Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Monterrey

School of Engineering and Sciences

Implementation of a Thermo Energetic Model of an Electric Arc Furnace Using Static Approach.

A thesis presented by

Armando Torres Vázquez

Submitted to the
School of Engineering and Sciences
in partial fulfillment of the requirements for the degree of

Master of Science

in

Energy Engineering

Monterrey, Nuevo León, May, 2019

Dedication

To my parents Mario and Armida, who always have been by my side.

To my sister Maricarmen, who encouraged me to achieve all my goals.

Acknowledgements

I thank Instituto Tecnológico y de Estudios Superiores de Monterrey for the tuition and SENER-CONACyT for the stipend. With their help this research have been made.

I want to thank Dr. José Luis López, for all the knowledge he shared to me, and for help me to develop my skills.

Thanks to Dr. Alejandro García and Dr. Carlos Rivera, who with their recommendations and advice helped to obtained a better thesis.

I appreciate the support provided by Ternium, specially to Dr. Eder Trejo, who motivated me to made this project.

Thanks to Dr. Osvaldo Micheloud, Dr. Armando Llamas and Dr. Federico Viramontes, for give me the opportunity to study the master degree in an incredible institution.

Finally, thanks to my partners and professors at Tecnológico de Monterrey, who made the stay in a foreign city more enjoyable.

Implementation of a Thermo Energetic Model of an Electric Arc Furnace Using Static Approach.

by

Armando Torres Vázquez

Abstract

In this work two mathematical models that describe the performance of an Electric Arc Furnace (EAF) are presented: Mass and energy balances and the heat rate transfer model. The mass and energy balances use real data taken from a process of a siderurgical and is able to calculate the amount of products of the actual process. The objective of this model is to make a diagnostic of the energetic performance of the EAF and to perform a detailed description of the energy distribution within the furnace. The mass and energy balance models were validated comparing the amount of hot heel and slag calculated with the obtained during the process. In order to the model works properly, it is necessary that the input data be accurate enough. The heat transfer model predicts the temperature distributions and energy flows between the diferent elements of the EAF, from the input energy rates and mass flows of the process. This model has an analytical approach and considers the different mechanisms of heat transfer: conduction, convection and radiation. The robustness and flexibility of the model was tested varying the electric power supplied to the furnace, the arc coverage index, the slag layer thickness and the process time. The model produces similar results to the results presented in other works. Both models require relatively few computational resources for their operation.

Contents

List of Figures

List of Tables

Nomenclature

A	Area
A_{btm}	Area of the furnace bottom
A_{roof}	Area of the furnace roof
$\boldsymbol{c_{in}}$	Mass fractions of compounds in each inlet stream
$\boldsymbol{c_{out}}$	Mass fractions of compounds in each outlet stream
cov	Arc coverage index
cp	Specific heat
d_{pipe}	Diameter of cooling panels pipes
Eb	Black body radiation
$\boldsymbol{ED}$	Element distribution matrix
f_{pn}	Friction factor of each cooling panel
F_{mn}	View factor from surface m to n
$\hat{H}$	Molar enthalpy
$\hat{\boldsymbol{H}}_{\boldsymbol{T}}$	Molar enthalpy vector
h_{air}	Convection heat transfer coefficient of surrounding air
h_{pn}	Internal heat transfer coefficient of each cooling panel
h_g	Convective heat transfer coefficient of the cooling panels
H_S	Sensible energy of the inlet and outlet streams
J	Radiosity
k_{ref}	Thermal conductivity of refractory brick
k_{shell}	Thermal conductivity of EAF shell
k_{slg}	Thermal conductivity of slag
k_{steel}	Thermal conductivity of steel
k_{water}	Thermal conductivity of water
L_{pipe}	Length of cooling panels pipes
$\dot{m}$	Mass flow
$\dot{m}_{ce}$	Mass flow of electrode cooling water
$\dot{m}_w$	Mass flow of cooling panel water
$\boldsymbol{m_{comp}}$	Total mass of each compound vector
$\boldsymbol{m_{in}}$	Mass of the inlet streams
$\boldsymbol{M_{in}}$	Mass of each compound that comforms each inlet stream
$\boldsymbol{m_{out}}$	Mass of the outlet streams
$\boldsymbol{M_{out}}$	Mass of each compound that comforms each outlet streams
$\boldsymbol{n_{comp}}$	Moles of each compound in the inlet or outlet streams
$\boldsymbol{N_{rx}}$	Moles of each compound reacted during the process

OF	Objective function
$\boldsymbol{PM_c}$	Molecular weight of each compound
Pr	Prandtl number
Q_{btm}	Heat loss in the bottom of EAF
Q_{comb}	Heat delivered by combustion of carbon
Q_{eloss}	Electrical losses
Q_{fo}	Heat loss when the furnace is open
Q_{gn}	Heat delivered by combustion of natural gas
Q_{loss}	Heat losses of the process
Q_{oxy}	Heat delivered by oxydation of iron
Q_{sd}	Heat losses in the slag door
Q_{cw}	Heat losses in the cooling panels
$\boldsymbol{R}$	Residuals of the mass balances
Re	Reynolds number
T	Temperature
t	Time
T_{cool}	Temperature of the cooling panels
T_{OS}	Temperature of the ambient air
T_{ref}	Reference temperature
T_{win}	Temperature of the inlet cooling water
T_{wout}	Temperature of the outlet cooling water
T_g	Temperature of the gases inside the furnace
t_{CH}	Charging time
t_{IN}	Inspection time
t_{PO}	Power on time
t_{TP}	Tapping time
t_{TT}	Tap to tap time
U_m	Global heat transfer coefficient of the cooling panels
$\dot{W}_e$	Electric power
x	Raw measurement data
$\hat{x}$	Reconciled measurement data
$\hat{x}$	Reconciled not measurement variable
$\Delta\hat{H}$	Transition latent heat
$\Delta\hat{H}_f^\circ$	Standard heat of formation
ΔH_r	Heat of reaction
Δx_{pipe}	Thickness of the cooling panels pipes
Δx_{ref}	Thickness of the refractory layer
Δx_{shell}	Thickness of the shell of the furnace
Δx_{slg}	Thickness of the slag layer
ϵ	Emissivity
η_e	Efficiency of the electric system
μ_w	Water dynamic viscosity
σ	Stefan-Boltzmann constant

Chapter 1

Introduction

The steelmaking industry bases its process on the refining of pig iron or metal scrap, or a combination of both. In the 1980s and 1990s there was a revolution in the methods used for the steel production. In 1963, 80% of the steel production came from basic open hearths. In 1985, this process fell to 7%, due to the implementation of more efficient processes, such as basic oxygen converters and electric arc furnaces (EAF) [9].

In the steelmaking industry, the EAF are widely used to convert metal scrap or direct reduction iron (DRI) into cast steel. In 2016, 407,105 thousand tons of steel were produced in the world through this process, representing 25.2% of total steel production in the world [34].

The EAF is formed by a steel shell with a cylindrical shape and a roof with a domed construction. The bottom and the lower sidewalls of the shell are covered by refractory brick, because this part of the furnace is in contact with the molten steel. The roof and the highest sidewalls of the shell are covered by water-cooled panels. The metals are melted by an electric arc formed between two graphite electrodes located in the middle of the furnace [33]. The arc is ionized gases that reach temperatures around of 12000°C when an electric current of great magnitude passes by a non conductor [30].

The electricity is the main energy source for this process. A ton of molten steel requires 450 kWh of electrical energy approximately [18], and the energy required for the process 60% comes from electric power and another 40% comes from the chemical energy resulting from the combustion of material and others chemical reactions ocurring in the oven [26]. However, the energy feed to the EAF is always higher than this value due to the energy losses that occur in industrial practice [31]. In total, the electricity needed for an EAF is enough to power a town with a population of 100,000 [5]. Due this high energy consumption, the study of this process are relevant because it can contribute to reduce the energy consumption, and consequently the production costs.

1.1 Process Description

The EAF process description of MacRosty and Swartz [24] and Trejo [30] is presented below. The melt of steel in an EAF is a batch process, and each lot is called "heat". The heat involves a series of different stages, specifically charging, melting and tapping (emptying of molten steel from the furnace).

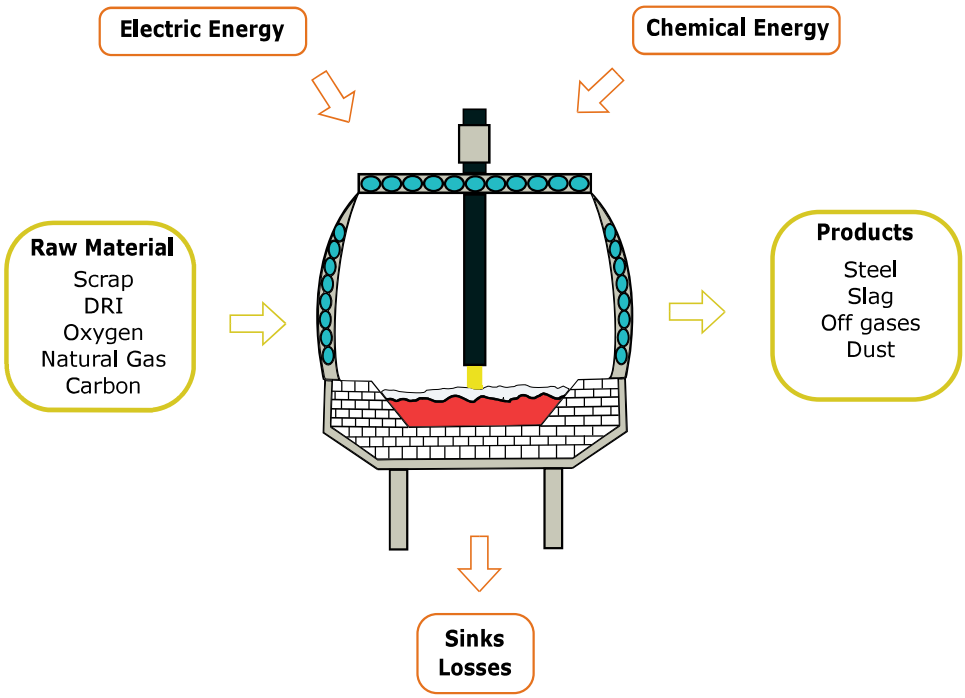

Figure 1.1: Mass and energy flows during the EAF process

The process begins when a scrap basket is loaded inside the furnace. Subsequently, the roof of the furnace is closed and the electrodes are lowered into the furnace. The arc is switched on, while the electrodes bore in the scrap, to heat and melt the load. This process can be accelerated using natural gas burners. For steels with strict specifications, DRI is fed continously to increase the quality of the product. In this stage liquid steel and slag are produced. The slag is a mixture of oxides and impurities of the metal charge which floats on the steel. Also, some additives such as lime are fed to the furnace in order to obtain a proper slag composition. Once the material is melted off, the furnace is loaded with new scrap baskets, until the capacity of the furnace is completed. Tipically, the furnace capacity is in the order of 100 tons.

At the end of the heat, the slag layer is foamed by injecting carbon and oxygen to obtain CO gas, which bubbles go through the slag. The foaming slag covers the arc, protecting the furnace walls from the arc radiation and also improving the energy transfer to the steel and hence the energy efficiency. In the furnace also occur oxidation and combustion reactions that serve as a source of energy.

After a certain time, the bath temperature and carbon content are measured. This information indicates what additions need to be made to reach the end point specifications. Once the desired composition and temperature have been obtained in the furnace, the tap hole is opened and the molten steel is poured into the ladle for transport to the next operation.

The EAF have cooling panels in the walls and the roof to avoid damages due to the high temperatures that the furnace handles. Each heat takes about 60 minutes.

1.2 Problem Statement

As discussed earlier, the energy consumption of the EAF is very high. This consumption depends on several variables, like the quality of the charge materials, the amount and quality of the slag, the recipes of material addition, the geometry of the EAF, the time of the heat, the power and others. The effect of this variables in the process can be studied in order to save energy and money, and reduce the production costs. However the EAF process is very complex, characterized by having a low automation index and a high level of operational involvement.

The energy savings are very important because a little save of electricity implies a big save in money. Also, on this respect exists a big opportunity area: the actual efficiencies of the EAFs reach 60% [30]. Almost the half of the energy goes to waste through the cooling panels, slag and off gases. The analysis of the energy distribution in the EAF during the process will allow to know in detail the furnace performance, and it will be a tool that helps to make decisions in order to increase te energy savings

1.3 Scope of Research

The high energy consumption of the EAF leads the development of optimization strategies that reduces production costs, keeping the desired product quality (steel grade) and the enviromental standars (carbon emissions). However, as mentioned earlier, the process is very complex, and most of the operating process used in the furnace have an empirical origin. Therefore, it is important represent the knowledge of the process in the form of a mathematical model, which describes the interactios of energy and matter in the process from the operational data of the process, like the amount and quality of raw material and the electricity supplied. This model will allow provide information that will help to improve the qualitity and energetic performance of the EAF. In particular, this work consists in the develop and validation of a tool whose made a diagnosis of the global performance of the EAF, using mass and energy balances. Also, this work proposes a heat transfer model, whose purpose is describe the flow and interaction of energy during the the process, which have flexibility, enough robustness and requires low computational resources in order to be adapted to other models in the future. For the validation of the model, it was used operative data from the EAF "Fuchs", located in the company Ternium, Guerrero plant, in Monterrey, México.

1.4 Objective

Implement a mathematical model that calculates the energy flows during the steel melting process in an EAF.

1.4.1 Specific Objectives

- Validate a model to solve the mass balance.
- Improve and modify the existing mathematical model model to solve the energy balance, identifying the sources, sinks, losses, infiltrations and interactions of energy.
- Develop a heat transfer model considering all of the heat transfer mechanisms, prioritizing radiation.

1.5 Thesis organization

The present document is organized in five chapters to get a better understanding of the project.

In Chapter 1 the introduction and the background of the project are presented. The problem statement, th scope of research and the general and specific objectives are also presented.

The Chapter 2 includes some theoretical concepts used in this work. Specifically, the description of the methods used to solve a mass and energy balance, and the fundamental concepts of the heat transfer mechanisms. Also, the characteristics of previous EAF models developed are presented.

The mass and energy balance model development are shown in Chapter 3. The energy balance model development includes the procedure to calculate the energy of the mass flows and the energy losses that happen during the process. The model validation results are also presented in this chapter.

Chapter 4 shows the development of the heat transfer model. This chapter contains the detailed description of the heat transfer mechanisms that occur in the different elements of the furnace, as the cooling panels, the electrode and the refractory lines, and the description of the energy balances in the molten zone and the off gases. This chapter also describes the methodology followed to obtain the view factors used in the radiative heat transfer. The results of the tests performed on the model are presented in the final section of this chapter.

In the Chapter 5 the conclusions and future work of the thesis are presented.

Finally, in the annexes of the project are included the properties of the compounds considered in the models, the process data used in the mass and energy balnce model and the view factors obtained in this work.

Chapter 2

Theoretical framework and literature review

In this chapter are presented the theoretical concepts on which the operation of the mass and energy balances are based, as well as the fundamentals of heat transfer used in this work. Subsequently, the characteristics of some models previously elaborated by different research groups are mentioned.

2.1 Mass and energy balances

When a process is analyzed or designed, the restrictions imposed by nature on the process must be taken into account. One of these restrictions are the matter and energy conservation laws, stating that matter and energy are not created nor destroyed. The equations based in the matter and energy conservation laws are called balances.

The balance of a quantity that is conserved in a system can be written in the general way:

$$input + generation - output - consumption = accumulation \tag{2.1}$$

There are two types of balances [10]:

- The differential balances indicate what is happening in a system in time. Each term of the balance equation is a rate and has units of the balanced quantity per time unit.
- The integral balances describe what happens between two instants of time. Each term of the equation is an amount of the balanced quantity.

2.1.1 Mass balance

In the process, the occurrence of a chemical reactions bring several complications into the material balance, because a material balance on a reactive sustance does not have the simple form input=output, it must be include a generation or consumption term. To solve a material balance in processes where chemical reactions occurs, there are three methods of solution [10].

- The molecular species balance method considers the generation or consumption of the reactive species. This terms of generation or consumption may be determined using the stoichiometric equations of the chemical reactions. This method is recommended to use in simple systems involving one reaction.
- The atomic species balances take the form $input = output$, since the atomic species can be neither generated nor consumed in chemical reactions. The atomic species balances are the most straightforward solution procedure, especially when more than one reaction is involved.
- The extent of reaction method consists in writing expressions for multiple reaction systems in terms of extents of reaction, and solve the extents of reaction and the unknown reactive species flow rate from known feed and product flow rates. This method is convenient to use when all the reactions that occur in the process are known.

2.1.2 Energy balance

The energy balances are useful to account for the energy that flows into or out of a process, to determine the requeriment of net energy for the process, and to design ways to reduce the energy requirement and improve process profitability. The principle that underlines all energy balances is the law of conservation of energy. This law is also known as the first law of thermodynamics. [10]

The equation of the first law of thermodynamics is:

$$\frac{d(m\hat{E})}{dt} = \sum_{in} \dot{m}_i(\hat{H}_{T,i} - \hat{E}_{Ki} + gz_i) + \sum_{out} \dot{m}_j(\hat{H}_{T,j} + \hat{E}_{Kj} + gz_j) + \dot{Q} - \dot{W} \tag{2.2}$$

where $\hat{E}$ is the specific total energy of the system, $\hat{H}$ is the specific enthalpy, $\hat{E}_K$ is the specific kinectic energy, gz is the specific potential energy, $\dot{Q}$ is the heat rate added to the system and $\dot{W}$ is the power produced by the system.

When energy balances on a reactive chemical process are present, two procedures are followed in the calculation of the change of enthalpy ($\sum_{in} \dot{m}_i \hat{H}_{T,i} - \sum_{out} \dot{m}_j \hat{H}_{T,j}$) that differ in the choice of reference states for enthalpy calculations. In the heat of reaction method, the references are the reactant and product species at 298.15 K and 1 atm in the phases for which the heat of reaction is know. In the heat of formation method, the references are the elemental species that constitute the reactant and product species at 298.15 K and 1 atm. [10]

When the heat of reaction method is used, the specific enthalpy of each species in every feed or product stream is calculated by choosing a process path from the reference state to the process state. When specific enthalpies have been calculated for all species in alll of their inlet and outlet states the change in the enthalpy is calculated as:

$$\sum_{in} \dot{m}_i \hat{H}_{T,i} - \sum_{out} \dot{m}_j \hat{H}_{T,j} = -\dot{\xi} \Delta \hat{H}_r^\circ + \sum_{in} \dot{m}_i \hat{H}_i - \sum_{out} \dot{m}_j \hat{H}_j \tag{2.3}$$

where $\dot{\xi}$ is the rate of reaction.

When the heat of formation method is used, the specific enthalpy of a species in a feed or product stream is calculated by choosing a process path from the reference state to the process state, begginning with the formation of the species from the elements. When specific enthalpies have been calculated for all species in all of their inlet and outlet states, the change in the enthalpy is calculated for:

$$\sum_{in} \dot{m}_i \hat{H}_{T,i} - \sum_{out} \dot{m}_j \hat{H}_{T,j} = \sum_{in} \dot{m}_i \hat{H}_i^* - \sum_{out} \dot{m}_j \hat{H}_j^* \tag{2.4}$$

2.2 Heat transfer fundamentals

The heat is the form of energy that can be transferred from one system to another as a result of a temperature difference. The heat transfer is the science that deals with the determination of the rates of this energy transfer. The transfer of heat is always from the higher temeprature body to the lower temperature one, and heat transfer stops when the two body reach the same temperature [4].

The heat can be transferred by three different mechanisms: conduction, convection and radiation. Below is a brief description of each mechanism.

2.2.1 Conduction

Conduction is the energy transfer mechanism from more energetic particles to the adjacent less energetic particles. The rate of heat conduction depends of the geometry, the thickness, the material of the body and the temperature difference across the body. The rate of heat conduction is described by the Fourier's law of heat conduction [4].

$$\dot{Q} = -kA\frac{dT}{dx} \tag{2.5}$$

where dT/dx is the temperature gradient, A is the normal heat transfer area and k is the thermal conductivity of the material, and is a measure of the ability of the material to conduct heat.

2.2.2 Convection

Convection is the heat transfer mechanism between a solid surface and the adjacent fluid that is in motion, and involves the combined effects of conduction and advection. The rate of convection heat transfer is proportional to the temperature difference between the solid surface and the bulk fluid temperature, and is expressed by Newton's law of cooling as

$$\dot{Q} = hA_s(T_s - T_\infty) \tag{2.6}$$

where h is the convection heat transfer coefficient, A_s is the surface area, T_s is the surface temperature and T_∞ is the temperature of the fluid far from the surface [4].

The convection heat transfer coefficient (h) is not a property of the fluid, and its value depends of many variables such as the surface geometry, the nature of fluid motion and the properties of the fluid [4].

2.2.3 Radiation

Radiation is the energy emitted by a body in the form of electromagnetic waves. The transfer of energy by radiation does not require the presence of an intervening medium. The maximum rate of radiation emitted from a surface is given by the Stefan-Boltzmann law as

$$\dot{Q}_{emit} = \sigma A_s T_s^4 \tag{2.7}$$

where σ is the Stefan-Boltzmann constant, A_s is the surface area and T_s is the absolute temperature of the body. The idealized surface that emits radiation at his maximum is called a blackbody. The radiation emitted by all real surfaces are less than the radiation emitted by a blackbody, and the concept of emissivity (ϵ) is used to measure how closely a surface approximate to a blackbody [4].

The difference between the rates of radiation emitted by the surface and the radiation absorbed is the net radiation heat transfer. When a surface is completely enclosed by another surface at absolute temperature T_{sur}, the net rate of radiation heat transfer is given by:

$$\dot{Q} = \sigma A_s \epsilon (T_s^4 - T_{sur}^4) \tag{2.8}$$

2.3 Data reconciliation and validation

The calculation of the mass and energy balances of a chemical process is a basic tool for monitoring its performance. To calculate these balances, one collects as many measurements as flow rates, concentrations and temperatures. These raw measurement data are subject to random and gross errors so that they may not be consistent with the conservation of mass and energy. The data must then be adjusted so that the adjusted values obey the conservation laws [7].

The advanced data validation and reconciliation method is the procedure of optimally adjusting measured variables in such way that the adjusted values of these measurements satisfy the laws of conservation and other constraints. In general, it is formulated by the constrained weighted least-squares optimization problem results from the maximization of Gauss likelihood function

$$min\{\sum_{i=1}^{m}(\frac{\hat{x}_i - x_i}{\sigma_i})^2\} \tag{2.9}$$

where σ is the standard uncertainty of raw measurement data. Equation 2.9 is subject to

$$g_l(\hat{x}_i, \hat{y}_j) = 0 \; for \; l = 1, ..., r \tag{2.10}$$

Equation 2.10 defines the set of constraints. In thermal analyses, these constraints are mass and energy balances [29].

Application of the advanced data validation and reconciliation method in process permits to achieve the following aims [29]:

- calculation of the most reliable measurements values
- unique solution of the unknown quantities
- accuracy of the validated results of measurements and of calculated unknown quantities
- a reduction of uncertainty of measured quantites
- a control of fullfillingof the assumed measurements uncertainty.

2.4 Literature review

During the last years, several mathematical models of the EAF have been developed with the purpose of optimizing and controlling this kind of processes. These models are based on the solution of mass and energy balances, and the models are classified in two types: “static” and “dynamic” models [2]. The “static” models use difference equations for mass and energy requirements, and this equations are obtained from the integration of rate equations. The “dynamic” models are formed by rate goberning equations for energy and mass transport or transformation.

Many researchers have developed models with different degrees of complexity, depending on how many details of the process they want to describe. Most of the models divide the EAF into zones in order to describe the mass and energy transfer within the zones and between one zone and another [26].

One of the first models which describes the EAF process was developed by Bekker et al. [2], who developed a nonlinear dynamic model consisting of 14 ordinary differential equations based on the first principles of thermodynamics. This model divides the EAF in two zones; a zone of fluids where the steel, the molten slag and the exahust gases are found, and another solid state zone which contain the steel and components of the ungrounded slag. In adittion, it only considers some reactions involving C, Si and Fe, as well as the reduction of FeO.

Morales et al. [25] elaborated a “dynamic” EAF model based on a “static mass” and energy balance, in order to generate the boundary conditions for the ordinary differential equations proposed. This model divided the EAF into three zones, corresponding to gases, metal and slag. The gas zone was modeled usel the ideal gas law, and for the metal and the slag were

used more complex models.

The static model from Çamdali et al. [8] uses, for the mass balance, an analysis by element, in order to know the precise quantities of the input components so that the products have a specific composition. They also establish that up to 27 different reactions can occur in the furnace [32]. However, it only considers some (about 13 reactions) because they present in a greater proportion and dominate the obtained result [8]. For the energy balance model, they use the first law of thermodynamics, using as inputs energies the electricity and the energy of the input flows, while for the output energy they use the energy of the output flows and the heat losses, in which they consider one for conduction, three for convection and three for radiation [3].

Arnout et al. [1] developed a dynamic model considering the EAF as a homogeneus in equilibrium, making the mass and energy balance at each step of time. The energy losses are classified into losses in cooling water and other losses. They only consider some elements, such as Fe, C, Mn, Si, Al, Cr, Mo, Ni, Ca, Mg, O and F. The EAF is divided in 4 zones: liquid slag, liquid steel, silicates and gases.

MacRosty and Swartz [24] modeled the EAF as a dynamic system of four zones of equilibrium. The zones approximates the behavior of each section of the furnace: gases, slag-metal interaction zone, molten steel and solid scrap. They assume that chemical equilibrium exists in the slag and gas zones, and all reactions are limited by mass transfer. This model includes for first time the radiation model of Guo and Irons. This radiation model allows to quantify the distribution of energy inside the furnace. It also determines the amount of heat extracted for the formation od slag on the wall and other parts of the furnace [13]. This work have been included after its publication in several models developed.

One of the most complete dynamic models was developed by Logar et al. [22]. This model includes a combination of the electrical, hydraulical, chemical and thermical submodels [18], and uses the principles developed by Bekker and MacRosty. In addition, the model considers the conductive and convective heat transfer between the different zones. In the model, the EAF is divided in seven different zones [19], and it considers fifteen different oxidation and reduction reactions that have influence in the mass and energy balance. The reaction kinetics are only considered in three zones of the furnace [20]. For the radiation model, the furnace is divided in five zones with simple geometric shapes, in order to simplify the calculation of the heat transferred by radiation [23]. In the same way, they developed a submodel that allows to calculate the transferred heat between the solid steel and liquid steel [21].

In the dynamic model of Kirschen et al. [17], the energy balance in the EAF considers the energy coming from the electricity, the oxygen injection and the burners of natural gas, while the energy outputs are given by the energy of liquid steel and slag, exhaust gases, heat absorbed by the cooling system, and other losses such as radiation. For the mass balance, the model are formed by the total mass and the mass of the most important metallic elements and the elements involved in the formation of the slag.

Over time, it have been created new numerical tools, such as computational fluid dynamics (CFD), that allow the development of more precise models. Gruber et al. [12] developed a model using CFD, in order to optimize the electrical and chemical energy in the furnace. This model includes zones that had not considered previously, such as electric arc and the electrodes.

Finally, Opitz and Treefinger [27] developed a complete EAF model, which is divided into four submodels: vessel, electric system, electrode regulation and off-gas system. The vessel submodels consists in three different phases: solid scrap, liquid cast and gaseous phase that include burners and electric arcs. To facilitate the analysis of the solid phase, it was divided into finite volumes, and to describe the heat transfer between the three phases, valid Nusselt correlations were used to avoid the adjusting parameters.

Chapter 3

Mass and Energy Balance Models

To make a static model for an steelmaking process in a EAF, it is necessary that the model satisfies the laws of conservation of mass and energy, which are represented in equations called balances. The models proposed here are based in the work of Trejo [30].

3.1 Model development

3.1.1 Mass balance

The mass balance allows to determine the distribution of input and output materials of the electric arc furnace from other mass known quantities and compositions, always satisfying the conservation law of mass. The mass balance was written in a vector form. The vectors, named $\boldsymbol{m_{in}}$ and $\boldsymbol{m_{out}}$, represents the mass values of the input and output streams, respectively, where their components i and j represent each stream in the process. The streams considered in this model for input and output in the furnace are presented in the table 3.1.

$$\boldsymbol{m_{in}} = \begin{bmatrix} m_{in,1} & m_{in,2} & \dots & m_{in,i} \end{bmatrix} \tag{3.1}$$

$$\boldsymbol{m_{out}} = \begin{bmatrix} m_{out,1} & m_{out,2} & \dots & m_{out,j} \end{bmatrix} \tag{3.2}$$

The concentrations of each input and output stream of the EAF are represented in a matrix $\boldsymbol{c_{in}}$ and $\boldsymbol{c_{out}}$ where the rows i and j represent to each diferent streams of the furnace and the columns k correspond to each chemical species considered in the model. The values of the matrix represent the mass fraction of each chemical species in the stream. The chemical species considered in this model are: 1) O_2, 2) CH_4, 3) CO, 4) CO_2, 5) H_2O, 6) H_2, 7) N_2, 8) C_9H_{20}, 9) Fe, 10) FeO, 11) Si, 12) SiO_2, 13) Mn, 14) MnO, 15) C, 16) Al, 17) Al_2O_3, 18) Fe_2O_3, 19) CaO, 20) MgO, 21) P, 22) P_2O_5, 23) S, 24) SO_2, 25) Cr, 26) Cr_2O_3, 27) Ti, 28) TiO_2, 29) $CaCO_3$, 30) $MgCO_3$, 31) Mo, 32) Cu and 33) Zn. Also, a molecular weight vector

Table 3.1: Input and output streams in the EAF process

Stream	Description
Input streams	
Oxygen	Added oxygen used for the decarburization of steel.
Natural gas	Burned fuel to preheat the scrap.
Other fuels in the load	Organic matter present in the scrap, as oils and wood.
Air	Air infiltrated in the furnace during the process.
Water in the load	Humidity present in the scrap.
Carbon added in the basket	Carbon added with the scrap.
Insufflated carbon	Graphite added during the process.
Anthracite	Carbon added during the process.
Slag foamer	Additive used to foam the slag.
Dolomitic lime	Additive used to mantain a specific chemical composition of the slag.
Lime	Additive used to mantain a specific chemical composition of the slag.
Calcium carbonate	Additive used to foam the slag.
Magnesium carbonate	Additive used to foam the slag.
Hot heel steel at the beginning	Remaining steel in the furnace from the previous “heat”.
Hot heel slag at the beginning	Remaining slag in the furnace from the previous “heat”.
Cold Direct Reduced Iron (DRI)	Iron obtained from iron ore reduced by a reducing gas, which is found at ambient temperature.
Hot DRI	Iron obtained from iron ore reduced by a reducing gas, which is found at 320 K.
Pig iron	High carrbon content iron.
Hot Briquetted Iron (HBI)	DRI that has been compacted.
Scrap (17 types)	Pieces of steel that will be melted, each type with different composition.
Output streams	
Liquid steel	Principal product of the process.
Slag	Mixture of oxides and secundary product of the process.
Off gases	Combustion gases produced during the process.
Dust	Solid particles suspended in the off gases.
Hot heel steel at the end	Remaining steel after tapping the furnace.
Hot heel slag at the end	Remaining slag after tapping the furnace.

for the species was defined $\boldsymbol{PM_c}$.

$$\boldsymbol{c_{in}} = \begin{bmatrix} c_{in,1,1} & c_{in,1,2} & \dots & c_{in,1,k} \\ c_{in,2,1} & c_{in,2,2} & \dots & c_{in,2,k} \\ \vdots & \vdots & \ddots & \vdots \\ c_{in,i,1} & c_{in,i,2} & \dots & c_{in,i,k} \end{bmatrix} \tag{3.3}$$

$$\boldsymbol{c_{out}} = \begin{bmatrix} c_{out,1,1} & c_{out,1,2} & \dots & c_{out,1,k} \\ c_{out,2,1} & c_{out,2,2} & \dots & c_{out,2,k} \\ \vdots & \vdots & \ddots & \vdots \\ c_{out,j,1} & c_{out,j,2} & \dots & c_{out,j,k} \end{bmatrix} \tag{3.4}$$

$$\boldsymbol{PM_c} = \begin{bmatrix} PM_{c,1} & PM_{c,2} & \dots & PM_{c,k} \end{bmatrix} \tag{3.5}$$

With the objective to perform an element mass balance, an element distribution matrix $\boldsymbol{ED}$ was constructed, where the rows k represent each chemical species and the columns l corresponds to the elements that form this species.

$$\boldsymbol{ED} = \begin{bmatrix} ED_{1,1} & ED_{1,2} & \dots & ED_{1,l} \\ ED_{2,1} & ED_{2,2} & \dots & ED_{2,l} \\ \vdots & \vdots & \ddots & \vdots \\ ED_{k,1} & ED_{k,2} & \dots & ED_{k,l} \end{bmatrix} \tag{3.6}$$

According to the variables described previously, the global mass balance is:

$$R_1 = \sum_i m_{in,i} - \sum_j m_{out,j} \tag{3.7}$$

To perform the balance by element, first two vectors are obtained from Eq. 3.8 and 3.9. The components of this vectors indicate the total mass of each compound that enter and leave the furnace.

$$\boldsymbol{m_{comp_{in}}} = \boldsymbol{m_{in}} \boldsymbol{c_{in}} \tag{3.8}$$

$$\boldsymbol{m_{comp_{out}}} = \boldsymbol{m_{out}} \boldsymbol{c_{out}} \tag{3.9}$$

Later, the vectors $\boldsymbol{m_{comp_{in}}}$ and $\boldsymbol{m_{comp_{out}}}$ are transformed to moles using the molecular weight.

$$n_{comp_{in,k}} = \frac{m_{comp_{in},k}}{PM_{c,k}} \tag{3.10}$$

$$n_{comp_{out},k} = \frac{m_{comp_{out},k}}{PM_{c,k}} \tag{3.11}$$

Finally, the balance for each element is:

$$R_{1+l} = \boldsymbol{n_{comp_{in}}ED}_{k,l} - \boldsymbol{n_{comp_{out}}ED}_{k,l} \tag{3.12}$$

In this model, 17 elements conform all the species present in the EAF process, forming a system of 18 equations. According to de data obtained from the plant, the unknowns of the systems are 1) Other fuels in the load, 2)Air, 3) Water in the load, 4) Slag, 5) Off gases, 6)Dust, 7) Hot heel steel at the end and 8) Hot heel slag at the end. As can be seen, the number of unknowns is less than the number of equations. Because of this reason, it's not convenient to use a direct method to solve the system, and it is better use an optimization method, in order to keep all the information of the heat. This optimization method consist in minimize the sum of the residual of the mass balances, to obtain the values of the unknowns. The objective function to minimize are expressed in the Eq. 3.13.

$$OF = \sum_i \sqrt{R_i^2} \tag{3.13}$$

subject to

$$m_{in,i} > 0 \tag{3.14}$$

$$m_{out,j} > 0 \tag{3.15}$$

3.1.2 Energy balance

The energy balance allows to determine the distribution of the energy. Using the first law of thermodynamics, the Eq. 3.16 represents the general energy balance for the EAF.

$$\frac{d(m\hat{E})}{dt} = \sum_i \dot{m}_{in,i}\hat{H}_i - \sum_j \dot{m}_{out,j}\hat{H}_j + \dot{W}_e - \dot{Q}_{loss} \tag{3.16}$$

where $\frac{d(m\hat{E})}{dt}$ is the energy accumulation in the system, $\dot{m}_{in,i}\hat{H}_i$ and $\dot{m}_{out,j}\hat{H}_j$ are the energy rate of the input and output flows, respectively, $\dot{W}_e$ is the electric power added to the process and $\dot{Q}_{loss}$ is the energy rate losses of the process.

