine simulates the on-blast cycle

- `subroutine blowdwn`

  This subroutine simulates the blow-down cycle

12.4.39 unary.f

The file unary.f contains the SPARSKIT sparse matrix unary matrix routines. These subroutines comprise the unary sparse matrix operation algorithms for the SPARSKIT software package discussed in Section 12.1.3.

Bibliography

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenny, S. Ostrouchov, and D. Sorensen. *LAPACK User Guide.* SIAM, Philidelphia, 1992.

[2] D. P. Bertsekas. *Nonlinear Programming.* Athena Scientific, Belmont, Mass., 1995.

[3] M. S. Bhatti and R. K. Shah. Turbulent and transition flow convective heat transfer in ducts. In S. Kakac, R. K. Shah, and W. Aung, editors, *Handbook of Single-Phase Convective Heat Transfer*, New York, 1987. Wiley.

[4] R. B. Bird, W. E. Stewart, and E. N. Lightfoot. *Transport Phenomena.* Wiley, New York, 1960.

[5] P. Butterfield, J. Schofield, and P. A. Young. Hot blast stoves: Part II. *J. Iron and Steel Inst.*, 201:497–508, 1963.

[6] J. Dennis and R. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Computational Mathematics Series. Prentice–Hall, Englewood Cliffs, NJ, 1983.

[7] J. J. Dongarra, C. B. Moler, J. R. Bunch, and G. W. Stewart. *LINPACK User Guide.* SIAM, Philadelphia, 1979.

[8] V. M. Faires. *Thermodynamics.* MacMillan, New York, 4th edition, 1962.

[9] R. Fletcher. *Practical Methods of Optimization.* John Wiley & Sons, New York, 1987.

[10] R. W. Fox and A. T. McDonald. *Fluid Mechanics.* Wiley, New York, 3rd edition, 1985.

[11] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization.* Academic Press, London, 1981.

[12] H. Hausen and J. A. Binder. Vereinfachte berechnung der wärmeübertragung durch strahlung von einem gas an eine wand. *Int. Journal Heat and Mass Transfer*, 5:317–327, 1962.

[13] J. Hilsenrath, C. W. Beckett, W. S. Benedict, L. Fano, H. J. Hoge, J. F. Masi, R. L. Nuttall, Y. S. Touloukian, and H. W. Woolley. Tables of thermal properties of gases. Technical Report 564, National Bureau of Standards, 1955.

[14] H. C. Hottel. Radiant heat transmission. In *Heat Transmission*, chapter 4. McGraw–Hill, New York, 3rd edition, 1954.

[15] J. W. Howse, G. A. Hansen, D. J. Cagliostro, and K. R. Muske. Implicit Newton-Krylov methods for modeling blast furnace stoves. In *Proceedings of the AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, pages 283–290, 1998.

[16] A. K. Jain. Accurate explicit equation for friction factor. *ASCE J. Hydrualic Div.*, 102:674–677, 1976.

[17] A. J. Jazwinski. *Stochastic Processes and Filtering Theory.* Academic Press, New York, 1970.

[18] C. P. Jeffreson. Feedforward control of blast furnace stoves. *Automatica*, 15:149–159, 1979.

[19] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations.* Number 16 in Frontiers in Applied Mathematics. SIAM, Philadelphia, 1995.

[20] H. Kwakernaak, R. C. W. Strijbos, and P. Tijssen. Optimal operation of thermal regenerators. *IEEE Trans. Auto. Control*, 14:728–731, 1969.

[21] H. Kwakernaak, P. Tijssen, and R. C. W. Strijbos. Optimal operation of blast furnace stoves. *Automatica*, 6:33–40, 1970.

[22] G. A. Labossiere and P. L. Lee. Model-based control of a blast furnace stove rig. *J. Process Control*, 1(4):217–227, 1991.

[23] G. A. Labossiere and P. L. Lee. On-line optimization and control of a blast furnace stove rig. *J. Process Control*, 3(1):3–15, 1993.

[24] L. S. Lasdon and A. D. Warren. *GRG2 User Guide.* Optimal Methods, Inc., 1997.

[25] L. S. Lasdon, A. D. Warren, A. Jain, and M. Ratner. Design and testing of a generalized reduced gradient code for nonlinear programming. *ACM Trans. Mathematical Software*, 4:34–50, 1978.

[26] H. W. Liepmann and A. Roshko. *Elements of Gas Dynamics.* Wiley, New York, 1966.