Considering a quasi-steady state, and integrating the Eq. 3.16, it was obtained the following expression:

$$\sum_i m_{in,i}\hat{H}_i - \sum_j m_{out,j}\hat{H}_j + W_e - Q_{loss} = 0 \tag{3.17}$$

The estimation of the energy of the input and output flows and the energy losses of the process are explained below.

Energy of the mass flows

The principal results of the mass balance are two matrices $\boldsymbol{M_{in}}$ and $\boldsymbol{M_{out}}$, which contain the mass value of each compound (represented in the columns) in each stream (represented in the rows). This two matrices are used to estimate the enthalpy of each stream.

$$M_{in,i,k} = m_{in,i} c_{in,i,k} \tag{3.18}$$

$$M_{out,j,k} = m_{out,j} c_{out,j,k} \tag{3.19}$$

The enthalpy of each species was estimated adding the heat of formation at standard conditions, the sensible heat and the latent heat. Each compound has different phases depending of the temperature. Each phase has a different specific heat and the change from one phase to another requires different latent heat. For example, the properties of the iron (Fe) are shown on Table 3.2.

Table 3.2: Specific heat capacities and latent heats of iron. Temperatures in K.

Phase	Specific heat cp (J mol^{-1} K^{-1})	Change phase temperature (K)	Transition latent heat (J mol^{-1})
Fe_α	$12.72 + 31.71 \times 10^{-3}T$ $+2.51 \times 10^{5}T^{-2}$	1033	$\Delta\hat{H}_{\alpha\beta} = 1363.98$
Fe_β	46.56	1183	$\Delta\hat{H}_{\beta\gamma} = 899.56$
Fe_γ	$24.27 + 8.28 \times 10^{-3}T$	1673	$\Delta\hat{H}_{\gamma\delta} = 690.36$
Fe_δ	$28.20 + 6.69 \times 10^{-3}T$	1812	$\Delta\hat{H}_{\delta l} = 15355.28$
Fe_l	$40.87 + 1.67 \times 10^{-3}T$		

So, the enthalpy of the iron at temperature T_{liq} ($T_{liq} > 1812$ K) is:

$$\begin{aligned}\hat{H}_{T_{Fe}}(T_{liq}) = \Delta\hat{H}^{\circ}_{f_{Fe}} + \int_{298.15}^{1033} cp_{Fe_\alpha} dT + \Delta\hat{H}_{\alpha\beta} + \int_{1033}^{1183} cp_{Fe_\beta} dT + \Delta\hat{H}_{\beta\gamma} + \int_{1183}^{1673} cp_{Fe_\gamma} dT \\ + \Delta\hat{H}_{\gamma\delta} + \int_{1673}^{1812} cp_{Fe_\delta} dT + \Delta\hat{H}_{\delta l} + \int_{1812}^{T_{liq}} cp_{Fe_l} dT\end{aligned} \tag{3.20}$$

However, the Eq. 3.20 it is not a mathematical function when the temperature is equal to one of the change phase temperatures, causing solution problems in other models where this

equation is used. This is the reason why the equation is modified in the following way:

$$\begin{aligned}\hat{H}_{T_{Fe}}(T_{liq}) =& \Delta\hat{H}^{\circ}_{f_{Fe}} + \int_{298.15}^{1033-\Delta T} cp_{Fe_\alpha} dT + \int_{1033-\Delta T}^{1033} \frac{\Delta\hat{H}_{\alpha\beta}}{\Delta T} dT + \int_{1033}^{1183-\Delta T} cp_{Fe_\beta} dT \\ &+ \int_{1183-\Delta T}^{1183} \frac{\Delta\hat{H}_{\beta\gamma}}{\Delta T} dT + \int_{1183}^{1673-\Delta T} cp_{Fe_\gamma} dT + \int_{1673-\Delta T}^{1673} \frac{\Delta\hat{H}_{\gamma\delta}}{\Delta T} dT \\ &+ \int_{1673}^{1812-\Delta T} cp_{Fe_\delta} dT + \int_{1812-\Delta T}^{1812} \frac{\Delta\hat{H}_{\delta l}}{\Delta T} dT + \int_{1812}^{T_{liq}} cp_{Fe_l} dT\end{aligned} \tag{3.21}$$

where ΔT is a little change on the temperature, and its purpose is generate a small slope during the phase change. It is recommended that ΔT have a value of 0.1 K. To solve the problem described above, it was also considered the use of the function of distribution of Cauchy-Lorentz, but this method increases the computing time significantly and therefore it use was avoided.

The Eq. 3.21 was defined for each of the 33 compounds considered in the model and grouped in a vector $\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T)$. The temperatures of each inlet stream and outlet stream are placed in two vectors named $\boldsymbol{T_{in}}$ and $\boldsymbol{T_{out}}$, respectively.

$$\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T) = \begin{bmatrix}\hat{H}_{T_1}(T) & \hat{H}_{T_2}(T) & \dots & \hat{H}_{T_k}(T)\end{bmatrix} \tag{3.22}$$

$$\boldsymbol{T_{in}} = \begin{bmatrix}T_{in,1} & T_{in,2} & \dots & T_{in,i}\end{bmatrix} \tag{3.23}$$

$$\boldsymbol{T_{out}} = \begin{bmatrix}T_{out,1} & T_{out,2} & \dots & T_{in,j}\end{bmatrix} \tag{3.24}$$

The enthalpy of each inlet and outlet stream are located in the matrix $\boldsymbol{H_{T,in}}$ and $\boldsymbol{H_{T,out}}$, where the rows correspond to a stream and the columns corresponds to a compound, and they are calculated with the equations 3.25 and 3.26.

$$H_{T,in,i,k}(T) = \frac{M_{in,i,k}\hat{H}_{T,k}(T)}{PM_{c,k}} \tag{3.25}$$

$$H_{T,out,j,k}(T) = \frac{M_{out,j,k}\hat{H}_{T,k}(T)}{PM_{c,k}} \tag{3.26}$$

The energy provided by the inlet streams and the energy acquired by the outlet streams are:

$$H_{S,in,i} = \sum_k H_{T,in,i,k}(T_{in,i}) - \sum_k H_{T,in,i,k}(T_{ref}) \tag{3.27}$$

$$H_{S,out,j} = \sum_k H_{T,out,j,k}(T_{out,j}) - \sum_k H_{T,out,j,k}(T_{ref}) \tag{3.28}$$

where T_{ref} is the reference temperature of the enthalpies, which is 298.15 K.

To calculate the energy provided or consumed by the chemical reactions, first the amount of compounds generated or consumed during the reactions is calculated using the Eq. 3.29

$$N_{rx,k} = \frac{\sum_j M_{out,j,k} - \sum_i M_{in,i,k}}{PM_{c,k}} \tag{3.29}$$

when the value of the moles of one compound is positive, it means that this compound is a product of the reactions, and if the value is negative, the compound is a reagent of the reactions. Therefore, the energy provided or consumed by the reaction is:

$$\Delta H_r = \sum_k N_{rx,k} \bar{H}_{T,k}(T_{ref}) \tag{3.30}$$

If the sign of ΔH_r is negative, it means that the reactions are providing energy to the system. The Eq. 3.30 includes all the reactions that happen in the furnace. However, there are reactions that provide more energy than others. In this model, these reactions are the combustion of natural gas, carbon and other fuels, and also the oxidation of iron.

The combustion can produce, besides water, two different gases: CO and CO_2. Since there are not measurements of the amount of these two gases produced specifically by each fuel, the model assumes that the selectivity is equal in each all reactions producing CO and CO_2.

$$y = \frac{N_{rx,CO_2}}{N_{rx,CO_2} + N_{rx,CO}} \tag{3.31}$$

The energy delivered by the combustion of natural gas is:

$$\Delta H_{r,GN} = -N_{rx,CH_4}(y\Delta\hat{H}^\circ_{f_{CO_2}} + (1-y)\Delta\hat{H}^\circ_{f_{CO}} + 2\Delta\hat{H}^\circ_{f_{H_2O}} - (\Delta\hat{H}^\circ_{f_{CH_4}} + (\frac{3}{2} + \frac{1}{2}y)\Delta\hat{H}^\circ_{f_{O2}})) \tag{3.32}$$

The energy produced by the combustion of carbon is:

$$\Delta H_{r,C} = -N_{rx,C}(y\Delta\hat{H}^\circ_{f_{CO_2}} + (1-y)\Delta\hat{H}^\circ_{f_{CO}} - (\Delta\hat{H}^\circ_{f_C} + (\frac{1}{2} + \frac{1}{2}y)\Delta\hat{H}^\circ_{f_{O2}})) \tag{3.33}$$

The energy delivered by the combustion of other fuels is:

$$\begin{aligned}\Delta H_{r,OF} = -N_{rx,C_9H_{20}}(9y\Delta\hat{H}^\circ_{f_{CO_2}} + 9(1-y)\Delta\hat{H}^\circ_{f_{CO}} + 10\Delta\hat{H}^\circ_{f_{H_2O}} \\ - (\Delta\hat{H}^\circ_{f_{C_9H_{20}}} + (\frac{19}{2} + \frac{9}{2}y)\Delta\hat{H}^\circ_{f_{O2}}))\end{aligned} \tag{3.34}$$

The iron oxidation is an important reaction that delivers energy to the process. This energy is given by:

$$\Delta H_{r,oxFe} = -N_{rx,Fe}(\Delta\hat{H}^\circ_{f_{FeO}} - (\Delta\hat{H}^\circ_{f_{Fe}} + \frac{1}{2}\Delta\hat{H}^\circ_{f_{O_2}})) \tag{3.35}$$

And the energy deliverd by other reactions is:

$$\Delta H_{r,oth} = \Delta H_r - \Delta H_{r,GN} - \Delta H_{r,C} - \Delta H_{r,OF} - \Delta H_{r,oxFe} \tag{3.36}$$

The term corresponding to the energy of the input and output mass flows of the Eq. 3.17 is:

$$\sum_i m_{in,i}\hat{H}_i - \sum_j m_{out,j}\hat{H}_j = \sum_i H_{S,in,i} - \sum_j H_{S,out,j} + (\Delta H_{r,GN} + \Delta H_{r,C} + \Delta H_{r,OF} + \Delta H_{r,oxFe} + \Delta H_{r,oth}) \tag{3.37}$$

Energy losses

The energy losses considered in this model are:

- Electrical losses
- Heat removed by the EAF cooling system
- Heat loss in the bottom of the EAF
- Heat loss when the EAF is open
- Heat loss when the slag door is open
- Heat removed by the electrode cooling system

According to Trejo et al. [31], the electrical losses are 6% of the electrical energy supplied to the furnace

$$Q_{eloss} = (1 - \eta_e)W_e \tag{3.38}$$

where η_e is the efficiency of the electric system, in this case is 0.94.

The EAF has cooling panels in the walls, in the roof an in the elbow of gas extraction. The estimation of the thermal losses in the cooling system is:

$$Q_{cw} = (\dot{m}_{wp}\Delta T_{wp} + \dot{m}_{rp}\Delta T_{rp} + \dot{m}_{ep}\Delta T_{ep})cp_{H_2O_{(l)}}t_{TT} \tag{3.39}$$

where $\dot{m}$ is the mass flow of the cooling water, ΔT is the difference between the in and out temperatures of the cooling water and t_{TT} is the tap to tap time of the "heat".

For the heat loss in the bottom of the furnace, a model was developed, considering the heat transfer mechanisms that occur during this phenomenon. To facilitate calculations, it was assumed that the bottom of the oven has a square shape, and the system has the geometry presented in the figure 3.1.

In the model, in first place the heat conduction at the bottom of the furnace is considered

$$\dot{Q}_{btm} = -k_{ref}A_{btm}\frac{T_{btm} - T_{steel}}{\Delta x_{ref}} \tag{3.40}$$

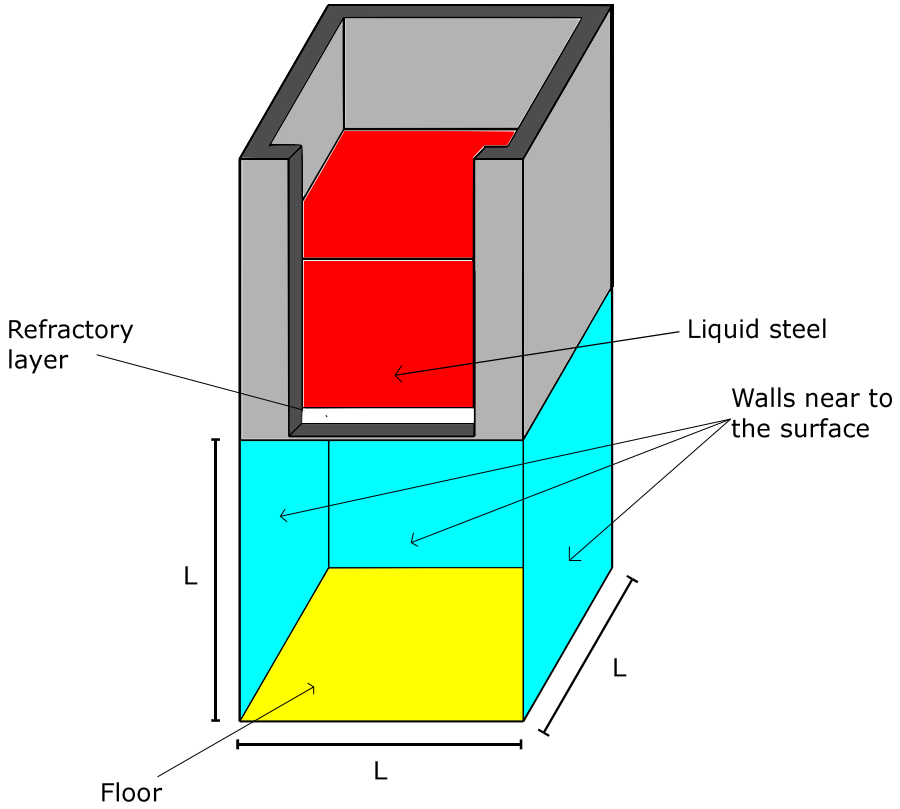

Figure 3.1: Geometry used to calculate the heat losses in the bottom of the EAF.

This heat is subsequently transferred by convection and radiation to the surroundings

$$\dot{Q}_{btm} = \dot{Q}_{conv} + \dot{Q}_{rad} \tag{3.41}$$

The heat transferred by convection is given by the following equation.

$$\dot{Q}_{conv} = h_{btm} A_{btm} (T_{btm} - T_{sur}) \tag{3.42}$$

where the convection heat transfer coefficient is calculated using a correlation of hot plate surface facing down (Eq. 3.43) [15].

$$\bar{Nu}_{L_c} = 0.27 Ra_{L_c}^{\frac{1}{4}} \tag{3.43}$$

$$Ra_{L_c} = \alpha (L_c^3)(T_{btm} - T_{sur}) \tag{3.44}$$

$$L_c = \frac{A_{btm}}{P_{btm}} \tag{3.45}$$

$$\bar{Nu}_{L_c} = \frac{h_{btm} L_c}{k_{air}} \tag{3.46}$$

For the heat transferred by radiation, four surfaces was considered: 1) EAF bottom, 2) walls near to the EAF, 3) floor and 4) distant surroundings. The radiation circuit is formed by the following equations:

$$\frac{Eb_m - J_m}{\frac{1-\epsilon_m}{\epsilon_m A_m}} = \sum_n \frac{J_m - J_n}{\frac{1}{A_m F_{mn}}} \tag{3.47}$$

$$Eb_m = \sigma T_m^4 \tag{3.48}$$

$$\frac{Eb_1 - J_1}{\frac{1-\epsilon_1}{\epsilon_1 A_1}} = \dot{Q}_{rad} \tag{3.49}$$

$$\frac{Eb_2 - J_2}{\frac{1-\epsilon_2}{\epsilon_2 A_2}} = \frac{Eb_3 - J_3}{\frac{1-\epsilon_3}{\epsilon_3 A_3}} = 0 \tag{3.50}$$

Solving the equation system (Eq. 3.40 to 3.50), it can obtain the heat rate loss in the furnace bottom ($\dot{Q}_{btm}$), and the total heat loss in the bottom is

$$Q_{btm} = \dot{Q}_{btm} t_{TT} \tag{3.51}$$

The energy losses that occur when the furnace and the slag door are open are manly caused by radiation. These losses are given by the following equations.

$$Q_{fo} = \sigma A_{roof}(T_{slag}^4 - T_{sur}^4)(t_{CH} + t_{TP} + t_{IN}) \tag{3.52}$$

$$Q_{sd} = \sigma A_{sd}(T_{slag}^4 - T_{sur}^4) t_{TT} \tag{3.53}$$

where t_{CH}, t_{TP} and t_{IN} are the charging time, the tapping time and the inspection time, respectively.

The electrode coolling system consists in a water sprinkler that keeps the electrode wet. The water evaporates with the heat released by the electrode. So, the energy lost in this system is:

$$Q_{ce} = \dot{m}_{ce}(cp_{H_2O}(T_{out,we} - T_{in,we}) + \Delta\hat{H}_{vap,H_2O}) t_{PO} \tag{3.54}$$

where t_{PO} is the power of time of the furnace, and $T_{out,we}$ is the temperature vaporization of the cooling water.

Therefore, the term of the energy losses of Eq. 3.17 is

$$Q_{loss} = Q_{eloss} + Q_{cw} + Q_{btm} + Q_{fo} + Q_{sd} + Q_{ce} \tag{3.55}$$

3.2 Results and discussion

The algorithm to solve the model proposed in the previous section was programmed in Matlab. The model was applied and validated using the process data obtained of ten heats from the Fuchs EAF of Ternium. This process data is presented on appendix.

In order to validate the mass balance, three parameters were evaluated: the residuals of the mass balances per element, the slag production during the process and the amount of hot heel remaining in the furnace.

The figure 3.2 shows the normalized average of the residuals obtained from the model for each element. The residuals were normalized using the total mass of each element loaded to the furnace. A positive value of the residual indicates the appearance of the elements during the process, while a negative value shows the disappearance of the element. The elements that presents a greater normalized residual are aluminum, zinc, manganese, phosphorus, sulfur, chromium, titanium, molybdenum, copper and hydrogen. These elements are found in a lower proportion during the process. These residuals arise because the concentrations of each compound in the mass flows are known in the process, and the unknown values of the mass flow are adapted to the compositions with the objective to satisfy the mass balance. The sum of this residuals is around of 6 tons, which represents 2.23% of the mass added to the furnace.

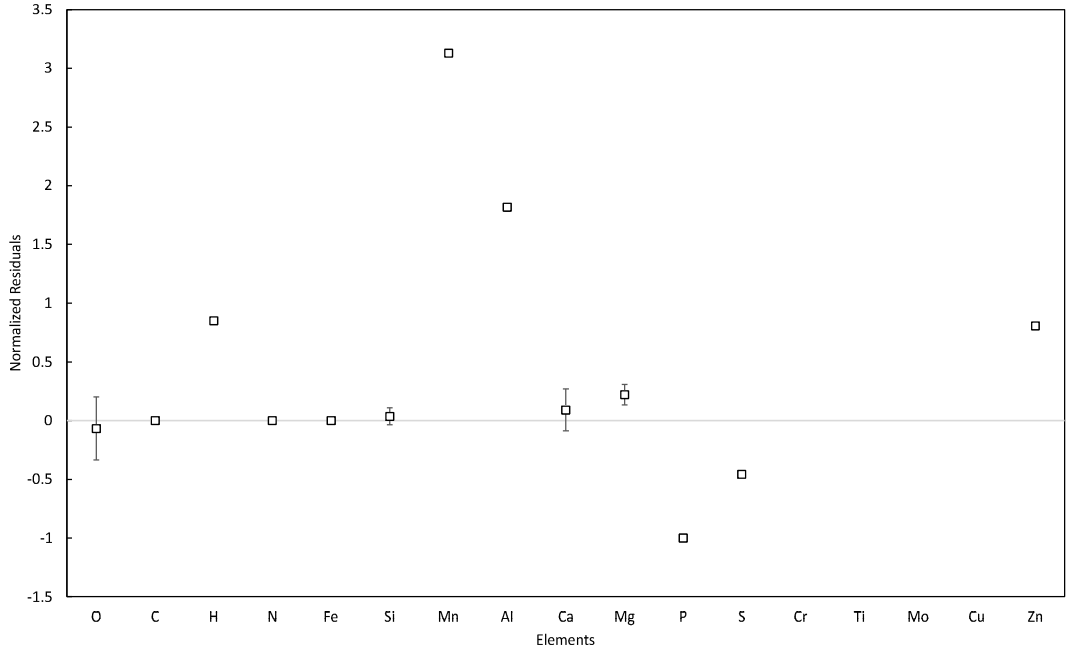

Figure 3.2: Mass residuals per element obtained from mass balance.

The figure 3.3 shows the comparision between the measured amount of slag obtained during the process and the amount of slag resulting from the model. As can be seen, in most cases the measured and simulated data are very similar. The average difference between this two types of data is 4.3 tons of slag produced.

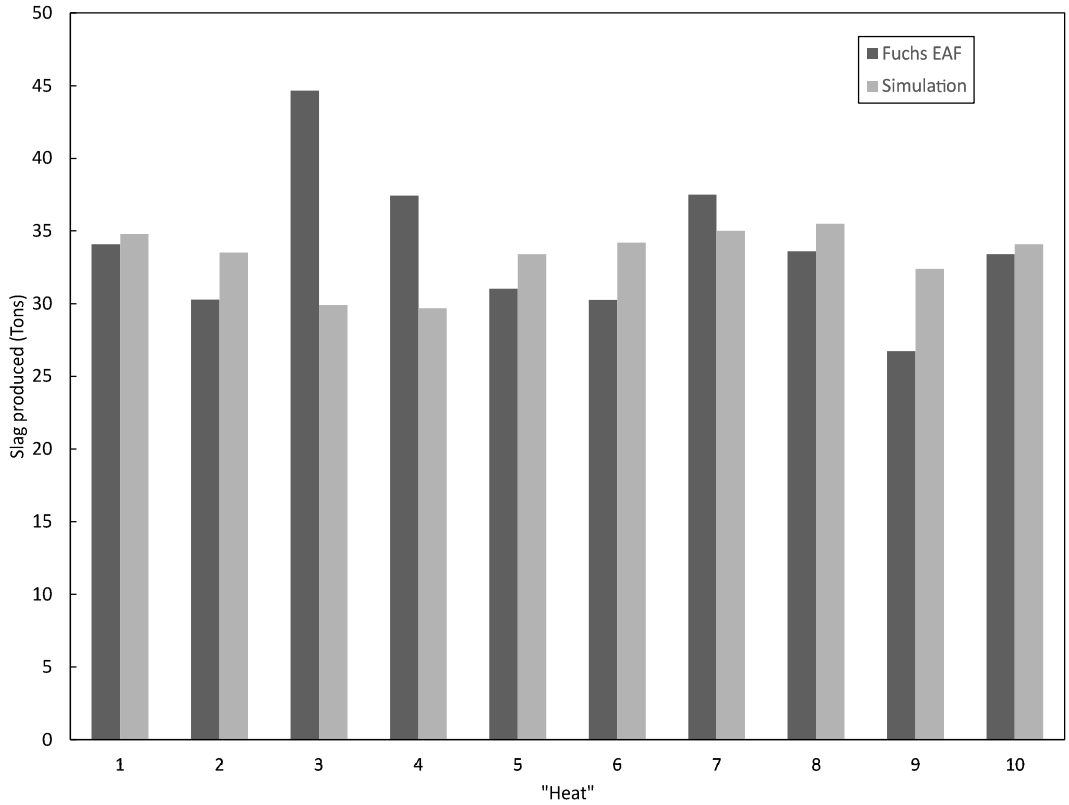

Figure 3.3: Mass of slag obtained in each “heat”.

The final parameter to validate the model is the amount the hot heel remaining in the furnace. The hot heel is formed by liquid steel and slag, which remain in the furnace after the tapping. The amount of hot heel is not measured regularly, but is know from experience that is around of 30 tons for liquid steel and 10 tons for slag, and its value does not change during the heats. In each simulation, the values above mentioned were considered as input flows, and it were compared with the output hot heel values obtained from the simulation. The results are shown in the figure 3.4. As can be seen, in most cases there is a decrease in the amount of hot heel, being more significant in the liquid steel. This result are contrary to the results obtained in the mass balances per element, where the iron is one of the elements that has better behavior in the mass balance.

In order to validate the energy balance, a new term was added to the equation 3.17, denominated unknown losses. This term allows to know the difference in magnitude between the energy sources and energy sinks during the process, and also identify posible errors in the energy balance. The figure 3.5 shows the sankey diagram of energy distribution of a heat in the EAF Fuchs.

As can be seen, the magnitude of the unknown losses are 127 kWh per ton of liquid steel produced, what represents 11.5% of the total energy suministred to the furnace. Another value that is above the expected is the energy provided by the other reactions, whose magnitude is

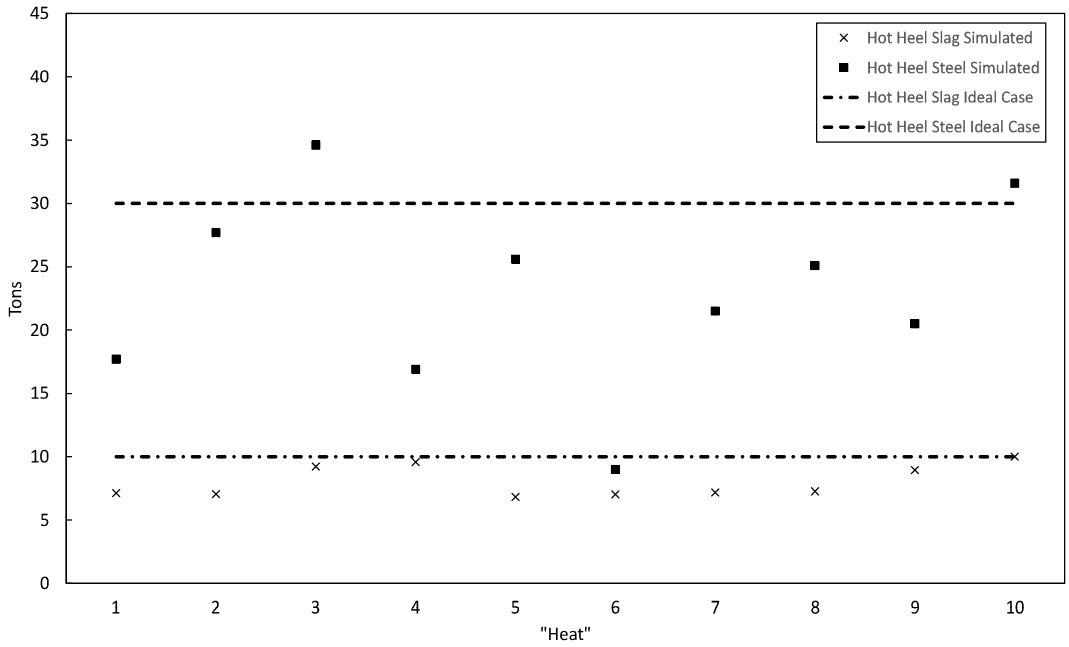

Figure 3.4: Mass of hot heel obtained in the simulation for each “heat”.

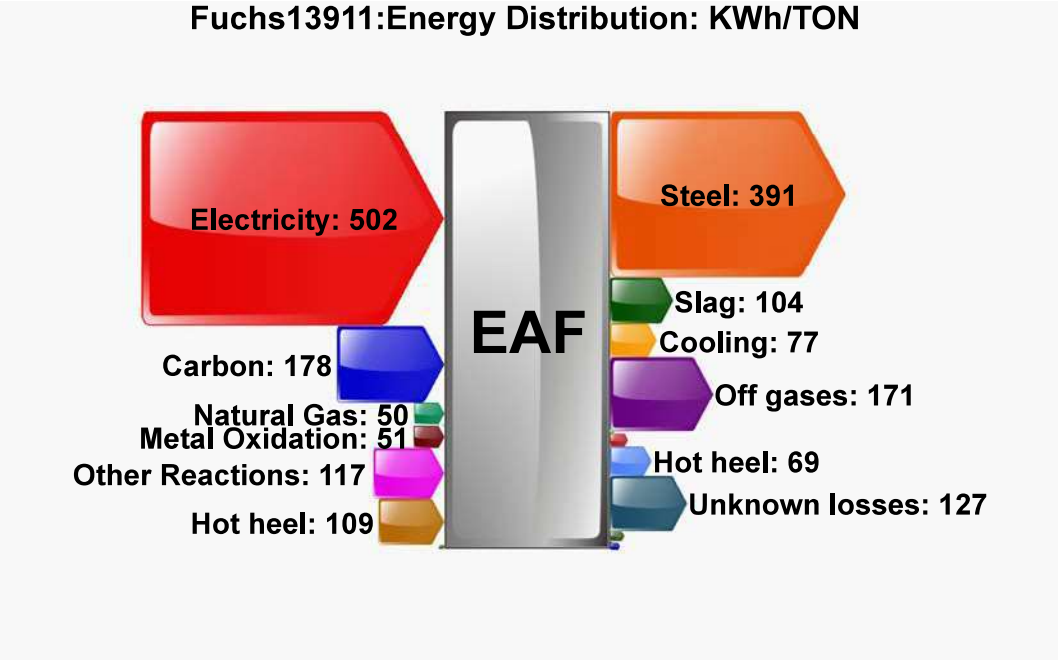

Figure 3.5: Energy distribution in the EAF, considering all the reactions.

similar to the unknown losses. The origin of the energy coming from the other reactions was analized and it was found that this energy comes from the formation of oxides of calcium, magnesium, manganese, aluminum and silicon. These elements are those present a greater appearance in the mass balance, and appear in the form of oxides. When the energy balance compares the amount of inlet and outlet compounds from the furnace, the model considers the appearance as a formation of these oxides inside the furnace, which causes the energy release

and the increase in the magnitude of the energy provided by the other reactions. This is the reason why the energy provided by the other reactions was considered zero in the energy balance in the later calculations. The table 3.3 shows the energy contributed by this reactions

Table 3.3: Energy contributed by elements that appear in the mass balance in “heat” 1 of EAF Fuchs.

Element	Element mass (Tons)	Formed oxide mass (Tons)	Energy contributed (kWh/TLS)
Si	0.2241	0.4796	12
Mn	0.4202	0.5425	5.57
Al	0.7256	1.3711	42.53
Ca	0.4816	0.6738	14.46
Mg	0.4417	0.7325	20.71

The figure 3.6 show the sankey diagram of energy distribution of the same heat in figure 3.5, now considering the energy provided by other reactions as zero. As can be seen , the unknown losses have practically disappeared. The figure 3.7 presents the unknown losses obtained from the energy balances of many heats, and also the percentage that represent the unknown losses of the total energy supplied to the furnace. The unknown losses do not exceed the 25 kWh/ton of liquid steel, and represent on average 1.74% of the total energy supplied to the furnace.

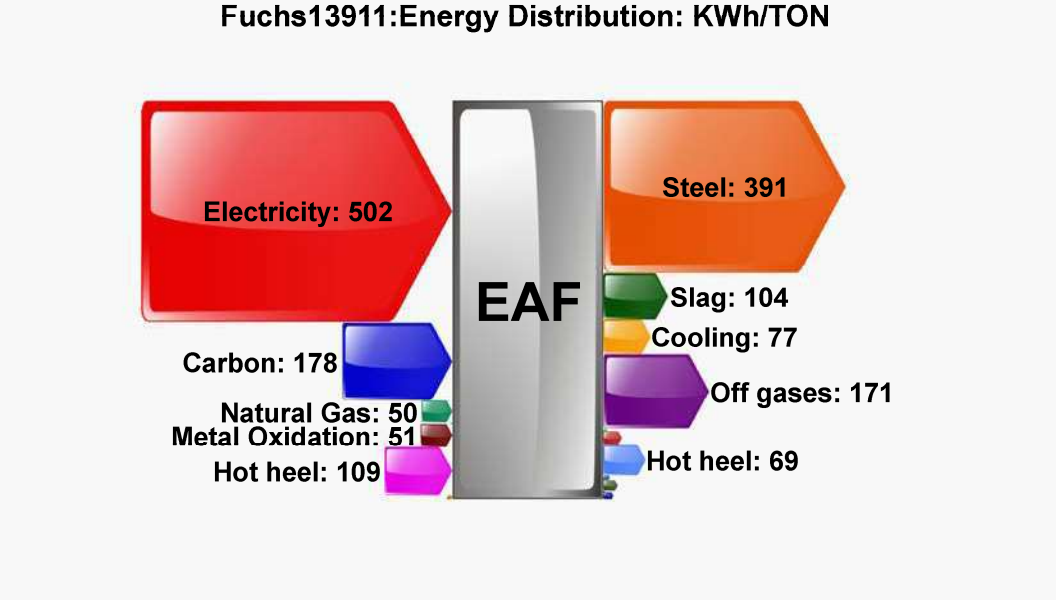

Figure 3.6: Energy distribution in the EAF, considering only selected reactions.

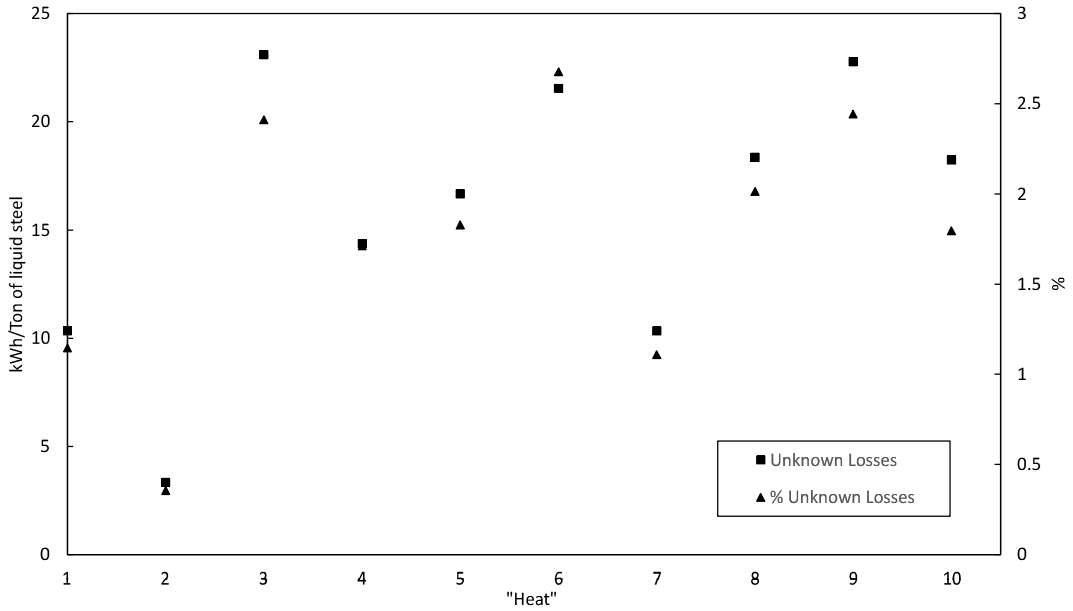

Figure 3.7: Unknown losses from each “heat” and percentage that represent from total energy supplied.

Chapter 4

Heat Rate Transfer Model

4.1 Model development

The heat transfer model is composed of the energy transport equations by radiation, conduction and convection. The assumptions considered for this model are as follows:

- Transient state.
- The model is made for flat bath conditions.
- It considers 27 isothermal internal surfaces in the EAF.
- The electric arc is a cylinder with a diameter of 0.1 m and a height of 0.45 m.
- All energy sources, chemical and electrical, are considered as inputs.
- The radiation of the gas atmosphere of the furnace was neglected.
- The heat received by the cooling panels are completely removed by the cooling system.
- The molten surface includes the molten slag and the liquid steel.

4.1.1 Radiative heat transfer

The heat transfer by radiation is a phenomenon that depends of the geometry of the EAF, and can be represented as an electrical network, as shown in the figure 4.1.

To build an electrical network for a specific problem of radiation, a surface resistance $\frac{1-\epsilon_m}{\epsilon_m A_m}$ is added to each surface m on the geometry and a resistance of space $\frac{1}{A_m F_{mn}}$ is connected among potencial radiosity (J) of surfaces m and n. The variable F is defined as view factor, and represent the fraction of radiant energy leaving a surface and reaching another surface of the system. The radiosity is the total radiation that leaves the surface per time unit and area unit. According to Holman [14], the recommended method to solve this network is the

application of Kirchoff's current law for the circuit. The equations for the radiation circuit are:

$$\frac{Eb_m - J_m}{\frac{1-\epsilon_m}{\epsilon_m A_m}} - \sum_n \frac{J_n - J_m}{\frac{1}{A_m F_{mn}}} = 0 \tag{4.1}$$

where Eb_m is the black body radiation of the surface m.

$$Eb_m = \sigma T_m^4 \tag{4.2}$$

The heat absorbed by radiation for a surface is

$$\dot{Q}_{rad,m} = \frac{J_m - Eb_m}{\frac{1-\epsilon_m}{\epsilon_m A_m}} \tag{4.3}$$

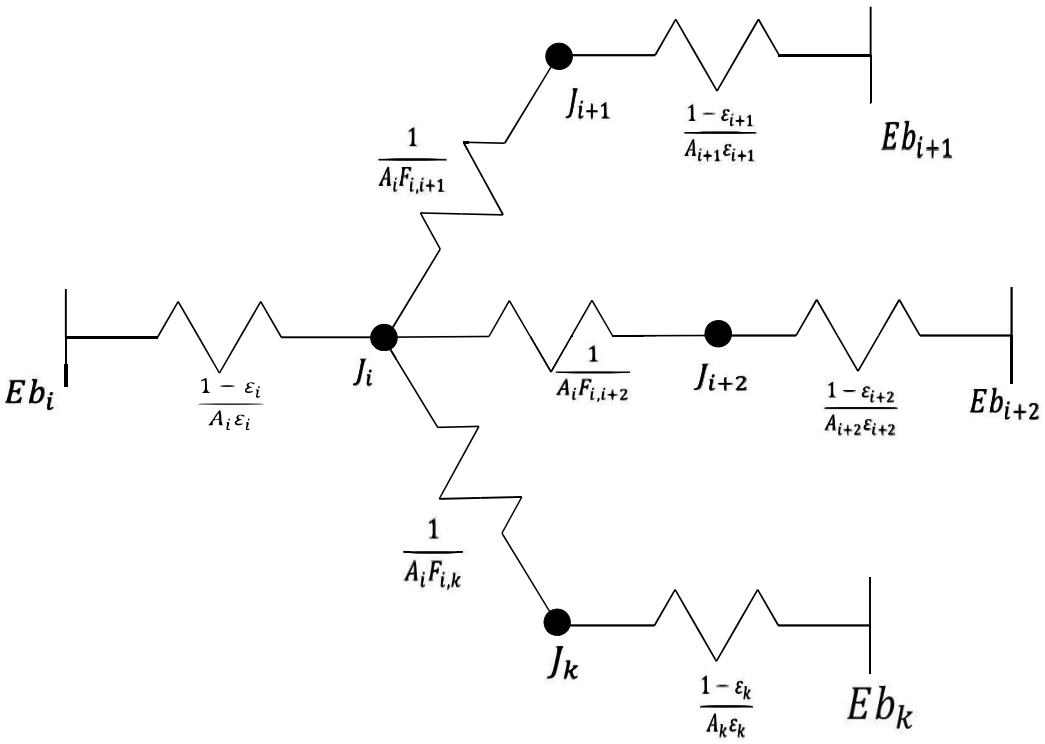

Figure 4.1: Radiative heat transfer represented as a circuit.

The surfaces considered in this model are: 1) to 14) Fourteen walls cooling panels, 15) to 20) Six roof cooling panels, 21) Out gases area, 22) Electrode, 23) Slag door, 24) Upper refractory line, 25) Lower refractory line, 26) Molten surface, 27) Arc. The geometry used is shown in figure 4.2.

4.1.2 Energy delivered by arc

According to Sánchez et al. [28], the electric power generated in the arc is transferred to the surroundings in three ways: by electrons ($\dot{Q}_e$), by radiation ($\dot{Q}_{rad}$) and by "convection" ($\dot{Q}_{conv}$). The energy transferred by electrons represents the 3.5% of the electric power, and

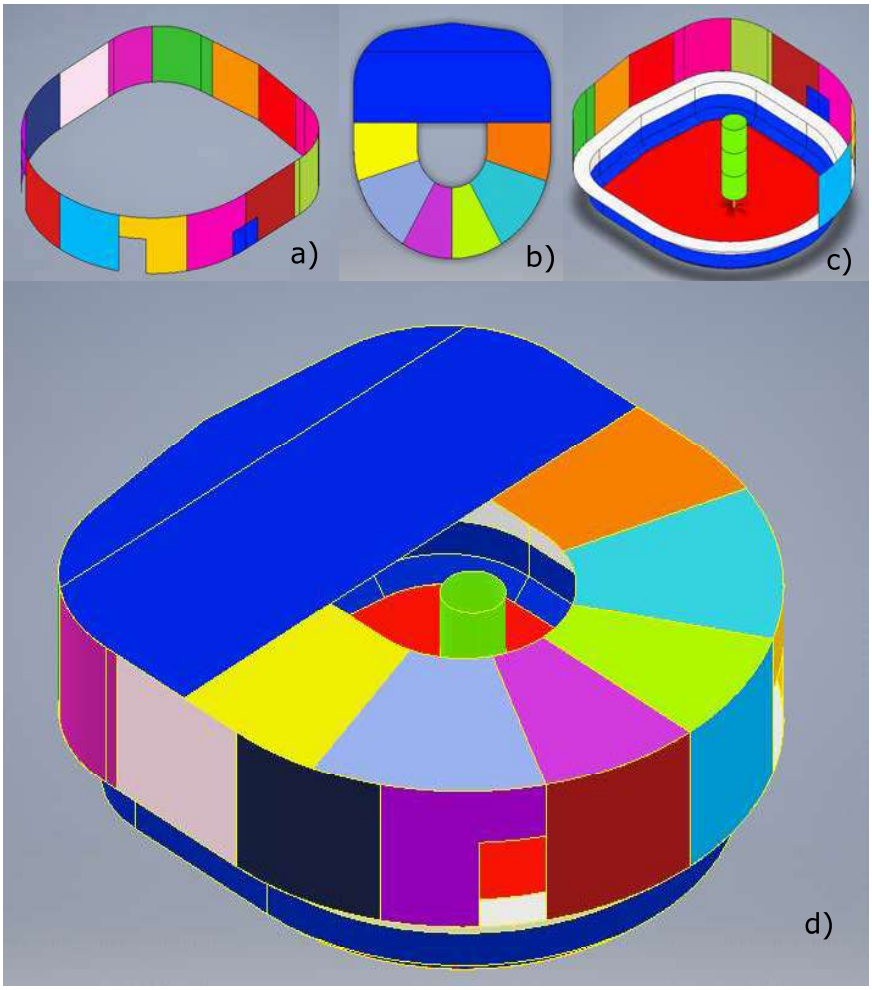

Figure 4.2: Geometry of the Fuchs EAF. a) The 14 cooling panels and the slag door. b) Top view of the furnace. It is shown the the six cooling panels of the roof and the out gases area. c) Interior view of the furnace. d) Complete geometry of the furnace.

its delivered directly to the molten slag and the liquid steel. The energy transferred by radiation represents the 24% of the electric power. This heat is delivered from the arc to the other surfaces of the EAF through the radiation circuit. The energy transferred by ”convection” represents the 72.5% of the electric power. The heat transferred by “convection” can be delivered to the molten slag and to the gases present in the furnace. The amount of heat delivered to each zone depends of the coverage index of the arc (cov). The coverage index of the arc indicates the percentage of the arc that is covered by the molten slag, and its value can be found between 0 and 1.

$$\dot{Q}_{conv,slg} = (cov)\dot{Q}_{conv,arc} \tag{4.4}$$

$$\dot{Q}_{conv,gas} = (1 - cov)\dot{Q}_{conv,arc} \tag{4.5}$$

4.1.3 Heat transfer in the cooling panels

The EAF have fourteen tubular panels in the walls, six tubular panels in the roof and tubular panels in the off gases elbow. These panels receive heat through radiation from other surfaces, and convection from the gases found inside of the furnace. The heat is subsequently removed by the water flowing through the pipes. In most of the EAF's, a layer of slag is formed on the sidewall and the roof may be formed. The thickness of the layer varies among furnaces and has a significant contribution to energy savings. The heat is transferred through this layer by one-dimensional conduction. The figure 4.3 shows the diagram of the cooling panel, and the heat transfer mechanisms involved in the panels.

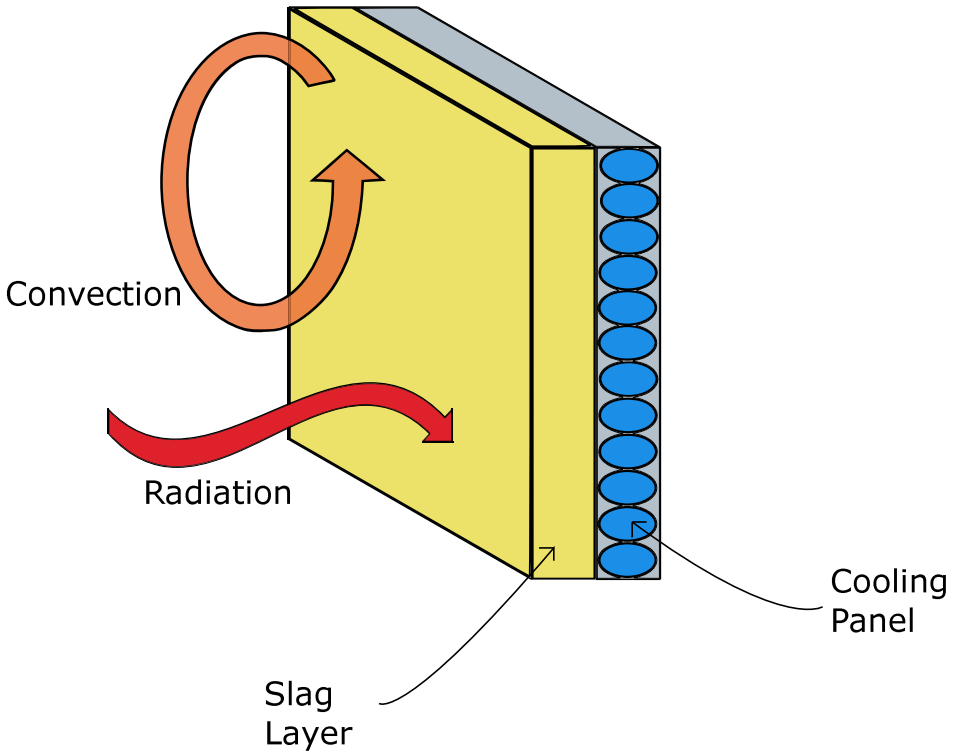

Figure 4.3: Heat transfer in cooling panels.

The equations for the heat transfer in the cooling panels are the following:

$$h_g A_m (T_g - T_m) + \dot{Q}_{rad,m} = \frac{A_m (T_m - T_{cool,m})}{\frac{\Delta x_{slg}}{k_{slg}}} \tag{4.6}$$

$$\frac{A_m (T_m - T_{cool,m})}{\frac{\Delta x_{slg}}{k_{slg}}} = U_m A_m (T_{cool,m} - \frac{T_{win,m} + T_{wout,m}}{2}) \tag{4.7}$$

$$U_m A_m (T_{cool,m} - \frac{T_{win,m} + T_{wout,m}}{2}) = \dot{m}_{w,m} cp_w (T_{wout,m} - T_{win,m}) \tag{4.8}$$

where h_g is the convection heat transfer coefficient for the panels, T_g is the gas temperature, T_m are the temperatures of the radiative surfaces, $T_{cool,m}$ are the temperatures of the panels,

Δx_{slg} is the thickness of the slag layer, k_{slg} is the thermal conductivity of the slag, U_m are the global heat transfer coefficients for each panel and $T_{win,m}$ and $T_{wout,m}$ are the inlet and outlet temperatures of cooling water, respectively.

To calculate the global heat transfer coefficient of each panel, the method proposed by Contreras [6] was used. The global heat transfer coefficient is given by the following equation:

$$U_m = \frac{1}{A_m(\frac{\Delta x_{pipe}}{k_{steel}A_m} + \frac{1}{h_{pn}\pi d_{pipe}L_{pipe}})} \tag{4.9}$$

where Δx_{pipe}, L_{pipe} and d_{pipe} are the thickness, the length and the internal diameter of the pipes in the panels, respectively, k_{steel} is the thermal conductivity of the steel, and h_{pn} is the internal heat transfer coefficient of each panel. This heat transfer coefficient is calculated using the Gnielinsky's correlation for internal flow (Eq. 4.10).

$$h_{pn}\frac{d_{pipe}}{k_{water}} = \frac{(\frac{f_{pn}}{8})(Re_{pn} - 1000)Pr_w}{1 + 12.7(\frac{f_{pn}}{8})^{0.5}(Pr_w^{\frac{2}{3}} - 1)} \tag{4.10}$$

where the Reynolds number for each panel (Re_{pn}), the Prandtl number for water (Pr_w) and the friction factor for each panel (f_{pn}) are calculated as follows:

$$Re_{pn} = \frac{\rho_w v_{pn} d_{pipe}}{\mu_w} \tag{4.11}$$

$$Pr_w = \frac{cp_w \mu_w}{k_w} \tag{4.12}$$

$$\frac{1}{f_{pn}^{0.5}} = -2\log_{10}(\frac{2.51}{Re_{pn}f_{pn}^{0.5}} + \frac{e}{3.7d_{pipe}}) \tag{4.13}$$

4.1.4 Other heat losses

The electrode receive heat from other surfaces by radiation and from the gases inside the furnace by radiation. The heat generated by Joule effect produced by circulating electricity through the electrode was neglected. The heat received by the electrode is dissipated by its cooling system, in which the water evaporates and is added to the gases inside the furnace.

$$h_g A_{22}(T_g - T_{22}) + \dot{Q}_{rad,22} = \dot{m}_{ce}(\hat{H}_{T,H_2O}(T_g) - \hat{H}_{T,H_2O}(T_{in,ce})) \tag{4.14}$$

While the energy received by the slag door is radiated to the surroundings.

$$\dot{Q}_{rad,23} = \sigma A_{23}(T_{23}^4 - T_{sur}^4) \tag{4.15}$$

The refractory lines are formed by a layer of slag, a layer of refractory brick and a metallic shell, as shown in fugure 4.4. The heat received from the radiation and the convection of the gases is transmitted by conduction in the three layers described above and is subsequently dissipated to the enviroment by convection and radiation.

$$h_g A_m (T_g - T_m) + \dot{Q}_{rad,m} = \frac{A_m (T_m - T_{os,m})}{\frac{\Delta x_{slg}}{k_{slg}} + \frac{\Delta x_{ref}}{k_{ref}} + \frac{\Delta x_{shell}}{k_{shell}}} \tag{4.16}$$

$$\frac{A_m (T_m - T_{os,m})}{\frac{\Delta x_{slg}}{k_{slg}} + \frac{\Delta x_{ref}}{k_{ref}} + \frac{\Delta x_{shell}}{k_{shell}}} = A_m h_{air} (T_{os,m} + T_{sur}) - A_m \sigma (T_{os,m}^4 - T_{sur}^4) \tag{4.17}$$

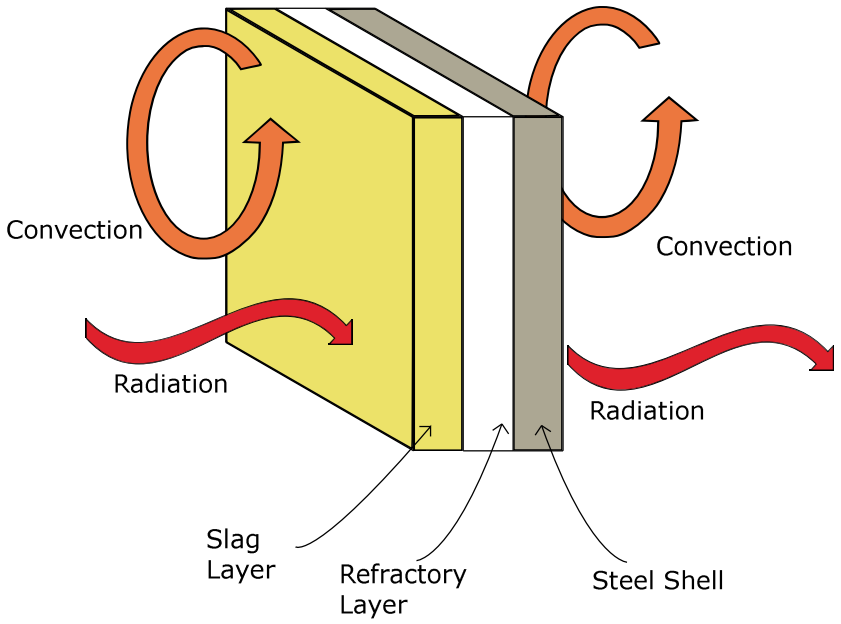

Figure 4.4: Heat transfer in refractory line.

4.1.5 Energy balances

An energy balance in the molten surface (liquid steel and molten slag) is performed to account for the electrical energy delivered in the furnace

$$\frac{d(m\hat{E})}{dt} = \sum_i \dot{m}_{in,i} \hat{H}_{in,i} - \sum_j \dot{m}_{out,j} \hat{H}_{out,j} + \sum \dot{Q} \tag{4.18}$$

specifying each of the heat flows that affect this zone , the equation 4.18 can be approximated as:

$$\begin{aligned} &\frac{\boldsymbol{M}_{f,26} \cdot (\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{26}) - \hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref})) - \boldsymbol{M}_{0,26} \cdot (\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{0,26}) - \hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))}{\Delta t} = \\ &\dot{\boldsymbol{M}}_{in,26} \cdot (\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{in}) - \hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref})) - \dot{\boldsymbol{M}}_{steel} \cdot (\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{26}) - \hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref})) \\ &- \dot{\boldsymbol{M}}_{slag} \cdot (\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{26}) - \hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref})) + \dot{Q}_{rad,m} + h_g A_{26} (T_g - T_{26}) + \dot{Q}_{conv,slg} + \dot{Q}_e \\ &+ \dot{Q}_{comb} + \dot{Q}_{oxy} - \dot{Q}_{btm} \end{aligned} \tag{4.19}$$

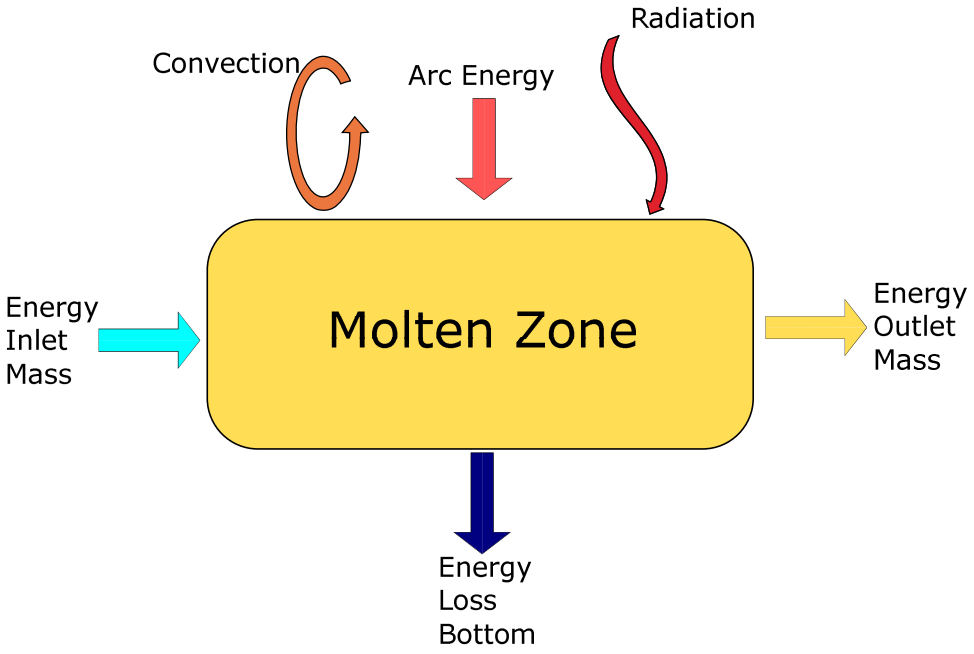

Figure 4.5: Energy balance in the molten zone.

and similarly, the energy balance for the gas zone is:

$$\begin{aligned}&\frac{\boldsymbol{M}_{f,g}\cdot(\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_g)-\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))-\boldsymbol{M}_{0,g}\cdot(\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{0,gaszone})-\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))}{\Delta t}=\\&\dot{\boldsymbol{M}}_{in,g}\cdot(\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{in})-\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))-\dot{\boldsymbol{M}}_{offgas}\cdot(\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_g)-\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))\\&-\dot{\boldsymbol{M}}_{dust}\cdot(\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_g)-\hat{\boldsymbol{H}}_{\boldsymbol{T}}(T_{ref}))-\sum_m h_g A_m(T_g-T_m)+\dot{Q}_{conv,gas}+\dot{Q}_{gn}\end{aligned}\tag{4.20}$$

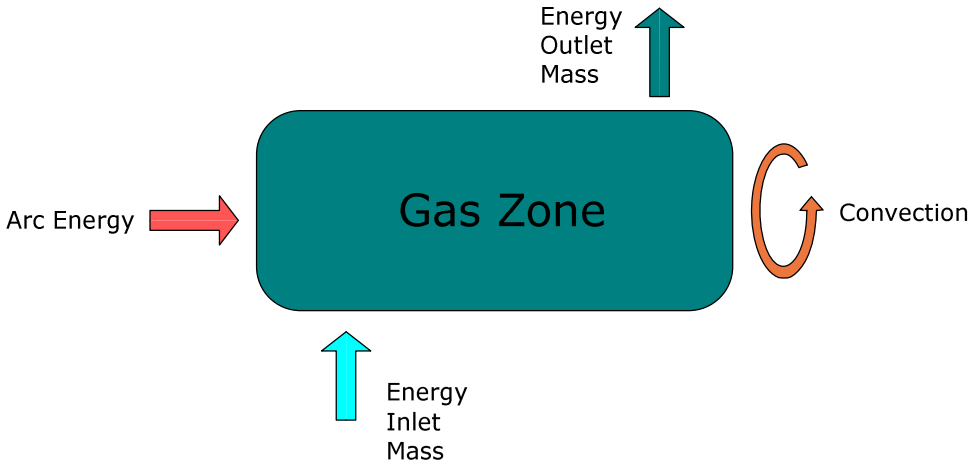

Figure 4.6: Energy balance in the gas zone.

The overall model is composed by the simultaneous solution of the radiative heat transfer, the heat transfer in the cooling panels, the other losses models and the energy balances of gases and molten surface. With this model it is possible to obtain the temperatures of each surface

in which the furnace is divided (T_m), the temperature of the gases (T_g), the temperatures of the cooling panels ($T_{cool,m}$) and the outlet temperatures of the cooling water of the panels ($T_{wout,m}$).

4.2 Results and discussion

To estimate the view factors in the furnace, the furnace surfaces involved in the radiative heat transfer were build in CAD format, and imported into a finite element , which is Steady State Thermal Analysis System, from Ansys®. The geometry was meshed using tetrahedral elements.

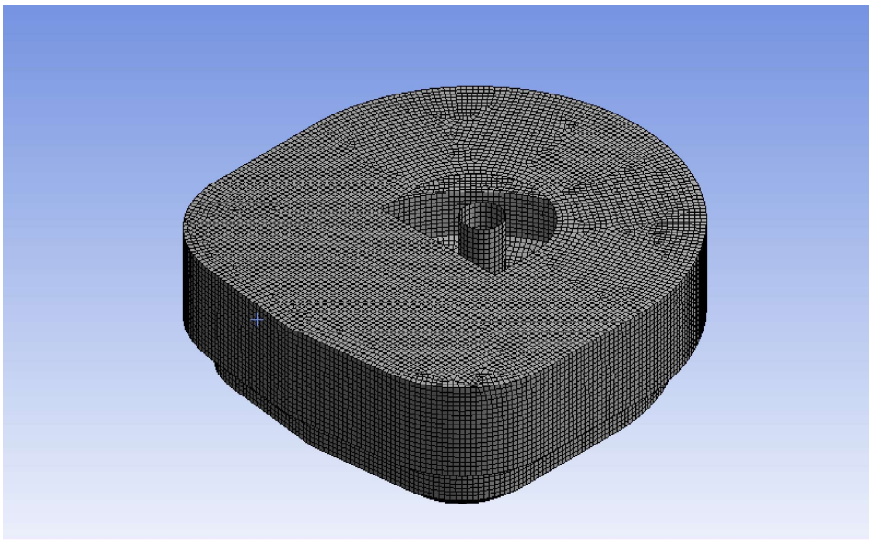

Figure 4.7: Meshed geometry used to estimate view factors.

To calculate the view factor between each pair of surfaces, it was adopted the method used by Guo and Irons, a constant temeperature boundary condition of 1000 K was set for one of the surface, while the temperatures of the other surfaces are 0 K. The condition for all the surfaces of the furnace was established as surface to surface radiation, and the emissivity of all the surfaces was one. The parameters used in the simulation are presented in table 4.1. After simulation, the software provides the total radiative energy flow emitted by the hot surface and the energy received for the cool surfaces. The view factors are the fraction of the energy emitted by the hot surface that each cool surface receives. This procedure was made for each surface in the furnace. The view factors estimated are presented in the appendix.
The algorithm to solve the model proposed in the previous section was programmed in Matlab. The model was applied using the data presented in table 4.2. The data used in this model are based in the process of Fuchs EAF of Ternium. The characteristics of the panels are presented in table B.2.

With the model presented previously, the effects of some variables on the energetic performance were analyzed, in order to verify the robustness of the model. These variables are the

Table 4.1: Parameters used to estimate the view factors.

Parameter	Value
Maximum face size	0.075 m
Emissivity	1
Correlation	Surface to Surface
Hemicube resolution	100
Temperature of hot surface	1000 K
Temperature of cold surfaces	0 K

Table 4.2: Parameters used in the heat transfer model.