[27] D. G. Luenberger. *Linear and Nonlinear Programming.* Addison–Wesley, Reading, Mass., 2nd edition, 1984.

[28] Y. Matoba, K. Otsuka, Y. Ueno, and M. Onishi. Mathematical model and automatic control system of hot blast stove. In *Proceedings of the IFAC Symposium on Automation in Mining,Mineral and Metal Processing*, number 10, pages 437–442, 1987.

[29] H. Mohnkern, M. Voss, and J. Janz. Neue optimierende winderhitzer-prozeßsteuerung bei klöckner stahl bremen. *Stahl und Eisen*, 114(5):51–57, 1994.

[30] K. R. Muske and T. F. Edgar. Nonlinear state estimation. In M. A. Henson and D. E. Seborg, editors, *Nonlinear Process Control.* Prentice–Hall, New York, 1996.

[31] K. R. Muske, J. W. Howse, Glen A. Hansen, Dominic J. Cagliostro, Pinakin C. Chaubal, Paul E. Quisenberry, and Michael J. Washo. Hot blast stove process model and model-based controller. *Iron and Steel Eng.*, 23:56–62, 1999.

[32] K. R. Muske and J. B. Rawlings. Nonlinear moving horizon state estimation. In R. Berber, editor, *Methods of Model Based Process Control*, NATO Advanced Study Institute Series, pages 349–365, 1995.

[33] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow.* Hemisphere Publishing, Washington, 1980.

[34] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipies in C.* Cambridge University Press, Cambridge, 2nd edition, 1992.

[35] R. C. Reid, J. M. Prausnitz, and B. E. Poling. *The Properties of Gases and Liquids.* McGraw–Hill, New York, 4th edition, 1987.

[36] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing Company, Boston, 1996.

[37] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. of Scientific and Statistical Computing*, 7:856–869, 1986.

[38] J. Schofield, P. Butterfield, and P. A. Young. Hot blast stoves. *J. Iron and Steel Inst.*, 199:229–240, 1961.

[39] J. R. Singham. Tables of emissivity of surfaces. *Int. Journal Heat and Mass Transfer*, 5:67–76, 1962.

[40] E. M. Sparrow and R. D. Cess. *Radiation Heat Transfer.* McGraw–Hill, New York, 1978.

[41] C. L. Tien. Thermal radiation properties of gases. In T. F. Irvine and J. P. Hartnett, editors, *Advances in Heat Transfer*, volume 5, New York, 1968. Academic Press.

[42] G. Walsh and A. Mitterer. Cost effective operation of a blast furnace stove system. In *Proceedings of the 1997 ACC*, pages 3765–3769, 1997.

[43] A. J. Willmott. Simulation of a thermal regenerator under conditions of variable mass flow. *Int. Journal Heat and Mass Transfer*, 11:1105–1116, 1968.

[44] A. J. Willmott. The regenerative heat exchanger computer representation. *Int. Journal Heat and Mass Transfer*, 12:997–1014, 1969.

[45] P. Zuidema. Non-stationary operation of a staggered parallel system of blast furnace stoves. *Int. Journal Heat and Mass Transfer*, 15:433–442, 1972.

Publication List

The following publications were based on the Phase One effort of this project.

Muske, K. R., J. W. Howse, G. A. Hansen, D. J. Cagliostro, P. C. Chaubal, P. E. Quisenberry, and M. J. Washo, "Hot blast stove process model and model-based controller." *Iron and Steel Eng.*, (23)56–62, 1999.

Muske, K. R., J. W. Howse, G. A. Hansen, D. J. Cagliostro, and P. C. Chaubal, "Hot blast stove process model and model-based controller." *Proceedings of the 1998 AISE Annual Conference*, paper 48, 1998.

Howse, J. W., G. A. Hansen, D. J. Cagliostro, and K. R. Muske, "Implicit Newton–Krylov methods for modeling blast furnace stoves." *Proceedings of the 1998 AIAA/ ASME Joint Thermophysics and Heat Transfer Conference*, 283–290, 1998.

Muske, K. R., G. A. Hansen, J. W. Howse, D. J. Cagliostro, and P. C. Chaubal, "Blast furnace stove control." *Proceedings of the 1998 American Control Conference*, 3809–3810, 1998.

Muske, K. R., G. A. Hansen, J. W. Howse, D. J. Cagliostro, and P. C. Chaubal, "Model-based hot blast stove control and optimization." Presented at the *1997 AIChE National Meeting*, Los Angeles, paper 197c.