Parameter	Value
Arc length	0.45 m
Initial temperature of molten zone	1815 K
Initial and final mass in molten zone	150 ton of liquid steel, 10 ton of slag
Outlet mass flow of slag	8.33 kg/s
Inlet mass flow temperature to molten zone	298.15 K
Initial temperature of gases	1400 K
Initial and final mass of gases	15.1 kg
Outlet mass flow of gases	16.17 kg/s
Inlet mass flow temperature to gases	298.15 K
Flow of cooling water	900, 500 and 200 m^3/s for wall, roof and elbow, 0.833 kg/s for electrode
Inlet temperature of cooling water	300.15 K
Convection coefficient for internal gas	3700 W/m^2K
Convection coefficient for ambient air	20 W/m^2K
Thickness of refractory layer	0.5 m
Thickness of metal layer	0.01109 m
Thermal conductivity of refractory	2.4 W/m K
Thermal conductivity of slag	2.2 W/m K
Thermal conductivity of metal layer	60.5 W/m K
Emissivity of the internal surfaces	0.7
Heat rate from carbon combustion	8458.33 W
Heat rate from oxydation	1375 W
Heat rate from gas natural combustion	2625 W

electric power of the arc, the coverage index of the arc and the slag layer thickness.

The Figure 4.8 shows the energy rate gained or lost by the molten zone, the gases and the cooling system, with a coverage index of 0.9 and a slag layer thickness of 0.015 m for one second of operation. As can be seen, the electric power has no influence in the heat rate of the cooling system and it slightly increases the amount of energy rate gained by the gases inside the furnace. However, the electric power has a significant effect on the energy rate of the

molten zone. The molten zone yields a large amount of energy to the other surfaces because it has the greater surface area and the higher temperature inside the furnace. This energy loss is subsequently replaced by the energy coming from the arc. If the electric power is not high enough, the molten zone will continue lost energy and the process will have negative efficiencies. It can also be observed the existence of a minimun electrical power in which the molten zone does not lose energy, and above that electric power the molten zone gains energy. The Figure 4.9 shows the percentage of electric power suministred to the furnace gained by the molten zone depends of the electrical power to which the furnace operates. A greater electrical power means that the liquid steel gains a higher percentage of the electric energy supplied and therefore the efficiency of the furnace increases.

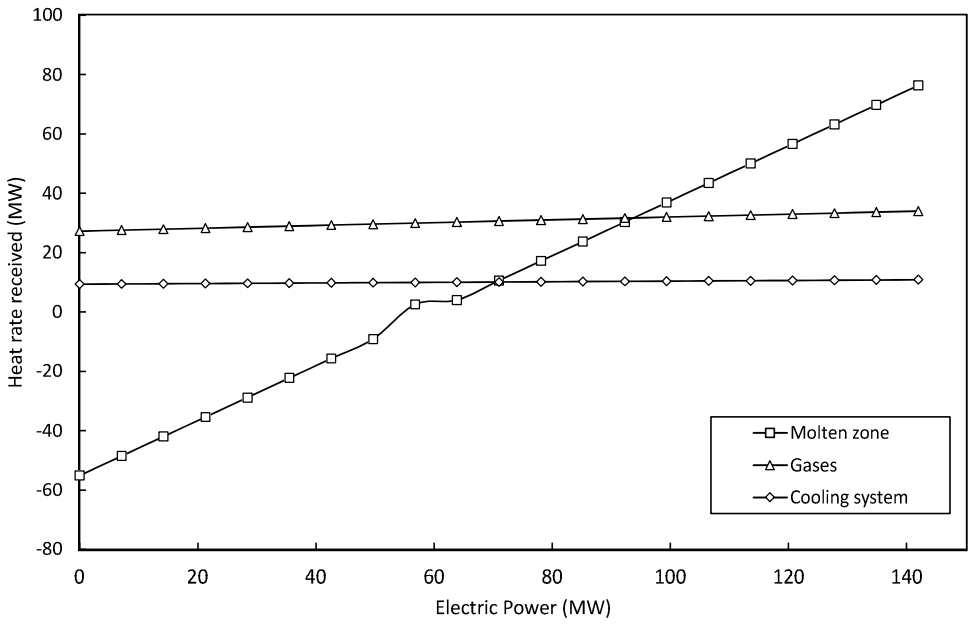

Figure 4.8: Heat rate received by the molten zone, the gases and the cooling system at different electric power, with a coverage index of 0.9 and slag layer thickness of 15 mm.

The Figure 4.10 shows the effect of the coverage index of the arc in the temperatures of the for gases, molten zone and wall cooling panels. The decrease in the coverage index causes an important increase in the temperature of the gases, because the gases obtain directly more energy from the arc. This increase in the temperature of the gases causes a greater heat transfer by convection between the gases and the cooling panels, which causes a slight increase in the temperature of the panel.

The most important variable in the heat transfer for the cooling panels is the slag layer that is formed on the surface of the panels. This slag layer work as an insulator and it brings two benefits to the process: prevents greater heat losses through the cooling system and protects the cooling panels from perforation caused by high temperatures in the pipes. The Figure 4.11 shows the temperatures of the gases, the molten zone and the wall cooling panels with different slag layer thickness. The increment of the slag layer thickness from 1 to 2 cm can decrease in 25 K the temperature of the cooling panels.

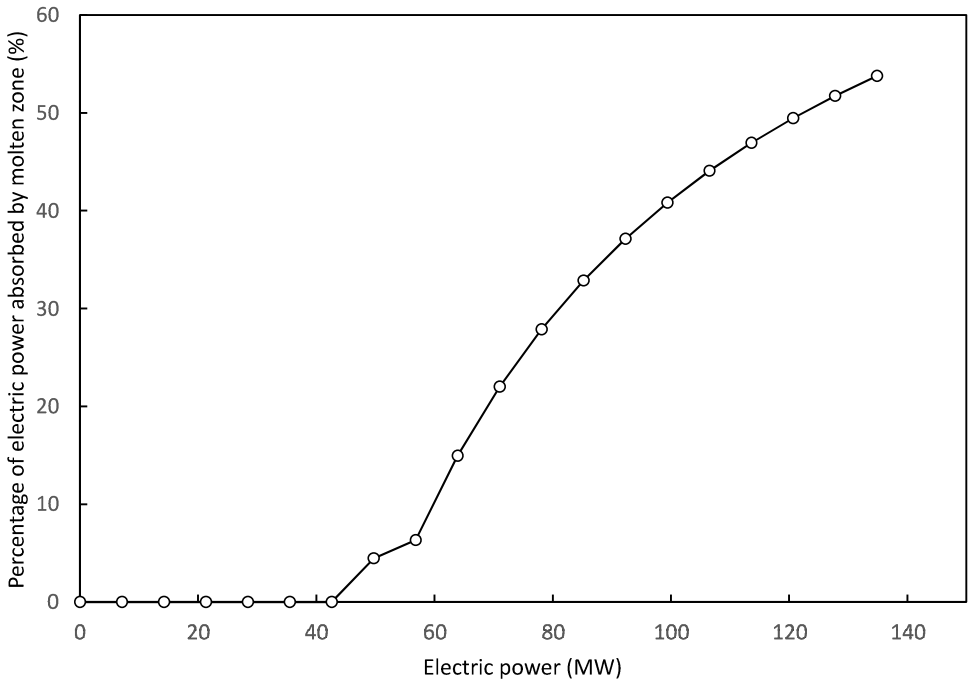

Figure 4.9: Percentage of electric power received by the molten zone at different electric power, with a coverage index of 0.9 and slag layer thickness of 15 mm.

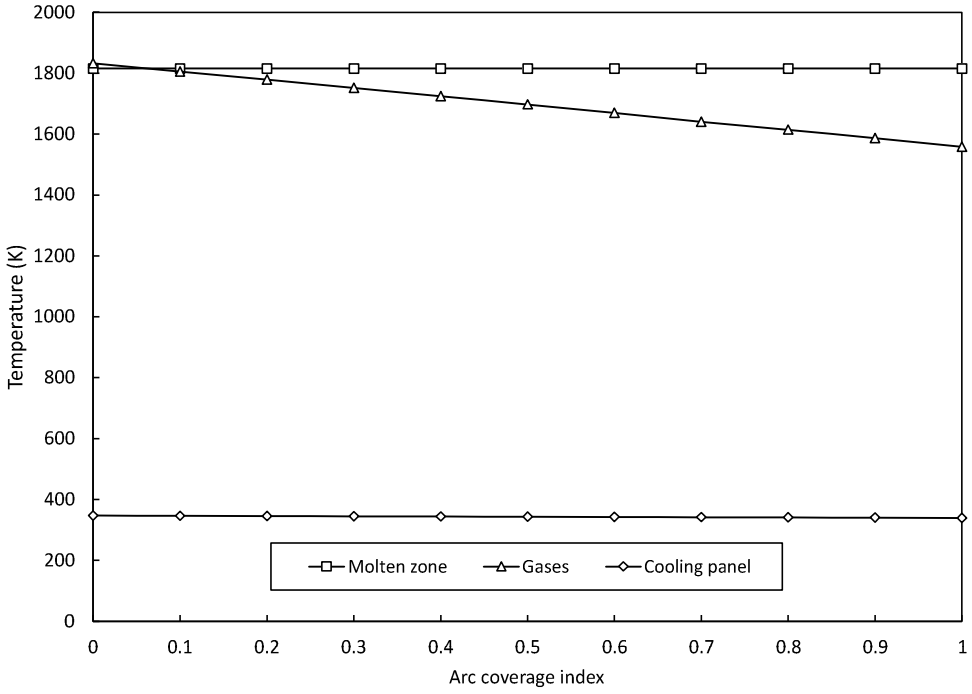

Figure 4.10: Temperatures of molten zone, gases and cooling panel at different arc coverage index, with an electric power of 71 MW and slag layer thickness of 15 mm.

An important aspect to consider for the slag layer in the cooling panels are their temperature

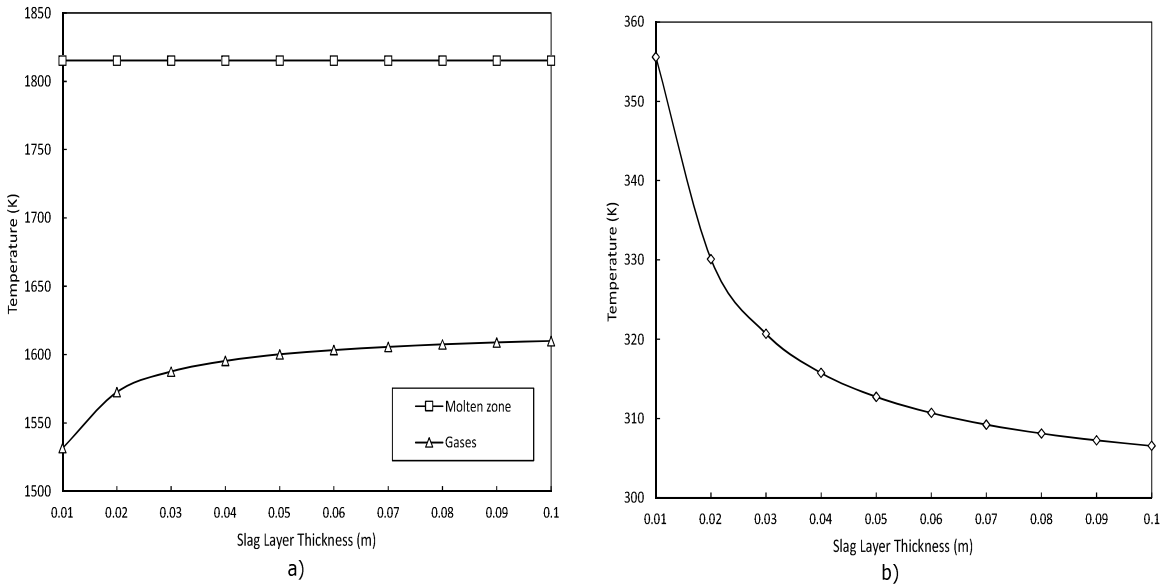

Figure 4.11: Temperatures of some surfaces of the EAF at different slag layer thickness, with an electric power of 71 MW and an arc coverage index of 1.0. a) Temperatures of molten zone and gases. b) Cooling panel temperature.

of the outer surface, because the solid slag start to melting around 1673.15 K [13]. If the outer surface temperature of the slag layer exceeds the slag melting point, it will be not possible to increase the thickness of the slag layer. The Figure 4.12 shows the impact of the arc coverage index and the slag layer thickness in the temperature of the outer surface of the slag layer. As can be seen, an completely covered arc favors the formation of the slag layer, causing it to increase its thickness to a maximum, where the slag starts to melt. With an arc coverage index of 0.8, can be formed a slag layer with a thickness of 5 cm.

The Figure 4.13 shows the temperature of all the wall cooling panels considering different slag layer thickness. As can be seen, the temperatures are similar in all the panels. This means that the position of the cooling panels inside the EAF Fuchs does not affect the heat flux received by each panel, and consequently does not exist a panel that is vulnerable to an increase in the temperature and continuous damage for this position in the furnace. Contreras [6] say that the global heat transfer coefficient of the panels depends highly to the water flow in the pipes of the tubular panel, and the water flow in each panel its affected by the losses in piping system. In this case, the global heat transfer coefficient of each panel (U_m) is 4.8893 $\mathrm{kW/m^2K}$. And the Figure 4.14 shows difference between the out and in temperatures of the cooling water with different slag thickness. It can be seen that some panels retires more heat than others. This is because these panels have a greater surface area and recives a greater heat flow.

Finally, the figure 4.15 shows the capacity of the model to calculate temperatures of the gases and the molten zone over time considering different slag layer thickness. It can be observed that the slag layer helps to increase the energy rate absorbed by the molten zone and the gases, which causes that the efficiency and the times of the process improves.

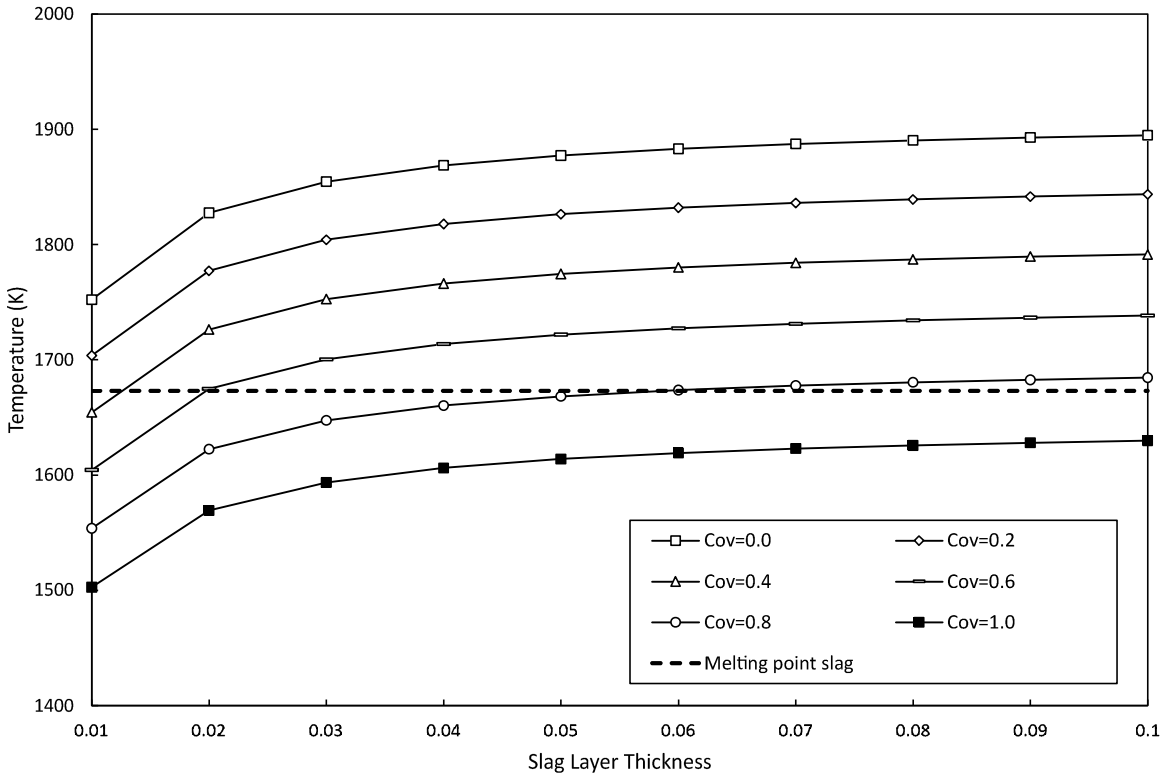

Figure 4.12: Temperatures of outer surface of the slag layer according to the slag layer thickness and the arc coverage.

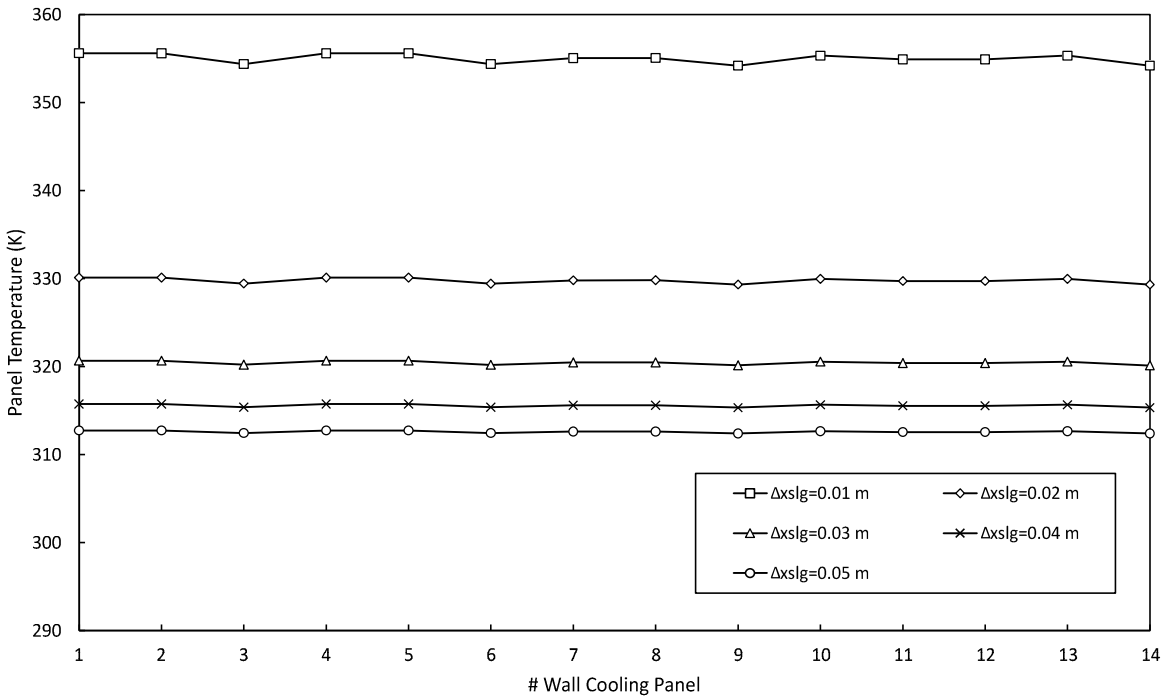

Figure 4.13: Temperatures of all wall cooling panels at different slag layer thickness.

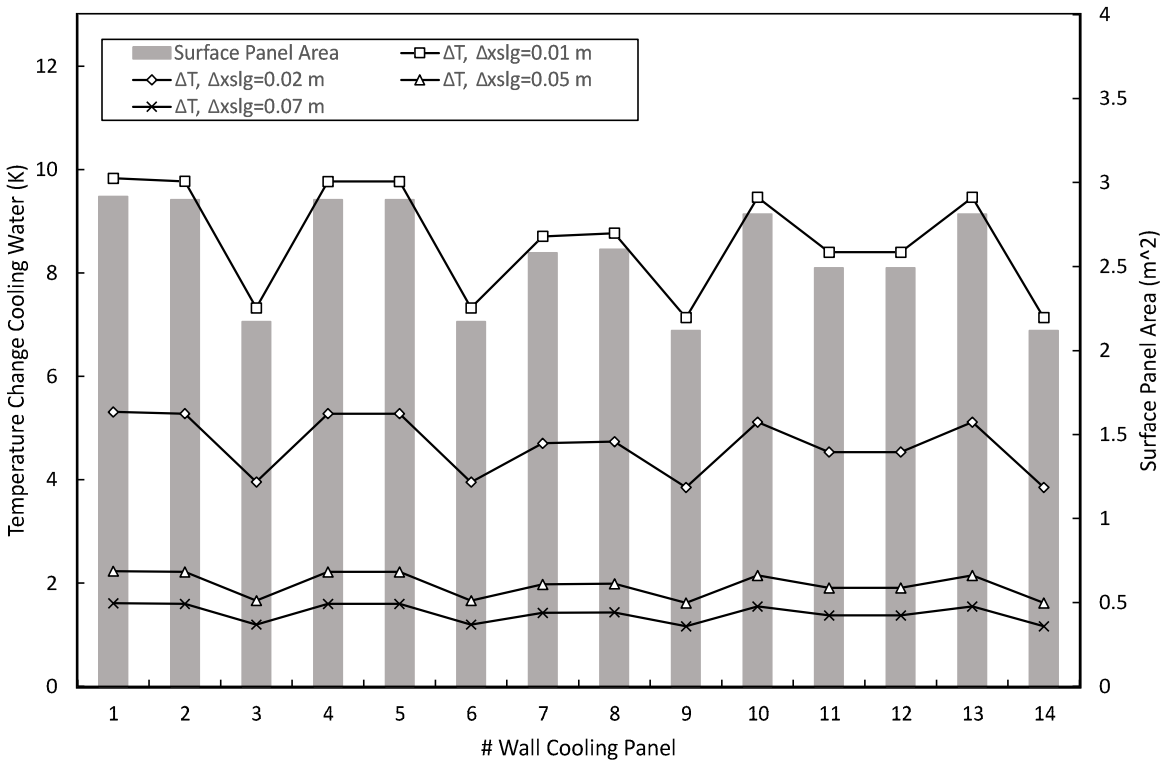

Figure 4.14: Surface area and temperature increase of the cooling water in each wall panel.

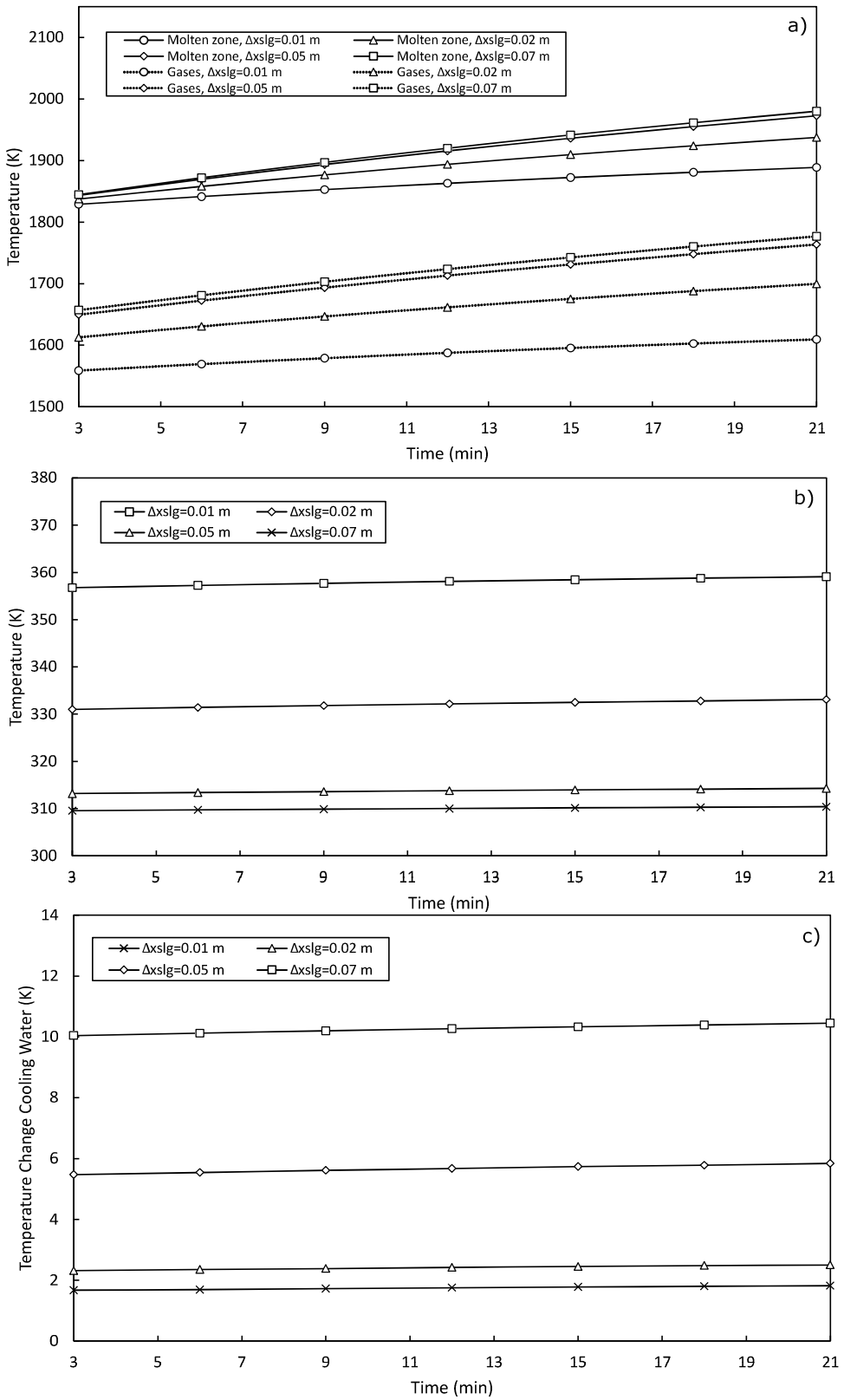

Figure 4.15: Temperatures of different EAF components during time, considering different slag layer thickness. a) Temperature of molten zone and gases. b) Temperature of wall cooling panel. c) Increase in cooling water temperature.

Chapter 5

Conclusions

In this work, two different models were developed in order to observe the interactions and consumption of energy during the steelmaking process in a EAF. The develop of this models are important because they can be used to optimize the energy consumption in the EAF, without compromising the quality of the product.

The objective of the mass and energy balance model is make a diagnosis of the energetic performance of the furnace, using data from the process. The mass flows that are not measured in the process are found using an optimization routine, which solve an equation system formed by a global mass balance and an element mass balance. The mass balance was validated using three parameters: the residuals of the mass balances by element, the amount of slag produced during the process and the amount of hot heel remaining in the furnace. The residuals of the mass balances by element represent a small amount of the total mass added to the process, and the elements that present the greatest residual are those species that are found in least amount during the process, as calcium, magnesium, aluminum and manganese. The amount of slag obtained with the model have good agreement with the amount of slag obtained from the process, maintaining an average difference between these two variables of 4.3 tons. The amount of hot heel obtained with the model is the variables that present greater differences with its simile of the process. This difference can be result of an erroneus lecture of the liquid steel produced, because the iron, principal component of the hot heel steel and the liquid steel, is one of the elements with less residual in the mass balance by element.

The energy balance was developed using the energy of the mass flows, the electric energy and the energy losses of the process. Also, the energy suministrated by the chemical reactions that occur in the furnace were calculated. The main energy sources of the EAF are the electric energy suministrated and the combustion of carbon. The principal energy losses in the furnace are the energy absorbed by the off gases and by the cooling system. In order to validate the energy balance, one variable was added to the equation of energy balance, denominated unknown losses. The unknown losses help to quantify the difference between energy input and output. It is to mention that the energy balance closes if the model only consider the important chemical reactions, as the combustion of carbon, natural gas and other fuels, and the oxydation of iron. Other reactions are not considered in this model because there are elements that appear spontaneously in the mass balance in the form of compounds,

causing an oversizing of the energy released by the chemical reactions during the process. As can be seen, these mass and energy balance models depends highly of the data obtained from the process, and they get better results if the data obtained from the process are more accurate.

The heat transfer model predicts temperature distributions and energy rates in different components of the EAF, from the incoming heat flows to the system and initial temperatures. The model was developed using the different heat transfer mechanisms that occur inside the oven. To calculate the heat transferred by radiation, view factors were obtained using a finite element software.

With the aim of observing its influence in the distribution of heat inside the furnace, thre variables were studied: the electric power, the arc coverage index and the slag layer thickness. The electric power mainly influences the energy rates obtained by the molten zone. The arc coverage index impacts in the energy rates obtained by the off gases, and na increase of of the slag layer thickness decreases the heat losses in the cooling system and also decreases the temperature of the cooling panels. Also, it was observed that the position of the panels inside the EAF Fuchs does not affect amount of energy rates received by each panel.

The robustness of the model was tested using extreme conditions for each of the variables above mentioned, as well as varying the duration time of the process. The model was able to solve the system of 99 equations and 99 unknowns proposde in less than five 5 minutes.

5.1 Future Work

The following activities are recomended in order to complement this research:

- Use the data validation and reconciliation method to solve the mass and energy balance
- Include the heat transfer model to a transient mass balance model of the process. This model allow a complete dynamic simulation of the EAF.
- Include gas radiation on the heat transfer model.
- Add to the model the ability to calculate by itself the slag layer thickness in the cooling panels.
- Consider the flow distribution of water in the cooling panels.
- Find the view factors of the furnace when the scrap is not completely melted.
- Analyze the heat transfer phenomena that occur in the electrode cooling system.
- Validate the heat transfer model using real data from the process.

Appendix A

Physical properties

Table A.1: Heats of formation at 298 K.

Compound	$\Delta\hat{H}_f^\circ$ (kJ/mol)	Compound	$\Delta\hat{H}_f^\circ$ (kJ/mol)
O_2	0	Fe_2O_3	-824.2
CH_4	-74.84	CaO	-635.6
CO	-110.5	MgO	-601.8
CO_2	-393.5	P	0
H_2O	-285.8	P_2O_5	-1506.24
H_2	0	S	0
N_2	0	SO_2	-296.9
C_9H_{20}	-266.05	Cr	0
Fe	0	Cr_2O_3	-1128.42
FeO	-270.37	Ti	0
Si	0	TiO_2	-519.9
SiO_2	-851	$CaCO_3$	-1206.87
Mn	0	$MgCO_3$	-1112.94
MnO	-384.9	Mo	0
C	0	Cu	0
Al	0	Zn	0
Al_2O_3	-1669.79		

Table A.2: Specific heat capacities and transformation enthalpy

$$cp = a + b \cdot T + c \cdot T^{-2} + d \cdot T^{2}$$

Units of cp are cal/(mol K)

Compound	a	$b \cdot 10^{3}$	$c \cdot 10^{5}$	$d \cdot 10^{-5}$	Temperature range (K)	Temperature transition (K)	$\Delta\hat{H}$ of transition (cal/mol)
O_2	7.16	1.0					
CH_4	5.65	1.44	-0.46				
CO	6.79	0.98	-0.11				
CO_2	10.57	2.1	-2.06				
$H_2O_{(l)}$	18.04				273.15 - 373.15	373.15	9718
$H_2O_{(g)}$	7.30	2.46					
H_2	6.25	0.78	0.12				
N_2	6.83	0.90	-0.12				
$C_9H_{20(l)}$	91.558	-0.2724		64.77	298.15 - 424	424	8919.603
$C_9H_{20(g)}$	8.77	158.9		-5.148			
$Fe_{(\alpha)}$	3.04	7.58	0.6		298.15 - 1033	1033	326
$Fe_{(\beta)}$	11.13				1033 - 1183	1183	215
$Fe_{(\gamma)}$	5.80	1.98			1183 - 1672	1672	165
$Fe_{(\delta)}$	6.74	1.6			1672 - 1812	1812	3670
$Fe_{(l)}$	9.77	0.4					
$FeO_{(s)}$	11.66	2.0	-0.67		298.15 - 1650	1650	7490
$FeO_{(l)}$	16.30						
$Si_{(s)}$	5.70	0.70	-1.04		298.15 - 1685	1685	12100
$Si_{(l)}$	6.1						
$SiO_{2(\alpha)}$	11.22	8.20	-2.70		298.15 - 848	848	290
$SiO_{2(\beta)}$	14.41	1.94					
$Mn_{(\alpha)}$	5.70	3.38	-0.37		298.15 - 1000	1000	535
$Mn_{(\beta)}$	8.33	0.66			1000 - 1374	1374	545
$Mn_{(\gamma)}$	10.70				1374 - 1410	1410	430

Continue next page

Continue Table A.2

Compound	a	$b \cdot 10^3$	$c \cdot 10^5$	$d \cdot 10^{-5}$	Temperature range (K)	Temperature transition (K)	$\Delta\hat{H}$ of transition (cal/mol)
$Mn_{(\delta)}$	11.30				1410 - 1517	1517	3500
$Mn_{(l)}$	11						
$MnO_{(s)}$	11.11	1.94	-0.88		273.15 - 1800	1800	55150
$MnO_{(g)}$	8.45	0.30	-0.87				
C	4.03	1.14	-2.04				
$Al_{(s)}$	4.94	2.96			298.15 - 932	932	10718
$Al_{(l)}$	7						
Al_2O_3	27.49	2.82	-8.38				
$Fe_2O_{3(\alpha)}$	23.49	18.6	-3.55		298.15 - 950	950	160
$Fe_2O_{3(\beta)}$	36				950 - 1050	1050	0
$Fe_2O_{3(\gamma)}$	31.71	1.76					
CaO	11.67	1.08	-1.56				
MgO	10.18	1.74	-1.48				
$P_{(s)}$	22.50				298.15 - 317.4	317.4	600
$P_{(l)}$	23.5				317.4 - 400	400	25600
$P_{(g)}$	4.97						
P_2O_5	40						
$S_{(\alpha)}$	3.58	6.24			298.15 - 368.6	368.6	85
$S_{(\beta)}$	6.20				368.6 - 392	392	335
$S_{(l)}$	8.73				392 - 717.8	717.8	2200
$S_{(g,\alpha)}$	5.26	-0.1	0.36		717.8 - 2400	2400	0
$S_{(g,\beta)}$	4.96	0.1	0.60				
SO_2	11.04	1.88	-1.84				
$Cr_{(\alpha)}$	4.16	3.62	0.3		298.15 - 2176	2176	5000
$Cr_{(l)}$	9.40						
Cr_2O_3	28.53	2.20	-3.74				
$Ti_{(\alpha)}$	5.25	2.52			298.15 - 1155	1155	950

Continue next page

Continue Table A.2

Compound	a	$b \cdot 10^3$	$c \cdot 10^5$	$d \cdot 10^{-5}$	Temperature range (K)	Temperature transition (K)	$\Delta\hat{H}$ of transition (cal/mol)
$Ti_{(\beta)}$	7.50				1155 - 1940	1940	4460
$Ti_{(l)}$	8						
TiO_2	17.97	0.28	-4.35				
$CaCO_3$	24.98	5.24	-6.20				
$MgCO_3$	18.62	13.80	-4.16				
$Mo_{(\alpha)}$	5.18	1.66			298.15 - 2890	2890	6550
$Mo_{(l)}$	10						
$Cu_{(\alpha)}$	5.41	1.5			298.15 - 1357	1357	3120
$Cu_{(l)}$	7.50						
$Zn_{(s)}$	5.35	2.40			298.15 - 692.7	692.7	1765
$Zn_{(l)}$	7.50				692.7 - 907	907	27430
$Zn_{(g)}$	4.97						

All the properties were obtained from [16], except the properties for C_9H_{20}, which were obtained from [11].

Appendix B

Process Data

Table B.1: Chemical compositions of input and output currents

Stream	Mass fraction
Input streams	
Oxygen	O_2=1
Natural gas	CH_4=1
Other fuels in the load	C_9H_{20}=1
Air	O_2=0.2217, CO_2=0.0152, H_2O=0.0062, N_2=0.7567
Water in the load	H_2O=1
Carbon added in the basket	C=0.9857, S=0.0066
Insufflated carbon	C=0.75
Anthracite	C=0.9857, S=0.0066
Dolomitic lime	H_2O=0.015, SiO_2=0.015, CaO=0.6, MgO=0.28, S=0.002
Hot heel steel at the beginning	Fe=0.9994, C=0.0006
Hot heel slag at the beginning	FeO=0.35, SiO_2=0.15, MnO=0.017, S=0.0003 Al_2O_3=0.05, CaO=0.21, MgO=0.102
Cold Direct Reduced Iron (DRI)	Fe=0.8189, FeO=0.0517, SiO_2=0.033, C=0.0397, Al_2O_3=0.002, CaO=0.009, MgO=0.001, P=0.0002
Hot DRI	Fe=0.831, FeO=0.0398, SiO_2=0.033, C=0.0381, Al_2O_3=0.002, CaO=0.009, MgO=0.001, P=0.0002
Scrap (17 types)	Fe=0.965, FeO=0.02, SiO_2=0.01
Output streams	
Liquid steel	Fe=0.9994, C=0.0006
Slag	FeO=0.35, SiO_2=0.15, MnO=0.017, S=0.0003 Al_2O_3=0.05, CaO=0.21, MgO=0.102
Off gases	O_2=3% Vol, CO=15% Vol, CO_2=9% Vol, H_2O=7% Vol, H_2=2% Vol
Dust	FeO=0.8, CaO=0.0289, SiO_2=0.0001, MgO=0.0307, Cr_2O_3=0.0013, Zn=0.15
Hot heel steel at the end	Fe=0.9994, C=0.0006
Hot heel slag at the end	FeO=0.35, SiO_2=0.15, MnO=0.017, S=0.0003 Al_2O_3=0.05, CaO=0.21, MgO=0.102

Table B.2: Dimensions of the tubular wall panels.

Wall panel	1	2	3	4	5	6	7	8	9	10	11	12	13	14
L_{pipe} (m)	43.21	42.93	32.17	42.93	42.93	32.17	38.26	38.55	31.39	41.66	36.91	36.91	41.66	31.39
A_m (m^2)	2.91	2.89	2.17	2.89	2.89	2.17	2.58	2.60	2.11	2.81	2.49	2.49	2.81	2.11
Internal diameter (d_{pipe}) = 0.0677 m								Pipes roughness (e) = 0.045×10^{-3} m						

Table B.3: View factors from all surfaces to wall cooling panels

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14
P1		0.0036	0.0071	0.0130	0.0149	0.0123	0.0156	0.0166	0.0129	0.0140	0.0157	0.0211	0.0272	0.0042
P2	0.0036		0.0107	0.0180	0.0186	0.0141	0.0169	0.0157	0.0112	0.0134	0.0134	0.0147	0.0146	0.0036
P3	0.0093	0.0141		0.0155	0.0180	0.0140	0.0168	0.0146	0.0104	0.0140	0.0138	0.0146	0.0154	0.0062
P4	0.0130	0.0180	0.0118		0.0150	0.0135	0.0166	0.0132	0.0093	0.0146	0.0141	0.0143	0.0151	0.0080
P5	0.0146	0.0182	0.0135	0.0146		0.0116	0.0160	0.0144	0.0081	0.0151	0.0143	0.0141	0.0146	0.0093
P6	0.0167	0.0191	0.0145	0.0183	0.0160		0.0126	0.0083	0.0062	0.0154	0.0146	0.0138	0.0140	0.0104
P7	0.0174	0.0188	0.0143	0.0185	0.0182	0.0104		0.0033	0.0038	0.0146	0.0146	0.0134	0.0134	0.0122
P8	0.0205	0.0192	0.0138	0.0162	0.0144	0.0075	0.0037		0.0044	0.0278	0.0214	0.0159	0.0142	0.0130
P9	0.0198	0.0172	0.0121	0.0142	0.0126	0.0070	0.0052	0.0055		0.0521	0.0233	0.0145	0.0125	0.0132
P10	0.0160	0.0152	0.0121	0.0166	0.0175	0.0129	0.0149	0.0257	0.0388		0.0052	0.0051	0.0060	0.0094
P11	0.0202	0.0172	0.0135	0.0181	0.0189	0.0139	0.0168	0.0225	0.0196	0.0059		0.0042	0.0055	0.0119
P12	0.0272	0.0189	0.0143	0.0185	0.0186	0.0131	0.0155	0.0166	0.0122	0.0508	0.0042		0.0053	0.0188
P13	0.0310	0.0166	0.0134	0.0171	0.0170	0.0117	0.0136	0.0131	0.0093	0.0060	0.0048	0.0046		0.0393
P14	0.0060	0.0052	0.0068	0.0116	0.0137	0.0111	0.0144	0.0153	0.0125	0.0120	0.0134	0.0211	0.0500	
R1	0.1168	0.1501	0.0221	0.0107	0.0063	0.0028	0.0032	0.0028	0.0020	0.0028	0.0040	0.0068	0.0122	0.0136
R2	0.0145	0.0999	0.1502	0.0464	0.0135	0.0040	0.0038	0.0024	0.0015	0.0021	0.0024	0.0031	0.0043	0.0035
R3	0.0057	0.0149	0.0344	0.2184	0.0599	0.0088	0.0064	0.0029	0.0016	0.0022	0.0022	0.0023	0.0027	0.0020
R4	0.0039	0.0073	0.0091	0.0546	0.2114	0.0362	0.0136	0.0031	0.0019	0.0027	0.0023	0.0022	0.0022	0.0017
R5	0.0032	0.0045	0.0042	0.0127	0.0450	0.1455	0.0998	0.0111	0.0033	0.0044	0.0032	0.0024	0.0021	0.0016
R6	0.0034	0.0037	0.0028	0.0058	0.0104	0.0192	0.1278	0.0968	0.0124	0.0118	0.0066	0.0040	0.0027	0.0021
OG	0.0317	0.0071	0.0030	0.0034	0.0035	0.0028	0.0056	0.0262	0.0319	0.0438	0.0405	0.0405	0.0439	0.0337
El	0.0601	0.0712	0.0559	0.0757	0.0775	0.0540	0.0638	0.0491	0.0286	0.0380	0.0391	0.0392	0.0381	0.0301
SD	0.0139	0.0076	0.0136	0.0175	0.0173	0.0108	0.0073	0.0001	0.0014	0.0123	0.0134	0.0119	0.0115	0.0108
RL1	0.0293	0.0297	0.0207	0.0299	0.0308	0.0197	0.0246	0.0211	0.0193	0.0259	0.0218	0.0218	0.0258	0.0204
RL2	0.0151	0.0141	0.0106	0.0140	0.0144	0.0102	0.0125	0.0123	0.0089	0.0120	0.0117	0.0177	0.0121	0.0094
LS	0.0227	0.0223	0.0164	0.0216	0.0222	0.0158	0.0197	0.0183	0.0131	0.0176	0.0177	0.0177	0.0177	0.0138
Arc	0.0152	0.0173	0.0126	0.0180	0.0181	0.0127	0.0150	0.0130	0.0085	0.0114	0.0113	0.0114	0.0114	0.0085

Table B.4: View factors from all surfaces to the rest of the surfaces

	R1	R2	R3	R4	R5	R6	OG	El	SD	RL1	RL2	LS	Arc
P1	0.1260	0.0195	0.0052	0.0036	0.0043	0.0039	0.2190	0.0336	0.0031	0.1757	0.0691	0.2341	0.0008
P2	0.1626	0.1346	0.0138	0.0067	0.0060	0.0042	0.0493	0.0398	0.0017	0.1791	0.0649	0.2315	0.0009
P3	0.0313	0.2648	0.0416	0.0109	0.0073	0.0042	0.0270	0.0408	0.0040	0.1630	0.0638	0.2218	0.0008
P4	0.0116	0.0625	0.2020	0.0504	0.0172	0.0066	0.0233	0.0423	0.0040	0.1800	0.0647	0.2241	0.0009
P5	0.0067	0.0177	0.0541	0.1997	0.0593	0.0115	0.0235	0.0424	0.0038	0.1815	0.0647	0.2246	0.0009
P6	0.0041	0.0073	0.0111	0.0452	0.2653	0.0297	0.0262	0.0408	0.0033	0.1606	0.0638	0.2209	0.0009
P7	0.0039	0.0057	0.0066	0.0140	0.1499	0.1623	0.0429	0.0398	0.0018	0.1655	0.0644	0.2274	0.0008
P8	0.0037	0.0040	0.0033	0.0036	0.0183	0.1354	0.2227	0.0337	0.0000	0.1564	0.0694	0.2327	0.0008
P9	0.0034	0.0031	0.0022	0.0027	0.0069	0.0215	0.3378	0.0245	0.0005	0.1777	0.0628	0.2075	0.0007
P10	0.0034	0.0031	0.0023	0.0028	0.0067	0.0152	0.3446	0.0243	0.0032	0.1770	0.0626	0.2077	0.0007
P11	0.0056	0.0042	0.0026	0.0028	0.0055	0.0097	0.3613	0.0282	0.0039	0.1689	0.0692	0.2358	0.0007
P12	0.0095	0.0054	0.0028	0.0026	0.0042	0.0058	0.3616	0.0283	0.0035	0.1691	0.0691	0.2363	0.0007
P13	0.0150	0.0066	0.0028	0.0023	0.0032	0.0035	0.3452	0.0243	0.0029	0.1768	0.0630	0.2084	0.0007
P14	0.0213	0.0069	0.0027	0.0022	0.0031	0.0034	0.3377	0.0245	0.0035	0.1777	0.0626	0.2071	0.0006
R1		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0275	0.0004	0.1658	0.1140	0.3717	0.0007
R2	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0283	0.0011	0.1692	0.1069	0.3389	0.0006
R3	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0299	0.0018	0.1706	0.1074	0.3314	0.0007
R4	0.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0299	0.0029	0.1707	0.1074	0.3319	0.0007
R5	0.0000	0.0000	0.0000	0.0000		0.0000	0.0000	0.0285	0.0089	0.1694	0.1069	0.3382	0.0006
R6	0.0000	0.0000	0.0000	0.0000	0.0000		0.0000	0.0278	0.0245	0.1663	0.1141	0.3720	0.0006
OG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		0.0192	0.0033	0.1667	0.1133	0.3596	0.0005
El	0.0530	0.0680	0.0493	0.0493	0.0683	0.0562	0.2349		0.0160	0.2225	0.2873	0.8154	0.0136
SD	0.0018	0.0066	0.0072	0.0120	0.0529	0.1238	0.1011	0.0400		0.2358	0.0726	0.2572	0.0010
RL1	0.0298	0.0378	0.0262	0.0262	0.0379	0.0314	0.1918	0.0208	0.0088		0.0460	0.1361	0.0006
RL2	0.0268	0.0313	0.0216	0.0216	0.0313	0.0282	0.1706	0.0363	0.0036	0.0602		0.2643	0.0013
LS	0.0389	0.0441	0.0296	0.0296	0.0440	0.0409	0.2405	0.0489	0.0056	0.0792	0.1174		0.0022
Arc	0.0145	0.0171	0.0119	0.0121	0.0170	0.0144	0.0668	0.1917	0.0043	0.0737	0.1218	0.4479	

Appendix C

Programs

C.1 Mass and energy balances program

Code	Function
`% Program`	
`Purpose:  Obtain the mass and energy balance`	
`of a EAF`	
`----------------------------------------`	
`Dat = get(handles.tab_dat1,'Data');`	Read the input variables
`NDat=cell2mat(Dat(:,2));`	Assign the numerical data to a variable
`----------------------------------------`	
`DT_in=NDat(51:87)`	Assign the temperatures of the input streams to a variable
`DT_out=NDat(89:94)`	Assign the temperatures of the output streams to a variable
`----------------------------------------`	
`[comp,elem, PM, M_in,M_out,c_in,c_out,y,` `fval,bal,TON,inlet,outlet]=` `mass_bal_4_p(NDat,NDat_p);`	Call the mass balance function and obtains the number of chemical species, the number of elements, the molecular weight vector, the mass flows of the streams per chemical species, the concentrations of the streams and the residuals of the balances.
`[m,n]=size(M_in);`	
`T_in=ones(m,n);` `for i=1:numel(DT_in)` `T_in(i,:)=DT_in(i)+273.15;`	Assign the temperatures of the input streams in a matrix and convert them in K.
`end`	
`P_in=ones(m,n)*101350;`	Pressure of the inlet streams in Pa.
`Q_in=0;`	

Continue next page

Continue Code

Code	Function
`EE=NDat(139);`	Assign the input electric energy to a variable in kWh per ton of liquid steel.
`W_in= EE;`	
`[m,n]=size(M_out);`	
`T_out=ones(m,n);`	
`for i=1:numel(DT_out))`	
`T_out(i,:)=DT_out(i)+273.15;`	Assign the temperatures of the output streams in a matrix and convert them in K.
`end`	
`P_out=ones(m,n)*101350;`	Pressure of the inlet streams in Pa.
`POT=NDat(140);`	Assign the power on time in minutes to a variable.
`TTT=NDat(141);`	Assign the tap to tap time in minutes to a variable.
`CHT=NDat(142);`	Assign the charge time in minutes to a variable.
`TPT=NDat(143);`	Assign the tapping time in minutes to a variable.
`INT=NDat(144);`	Assign the inspection time in minutes to a variable.
`%Flujos de agua enfriamiento por paneles`	
`WF_w=NDat(46);`	Assign the flow of cooling water in the wall panels in m^3/s to a variable.
`DT_w=NDat(95)`	Assign the increase of temperature of the cooling water in the wall panels in K to a variable.
`WF_r=NDat(47);`	Assign the flow of cooling water in the roof panels in m3/s to a variable.

Continue next page

Continue Code

Code	Function
`DT_r=NDat(96);`	Assign the increase of temperature of the cooling water in the roof panels in K to a variable.
`WF_c=NDat(48);`	Assign the flow of cooling water in the elbow in m3/s to a variable.
`DT_c=NDat(97);`	Assign the increase of temperature of the cooling water in the elbow in K to a variable.
`%calor removido por el sistema de enfriamiento`	
`Q_cool=4.184*(WF_w*DT_w+WF_r*DT_r+WF_c*DT_c)* (1000/3600)*(TTT/60)/TON;`	Calculates the energy withdrawn by the cooling system in kWh per ton of liquid steel.
`%perdidas electricas`	
`Q_pe=EE*0.06;`	Calculates the electric losses of the furnace in kWh per ton of liquid steel.
`%calor por conduccion en el fondo del horno`	
`A_fond=54.37;`	Area of the furnace bottom in m^2.
`P_fond=27.542`	Perimeter of the furnace bottom in m.
`h_fond=9.6;`	Elevation of the oven on the ground in m.
`T_steel=T_out(1,1);`	Temperature of the liquid steel in K.
`Talr=323.15;`	Temperature of the surroundings of the furnace in K.
`dx=1.2;`	Thickness of the refractory layer of the bottom in m.
`kref=0.1;`	Thermal conductivity of the refractory brick in W/mK.
`Q_cd=FondoHorno(TTT,A_fond,P_fond,h_fond, T_steel,Talr,dx,kref,TON);`	Call the heat loss in the bottom of the furnace function. The function obtains the heat loss in kWh per ton of liquid steel.
`%calor transferido al destapar el horno`	

Continue next page

Continue Code

Code	Function
`T_slag=DT_in(15)+273.15;`	Temperature of the liquid slag in K.
`T_surr=323.15;`	Temperature of the surroundings of the furnace in K.
`emiss=1;`	Emissivity of the slag.
`sigma=5.67e-8;`	Stefan-Boltzmann constant in W/m^2K^4
`A_lo=54.37;`	Surface area of the furnace in m^2.
`t_abierto=CHT+TPT+INT;`	Duration of the furnace opening in min.
`F_vis=0.58;`	View factor of the slag to the surroundings during the furnace opening.
`Q_lo=0.58*(sigma*emiss*A_lo*(T_slag^4-T_surr^4)*(t_abierto*(60))/3.6e6)/TON;`	Calculation of the heat loss during the furnace opening in kWh per ton of liquid steel
`%calor que sale por la puerta de escoria`	
`A_sd=0.9*0.7;`	Area of the slag door in m^2
`Q_sd=(sigma*emiss*A_sd*(T_slag^4-T_surr^4)*(TTT*60)/3.6e6)/TON;`	Calculation of the heat loss through the slag door in kWh per ton of liquid steel
`%calor en el agua que se evapora en el enfriamiento del electrodo`	
`WF_e=NDat(49);`	Amount of cooling water for the electrode in ton/h.
`T_e=NDat(98);`	Inlet temperature of the cooling water for the electrode in K.
`h_ein=enthalpy(T_e);`	Calculation of the enthalpy at the inlet temperature of the cooling water for the electrode.
`h_eout=enthalpy(374);`	Calculation of the enthalpy at the boiling water temperature.
`h_ein=h_ein(5);`	Assign the inlet enthalpy of the cooling water for the electrode to a variable.
`h_eout=h_eout(5);`	Assign the outlet enthalpy of the cooling water for the electrode to a variable.

Continue next page

Continue Code

Code	Function
`Q_ee=(h_eout-h_ein)*(WF_e*1000/PM(5))*(POT/60)/(3600*TON);`	Calculation of the heat absorbed by the cooling system of the electrode in kWh per ton of liquid steel.
`Q_out=Q_cool+Q_pe+Q_cd+Q_lo+Q_sd+Q_ee;`	Calculation of the total heat losses in the furnace during the process.
`W_out=0;`	
`[ED_in_s,ED_in_c,ED_out_s,En_rxn_imp,OL,En_ox,En_C_flux,En_rxn_oth]=en_bal_2(comp,PM,M_in,M_out,T_in,T_out,P_in,P_out,Q_in,Q_out,W_in,W_out);`	Call the energy balance function. The input variables are the chemical species considered, the molecular weight, the mass and temperatures of the inlet and outlet streams, and the heat and work suministred and lost during the process. The function obtains the sensible heat of each stream, the energy suministred by the reactions and the unknown energy losses.
`E_in=sum(ED_in_s)+En_rxn_imp+W_in;`	Total input energy suministred to the furnace.
`E_out=sum(ED_out_s)+Q_out+OL;`	Total energy that comes out of the furnace.
`E_CH4=ED_in_c(1);`	Energy supplied by the combustion of natural gas.
`E_OC=ED_in_c(2);`	Energy supplied by the other fuels in the load.
`E_C=sum(En_C_flux);`	Energy supplied by the combustion of the carbon.0
`E_HH_st_in=ED_in_s(14);`	Sensible heat of the hot heel steel at the beggining.
`E_HH_sl_in=ED_in_s(15);`	Sensible heat of the hot heel slag at the end.
`E_oxy=sum(En_ox);`	Energy supplied by the oxidation of iron.
`E_oth=sum(En_rxn_oth)`	Energy supplied by the other reactions that occur during the process.

Continue next page

Continue Code

Code	Function
`E_sens=sum(ED_in_s)-E_HH_st_in-E_HH_sl_in;`	Sensible heat of the input streams except the hot heel steel and hot heel slag.
`ed_in_1={W_in E_C E_CH4 E_OC E_oxy E_oth E_HH_st_in+E_HH_sl_in E_sens 'Electricity' 'Carbon' 'Natural Gas' 'Other Fuels' 'Metal Oxidation' 'Other Reactions' 'Hot heel' 'Hot Load'};`	Energy sources of the furnace.
`E_st=ED_out_s(1);`	Sensible energy of the liquid steel.
`E_sl=ED_out_s(2);`	Sensible energy of the slag.
`E_g=ED_out_s(3);`	Sensible energy of the off gases.
`E_pw=ED_out_s(4);`	Sensible energy of the dust.
`E_HH_st_out=ED_out_s(5);`	Sensible energy of the hot heel steel at the end.
`E_HH_sl_out=ED_out_s(6);`	Sensible energy of the hot heel slag at the end.
`E_oth=OL;`	Unknown heat losses in the process.
`ed_out_1={E_st E_sl Q_cool E_g E_pw Q_pe E_HH_st_out+E_HH_sl_out E_oth Q_cd Q_lo Q_sd Q_ee 'Steel' 'Slag' 'Cooling' 'Off gases' 'Dust' 'Electrical losses' 'Hot heel' 'Unknown losses' 'EAF bottom' 'EAF load' 'Slag door' 'Electrode cooling'};`	Energy sinks and energy losses of the EAF process.
`plot_2(ed_in_1,ed_out_1,[nam_file num2str(num_file)]);`	Plot of the energy balance of the EAF process.

C.2 Mass balance function

Code	Function
`% Mass balance function`	
`function [comp,elem,PM_c,Mc_in,Mc_out,c_in, c_out,y,fval,bal,TON,inlet,outlet] =mass_bal_4_p(NDat,NDat_p)`	
`comp={'O2' 'CH4' 'CO' 'CO2' 'H2O' 'H2' 'N2' 'C9H20' 'Fe' 'FeO' 'Si' 'SiO2' 'Mn' 'MnO' 'C' 'Al' 'Al2O3' 'Fe2O3' 'CaO' 'MgO' 'P' 'P2O5' 'S' 'SO2' 'Cr' 'Cr2O3' 'Ti' 'TiO2' 'CaCO3' 'MgCO3' 'Mo' 'Cu' 'Zn'};`	Chemical species considered in the model.
`ncc=numel(comp);`	Number of chemical species.
`elem={'O' 'C' 'H' 'N' 'Fe' 'Si' 'Mn' 'Al' 'Ca' 'Mg' 'P' 'S' 'Cr' 'Ti' 'Mo' 'Cu' 'Zn'};`	Elements considered in the model.
`nec=numel(elem);`	Number of elements.
`inlet={'Oxy' 'GN' 'OC' 'Ai' 'WL' 'CC' 'Ci' 'Ant' 'ESP' 'CD' 'CS' 'CAC' 'CAM' 'PST_in' 'PSL_in' 'DRI_F' 'DRI_T' 'DRI_C' 'AR' 'HBI' 'S1' 'S2' 'S3' 'S4' 'S5' 'S6' 'S7' 'S8' 'S9' 'S10' 'S11' 'S12' 'S13' 'S14' 'S15' 'S16' 'S17'};`	Inlet currents of the furnace.
`outlet='ST' 'SL' 'OG' 'PW' 'PST_out' 'PSL_out';`	Outlet currents of the furnace.

Continue next page

Continue Code

Code	Function
`% Molecular weight chemical species`	Molecular weight in g/mol.
`PM_O2=32;`	
`PM_CH4=16;`	
`PM_CO=28`	
`PM_CO2=44;`	
`PM_H2O=18;`	
`PM_H2=2;`	
`PM_N2=28;`	
`PM_C9H20=128;`	
`PM_Fe=55.85;`	
`PM_FeO=71.85;`	
`PM_Si=28.08;`	
`PM_SiO2=60.08;`	
`PM_Mn=54.94;`	
`PM_MnO=70.94;`	
`PM_C=12;`	
`PM_Al=26.98;`	
`PM_Al2O3=101.96;`	
`PM_Fe2O3=159.69;`	
`PM_CaO=56.08;`	
`PM_MgO=40.3;`	
`PM_P=30.97;`	
`PM_P2O5=141.94;`	
`PM_S=32;`	
`PM_SO2=64;`	
`PM_Cr=52;`	
`PM_Cr2O3=152;`	
`PM_Ti=47.9;`	

Continue next page

Continue Code

Code	Function
`PM_TiO2=79.9;`	
`PM_CaCO3=100;`	
`PM_MgCO3=84.31;`	
`PM_Mo=95.95;`	
`PM_Cu=63.5;`	
`PM_Zn=65.4;`	
`for i=1:ncc`	Assignement of the molecular weight to a vector.
`pm=['PM_' cell2mat(comp(i))];`	
`PM_c(i)=eval(pm);`	
`end`	
`% Molecular weight elements`	Molecular weight in g/mol.
`PM_O=16;`	
`PM_C=12;`	
`PM_H=1;`	
`PM_N=14;`	
`PM_Fe=55.8;`	
`PM_Si=28.08;`	
`PM_Mn=54.94;`	
`PM_Al=26.98;`	
`PM_Ca=40.08;`	
`PM_Mg=24.3;`	
`PM_P=30.97;`	
`PM_S=32;`	
`PM_Cr=52;`	
`PM_Ti=47.9;`	
`PM_Mo=95.95;`	

Continue next page

Continue Code

Code	Function
`PM_Cu=63.5;` `PM_Zn=65.4;`	

`for i=1:nec`
`  pm=['PM_' cell2mat(elem(i))];`
`  PM_e(i)=eval(pm);`
`end`
	Assignement of the molecular weight of the elements to a vector.

`----------------------------------------`	

`% Composition data`
`cdat='comp_hist';`
`c_in=csvread([cdat '.csv']);`
	Assignement of the inlet streams compositions to a matrix.

`%Composition Liquid steel`
`C_stl_out=NDat(101:111);`
`ppm_FeO_ST=C_stl_out(1);`
`ppm_CaO_ST=C_stl_out(2);`
`ppm_MnO_ST=C_stl_out(3);`
`ppm_MgO_ST=C_stl_out(4);`
`ppm_SiO2_ST=C_stl_out(5);`
`ppm_Al2O3_ST=C_stl_out(6);`
`ppm_C_ST=C_stl_out(7);`
`ppm_S_ST=C_stl_out(8);`
`ppm_Cr_ST=C_stl_out(9);`
`ppm_Si_ST=C_stl_out(10);`
`ppm_Al_ST=C_stl_out(11);`
	Read the compositions of the liquid steel in parts per million.

`f_FeO_ST=ppm_FeO_ST*10^-6;`	Convert the composition of liquid steel in mass fraction.

Continue next page

Continue Code

Code	Function
`f_CaO_ST=ppm_CaO_ST*10^-6;`	
`f_MnO_ST=ppm_MnO_ST*10^-6;`	
`f_MgO_ST=ppm_MgO_ST*10^-6;`	
`f_SiO2_ST=ppm_SiO2_ST*10^-6;`	
`f_Al2O3_ST=ppm_Al2O3_ST*10^-6;`	
`f_C_ST=ppm_C_ST*10^-6;`	
`f_S_ST=ppm_S_ST*10^-6;`	
`f_Cr_ST=ppm_Cr_ST*10^-6;`	
`f_Si_ST=ppm_Si_ST*10^-6;`	
`f_Al_ST=ppm_Al_ST*10^-6;`	
`f_Fe_ST=1-(f_FeO_ST+f_S_ST+f_SiO2_ST+f_CaO_ST +f_MgO_ST+f_Al2O3_ST+f_C_ST+f_Cr_ST+f_Si_ST +f_Al_ST);`	Determination of the mass fraction of iron in liquid steel.
`c_ST=zeros(1,ncc);`	
`c_ST(9)=f_Fe_ST;`	
`c_ST(10)=f_FeO_ST;`	
`c_ST(11)=f_Si_ST;`	
`c_ST(12)=f_SiO2_ST;`	
`c_ST(14)=f_MnO_ST;`	
`c_ST(15)=f_C_ST;`	
`c_ST(16)=f_Al_ST;`	
`c_ST(17)=f_Al2O3_ST;`	
`c_ST(19)=f_CaO_ST;`	
`c_ST(20)=f_MgO_ST;`	
`c_ST(23)=f_S_ST;`	
`c_ST(25)=f_Cr_ST;`	

Continue next page

Continue Code

Code	Function

`% Composition slag`
`C_slag_out=NDat(113:122);`
`f_FeO_SL=C_slag_out(1);`
`f_CaO_SL=C_slag_out(2);`
`f_MgO_SL=C_slag_out(3);`
`f_SiO2_SL=C_slag_out(4);`
`f_Al2O3_SL=C_slag_out(5);`
`f_MnO_SL=C_slag_out(6);`
`f_S_SL=C_slag_out(7);`
`f_Cr_SL=C_slag_out(8);`
`f_Si_SL=C_slag_out(9);`
`f_Al_SL=C_slag_out(10);`
 	
`c_SL=zeros(1,ncc);`
`c_SL(10)=f_FeO_SL;`
`c_SL(11)=f_Si_SL;`
`c_SL(12)=f_SiO2_SL;`
`c_SL(14)=f_MnO_SL;`
`c_SL(16)=f_Al_SL;`
`c_SL(17)=f_Al2O3_SL;`
`c_SL(19)=f_CaO_SL;`
`c_SL(20)=f_MgO_SL;`
`c_SL(23)=f_S_SL;`
`c_SL(25)=f_Cr_SL;`
	Read the compositions of the slag in mass fraction.

`%Composition off gases` `C_gases=NDat(124:128);`	Read the compositions of the off gases in volume percentage.

Continue next page

Continue Code

Code	Function
`pv_O2_OG=C_gases(1);`	
`pv_CO_OG=C_gases(2);`	
`pv_CO2_OG=C_gases(3);`	
`pv_H2O_OG=C_gases(4);`	
`pv_H2_OG=C_gases(5);`	
`pv_N2_OG=100-(pv_O2_OG+pv_CO_OG+pv_CO2_OG +pv_H2O_OG+pv_H2_OG);`	Determination of the volume percentage of nitrogen in off gases.
`m_O2_OG=pv_O2_OG*PM_O2;`	Conversion of the volume percentage in mass fraction.
`m_CO_OG=pv_CO_OG*PM_CO;`	
`m_CO2_OG=pv_CO2_OG*PM_CO2;`	
`m_H2O_OG=pv_H2O_OG*PM_H2O;`	
`m_H2_OG=pv_H2_OG*PM_H2;`	
`m_N2_OG=pv_N2_OG*PM_N2;`	
`m_t_OG=m_O2_OG+m_CO_OG+m_CO2_OG +m_H2O_OG+m_H2_OG+m_N2_OG;`	
`f_O2_OG=m_O2_OG/m_t_OG;`	
`f_CO_OG=m_CO_OG/m_t_OG;`	
`f_CO2_OG=m_CO2_OG/m_t_OG;`	
`f_H2O_OG=m_H2O_OG/m_t_OG;`	
`f_H2_OG=m_H2_OG/m_t_OG;`	
`f_N2_OG=m_N2_OG/m_t_OG;`	
`c_OG=zeros(1,ncc)`	
`c_OG(1)=f_O2_OG;`	
`c_OG(3)=f_CO_OG;`	
`c_OG(4)=f_CO2_OG;`	

Continue next page

Continue Code

Code	Function
`c_OG(5)=f_H2O_OG;`	
`c_OG(6)=f_H2_OG;`	
`c_OG(7)=f_N2_OG;`	
`% Composition dust`	Read the compositions of dust in mass fraction.
`C_pow=NDat(130:137);`	
`f_FeO_PW=C_pow(1);`	
`f_CaO_PW=C_pow(2);`	
`f_SiO2_PW=C_pow(3);`	
`f_MgO_PW=C_pow(4);`	
`f_TiO2_PW=C_pow(5);`	
`f_Cr2O3_PW=C_pow(6);`	
`f_Cu_PW=C_pow(7);`	
`f_Zn_PW=C_pow(8);`	
`c_PW=zeros(1,ncc);`	
`c_PW(10)=f_FeO_PW;`	
`c_PW(19)=f_CaO_PW;`	
`c_PW(12)=f_SiO2_PW;`	
`c_PW(20)=f_MgO_PW;`	
`c_PW(28)=f_TiO2_PW;`	
`c_PW(26)=f_Cr2O3_PW;`	
`c_PW(32)=f_Cu_PW;`	
`c_PW(33)=f_Zn_PW;`	
`%Composition hot heel steel`	Read the compositions of hot heel steel at the end.
`c_PST_out=c_ST`	

Continue next page

Continue Code

Code	Function
`%Composition hot heel slag` `c_PSL_out=c_SL;`	Read the compositions of hot heel slag at the end.

`%Matrix outlet streams compositions`
`for i=1:numel(outlet)`
`  c=['c_' cell2mat(outlet(i))];`
`  c_out(i,:)=eval(c);`
`----------`
	Assign the composition of the outlet streams to a matrix.

`Atom distribution in chemical species`
 	
`d_O2=zeros(1,nec);`
`d_O2(1)=2;`
 	
`d_CH4=zeros(1,nec);`
`d_CH4(2)=1;`
`d_CH4(3)=4;`
 	
`d_CO=zeros(1,nec);`
`d_CO(2)=1;`
`d_CO(1)=1;`
 	
`d_CO2=zeros(1,nec);`
`d_CO2(2)=1;`
`d_CO2(1)=2;`
 	
`d_H2O=zeros(1,nec);`
`d_H2O(3)=2;`
`d_H2O(1)=1;`
	Determine the number of each element that form each of the chemical species.

Continue next page

Continue Code

Code | Function

```
d_H2=zeros(1,nec);
d_H2(3)=2;

d_N2=zeros(1,nec);
d_N2(4)=2;

d_C9H20=zeros(1,nec);
d_C9H20(2)=9;
d_C9H20(3)=20;

d_Fe=zeros(1,nec);
d_Fe(5)=1;

d_FeO=zeros(1,nec);
d_FeO(5)=1;
d_FeO(1)=1;

d_Si=zeros(1,nec);
d_Si(6)=1;

d_SiO2=zeros(1,nec);
d_SiO2(6)=1;
d_SiO2(1)=2;

d_Mn=zeros(1,nec);
d_Mn(7)=1;
```

Continue next page

Continue Code

Code | Function

```
d_MnO=zeros(1,nec);
d_MnO(7)=1;
d_MnO(1)=1;

d_C=zeros(1,nec);
d_C(2)=1;

d_Al=zeros(1,nec);
d_Al(8)=1;

d_Al2O3=zeros(1,nec);
d_Al2O3(8)=2;
d_Al2O3(1)=3;

d_Fe2O3=zeros(1,nec);
d_Fe2O3(5)=2;
d_Fe2O3(1)=3;

d_CaO=zeros(1,nec);
d_CaO(9)=1;
d_CaO(1)=1;

d_MgO=zeros(1,nec);
d_MgO(10)=1;
d_MgO(1)=1;

d_P=zeros(1,nec);
d_P(11)=1;
```

Continue next page

Continue Code

Code	Function

```
d_P2O5=zeros(1,nec);
d_P2O5(11)=2;
d_P2O5(1)=5;

d_S=zeros(1,nec);
d_S(12)=1;

d_SO2=zeros(1,nec);
d_SO2(11)=1;
d_SO2(1)=2;

d_Cr=zeros(1,nec);
d_Cr(13)=1;

d_Cr2O3=zeros(1,nec);
d_Cr2O3(13)=2;
d_Cr2O3(1)=3;

d_Ti=zeros(1,nec);
d_Ti(14)=1;

d_TiO2=zeros(1,nec);
d_TiO2(14)=1;
d_TiO2(1)=2;

d_CaCO3=zeros(1,nec);
d_CaCO3(9)=1;
```

Continue next page

Continue Code

Code	Function
`d_CaCO3(2)=1;` `d_CaCO3(1)=3;`	

`d_MgCO3=zeros(1,nec);`
`d_MgCO3(10)=1;`
`d_MgCO3(2)=1;`
`d_MgCO3(1)=3;`	

`d_Mo=zeros(1,nec);` `d_Mo(15)=1;`	
`d_Cu=zeros(1,nec);` `d_Cu(16)=1;`	
`d_Zn=zeros(1,nec);` `d_Zn(17)=1;`	
`%Distribution matrix construction`	Assign the numbers of each element that form each chemical species to a matrix.

`for i=1:numel(comp)`
`d=['d_' cell2mat(comp(i))];`
`dist(i,:)=eval(d);`
`end`	

`% Stream data`	Read the mass of the streams considered in the process in tons.
`%Outlets` `m_ST=NDat(40);`	Read the mass of the liquid steel.
`m_SL=12;`	The mass of slag is unknown.

Continue next page

Continue Code

Code	Function
`m_OG=20;`	The mass of off gases is unknown.
`m_PW=1;`	The mass of dust is unknown.
`m_PST_out=10;`	The mass of hot heel steel is unknown.
`m_PSL_out=3;`	The mass of hot heel slag is unknown.
`for i=1:numel(outlet)`	Assign the mass of the outlet streams to a vector.
`m=['m_' cell2mat(outlet(i))];`	
`m_out(i)=eval(m);`	
`end`	
`TAL=m_ST;`	
`%inlets`	
`m_DRI_F=NDat(17);`	Read the mass of cold DRI added to the process in tons.
`m_DRI_T=NDat(18);`	Read the mass of warm DRI added to the process in tons.
`m_DRI_C=NDat(19);`	Read the mass of hot DRI added to the process in tons.
`m_HBI=NDat(20);`	Read the mass of HBI added to the process in tons.
`m_AR=NDat(21);`	Read the mass of pig iron added to the process in tons.
`m_S1=NDat(22);`	Read the mass of scrap added to the process in tons.
`m_S2=NDat(23);`	
`m_S3=NDat(24);`	
`m_S4=NDat(25);`	
`m_S5=NDat(26);`	
`m_S6=NDat(27);`	
`m_S7=NDat(28);`	
`m_S8=NDat(29);`	
`m_S9=NDat(30);`	
`m_S10=NDat(31);`	

Continue next page

Continue Code

Code	Function
`m_S11=NDat(32);`	
`m_S12=NDat(33);`	
`m_S13=NDat(34);`	
`m_S14=NDat(35);`	
`m_S15=NDat(36);`	
`m_S16=NDat(37);`	
`m_S17=NDat(38);`	
`TCM=m_DRI_F+m_DRI_T+m_DRI_C+m_AR+m_HBI;`	Calculate the mass of metallic charge added to the process.
`for i=1:17`	Add the mass of scrap to the mass of the metallic charge.
`sc=['m_S' num2str(i)];`	
`m_s=eval(sc);`	
`TCM=TCM+m_s;`	
`end`	
`TON=TAL;`	
`m_OC=NDat(4);`	Read the mass of other fuels in the load in tons.
`m_WL=NDat(5);`	Read the mass of water in the load in tons.
`m_CC=NDat(7);`	Read the mass of carbon in the basket added in tons.
`m_Ci=NDat(8);`	Read the mass of carbon insufflated added in tons.
`m_Ant=NDat(9);`	Read the mass of anthracite added in tons.
`m_ESP=NDat(10);`	Read the mass of slag foamer added in tons.
`m_CD=NDat(11);`	Read the mass of dolomitic lime added in tons.
`m_CS=NDat(12);`	Read the mass of lime added in tons.
`m_CAC=NDat(13);`	Read the mass of calcium carbonate added in tons.
`m_CAM=NDat(14);`	Read the mass of magnesium carbonate added in tons.

Continue next page

Continue Code

Code	Function
`m_PST_in=NDat(15);`	Read the mass of hot heel steel at the beggining in tons.
`m_PSL_in=NDat(16);`	Read the mass of hot heel slag at the beggining in tons.
`m_Ai=0;`	The mass of air is unknown
`V_O2=NDat(2);`	Read the volume of oxygen added to the procces in Nm^3 per ton of liquid steel.
`m_Oxy=((V_O2*1000*1*PM_O2/(0.082*273.15))/10^6)*TON;`	Convert the volume of oxygen added to mass using the ideal gas law.
`V_CH4=NDat(3)`	Read the volume of natural gas added to the process in Nm^3 per ton of liquid steel.
`m_GN=((V_CH4*1000*1*PM_CH4/(0.082*273.15))/10^6)*TON;`	Convert the volume of natural gas added to mass using the ideal gas law.

`for i=1:numel(inlet)`
` m=['m_' cell2mat(inlet(i))];`
` m_in(i)=eval(m);`
`end`
	Assign the mass of the inlet streams to a vector.

`%Initial values of the unknowns`	Determine the initial values of the unknowns.
`m_c=m_in*c_in;`	
`x0(1)=TON*0.1;`	Initial value of the mass of air.
`x0(2)=TON*0.5;`	Initial value of the mass of off gases.
`x0(3)=TON*0.02;`	Initial value of the mass of dust.
`x0(4)=1;`	Initial value of the mass of water in the load.
`x0(5)=(m_c(12)+m_c(14)+m_c(17)+m_c(19)+m_c(20)+m_c(29)+m_c(30))*1.3;`	Initial value of the mass of slag.
`x0(6)=0.0001;`	Initial value of the mass of other fuels in the load.
`x0(7)=TON*0.2;`	Initial value of mass of hot heel steel at the end.

Continue next page

Continue Code

Code	Function
`x0(8)=TON*0.2*0.2;`	Initial value of mass of hot heel slag at the end.
`options= optimset('MaxFunEvals',10000, 'MaxIter',15000,'Algorithm', 'interior-point');`	Options of the solution method of the equation system.
`%Lower limits`	Determination of the lower limits of the unknowns.
`lb=zeros(numel(x0));`	
`lb(3)=TON*0.02*0.3;`	Lower limit of the mass of dust.
`lb(4)=0.001;`	Lower limit of the water in the load.
`lb(5)=m_c(12)+m_c(14)+m_c(17)+m_c(19)+m_c(20) +m_c(29)+m_c(30);`	Lower limit of the mass of slag.
`lb(6)=0.001;`	Lower limit of the other fuels in the load.
`%Upper limits`	Determination of the upper limits of the unknowns.
`ub=TON*ones(numel(x0));`	
`ub(3)=TON*0.02*1.5;`	Upper limit of the mass of dust.
`ub(4)=4;`	Upper limit of the mass of water in the load.
`ub(6)=0.1;`	Upper limit of the mass of other fuels in the load.
`ub(7)=TON*0.4;`	Upper limit of the mass of hot heel steel at the end.
`ub(8)=lb(5);`	Upper limit of the mass of hot heel slag at the end.
`%solucion del sistema`	
`[y,fval]=fmincon(@(x)(ecs(x,m_in,m_out,c_in, c_out,PM_c,PM_e,dist)),x0,[],[],[],[],lb,ub,[], options);`	Minimization algorithm to solve the equation system of the mass balance.
`%Found values assignement`	

Continue next page

Continue Code

Code	Function
`m_in(4)=y(1);`	Result of the mass of air.
`m_out(3)=y(2);`	Result of the mass of off gases.
`m_out(4)=y(3);`	Result of the mass of dust.
`m_in(5)=y(4);`	Result of the mass of water in the load.
`m_out(2)=y(5);`	Result of the mass of slag.
`m_in(3)=y(6);`	Result of the mass of other fuels in the load.
`m_out(5)=y(7);`	Result of the mass of hot heel steel at the end.
`m_out(6)=y(8);`	Result of the mass of hot heel slag at the end.
`'Comprobacion del balance de materia por elemento (Mega mol)'`	
`bal_mol=(m_in*c_in).*PM_c.^-1*dist-(m_out*c_out).*PM_c.^-1*dist;`	Moles balance of each element.
`mass_in=((m_in*c_in).*PM_c.^-1*dist).*PM_e;`	Inlet mass of each element.
`mass_out=((m_out*c_out).*PM_c.^-1*dist).*PM_e;`	Outlet mass of each element.
`er=mass_out-mass_in;`	Difference between the inlet and outlet mass of each element.
`O=er(1)`	
`C=er(2)`	
`H=er(3)`	
`N=er(4)`	
`Fe=er(5)`	
`Si=er(6)`	
`Mn=er(7)`	
`Al=er(8)`	
`Ca=er(9)`	
`Mg=er(10)`	
`P=er(11)`	
`S=er(12)`	

Continue next page

Continue Code

Code	Function

`Cr=er(13)`
`Ti=er(14)`
`Mo=er(15)`
`Cu=er(16)`
`Zn=er(17)`	

`Total=sum(mass_in)-sum(mass_out)`
`'Residuo:  error acumulado en el balance (toneladas)'`
`fval`
	Difference between the inlet and outlet mass to the process.

`bal=bal_mol.*PM_e;`	

`for i=1:numel(m_in)`
`Mc_in(i,:)=c_in(i,:)*m_in(i);`
`end`
	Matrix of mass of each compound in each inlet stream.

`for i=1:numel(m_out)`
`Mc_out(i,:)=c_out(i,:)*m_out(i);`
`end`
	Matrix of mass of each compound in each outlet stream.

`%Equation system` `function f=ecs(x,m_in,m_out,c_in,c_out,PM_c,PM_e,dist)`	Equation system of the mass balance. The input variables of this function are the mass of the inlet and outlet stream. the compositions of each stream, the molecular weight and the element distribution matrix.
`m_in(4)=x(1);` `m_out(3)=x(2);`	Establishes the unknowns of the equation system.

Continue next page

Continue Code

Code	Function
`m_out(4)=x(3);`	
`m_in(5)=x(4);`	
`m_out(2)=x(5);`	
`m_in(3)=x(6);`	
`m_out(5)=x(7);`	
`m_out(6)=x(8);`	
`% balance por elemento:`	
`% 17 ecuaciones`	
`f_p=((m_in*c_in).*PM_c.^-1*dist-(m_out*c_out).*PM_c.^-1*dist).*PM_e;`	Mass balance per element.
`% balance global:`	
`% 1 ecuacion`	
`f_p(18)=sum(m_in)-sum(m_out);`	Total mass balance.
`f=sum(sqrt(f_p.^2));`	Objective function.

C.3 Energy balance function

Code	Function
`%Energy balance Function`	
`function [ED_in_s,ED_in_c,ED_out_s,En_rxn_imp,OL, En_ox,En_C_flux,En_rxn_oth]=en_bal_2(comp,PM, M_in,M_out,T_in,T_out,P_in,P_out,Q_in,Q_out, W_in,W_out)`	Energy balance function. The input variables of this function are the chemical species considered in the model, the mass of each inlet and oulet stream, the tempearture of each stream, the heat losses and the electric energy added to the process.
`[m,n]=size(M_in);`	
`TAL=sum(M_out(1,:));`	Tons of liquid steel produced.
`TON=TAL;`	
`T_ref=298.15;`	Reference temperature in K.
`for i=1:m H_in(i,:)=M_in(i,:).*(enthalpy(T_in(i,1)) -enthalpy(T_ref)).*(PM.^-1)*(1000/3600)/TON; end`	Enthalpy of the inlet streams in kWh per ton of liquid steel.
`for i=1:m ED_in_s(i)=sum(H_in(i,:)); end`	Sensible energy of the inlet streams in kWh per ton of liquid steel.
`[m,n]=size(M_out);`	
`for i=1:m`	

Continue next page

Continue Code

Code	Function
`H_out(i,:)=M_out(i,:).*(enthalpy(T_out(i,1))-enthalpy(T_ref)).*(PM.^-1)*(1000/3600)/TON;`	Enthalpy of the outlet streams in kWh per ton of liquid steel.
`end`	
`for i=1:m`	
`ED_out_s(i)=sum(H_out(i,:));`	Sensible energy of the outlet streams in kWh per ton of liquid steel.
`end`	
`%Mass of reactive chemical species`	
`M_rx=(sum(M_out)-sum(M_in));`	Mass of chemical species that react during the process.
`N_rx=(sum(M_out)-sum(M_in)).*PM.^-1;`	Moles of chemical species that react during the process.
`%Heat of formation of chemical species`	
`Dhf=enthalpy(T_ref);`	Heat of formation of chemical species at reference temperature.
`%Combustion`	
`N_rx_CH4=-N_rx(1,2);`	Moles of reacted natural gas.
`N_rx_C9H20=-N_rx(1,8);`	Moles of reacted nonane.
`N_rx_C=-N_rx(1,15);`	Moles of reacted carbon.
`%CO and CO2 fraction`	
`N_rx_CO2=N_rx(4);`	Moles of carbon dioxide produced.
`N_rx_CO=N_rx(3);`	Moles of carbon monoxide produced.
`y=N_rx_CO2/(N_rx_CO2+N_rx_CO);`	Fraction of carbon dioxide respect the carbon dioxide and carbon monoxide produced.
`%Heat of combustion`	

Continue next page

Continue Code

Code	Function
`Comb_CH4=N_rx_CH4*(y*Dhf(1,4)+(1-y)*Dhf(1,3)+ 2*Dhf(1,5)-(Dhf(1,2)+((3/2)+(1/2)*y)* Dhf(1,1)))*(1000/3600)/TON;`	Heat released by combustion of natural gas.
`Comb_C9H20=N_rx_C9H20*(9*y*Dhf(1,4)+9*(1-y)* Dhf(1,3)+10*Dhf(1,5)-(Dhf(1,8)+((19/2)+ (9/2)*y)*Dhf(1,1)))*(1000/3600)/TON;`	Heat released by the combustion of nonane.
`Comb_C=N_rx_C*(y*Dhf(1,4)+(1-y)*Dhf(1,3)- (Dhf(1,15)+((1/2)+(1/2)*y)*Dhf(1,1)))* (1000/3600)/TON;`	Heat released by the combustion of carbon.
`En_comb=-(Comb_CH4+Comb_C9H20+Comb_C);`	Total heat released by the combustion.
`%Energy of oxydation`	
`N_rx_Fe=N_rx(1,10);`	Moles of reacted iron.
`Ox_Fe=N_rx_Fe*(Dhf(1,10)-(Dhf(1,9)+(1/2)* Dhf(1,1)))*(1000/3600)/TON;`	Heat realesed by the oxidation of iron.
`En_ox=-(Ox_Fe);`	
`%Important reactions`	
`En_rxn_imp=(En_comb+En_ox);`	Energy supplied by the reactions during the process.
`%All reactions`	
`En_rxn=(N_rx.*Dhf)*(1000/3600)/TON;`	Heat of formation of the reacted chemical species.
`En_rxn_tot=-sum(En_rxn);`	Energy released or absorbed by all the reactions that occur in the furnace.
`En_rxn_oth=En_rxn_tot-En_rxn_imp;`	Energy released or absorbed by all the reactions except the combustion and oxidation of iron.

Continue next page

Continue Code

Code	Function
`En_rxn_oth=0;`	Energy released or absorbed by all the reactions except the combustion and oxidation of iron in reality.
`%Separating the combustion reactions`	
`ED_in_c(1)=-Comb_CH4;`	Energy supplied by the combustion of natural gas
`ED_in_c(2)=-Comb_C9H20;`	Energy supplied by the combustion of nonane
`C_tot=0;`	
`j=0;`	
`for i=6:9`	
` j=j+1;`	
` C(j)=M_in(i,15);`	
` C_tot=C_tot+M_in(i,15);`	Total carbon added to the furnace.
`end`	
`j=j+1;`	
`C(j)=0;`	
`for i=16:37`	
` C(j)=C(j)+M_in(i,15);`	
` C_tot=C_tot+M_in(i,15);`	Total carbon added to the furnace, considering the carbon added in the metal load.
`end`	
`Frac_C=C/C_tot;`	Fraction of carbon contained in each stream, with respect to the total carbon added.
`En_C_flux=-Comb_C*Frac_C;`	Energy supplied by the carbon in each stream.

Continue next page

Continue Code

Code	Function

`%Determination of unknown losses`
`OL=sum(Q_in)-sum(Q_out)+sum(W_in)-sum(W_out)+`
`sum(ED_in_s)+sum(En_rxn_imp)-sum(ED_out_s);`
	Energy balance of the EAF Fuchs and determination of the unknown energy losses in the furnace.

C.4 Enthalpy function

Code	Function
`% Function enthalpy`	
`function h=enthalpy(T_alim)`	Function enthalpy. The input variable is the temperature of the stream.
`Tr=zeros(33,7);`	Reference temperature matrix.
`H_cf=zeros(33,7);`	Heat of phase change matrix.

`for i=1:33`
`Tr(i,1)=298.15;`
`end`
	Reference temperature in K.

`%Heats of formation`	Heats of formation of each chemical species considered in the model, in J/mol.
`H_cf(1,1)=0;`	
`H_cf(2,1)=-74.84*1000;`	
`H_cf(3,1)=-110.5*1000;`	
`H_cf(4,1)=-393.5*1000;`	
`H_cf(5,1)=-285.8*1000;`	
`H_cf(6,1)=0;`	
`H_cf(7,1)=0;`	
`H_cf(8,1)=-266059.61;`	
`H_cf(9,1)=0;`	
`H_cf(10,1)=-270.37*1000;`	
`H_cf(11,1)=0;`	

Continue next page

Continue Code

Code	Function

```
H_cf(12,1)=-851*1000;
H_cf(13,1)=0;
H_cf(14,1)=-384.9*1000;
H_cf(15,1)=0;
H_cf(16,1)=0;
H_cf(17,1)=-1669.792*1000;
H_cf(18,1)=-824.2*1000;
H_cf(19,1)=-635.6*1000;
H_cf(20,1)=-601.8*1000;
H_cf(21,1)=0;
H_cf(22,1)=-1506.24*1000;
H_cf(23,1)=0;
H_cf(24,1)=-296.9*1000;
H_cf(25,1)=0;
H_cf(26,1)=-1128424.8;
H_cf(27,1)=0;
H_cf(28,1)=-519.95*1000;
H_cf(29,1)=-1206.8748*1000;
H_cf(30,1)=-1112944;
H_cf(31,1)=0;
H_cf(32,1)=0;
H_cf(33,1)=0;

for i=1:33
 H_cf(i,1)=H_cf(i,1)/4.184;          Conversion of J/mol to cal/mol.
end

%Temperatures
```

Continue next page

Continue Code

Code	Function
`T=zeros(33,8);`	
`for i=1:33`	
`T(i,1)=T_alim;`	Temperature of each compound.
`end`	
`% Delta T change phase`	
`delta=0.1;`	Delta T used for give a little slope when exist phase change in the chemical species.
`%Reference Kelley/ Mines Bureau Bulletin`	
`%1.-O2`	
`n(1)=1;`	Number of phase.
`%Gas`	Phase
`a(1,1)=7.16;`	Coefficient “a”
`b(1,1)=1e-3;`	Coefficient “b”
`c(1,1)=-0.40e5;`	Coefficient “c”
`d(1,1)=0;`	Coefficient “d”
`%2.-CH4`	
`n(2)=1;`	
`%Gas`	
`a(2,1)=5.65;`	
`b(2,1)=11.44e-3;`	
`c(2,1)=-0.46e5;`	
`d(2,1)=0;`	

Continue next page

Continue Code

Code	Function
`%3.-CO`	
`n(3)=1;`	
`%Gas`	
`a(3,1)=6.79;`	
`b(3,1)=0.98e-3;`	
`c(3,1)=-0.11e5;`	
`d(3,1)=0;`	
`%4.-CO2`	
`n(4)=1;`	
`%Gas`	
`a(4,1)=10.57;`	
`b(4,1)=2.1e-3;`	
`c(4,1)=-2.06e5;`	
`d(4,1)=0;`	
`%5.-H2O`	
`%Liquid`	Phase
`n(5)=1;`	Number of phase.
`a(5,1)=18.04;`	Coefficient “a”
`b(5,1)=0;`	Coefficient “b”
`c(5,1)=0;`	Coefficient “c”
`d(5,1)=0;`	Coefficient “d”
`Tv=373.15-delta;`	Temperature of phase change minus delta T.
`if T(5,1)>Tv`	The temperature of the stream is greater than the temperature of phase change.

Continue next page

Continue Code

Code	Function
`n(5)=2;`	Number of phase.
`T(5,2)=T(5,1);`	Upper temperature of the new phase.
`T(5,1)=Tv;`	Upper temperature of the previous phase.
`Tr(5,2)=Tv;`	Lower temperature of the new phase.
`%Cambio de estado`	Phase change.
`H_cf(5,2)=0;`	
`a(5,2)=9718/delta;`	Latent heat divided by the delta T.
`b(5,2)=0;`	
`c(5,2)=0;`	
`d(5,2)=0;`	
`end`	
`Tv=373.15;`	Temperature of phase change.
`if T(5,2)>Tv`	The temperature of the stream is greater than the temperature of phase change.
`n(5)=3;`	Number of phase.
`T(5,3)=T(5,2);`	Upper temperature of the new phase.
`T(5,2)=Tv;`	Upper temperature of the previous phase.
`Tr(5,3)=Tv;`	Lower temperature of the new phase.
`%Gas`	Phase
`H_cf(5,3)=0;`	
`a(5,3)=7.30;`	Coefficient “a”
`b(5,3)=2.46e-3;`	Coefficient “b”
`c(5,3)=0;`	Coefficient “c”
`d(5,3)=0;`	Coefficient “d”
`end`	

Continue next page

Continue Code

Code | Function

```
%6.-H2
%Gas
n(6)=1;
a(6,1)=6.25;
b(6,1)=0.78e-3;
c(6,1)=0.12e5;
d(6,1)=0;

%7.-N2
%Gas
n(7)=1;
a(7,1)=6.83;
b(7,1)=0.90e-3;
c(7,1)=-0.12e5;
d(7,1)=0;

%8.-C9H20 Perry
%Lquido
n(8)=1;
a(8,1)=91.558;
b(8,1)=-0.2724;
c(8,1)=0;
d(8,1)=6.477e-4;

Tv=424-delta;

if T(8,1)>Tv
 n(8)=2;
```

Continue next page

Continue Code

Code | Function

```
 T(8,2)=T(8,1);
 T(8,1)=Tv;
 Tr(8,2)=Tv;
 %Cambio de estado
 H_cf(8,2)=0;
 a(8,2)=8919.603/delta;
 b(8,2)=0;
 c(8,2)=0;
 d(8,2)=0;
end

Tv=424;

if T(8,2)>Tv
 n(8)=3;
 T(8,3)=T(8,2);
 T(8,2)=Tv;
 Tr(8,3)=Tv;
 %Gas
 H_cf(8,3)=0;
 a(8,3)=8.7751;
 b(8,3)=158.9e-3;
 c(8,3)=0;
 d(8,3)=-51.486e-6;
end

%9.-Fe
%alfa
```

Continue next page

Continue Code

Code | Function

```
n(9)=1;
a(9,1)=3.04;
b(9,1)=7.58e-3;
c(9,1)=0.60e5;
d(9,1)=0;

Tv=1033-delta;

if T(9,1)>Tv
 n(9)=2;
 T(9,2)=T(9,1);
 T(9,1)=Tv;
 Tr(9,2)=Tv;
 %Cambio de fase
 H_cf(9,2)=0;
 a(9,2)=326/delta;
 b(9,2)=0;
 c(9,2)=0;
 d(9,2)=0;
end

Tv=1033;

if T(9,2)>Tv
 n(9)=3;
 T(9,3)=T(9,2);
 T(9,2)=Tv;
 Tr(9,3)=Tv;
```

Continue next page

Continue Code

Code | Function

```
 %Beta
 H_cf(9,3)=0;
 a(9,3)=11.13;
 b(9,3)=0;
 c(9,3)=0;
 d(9,3)=0;
end

Tv=1183-delta;

if T(9,3)>Tv
 n(9)=4;
 T(9,4)=T(9,3);
 T(9,3)=Tv;
 Tr(9,4)=Tv;
 %cambio de fase
 H_cf(9,4)=0;
 a(9,4)=215/delta;
 b(9,4)=0;
 c(9,4)=0;
 d(9,4)=0;
end

Tv=1183;

if T(9,4)>Tv
 n(9)=5;
 T(9,5)=T(9,4);
```

Continue next page

Continue Code

Code Function

```
 T(9,4)=Tv;
 Tr(9,5)=Tv;
 %Gamma
 H_cf(9,5)=0;
 a(9,5)=5.80;
 b(9,5)=1.98e-3;
 c(9,5)=0;
 d(9,5)=0;
end

Tv=1672-delta;

if T(9,5)>Tv
 n(9)=6;
 T(9,6)=T(9,5);
 T(9,5)=Tv;
 Tr(9,6)=Tv;
 %Cambio de fase
 H_cf(9,6)=0;
 a(9,6)=165/delta;
 b(9,6)=0;
 c(9,6)=0;
 d(9,6)=0;
end

Tv=1672;

if T(9,6)>Tv
```

Continue next page

Continue Code

Code Function

```
 n(9)=7;
 T(9,7)=T(9,6);
 T(9,6)=Tv;
 Tr(9,7)=Tv;
 %Delta
 H_cf(9,7)=0;
 a(9,7)=6.74;
 b(9,7)=1.60e-3;
 c(9,7)=0;
 d(9,7)=0;
end

Tv=1812-delta;

if T(9,7)>Tv
 n(9)=8;
 T(9,8)=T(9,7);
 T(9,7)=Tv;
 Tr(9,8)=Tv;
 %Cambio de fase
 H_cf(9,8)=0;
 a(9,8)=3670/delta;
 b(9,8)=0;
 c(9,8)=0;
 d(9,8)=0;
end

Tv=1812;
```

Continue next page

Continue Code

Code | Function

```
if T(9,8)>Tv
 n(9)=9;
 T(9,9)=T(9,8);
 T(9,8)=Tv;
 %Lquido
 H_cf(9,9)=0;
 a(9,9)=9.77;
 b(9,9)=0.40e-3;
 c(9,9)=0;
 d(9,9)=0;
end

%10.-FeO
%solido
n(10)=1;
a(10,1)=11.66;
b(10,1)=2.00e-3;
c(10,1)=-0.67e5;
d(10,1)=0;

Tv=1650-delta;

if T(10,1)>Tv
 n(10)=2;
 T(10,2)=T(10,1);
 T(10,1)=Tv;
 Tr(10,2)=Tv;
```

Continue next page

Continue Code

Code | Function

```
 %Cambio de fase
 H_cf(10,2)=0;
 a(10,2)=7490/delta;
 b(10,2)=0;
 c(10,2)=0;
 d(10,2)=0;
end

Tv=1650;

if T(10,2)>Tv
 n(10)=3;
 T(10,3)=T(10,2);
 T(10,2)=Tv;
 Tr(10,3)=Tv;
 %Lquido
 H_cf(10,3)=0;
 a(10,3)=16.30;
 b(10,3)=0;
 c(10,3)=0;
 d(10,3)=0;
end

%11.-Si
%solido
n(11)=1;
a(11,1)=5.70;
b(11,1)=0.70e-3;
```

Continue next page

Continue Code

Code Function

```
c(11,1)=-1.04e5;
d(11,1)=0;

Tv=1685-delta;

if T(11,1)>Tv
 n(11)=2;
 T(11,2)=T(11,1);
 T(11,1)=Tv;
 Tr(11,2)=Tv;
 %Cambio de fase
 H_cf(11,2)=0;
 a(11,2)=12100/delta;
 b(11,2)=0;
 c(11,2)=0;
 d(11,2)=0;
end

Tv=1685;

if T(11,2)>Tv
 n(11)=3;
 T(11,3)=T(11,2);
 T(11,2)=Tv;
 Tr(11,3)=Tv;
 %Lquido
 H_cf(11,3)=0;
 a(11,3)=6.1;
```

Continue next page

Continue Code

Code Function

```
 b(11,3)=0;
 c(11,3)=0;
 d(11,3)=0;
end

%12.-SiO2
%Alfa
n(12)=1;
a(12,1)=11.22;
b(12,1)=8.20e-3;
c(12,1)=-2.70e5;
d(12,1)=0;

Tv=848-delta;

if T(12,1)>Tv
 n(12)=2;
 T(12,2)=T(12,1);
 T(12,1)=Tv;
 Tr(12,2)=Tv;
 %Cambio de fase
 H_cf(12,2)=0;
 a(12,2)=290/delta;
 b(12,2)=0;
 c(12,2)=0;
 d(12,2)=0;
end
```

Continue next page

Continue Code

Code | Function

```
Tv=848;

if T(12,2)>Tv
 n(12)=3;
 T(12,3)=T(12,2);
 T(12,2)=Tv;
 Tr(12,3)=Tv;
 %Beta
 H_cf(12,3)=0;
 a(12,3)=14.41;
 b(12,3)=1.94e-3;
 c(12,3)=0;
 d(12,3)=0;
end
%13.-Mn
%Alfa
n(13)=1;
a(13,1)=5.70;
b(13,1)=3.38e-3;
c(13,1)=-0.37e5;
d(13,1)=0;

Tv=1000-delta;

if T(13,1)>Tv
 n(13)=2;
 T(13,2)=T(13,1);
 T(13,1)=Tv;
```

Continue next page

Continue Code

Code Function

```
 Tr(13,2)=Tv;
 %Cambio de fase
 H_cf(13,2)=0;
 a(13,2)=535/delta;
 b(13,2)=0;
 c(13,2)=0;
 d(13,2)=0;
end

Tv=1000;

if T(13,2)>Tv
 n(13)=3;
 T(13,3)=T(13,2);
 T(13,2)=Tv;
 Tr(13,3)=Tv;
 %Beta
 H_cf(13,3)=0;
 a(13,3)=8.33;
 b(13,3)=0.66e-3;
 c(13,3)=0;
 d(13,3)=0;
end

Tv=1374;

if T(13,3)>Tv
 n(13)=4;
```

Continue next page

Continue Code

Code Function

```
 T(13,4)=T(13,3);
 T(13,3)=Tv;
 Tr(13,4)=Tv;
 %Cambio de fase
 H_cf(13,4)=0;
 a(13,4)=545/delta;
 b(13,4)=0;
 c(13,4)=0;
 d(13,4)=0;
end

Tv=1374;

if T(13,4)>Tv
 n(13)=5;
 T(13,5)=T(13,4);
 T(13,4)=Tv;
 Tr(13,5)=Tv;
 %Gamma
 H_cf(13,5)=0;
 a(13,5)=10.70;
 b(13,5)=0;
 c(13,5)=0;
 d(13,5)=0;
end

Tv=1410-delta;
```

Continue next page

Continue Code

Code Function

```
if T(13,5)>Tv
 n(13)=6;
 T(13,6)=T(13,5);
 T(13,5)=Tv;
 Tr(13,6)=Tv;
 %Cambio de fase
 H_cf(13,6)=0;
 a(13,6)=430/delta;
 b(13,6)=0;
 c(13,6)=0;
 d(13,6)=0;
end

Tv=1410;

if T(13,6)>Tv
 n(13)=7;
 T(13,7)=T(13,6);
 T(13,6)=Tv;
 Tr(13,7)=Tv;
 %Delta
 H_cf(13,7)=0;
 a(13,7)=11.30;
 b(13,7)=0;
 c(13,7)=0;
 d(13,7)=0;
end
```

Continue next page

Continue Code

Code | Function

```
Tv=1517-delta;

if T(13,7)>Tv
 n(13)=8;
 T(13,8)=T(13,7);
 T(13,7)=Tv;
 Tr(13,8)=Tv;
 %Cambio de fase
 H_cf(13,8)=0;
 a(13,8)=3500/delta
; b(13,8)=0;
 c(13,8)=0;
 d(13,8)=0;
end

Tv=1517;

if T(13,8)>Tv
 n(13)=9;
 T(13,9)=T(13,8);
 T(13,8)=Tv;
 Tr(13,9)=Tv;
 %Liquido
 H_cf(13,9)=0;
 a(13,9)=11;
 b(13,9)=0;
 c(13,9)=0;
 d(13,9)=0;
```

Continue next page

Continue Code

Code | Function

```
end

%14.-MnO
%Solido
n(14)=1;
a(14,1)=11.11;
b(14,1)=1.94e-3;
c(14,1)=-0.88e5;
d(14,1)=0;

Tv=1800-delta;

if T(14,1)>Tv
 n(14)=2;
 T(14,2)=T(14,1);
 T(14,1)=Tv;
 Tr(14,2)=Tv;
 %Cambio de fase
 H_cf(14,2)=0;
 a(14,2)=55150/delta;
 b(14,2)=0;
 c(14,2)=0;
 d(14,2)=0;
end

Tv=1800;

if T(14,2)>Tv
```

Continue next page

Continue Code

Code Function

```
 n(14)=3;
 T(14,3)=T(14,2);
 T(14,2)=Tv;
 Tr(14,3)=Tv;
 %Gas
 H_cf(14,3)=0;
 a(14,3)=8.45;
 b(14,3)=0.30e-3;
 c(14,3)=-0.87e5;
 d(14,3)=0;
end

%15.-C
%Solido
n(15)=1;
a(15,1)=4.03;
b(15,1)=1.14e-3;
c(15,1)=-2.04e5;
d(15,1)=0;

%16.-Al
%Solido
n(16)=1;
a(16,1)=4.94;
b(16,1)=2.96e-3;
c(16,1)=0;
d(16,1)=0;
```

Continue next page

Continue Code

Code | Function

```
Tv=932-delta;

if T(16,1)>Tv
 n(16)=2;
 T(16,2)=T(16,1);
 T(16,1)=Tv;
 Tr(16,2)=Tv;
 %Cambio de fase
 H_cf(16,2)=0;
 a(16,2)=10718/delta;
 b(16,2)=0;
 c(16,2)=0;
 d(16,2)=0;
end

Tv=932;

if T(16,2)>Tv
 n(16)=3;
 T(16,3)=T(16,2);
 T(16,2)=Tv;
 Tr(16,3)=Tv;
 %Lquido
 H_cf(16,3)=0;
 a(16,3)=7;
 b(16,3)=0;
 c(16,3)=0;
 d(16,3)=0;
```

Continue next page

Continue Code

Code | Function

```
end

%17.-Al2O3
%Alfa
n(17)=1;
a(17,1)=27.49;
b(17,1)=2.82e-3;
c(17,1)=-8.38e5;
d(17,1)=0;


%18.-Fe2O3
%Alfa
n(18)=1;
a(18,1)=23.49;
b(18,1)=18.60e-3
;c(18,1)=-3.55e5;
d(18,1)=0;

Tv=950-delta;

if T(18,1)>Tv
 n(18)=2;
 T(18,2)=T(18,1);
 T(18,1)=Tv;
 Tr(18,2)=Tv;
 %Cambio de fase
 H_cf(18,2)=0;
```

Continue next page

Continue Code

Code | Function

```
 a(18,2)=160/delta;
 b(18,2)=0;
 c(18,2)=0;
 d(18,2)=0;
end

Tv=950;

if T(18,2)>Tv
 n(18)=3;
 T(18,3)=T(18,2);
 T(18,2)=Tv;
 Tr(18,3)=Tv;
 %Beta
 H_cf(18,3)=0;
 a(18,3)=36.00;
 b(18,3)=0;
 c(18,3)=0;
 d(18,3)=0;
end

Tv=1050-delta;

if T(18,3)>Tv
 n(18)=4;
 T(18,4)=T(18,3);
 T(18,3)=Tv;
 Tr(18,4)=Tv;
```

Continue next page

Continue Code

Code Function

```
 %Cambio de fase
 H_cf(18,4)=0;
 a(18,4)=0/delta;
 b(18,4)=0;
 c(18,4)=0;
 d(18,4)=0;
end

Tv=1050;

if T(18,4)>Tv
 n(18)=5;
 T(18,5)=T(18,4);
 T(18,4)=Tv;
 Tr(18,5)=Tv;
 %Gamma
 H_cf(18,5)=0;
 a(18,5)=31.71;
 b(18,5)=1.76e-3;
 c(18,5)=0;
 d(18,5)=0;
end

%19.-CaO
%Solido
n(19)=1;
a(19,1)=11.67;
b(19,1)=1.08e-3;
```

Continue next page

Continue Code

Code | Function

```
c(19,1)=-1.56e5;
d(19,1)=0;

%20.-MgO
%Solido
n(20)=1;
a(20,1)=10.18;
b(20,1)=1.74e-3;
c(20,1)=-1.48e5;
d(20,1)=0;

%21.-P
%Solido
n(21)=1;
a(21,1)=22.50;
b(21,1)=0;
c(21,1)=0;
d(21,1)=0;

Tv=317.4-delta;

if T(21,1)>Tv
 n(21)=2;
 T(21,2)=T(21,1);
 T(21,1)=Tv;
 Tr(21,2)=Tv;
 %Cambio de fase
 H_cf(21,2)=0;
```

Continue next page

Continue Code

Code | Function

```
 a(21,2)=600/delta;
 b(21,2)=0;
 c(21,2)=0;
 d(21,2)=0;
end

Tv=317.4;

if T(21,2)>Tv
 n(21)=3;
 T(21,3)=T(21,2);
 T(21,2)=Tv;
 Tr(21,3)=Tv;
 %Liquido
 H_cf(21,3)=0;
 a(21,3)=23.5;
 b(21,3)=0;
 c(21,3)=0;
 d(21,3)=0;
end

Tv=400-delta;

if T(21,3)>Tv
 n(21)=4;
 T(21,4)=T(21,3);
 T(21,3)=Tv;
 Tr(21,4)=Tv;
```

Continue next page

Continue Code

Code | Function

```
 %Cambio de fase
 H_cf(21,4)=0;
 a(21,4)=25600/delta;
 b(21,4)=0;
 c(21,4)=0;
 d(21,4)=0;
end

Tv=400;

if T(21,4)>Tv
 n(21)=5;
 T(21,5)=T(21,4);
 T(21,4)=Tv;
 Tr(21,5)=Tv;
 %Gas
 H_cf(21,5)=0;
 a(21,5)=4.97;
 b(21,5)=0;
 c(21,5)=0;
 d(21,5)=0;
end

%22.-P2O5
%Solido
n(22)=1;
a(22,1)=40;
b(22,1)=0;
```

Continue next page

Continue Code

Code | Function

```
c(22,1)=0;
d(22,1)=0;

%23.-S
%Alfa
n(23)=1;
a(23,1)=3.58;
b(23,1)=6.24e-3;
c(23,1)=0;
d(23,1)=0;

Tv=368.6-delta;

if T(23,1)>Tv
 n(23)=2;
 T(23,2)=T(23,1);
 T(23,1)=Tv;
 Tr(23,2)=Tv;
 %Cambio de fase
 H_cf(23,2)=0;
 a(23,2)=85/delta;
 b(23,2)=0;
 c(23,2)=0;
 d(23,2)=0;
end

Tv=368.6;
```

Continue next page

Continue Code

Code Function

```
if T(23,2)>Tv
 n(23)=3;
 T(23,3)=T(23,2);
 T(23,2)=Tv;
 Tr(23,3)=Tv;
 %Beta
 H_cf(23,3)=0;
 a(23,3)=6.20;
 b(23,3)=0;
 c(23,3)=0;
 d(23,3)=0;
end

Tv=392-delta;

if T(23,3)>Tv
 n(23)=4;
 T(23,4)=T(23,3);
 T(23,3)=Tv;
 Tr(23,4)=Tv;
 %Cambio de fase
 H_cf(23,4)=0;
 a(23,4)=335/delta
; b(23,4)=0;
 c(23,4)=0;
 d(23,4)=0;
end
```

Continue next page

Continue Code

Code | Function

```
Tv=392;

if T(23,4)>Tv
 n(23)=5;
 T(23,5)=T(23,4);
 T(23,4)=Tv;
 Tr(23,5)=Tv;
 %Lquido
 H_cf(23,5)=0;
 a(23,5)=8.73;
 b(23,5)=0;
 c(23,5)=0;
 d(23,5)=0;
end

Tv=717.8-delta;

if T(23,5)>Tv
 n(23)=6;
 T(23,6)=T(23,5);
 T(23,5)=Tv;
 Tr(23,6)=Tv;
 %Cambio de fase
 H_cf(23,6)=0;
 a(23,6)=2200/delta;
 b(23,6)=0;
 c(23,6)=0;
 d(23,6)=0;
```

Continue next page

Continue Code

Code | Function

```
end

Tv=717.8;

if T(23,6)>Tv
 n(23)=7;
 T(23,7)=T(23,6);
 T(23,6)=Tv;
 Tr(23,7)=Tv;
 %Gas alfa
 H_cf(23,7)=0;
 a(23,7)=5.26;
 b(23,7)=-0.10e-3;
 c(23,7)=0.36e5;
 d(23,7)=0;
end

Tv=2400-delta;

if T(23,7)>Tv
 n(23)=8;
 T(23,8)=T(23,7);
 T(23,7)=Tv;
 Tr(23,8)=Tv;
 %Cambio de fase
 H_cf(23,8)=0;
 a(23,8)=0/delta;
 b(23,8)=0;
```

Continue next page

Continue Code

Code Function

```
 c(23,8)=0;
 d(23,8)=0;
end

Tv=2400;

if T(23,8)>Tv
 n(23)=9;
 T(23,9)=T(23,8);
 T(23,8)=Tv;
 Tr(23,9)=Tv;
 %Gas Beta
 H_cf(23,9)=0;
 a(23,9)=4.96;
 b(23,9)=0.10e-3;
 c(23,9)=0.60e5;
 d(23,9)=0;
end

%24.-SO2
%Gas
n(24)=1;
a(24,1)=11.04;
b(24,1)=1.88e-3;
c(24,1)=-1.84e5;
d(24,1)=0;

%25.-Cr
```

Continue next page

Continue Code

Code | Function

```
%Alfa
n(25)=1;
a(25,1)=4.16;
b(25,1)=3.62e-3;
c(25,1)=0.30e5;
d(25,1)=0;

Tv=2176-delta;

if T(25,1)>Tv
 n(25)=2;
 T(25,2)=T(25,1);
 T(25,1)=Tv;
 Tr(25,2)=Tv;
 %Cambio de fase
 H_cf(25,2)=0;
 a(25,2)=5000/del
ta;  b(25,2)=0;
 c(25,2)=0;
 d(25,2)=0;
end

Tv=2176;

if T(25,2)>Tv
 n(25)=3;
 T(25,3)=T(25,2);
 T(25,2)=Tv;
```

Continue next page

Continue Code

Code | Function

```
 Tr(25,3)=Tv;
 %Liquido
 H_cf(25,3)=0;
 a(25,3)=9.40;
 b(25,3)=0;
 c(25,3)=0;
 d(25,3)=0;
end

%26.-Cr2O3
%Solido
n(26)=1;
a(26,1)=28.53;
b(26,1)=2.20e-3;
c(26,1)=-3.74e5;
d(26,1)=0;

%27.-Ti
%Alfa
n(27)=1;
a(27,1)=5.25;
b(27,1)=2.52e-3;
c(27,1)=0;
d(27,1)=0;

Tv=1155-delta;

if T(27,1)>Tv
```

Continue next page

Continue Code

Code Function

```
 n(27)=2;
 T(27,2)=T(27,1);
 T(27,1)=Tv;
 Tr(27,2)=Tv;
 %Cambio de fase
 H_cf(27,2)=0;
 a(27,2)=950/delta;
 b(27,2)=0;
 c(27,2)=0;
 d(27,2)=0;
end

Tv=1155;

if T(27,2)>Tv
 n(27)=3;
 T(27,3)=T(27,2);
 T(27,2)=Tv;
 Tr(27,3)=Tv;
 %Beta
 H_cf(27,3)=0;
 a(27,3)=7.50;
 b(27,3)=0;
 c(27,3)=0;
 d(27,3)=0;
end

Tv=1940-delta;
```

Continue next page

Continue Code

Code | Function

```
if T(27,3)>Tv
 n(27)=4;
 T(27,4)=T(27,3);
 T(27,3)=Tv;
 Tr(27,4)=Tv;
 %Cambio de fase
 H_cf(27,4)=0;
 a(27,4)=4460/delta
; b(27,4)=0;
 c(27,4)=0;
 d(27,4)=0;
end

Tv=1940;

if T(27,4)>Tv
 n(27)=5;
 T(27,5)=T(27,4);
 T(27,4)=Tv;
 Tr(27,5)=Tv;
 %Liquido
 H_cf(27,5)=0;
 a(27,5)=8.00;
 b(27,5)=0;
 c(27,5)=0;
 d(27,5)=0;
end
```

Continue next page

Continue Code

Code | Function

```
%28.-TiO2
%Rutilo
n(28)=1;
a(28,1)=17.97;
b(28,1)=0.28e-3;
c(28,1)=-4.35e5;
d(28,1)=0;

%29.-CaCO3
%Alfa
n(29)=1;
a(29,1)=24.98;
b(29,1)=5.24e-3;
c(29,1)=-6.20e5;
d(29,1)=0;


%30.-MgCO3
%Alfa
n(30)=1;
a(30,1)=18.62;
b(30,1)=13.80e-3;
c(30,1)=-4.16e5;
d(30,1)=0;

%31.-Mo
%Alfa
```

Continue next page

Continue Code

Code | Function

```
n(31)=1;
a(31,1)=5.18;
b(31,1)=1.66e-3;
c(31,1)=0;
d(31,1)=0;

Tv=2890-delta;

if T(31,1)>Tv
 n(31)=2;
 T(31,2)=T(31,1);
 T(31,1)=Tv;
 Tr(31,2)=Tv;
 %Cambio de fase
 H_cf(31,2)=0;
 a(31,2)=6650/delta;
 b(31,2)=0;
 c(31,2)=0;
 d(31,2)=0;
end

Tv=2890;

if T(31,2)>Tv
 n(31)=3;
 T(31,3)=T(31,2);
 T(31,2)=Tv;
 Tr(31,3)=Tv;
```

Continue next page

Continue Code

Code Function

```
 %Liquido
 H_cf(31,3)=0;
 a(31,3)=10.00;
 b(31,3)=0;
 c(31,3)=0;
 d(31,3)=0;
end

%32.-Cu
%Alfa
n(32)=1;
a(32,1)=5.41;
b(32,1)=1.50e-3;
c(32,1)=0;
d(32,1)=0;

Tv=1357-delta;

if T(32,1)>Tv
 n(32)=2;
 T(32,2)=T(32,1);
 T(32,1)=Tv;
 Tr(32,2)=Tv;
 %Cambio de fase
 H_cf(32,2)=0;
 a(32,2)=3120/delta;
 b(32,2)=0;
 c(32,2)=0;
```

Continue next page

Continue Code

Code	Function

```
 d(32,2)=0;
end

Tv=1357;

if T(32,2)>Tv
 n(32)=3;
 T(32,3)=T(32,2);
 T(32,2)=Tv;
 Tr(32,3)=Tv;
 %Liquido
 H_cf(32,3)=0;
 a(32,3)=7.50;
 b(32,3)=0;
 c(32,3)=0;
 d(32,3)=0;
end

%33.-Zn
%Solido
n(33)=1;
a(33,1)=5.35;
b(33,1)=2.40e-3;
c(33,1)=0;
d(33,1)=0;

Tv=692.7-delta;
```

Continue next page

Continue Code

Code Function

```
if T(33,1)>Tv
 n(33)=2;
 T(33,2)=T(33,1);
 T(33,1)=Tv;
 Tr(33,2)=Tv;
 %Cambio de fase
 H_cf(33,2)=0;
 a(33,2)=1765/delta;
 b(33,2)=0;
 c(33,2)=0;
 d(33,2)=0;
end

Tv=692.7;

if T(33,2)>Tv
 n(33)=3;
 T(33,3)=T(33,2);
 T(33,2)=Tv;
 Tr(33,3)=Tv;
 %Liquido
 H_cf(33,3)=0;
 a(33,3)=7.50;
 b(33,3)=0;
 c(33,3)=0;
 d(33,3)=0;
end
```

Continue next page

Continue Code

Code | Function

```
Tv=907-delta;

if T(33,3)>Tv
 n(33)=4;
 T(33,4)=T(33,3);
 T(33,3)=Tv;
 Tr(33,4)=Tv;
 %Cambio de fase
 H_cf(33,4)=0;
 a(33,4)=27430/delta;
 b(33,4)=0;
 c(33,4)=0;
 d(33,4)=0;
end

Tv=907;

if T(33,4)>Tv
 n(33)=5;
 T(33,5)=T(33,4);
 T(33,4)=Tv;
 Tr(33,5)=Tv;
 %Gaseoso
 H_cf(33,5)=0;
 a(33,5)=4.97;
 b(33,5)=0;
 c(33,5)=0;
 d(33,5)=0;
```

Continue next page

Continue Code

Code	Function
`end`	
`%------------------------------`	
`%Equation`	
`%------------------------------`	
`h=zeros(33,1);`	
`for i=1:33`	Number of chemical species.
`h(i)=0;`	
`for j=1:n(i)`	Number of phase reached by the chemical species.
`h(i)=h(i)+H_cf(i,j)+integral(@(x)(a(i,j)+`	Enthalpy of each chemical species in cal/mol.
`b(i,j)*(x)+c(i,j)*(x.^(-2))+d(i,j)*(x.^2)),`	
`Tr(i,j),T(i,j));`	
`end`	
`end`	
`h=h'*4.184;`	Conversion of cal/mol to J/mol.

C.5 Heat loss in the bottom of the furnace function

Code	Function

`% Function EAF Bottom`
`%Determine the heat transfer rate in the`
`bootom of the furnace`	

`function q=FondoHorno(TTT,A_fond,P_fond,h_fond,` `T_steel,Talr,dx,kref,TON)`	Function EAF Bottom. The input variables are the tap to tap time, the area and perimeter of the furnace bottom, the elevation respect to the ground of the eaf, the temperature of the liquid steel, the temperature of the surroundings, the thickness of the refractory layer, the thermal conductivity of the refractory brick and the tons of liquid steel produced.

`%Surfaces considered`
`% 1.-Furnace bottom`
`% 2.-Walls near to the furnace`
`% 3.-Floor`
`% 4.-Surroundings`	

`%View factors` `F=zeros(4,4);`	View factors between the surfaces considered.

`F(1,2)=0.46465;`
`F(1,3)=0.1378;`
`F(1,4)=0.2155;`	

`F(2,1)=0.1655;`
`F(2,3)=0.1655;`
`F(2,4)=0.223;`	

Continue next page

Continue Code

Code	Function
`F(3,1)=0.1378;`	
`F(3,2)=0.6465;`	
`F(3,4)=0.2155;`	
`F(4,1)=0.1655;`	
`F(4,2)=0.669;`	
`F(4,3)=0.1655;`	
`%Area of the surfaces (Considering a space with square prism form under the furnace)`	
`A(1)=A_fond;`	Furnace bottom area.
`A(2)=3*sqrt(A_fond)*h_fond;`	Area of the walls near to the furnace.
`A(3)=A_fond;`	Area of the floor under the furnace
`A(4)=sqrt(A_fond)*h_fond;`	Area of the opening towards the surroundings.
`%Emissivity of the surfaces`	
`epsilon(1)=0.95;`	
`epsilon(2)=0.95;`	
`epsilon(3)=0.95;`	
`epsilon(4)=0.95;`	
`%Initial values`	
`x0(1)=329.1;`	Initial value of the temperature of the furnace bottom.
`x0(2)=310.3;`	Initial value of the temperature of the walls near to the furnace.
`x0(3)=310.7;`	Initial value of the temperature of the floor.
`x0(4)=658.5;`	Inital value of the radiosity of the furnace bottom.

Continue next page

Continue Code

Code	Function
`x0(5)=453.1;`	Initial value of the radiosity of the surroundings.
`%Upper limits`	
`ub(1)=T_steel;`	Upper limit of the temperature of the furnace bottom.
`ub(2)=T_steel;`	Upper limit of the temperature of the walls near to the furnace.
`ub(3)=T_steel;`	Upper limit of the temperature of the floor.
`ub(4)=1000000;`	Upper limit of the radiosity of the furnace bottom.
`ub(5)=1000000;`	Upper limit of the radiosity of the surroundings.
`%Lower limits`	
`lb(1)=293.15;`	Lower limit of the temperature of the furnace bottom.
`lb(2)=293.15;`	Lower limit of the temperature of the walls near to the furnace.
`lb(3)=293.15;`	Lower limit of the temperature of the floor.
`lb(4)=100;`	Lower limit of the radiosity of the furnace bottom.
`lb(5)=100;`	Lower limit of the radiosity of the surroundings.
`[y]=fmincon(@(x)(CondConvRad(x,kref,A_fond,T_steel,dx,P_fond,Talr,epsilon,A,F)),x0,[],[],[],[],lb,ub);`	Minimization algorithm used to solve the equation system of the heat transfer mechanisms.
`T(1)=y(1);`	Temperature of the furnace bottom.
`q_dot=-kref*A_fond*(T(1)-T_steel)/dx;`	Heat rate transferred in the furnace bottom in W.
`q=q_dot*(TTT*60)/(1000*3600)/TON;`	Total heat transferred in the furnace bottom during the process in kWh per ton of liquid steel.
`end`	

Continue next page

Continue Code

Code	Function

`%Equation system`
`function f=CondConvRad(x,k_ref,A_fond,T_steel,`
`dx,P_fond,Talr,epsilon,A,F)`
	Equation system for the heat transfer in the EAF Bottom

`%Unknowns`	List of unknowns of equation system
`T(1)=x(1);`	Temperature of the furnace bottom.
`T(2)=x(2);`	Temperature of the walls near to the furnace.
`T(3)=x(3);`	Temperature of the floor under the furnace.
`J(1)=x(4);`	Radiosity of the furnace bottom
`J(4)=x(5);`	Radiosity of the surroundings.
`T(4)=298.15; %K`	Temperature of the surroundings.
`%Conduction`	
`Q_cond=-k_ref*A_fond*(T(1)-T_steel)/dx;`	Conduction in the bottom of the furnace.
`%Convection`	Heat transfer by convection
`k_air=0.0283; %W/(m*K)`	Thermal conductivity of the air.
`L=A_fond/P_fond;`	Characteristic length.
`g=9.81; %(m/s^2)`	Gravity acceleration.
`beta=0.0033; %(1/K)`	Thermal expansion coefficient of the air.
`nu=15.89e-6; %(m^2/s)`	Kinematic viscosity of the air.
`alpha=22.5e-6; %(m^2/s)`	Thermal diffusivity of the air.
`Ra_L=((g*beta)/(nu*alpha))*(L^3)*(T(1)-Talr);`	Rayleigh number.
`Nu_L=0.27*Ra_L^(1/4);`	Nusselt number.
`h_1=Nu_L*k_air/L;`	Convection heat transfer coefficient.

Continue next page

Continue Code

Code	Function
`Q_conv=h_1*A_fond*(T(1)-Talr);`	
`%Radiation`	Heat transfer by radiation.
`sigma=5.67e-8;`	Stefan-Boltzmann constant.
`E(1)=sigma*T(1)^4;`	Black body radiation of furnace bottom.
`E(2)=sigma*T(2)^4;`	Black body radiation of walls near to the furnace.
`E(3)=sigma*T(3)^4;`	Black body radiation of floor.
`E(4)=sigma*T(4)^4;`	Black body radiation of surroundings.
`Q_rad=(E(1)-J(1))/((1-epsilon(1))/ (epsilon(1)*A(1)));`	Heat radiated by furnace bottom.
`Q_2=0;`	Heat absorbed by the walls near to the furnace.
`J(2)=Q_2*((1-epsilon(2))/(epsilon(2)*A(2))) +E(2);`	Radiosity of the walls near to the surface.
`Q_3=0;`	Heat oabsorbed by the floor.
`J(3)=Q_3*((1-epsilon(3))/(epsilon(3)*A(3))) +E(3);`	Radiosity of the floor.
`Q_4=(J(4)-E(4))/((1-epsilon(4))/(epsilon(4) *A(4)));`	Heat radiated by the surroundings.
`Q_12=(J(1)-J(2))/(1/(A(1)*F(1,2)));`	Heat transferred by radiation from surface 1 to 2.
`Q_13=(J(1)-J(3))/(1/(A(1)*F(1,3)));`	Heat transferred by radiation from surface 1 to 3.
`Q_14=(J(1)-J(4))/(1/(A(1)*F(1,4)));`	Heat transferred by radiation from surface 1 to 4.
`Q_23=(J(2)-J(3))/(1/(A(2)*F(2,3)));`	Heat transferred by radiation from surface 2 to 3.
`Q_24=(J(2)-J(4))/(1/(A(2)*F(2,4)));`	Heat transferred by radiation from surface 2 to 4.

Continue next page

Continue Code

Code	Function
`Q_34=(J(3)-J(4))/(1/(A(3)*F(3,4)));`	Heat transferred by radiation from surface 3 to 4.
`%Equation System, EAF Bottom`	
`f(1)=Q_rad-Q_12-Q_13-Q_14;`	Heat transferred by radiation in surface 1.
`f(2)=Q_12-Q_2-Q_23-Q_24;`	Heat transferred by radiation in surface 2.
`f(3)=Q_13+Q_23-Q_3-Q_34;`	Heat transferred by radiation in surface 3.
`f(4)=Q_14+Q_24+Q_34-Q_4;`	Heat transferred by radiation in surface 4.
`f(5)=Q_cond-Q_conv-Q_rad;`	Heat transferred from the furnace bottom to the surroundings by convection and radiation.

C.6 Heat Rate Transfer EAF Fuchs Program

Code	Function
`%Program Heat Transfer Rate`	
`%Determine the heat transfer rate in the interior of the EAF Fuchs`	
`%Superficies Consideradas`	List of surfaces considered.
`%1.-Panel 1`	
`%2.-Panel 2`	
`%3.-Panel 3`	
`%4.-Panel 4`	
`%5.-Panel 5`	
`%6.-Panel 6`	
`%7.-Panel 7`	
`%8.-Panel 8`	
`%9.-Panel 9`	
`%10.-Panel 10`	
`%11.-Panel 11`	
`%12.-Panel 12`	
`%13.-Panel 13`	
`%14.-Panel 14`	
`%15.-Techo 1`	
`%16.-Techo 2`	
`%17.-Techo 3`	
`%18.-Techo 4`	
`%19.-Techo 5`	
`%20.-Techo 6`	
`%21.-Salida de gases`	
`%22.-Electrodo`	

Continue next page

Continue Code

Code	Function
%23.-Puerta de escoria	
%24.-Refractario Superior	
%25.-Refractario Inferior	
%26.-Escoria-Acero Lquido	
%27.-Arco	
%Areas	Area of the surfaces in mm^2.
A(1)=2917242;	
A(2)=2898071.361;	
A(3)=2171722.468;	
A(4)=2898072.468;	
A(5)=2898071.935;	
A(6)=2171722.200;	
A(7)=2583072.162;	
A(8)=2602242.000;	
A(9)=2118834.873;	
A(10)=2812721.458;	
A(11)=2492065.744;	
A(12)=2492065.260;	
A(13)=2812721.159;	
A(14)=2118835.655;	
A(15)=3338084.723;	
A(16)=3920802.534;	
A(17)=2735443.628;	
A(18)=2735443.514;	
A(19)=3920802.649;	
A(20)=3338084.723;	
A(21)=119361795.444;	

Continue next page

Continue Code

Code	Function
`A(22)=5162538.032;`	
`A(23)=630000;`	
`A(24)=18370636.5;`	
`A(25)=14198897.31;`	
`A(26)=28939797.375;`	
`A(27)=141371.669;`	
`A=A/(1000^2); %m^2`	Conversion of the areas from mm^2 to m^2.
`%Emisividades`	Emissivities of the surfaces.
`epsilon(1)=0.7;`	
`epsilon(2)=0.7;`	
`epsilon(3)=0.7;`	
`epsilon(4)=0.7;`	
`epsilon(5)=0.7;`	
`epsilon(6)=0.7;`	
`epsilon(7)=0.7;`	
`epsilon(8)=0.7;`	
`epsilon(9)=0.7;`	
`epsilon(10)=0.7;`	
`epsilon(11)=0.7;`	
`epsilon(12)=0.7;`	
`epsilon(13)=0.7;`	
`epsilon(14)=0.7;`	
`epsilon(15)=0.7;`	
`epsilon(16)=0.7;`	
`epsilon(17)=0.7;`	
`epsilon(18)=0.7;`	

Continue next page

Continue Code

Code	Function
`epsilon(19)=0.7;`	
`epsilon(20)=0.7;`	
`epsilon(21)=0.7;`	
`epsilon(22)=0.7;`	
`epsilon(23)=0.7;`	
`epsilon(24)=0.7;`	
`epsilon(25)=0.7;`	
`epsilon(26)=0.7;`	
`epsilon(27)=0.7;`	
`%Factores de forma`	View factors between the surfaces.
`VF=xlsread('VF.xlsx'); %Lee los factores de vista del horno`	Read the view factors from a spreadsheet.
`sigma=5.67e-8; % W/(m^2 K^4);`	Stefan-Boltzmann constant.
`%Energa emitida por el arco`	Energy emitted by the arc.
`W_dot_arc=1*(75522537.9-4531352);%W`	Electric power suministrated.
`Cob=0.2; %Cobertura del arco`	Arc coverage index.
`Q_dot_rad_arc=0.24*W_dot_arc; %Energa emitida por radiacin`	Energy rate transferred by radiation from the arc.
`Q_dot_elec_arc=0.035*W_dot_arc; %Energa transferida por electrones`	Energy rate transferred by electrons from the arc to the molten zone.
`Q_dot_conv_arcst=(Cob*0.725+0.035)*W_dot_arc; %Energa emitida por conveccion y electrones a la escoria`	Energy rate transferred from the arc to the molten zone by "convection".

Continue next page

Continue Code

Code	Function
`Q_dot_conv_arcgas=(1-Cob)*0.725*W_dot_arc;` `%Energa emitida por conveccin a los gases`	Energy rate transferred from the arc to the gases by “convection”.
`%Clculo de Coeficientes globales de transferencia de calor`	Determination of the heat transfer global coefficient of the cooling panels.
`A_panel=A(1:20);`	Area of the cooling panels.
`U=HTGC(A_panel);`	Function heat transfer global coefficient. The input variable for this function is the area of the panels.
`%Valores iniciales de las incognitas del sistema de ecuaciones`	Initials values of the unknowns of the equation system.

`%Temperatura inicial`
`T_0=zeros(1,27);`
`for i=1:27`
`T_0(i)=2000;`
`end`
`T_0(27)=11000;`
	Initial temperatures of the surfaces.

`%Radiosidad inicial`
`J_0=zeros(1,27);`
`for i=1:27`
`J_0(i)=10000;`
`end`
	Initial radiosities of the surfaces.

`%Temperatura gases` `T_g0=1500;`	Initial gas temeperature.

Continue next page

Continue Code

Code	Function

`%Temperatura panel`
`T_cold0=zeros(1,21);`
`for i=1:21`
`T_cold0(i)=1000;`
`end`
	Initial temperatures of the cooling panels.

`%Temperatura de salida panel`
`T_outw0=zeros(1,21);`
`for i=1:21`
`T_outw0(i)=300;`
`end`
	Initial outlet temperatures of the cooling water.

`T_sur0=zeros(1,2);`
`for i=1:2`
`T_sur0(i)=300;`
`end`
	Initial temperatures of the exterior surfaces of refractory line.

`x_0=[T_0,J_0,T_g0,T_cold0,T_outw0,T_sur0];`	Initial value vector.

`options=optimset('MaxFunEvals',6000,'MaxIter'`
`,1000,'Algorithm','levenberg-marquardt','TolX'`
`,1e-30);`
	Options of the algorithm used to solve the equation system.

`[y,Fval]=fsolve(@(x)(ecsrad3(x,epsilon,VF,A,`
`sigma,Q_dot_rad_arc,Q_dot_elec_arc,`
`Q_dot_conv_arcst,Q_dot_conv_arcgas,U)),`
`x_0,options);`
	Function used to solve the equation system proposed to describe the heat rate transfer inside the furnace.

C.7 Heat Rate Transfer Equation System Function

Code	Function
`%Function Equation System Heat Rate Transfer` `%Equations of the heat rate transfer model`	

`function fp=ecsrad3(x,epsilon,VF,A,sigma,`
`Q_dot_rad_arc,Q_dot_elec_arc,Q_dot_conv_arcst,`
`Q_dot_conv_arcgas,U)`
	Function where the equation system of the heat rate transfer model is described. The input variables of the function are the emissivities, the view factor, the surface areas, the Stefan-Boltzmann constant, the energy rate emitted from the arc by radiation, electrons and "convection" and the heat transfer global coefficient of the panels.

`%Definicion de variables`
`T=zeros(1,27);`
`for i=1:27`
`T(i)=x(i);`
`end`
	Unknowns considered in this equation system.
	Temperature of the surfaces.

`J=zeros(1,27);`
`for i=1:27`
`J(i)=x(i+27);`
`end`
	Radiosity of the surfaces.

`T_g=x(55);`	Temperature of the gases inside the furnace.

`T_cold=zeros(1,21);`
`for i=1:21`
`T_cold(i)=x(i+55);`
	Temperature of the cooling panels.

Continue next page

Continue Code

Code	Function
`end`	
`T_outw=zeros(1,21);`	
`for i=1:21`	
`T_outw(i)=x(i+76);`	Outlet temperature of the cooling water from the panels.
`end`	
`T_sur=zeros(1,2);`	
`for i=1:2`	
`T_sur(i)=x(i+97);`	Temperature of the exterior surface of the refractory line.
`end`	
`fp=zeros(1,99);`	
`%Datos`	Operational data of the furnace
`T_inw=300.15; %K`	Inlet temperature of the cooling water from the panels.
`h_g=4000; %W/m^2 K`	Convective heat transfer coefficient of the gases inside the furnace.
`m_dot_we=0.833; %kg/s`	Mass flow of the cooling water in the electrode.
`m_dot_wp=((900*1000)/3600)/14; %kg/s`	Mass flow of the cooling water in the wall panels.
`m_dot_wt=((500*1000)/3600)/6; %kg/s`	Mass flow of the cooling water in the roof panels.
`m_dot_wc=((200*1000)/3600); %kg/s`	Mass flow of the cooling water in the elbow.
`%DELTAx_slg=0.04; %m`	Slag layer thickness.
`k_slg=2.2; %W/m K`	Thermal conductivity of the solid slag.
`T_alr=298.15; %K`	Temperature of the surroundings.
`T_ref=298.15; %K`	Reference temperature for enthalpy calculation.

Continue next page

Continue Code

Code	Function
`DELTAx_ref=0.5; %m`	Thickness of the refractory layer.
`k_ref=2.4; %W/mK`	Thermal conductivity of the refractory brick.
`DELTAx_met=0.01109; %m`	Thickness of the metal sheel of the furnace.
`k_met=60.5; %W/m K`	Thermal conductivity of the steel.
`comp={'O2' 'CH4' 'CO' 'CO2' 'H2O' 'H2' 'N2' 'C9H20' 'Fe' 'FeO' 'Si' 'SiO2' 'Mn' 'MnO' 'C' 'Al' 'Al2O3' 'Fe2O3' 'CaO' 'MgO' 'P' 'P2O5' 'S' 'SO2' 'Cr' 'Cr2O3' 'Ti' 'TiO2' 'CaCO3' 'MgCO3' 'Mo' 'Cu' 'Zn'};`	Chemical species considered in the model.
`%Peso molecular`	Molecular weight of chemical species considered.
`PM_O2=32;`	
`PM_CH4=16;`	
`PM_CO=28;`	
`PM_CO2=44;`	
`PM_H2O=18;`	
`PM_H2=2;`	
`PM_N2=28;`	
`PM_C9H20=128;`	
`PM_Fe=55.85;`	
`PM_FeO=71.85;`	
`PM_Si=28.08;`	
`PM_SiO2=60.08;`	
`PM_Mn=54.94;`	
`PM_MnO=70.94;`	
`PM_C=12;`	
`PM_Al=26.98;`	

Continue next page

Continue Code

Code	Function
`PM_Al2O3=101.96;`	
`PM_Fe2O3=159.69;`	
`PM_CaO=56.08;`	
`PM_MgO=40.3;`	
`PM_P=30.97;`	
`PM_P2O5=141.94;`	
`PM_S=32;`	
`PM_SO2=64;`	
`PM_Cr=52;`	
`PM_Cr2O3=152;`	
`PM_Ti=47.9;`	
`PM_TiO2=79.9;`	
`PM_CaCO3=100;`	
`PM_MgCO3=84.31;`	
`PM_Mo=95.95;`	
`PM_Cu=63.5;`	
`PM_Zn=65.4;`	
`PM_c=zeros(1,33);`	
`for i=1:33`	
`pm=['PM_' cell2mat(comp(i))];`	
`PM_c(i)=eval(pm);`	Construction of the vector of molecular weight.
`end`	
`%Ecuaciones`	Equation system.
`%Circuito de Radiacin`	Radiation circuit.
`Eb=zeros(1,27);`	
`for i=1:27`	

Continue next page

Continue Code

Code	Function
`Eb(i)=sigma*(T(i)^4);` `end`	Blackbody radiation of each surface.

`for i=1:27`
`fp(i)=(Eb(i)-J(i))/((1-epsilon(i))`
`/(epsilon(i)*A(i)));`
`for k=1:27`
`fp(i)=fp(i)+(J(k)-J(i))/(1/(A(i)*VF(i,k)));`
`end`
`end`
	Radiation emitted by each surface.
	Heat rate transferred by radiation to the other surfaces.

`%Condiciones frontera`
`%Paneles enfriados por agua`
`for i=1:14`
`fp(i+27)=(J(i)-Eb(i))/((1-epsilon(i))/`
`(epsilon(i)*A(i)))+h_g*A(i)*(T_g-T(i))-(A(i)*`
`(T(i)-T_cold(i)))/(DELTAx_slg/k_slg);`
`end`
	Heat rate transfer in the wall cooling panels.
	The heat rate is received by the panels through convection and radiation and is transferred in the slag layer.

`for i=1:14`
`fp(i+41)=(A(i)*(T(i)-T_cold(i)))/(DELTAx_slg/`
`k_slg)-U(i)*A(i)*(T_cold(i)-(T_inw+T_outw(i))/2);`
`end`
	The heat rate transferred in the slag layer is received by the cooling panels.

`h_in_wp=enthalpy(T_inw)*1000/18;`
`h_out_wp=zeros(1,21);`
`for i=1:14`
	Enthalpy of the inlet cooling water in the panels in J/kg.

Continue next page

Continue Code

Code	Function
`h_out_w=enthalpy(T_outw(i))*1000/18;`	Enthalpy of the chemical species at the temperature of the outlet cooling water in J/kg.
`h_out_wp(i)=h_out_w(5);`	Enthalpy of the outlet cooling water in the panels.
`fp(i+55)=U(i)*A(i)*(T_cold(i)-(T_inw+T_outw(i))/2)+m_dot_wp*(h_in_wp(5)-h_out_wp(i));`	Energy rate balance in the cooling panels.
`end`	
`%Paneles Techo`	Heat rate transfer in the roof cooling panels.
`n=1;`	
`for i=15:20`	
`fp(n+69)=(J(i)-Eb(i))/((1-epsilon(i))/(epsilon(i)*A(i)))+h_g*A(i)*(T_g-T(i))-(A(i)*(T(i)-T_cold(i)))/(DELTAx_slg/k_slg);`	The heat rate is received by the panels through convection and radiation and is transferred in the slag layer.
`n=n+1;`	
`end`	
`n=1;`	
`for i=15:20`	
`fp(n+75)=(A(i)*(T(i)-T_cold(i)))/(DELTAx_slg/k_slg)-U(i)*A(i)*(T_cold(i)-(T_inw+T_outw(i))/2);`	The heat rate transferred in the slag layer is received by the cooling panels.
`n=n+1;`	
`end`	
`n=1;`	
`for i=15:20`	
`h_out_w=enthalpy(T_outw(i))*1000/18;`	Enthalpy of the chemical species at the temperature of the outlet cooling water in J/kg.
`h_out_wp(i)=h_out_w(5);`	Enthalpy of the outlet cooling water in the panels.

Continue next page

Continue Code

Code	Function
`fp(n+81)=U(i)*A(i)*(T_cold(i)-(T_inw+T_outw(i))/2)+m_dot_wt*(h_in_wp(5)-h_out_wp(i));`	Energy rate balance in the cooling panels.
`n=n+1;`	
`end`	
`%Salida de Gases (Codo)`	Heat rate transfer in the elbow.
`A_codo=2.84+2.6+2.030; %Area del sistema de enfriamiento del codo`	Area of the elbow cooling system.
`fp(88)=(J(21)-Eb(21))/((1-epsilon(21))/(epsilon(21)*A(21)))+h_g*A(21)*(T_g-T(21))-(A_codo*(T(21)-T_cold(21)))/(DELTAx_slg/k_slg);`	The heat rate is received by the panels through convection and radiation and is transferred in the slag layer.
`fp(89)=(A_codo*(T(21)-T_cold(21)))/(DELTAx_slg/k_slg)-U(21)*A_codo*(T_cold(21)-(T_inw+T_outw(21))/2);`	The heat rate transferred in the slag layer is received by the cooling panels.
`h_out_w=enthalpy(T_outw(21))*1000/18;`	Enthalpy of the chemical species at the temperature of the outlet cooling water in J/kg.
`h_out_wp(21)=h_out_w(5);`	Enthalpy of the outlet cooling water in the panels.
`fp(90)=U(21)*A(21)*(T_cold(21)-(T_inw+T_outw(21))/2)+m_dot_wc*(h_in_wp(5)-h_out_wp(21));`	Energy rate balance in the cooling panels.
`%Electrodo`	Heat rate transfer in the electrode.
`T_in_e=298.15;%K`	Inlet temperature of the electrode cooling water.
`h_in_e=enthalpy(T_in_e)*1000/18;`	Enthalpy of the chemical species at the temperature of the inlet cooling water in the electrode.
`h_out_e=enthalpy(T_g)*1000/18;`	Enthalpy of the chemical species at the temperature of the outlet cooling water in the electrode.

Continue next page

Continue Code

Code	Function
`h_in_el=h_in_e(5);`	Enthalpy of the inlet cooling water in the electrode.
`h_out_el=h_out_e(5);`	Enthalpy of the outlet cooling water in the electrode.
`fp(91)=(J(22)-Eb(22))/((1-epsilon(22))/ (epsilon(22)*A(22)))+h_g*A(22)*(T_g-T(22))+ m_dot_we*(h_in_el-h_out_el);`	Heat rate received by the electrode through radiation and convection and transferred to the cooling water of the electrode.
`%Puerta de escoria`	Heat rate tranfer in the slag door.
`fp(92)=(J(23)-Eb(23))/((1-epsilon(23))/ (epsilon(23)*A(23)))-sigma*A(23)* (T(23)^4-T_alr^4);`	The heat rate received by the slag door through radiation is dissipated to the surroundings.
`%Refractario superior e inferior`	Heat rate transfer in refractory line.
`U_ref=1/(DELTAx_slg/k_slg+DELTAx_ref/k_ref+ DELTAx_met/k_met);`	Heat transfer global coefficient in the refractory line.
`n=1; for i=24:25 fp(n+92)=(J(i)-Eb(i))/((1-epsilon(i))/ (epsilon(i)*A(i)))+h_g*A(i)*(T_g-T(i))-U_ref* A(i)*(T(i)-T_sur(n)); n=n+1; end`	The heat rate is received by the refractory line through convection and radiation, and is transferred in through the layers of the refractory line.
`h_air=20; %W/m^2 K n=1; for i=24:25`	Convective heat transfer coefficient of the air outside the furnace.

Continue next page

Continue Code

Code	Function
`fp(n+94)=U_ref*A(i)*(T(i)-T_sur(n))-A(i)*h_air*(T_sur(n)-T_alr)-A(i)*sigma*((T_sur(n)^4)-(T_alr^4));`	The heat rate received by the refractory line is dissipated to the surroundings through convection.
`n=n+1;`	
`end`	
`%Balance Escoria-Acero lquido`	Energy rate balance in the molten zone.
`m_steel=150*1000; %kg`	Mass of steel in the molten zone.
`m_slag=10*1000; %kg`	Mass of slag in the molten zone.
`m_dot_slag=0.00833*1000; %kg/s`	Outlet mass flow of slag in the molten zone.
`dt=1;%s`	Time.
`c_steel=[0,0,0,0,0,0,0,0,0.965,0.02,0,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];`	Composition of steel in mass fraction.
`c_slag=[0,0,0,0,0,0,0,0,0,0.37,0,0.15,0,0.017,0,0,0.05,0,0.21,0.102,0,0,0.0003,0,0,0,0,0,0,0,0,0,0];`	Composition of slag in mass fraction.
`h_fin_steel=1000*(enthalpy(T(26))-enthalpy(T_ref))./PM_c;`	Final enthalpy of the steel.
`h_in_steel=1000*(enthalpy(1815)-enthalpy(T_ref))./PM_c;`	Initial enthalpy of the steel.
`h_fin_slag=1000*(enthalpy(T(26))-enthalpy(T_ref))./PM_c;`	Final enthalpy of the slag.
`h_in_slag=1000*(enthalpy(1815)-enthalpy(T_ref))./PM_c;`	Initial enthalpy of the slag.
`Q_dot_fondo=FondoHorno(T(26));`	Heat rate trasferred in the bottom of the furnace.
`Q_dot_comb=8458.33; %W`	Heat rate supplied by combustion of carbon.
`Q_dot_ox=1375; %W`	Heat rate supplied by oxidation of iron.

Continue next page

Continue Code

Code	Function
`fp(97)=(J(26)-Eb(26))/((1-epsilon(26))/ (epsilon(26)*A(26)))+h_g*A(26)*(T_g-T(26))+ Q_dot_conv_arcst+Q_dot_elec_arc+Q_dot_comb+Q_dot_ox-Q_dot_fondo+ (m_steel*sum(c_steel.*h_in_steel)-m_steel*sum (c_steel.*h_fin_steel)+m_slag*sum(c_slag.* h_in_slag)-m_slag*sum(c_slag.*h_fin_slag))/dt- m_dot_slag*sum(c_slag.*h_fin_slag);`	Energy rate balance in the molten zone.
`%Arco`	Electric arc
`fp(98)=(Eb(27)-J(27))/((1-epsilon(27))/ (epsilon(27)*A(27)))-Q_dot_rad_arc;`	Heat rate emitted through radiation by the electric arc.
`%Balance de energa gases`	Energy rate balance in the gases.
`m_dot_og=16.17; %kg/s`	Mass flow of off gases.
`m_gas=0.0151*1000; %kg`	Mass of gases inside the furnace
`c_og=[0.0338,0,0.1482,0.1397,0.0444,0.001411, 0.6323,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];`	Composition of the gases inside the furnace.
`h_init_og=1000*(enthalpy(1400)- enthalpy(T_ref))./PM_c;`	Initial enthalpy of the gases.
`h_fin_og=1000*(enthalpy(T_g)- enthalpy(T_ref))./PM_c;`	Final enthalpy of the gases.
`Q_dot_gn=2625; %W`	Heat rate supplied by the combustion of natural gas.

Continue next page

Continue Code

Code	Function
`fp(99)=Q_dot_conv_arcgas+Q_dot_gn-m_dot_og*sum (c_og.*h_fin_og)+(m_gas*sum(c_og.*h_init_og)- m_gas*sum(c_og.*h_fin_og))/dt;`	Energy balance in the gases.
`for i=1:22`	
`fp(99)=fp(99)+h_g*A(i)*(T(i)-T_g);`	Heat rate transferred through convection to the panels added to the energy balance in the gases.
`end`	
`for i=24:26`	
`fp(99)=fp(99)+h_g*A(i)*(T(i)-T_g);`	Heat rate transferred through convection to the refractory line added to the energy balance in the gases.

C.8 Heat Rate Transfer Equation System Function

Code	Function
`%Function Global Heat Transfer Coefficient` `%Determination of the global heat transfer coefficient of the cooling panels.`	
`function U=HTGC(A)`	Function used to calculate the global heat transfer coefficient of the cooling panels.
`V_dot_coraza=900/3600; %m^3/s`	Volumetric flow of water in the wall cooling panels.
`V_dot_domo=500/3600; %m^3/s`	Volumetric flow of water in the roof cooling panels.
`V_dot_codo=200/3600; %m^3/s`	Volumetric flow of water in the elbow cooling panels.
`d_it=0.0667; %m`	Internal diameter of the pipes.
`rugosity=0.045e-3; %m`	Rugosity of the pipes.
`A_it=(pi*d_it^2)/4;`	Internal area of the pipes in m^2.
`rho_w=1000; %kg/s`	Density of the water.
`mu_w=1e-3; %Pa s`	Dynamic viscosity of the water.
`Cp_w=4184; %J/kg K`	Specific heat of water.
`k_w=0.591;`	Thermal conductivity of water.
`%Velocidad agua coraza`	Velocity of water inside the wall cooling panels.
`for i=1:14`	
`V_pn(i)=(V_dot_coraza/14)/A_it;`	Velocity of cooling water inside the pipes.
`Re_pn(i)=(rho_w*V_pn(i)*d_it)/mu_w;`	Reynolds number of water flow inside the pipes.
`end`	

Continue next page

Continue Code

Code	Function
`%Factor de friccin paneles coraza`	Friction factor of wall cooling panels.
`for i=1:14`	
`f_0(i)=0.5;`	
`f_pn(i)=fsolve(@(f)(colebrook(f,Re_pn(i),rugosity,d_it)),f_0(i));`	Solution of colebrook equation to obtain the friction factor of each wall cooling panel.
`end`	
`%Velocidad agua domo`	Velocity of water inside the roof cooling panels.
`for i=1:6`	
`V_domo(i)=(V_dot_domo/6)/A_it;`	Velocity of cooling water inside the pipes.
`Re_domo(i)=(rho_w*V_domo(i)*d_it)/mu_w;`	Reynolds number of water flow inside the pipes.
`end`	
`%Factor de friccin paneles domo`	Friction factor of roof cooling panels.
`for i=1:6`	
`f_0(i)=0.5;`	
`f_domo(i)=fsolve(@(f)(colebrook(f,Re_domo(i),rugosity,d_it)),f_0(i));`	Solution of colebrook equation to obtain the friction factor of each roof cooling panel.
`end`	
`%Velocidad agua codo`	Velocity of water inside the elbow cooling panels.
`V_codo=V_dot_codo/A_it;`	Velocity of cooling water inside the pipes.
`Re_codo=(rho_w*V_codo*d_it)/mu_w;`	Reynolds number of water flow inside the pipes.
`%Factor de friccin`	
`f_0=0.5;`	
`f_codo=fsolve(@(f)(colebrook(f,Re_codo,rugosity,d_it)),f_0);`	Solution of colebrook equation to obtain the friction factor of each elbow cooling panel.

Continue next page

Continue Code

Code	Function
`%Areas` `A(21)=2.84+2.6+2.030;`	Area of the elbow cooling system.
`%Longitud Tubos` `L_T=A/0.0675;`	Length of the pipes.
`Pr_w=(Cp_w*mu_w)/k_w;`	Prandtl number of the water

`%Coeficiente transferencia de calor por conveccin agua`
`for i=1:14`
`h_pn(i)=(k_w/d_it)*((f_pn(i)/8)*(Re_pn(i)-1000)*Pr_w)/(1+12.7*((f_pn(i)/8)^0.5)*(Pr_w^(2/3)-1));`
`end`
	Convective heat transfer coefficient for the water flow inside the pipe.
	Gnielinski correlation for convective heat transfer coefficient for a fluid flow inside the pipe in the wall cooling panels.

`for i=1:6`
`h_domo(i)=(k_w/d_it)*((f_domo(i)/8)*(Re_domo(i)-1000)*Pr_w)/(1+12.7*((f_domo(i)/8)^0.5)*(Pr_w^(2/3)-1));`
`end`
	Gnielinski correlation for convective heat transfer coefficient for a fluid flow inside the pipe in the roof cooling panels.

`h_codo=(k_w/d_it)*((f_codo/8)*(Re_codo-1000)*Pr_w)/(1+12.7*((f_codo/8)^0.5)*(Pr_w^(2/3)-1));`	Gnielinski correlation for convective heat transfer coefficient for a fluid flow inside the pipe in the elbow cooling panels.
`k_met=60.5; %W/m K` `DELTAx_met=0.01109; %m`	Thermal conductivity of steel. Thickness of the pipes.

Continue next page

Continue Code

Code	Function

`for i=1:14`
`U_pn(i)=1/(DELTAx_met/k_met+A(i)/(h_pn(i)*pi*d_it*L_T(i)));`
`end`
	Global heat transfer coefficient of wall cooling panels.

`for i=1:6`
`U_domo(i)=1/(DELTAx_met/k_met+A(i+14)/(h_domo(i)*pi*d_it*L_T(i+14)));`
`end`
	Global heat transfer coefficient of roof cooling panels.

`U_codo=1/(DELTAx_met/k_met+A(21)/(h_codo*pi*d_it*L_T(21)));`	Global heat transfer coefficient of elbow cooling panels.
`U=[U_pn U_domo U_codo];`	Vector of global heat transfer coefficient of all cooling panels.

`end`
`function fp=colebrook(f,Re,rugosity,d_it)`
`fp=(1/(f^0.5))+2*log10(2.51/(Re*(f^0.5))+rugosity/(3.7*d_it));`
	Function used to solve the colebrook equation to obtain the friction factors.

Bibliography

[1] ARNOUT, S., VERHAEGHE, F., BLANPAIN, B., WOLLANTS, P., HENDRICKX, R., AND HEYLEN, G. A Thermodynamic Model of the EAF Process for Stainless Steel. *Steel Research International 77*, 5 (may 2006), 317–323.

[2] BEKKER, J. G., CRAIG, I. K., AND PISTORIUS, P. C. Modeling and simulation of an electric arc furnace process. *ISIJ INTERNATIONAL 39*, 1 (1999), 23–32.

[3] ÇAMDALI, Ü., AND TUNÇ, M. Exergy analysis and efficiency in an industrial AC electric ARC furnace. *Applied Thermal Engineering 23*, 17 (2003), 2255–2267.

[4] CENGEL, Y. A. *Heat & Mass Transfer: A Practical Approach*, third ed. McGraw-Hill, 2007.

[5] CHEREMISINOFF, N. P. *Handbook of Air Pollution and Control*. Elsevier, 2002.

[6] CONTRERAS SERNA, J. *Study of heat transfer in tubular-panel and spray cooling systems applied to the electric arc furnace walls*. PhD thesis, Instituto Tecnológico y de Estudios Superiores de Monterrey, 2018.

[7] CROWE, C. M., GARCIA CAMPOS, Y. A., AND HRYMAK, A. Reconciliation of Process Flow Rates by Matrix Projection. *AICHE JOURNAL 29*, 6 (1983), 881–888.

[8] EKMEKÇI, S., YETISKEN, Y., AND ÇAMDALI, Ü. Mass Balance Modeling for Electric Arc Furnace and Ladle Furnace System in Steelmaking Facility in Turkey. *Journal of Iron and Steel Research, International 14*, 5 (sep 2007), 1–55.

[9] ELSHENNAWY, A. K., AND WEHEBA, G. S. *Manufacturing Processes and Materials*, 5th ed. Society of Manufacturing Engineers (SME), 2015.

[10] FELDER, R. M., AND ROUSSEAU, R. W. *Elementary Principles of Chemical Processes*. Wiley series in chemical engineering. John Wiley, 2000.

[11] GREEN, D. W., AND PERRY, R. H. *Perry's Chemical Engineers' Handbook, Eight Edition*. McGraw Hill professional. McGraw-Hill Education, 2007.

[12] GRUBER, J. C., ECHTERHOF, T., AND PFEIFER, H. Investigation on the Influence of the Arc Region on Heat and Mass Transport in an EAF Freeboard using Numerical Modeling. *STEEL RESEARCH INTERNATIONAL 87*, 1 (2016), 15–28.

[13] GUO, D., AND IRONS, G. A. Modeling of Radiation Intensity in an EAF. In *Proceedings of the Third International Conference on CFD in the Minerals and Process Industries* (Melbourne, Australia, 2003), CSIRO, pp. 223–228.

[14] HOLMAN, J. P. *Heat Transfer*, 10th ed. 2010.

[15] INCROPERA, F. P., AND DEWITT, D. P. *Fundamentals of Heat and Mass Transfer*, 4th ed. Wiley, 1999.

[16] KELLEY, K. K. Contributions to the Data on Theoretical Metallurgy. Tech. rep., United States Goverment Printing Office Washington, Washington D.C., 1960.

[17] KIRSCHEN, M., BADR, K., AND PFEIFER, H. Influence of direct reduced iron on the energy balance of the electric arc furnace in steel industry. *Energy 36*, 10 (2011), 6146–6155.

[18] LOGAR, V., DOVZAN, D., AND SKRJANC, I. Mathematical Modeling and Experimental Validation of an Electric Arc Furnace. *ISIJ INTERNATIONAL 51*, 3 (2011), 382–391.

[19] LOGAR, V., DOVZAN, D., AND SKRJANC, I. Modeling and Validation of an Electric Arc Furnace: Part 1, Heat and Mass Transfer. *ISIJ INTERNATIONAL 52*, 3 (2012), 402–412.

[20] LOGAR, V., DOVZAN, D., AND SKRJANC, I. Modeling and Validation of an Electric Arc Furnace: Part 2, Thermo-chemistry. *ISIJ INTERNATIONAL 52*, 3 (2012), 413–423.

[21] LOGAR, V., FATHI, A., AND SKRJANC, I. A Computational Model for Heat Transfer Coefficient Estimation in Electric Arc Furnace. *STEEL RESEARCH INTERNATIONAL 87*, 3 (mar 2016), 330–338.

[22] LOGAR, V., AND SKRJANC, I. Development of an Electric Arc Furnace Simulator Considering Thermal, Chemical and Electrical Aspects. *ISIJ INTERNATIONAL 52*, 10 (2012), 1924–1926.

[23] LOGAR, V., AND SKRJANC, I. Modeling and Validation of the Radiative Heat Transfer in an Electric Arc Furnace. *ISIJ INTERNATIONAL 52*, 7 (2012), 1225–1232.

[24] MACROSTY, R. D. M., AND SWARTZ, C. L. E. Dynamic modeling of an industrial electric arc furnace. *INDUSTRIAL & ENGINEERING CHEMISTRY RESEARCH 44*, 21 (oct 2005), 8067–8083.

[25] MORALES, R. D., RODRIGUEZ-HERNANDEZ, H., AND CONEJO, A. N. A mathematical simulator for the EAF steelmaking process using direct reduced iron. *ISIJ INTERNATIONAL 41*, 5 (2001), 426–435.

[26] MOUSTAFA, Y. E. *Modeling, Optimization and Estimation in Electric Arc Furnace (EAF) Operation*. PhD thesis, McMaster University, 2013.

[27] OPITZ, F., AND TREFFINGER, P. Physics-Based Modeling of Electric Operation, Heat Transfer, and Scrap Melting in an AC Electric Arc Furnace. *METALLURGICAL AND MATERIALS TRANSACTIONS B-PROCESS METALLURGY AND MATERIALS PROCESSING SCIENCE 47*, 2 (2016), 1489–1503.

[28] SANCHEZ, J. L. G., RAMIREZ-ARGAEZ, M. A., AND CONEJO, A. N. Power Delivery from the Arc in AC Electric Arc Furnaces with Different Gas Atmospheres. *STEEL RESEARCH INTERNATIONAL 80*, 2 (feb 2009), 113–120.

[29] SZEGA, M. Extended applications of the advanced data validation and reconciliation method in studies of energy conversion processes. *ENERGY 161*, 1 (2018), 156–171.

[30] TREJO, E. *Mathematical Models in Electric Arc Furnaces*. PhD thesis, Instituto Tecnológico y de Estudios Superiores de Monterrey, 2012.

[31] TREJO, E., MARTELL, F., MICHELOUD, O., TENG, L., LLAMAS, A., AND MONTESINOS-CASTELLANOS, A. A novel estimation of electrical and cooling losses in electric arc furnaces. *ENERGY 42*, 1 (jun 2012), 446–456.

[32] TUNC, M., CAMDALI, U., AND ARASIL, G. Energy Analysis of the Operation of an Electric-Arc Furnace at a Steel Company in Turkey. *Metallurgist 59*, 5 (2015), 489–497.

[33] TUPKARY R. H., T. V. R. *Modern Steel Making Handbook*. Mercury Learning and Information, 2018.

[34] WORLD STEEL ASSOCIATION. Steel Statistical Yearbook 2017. Tech. rep., Brussels, 2017.

Curriculum Vitae

Armando Torres Vázquez was born in Torreón, Coahuila in 1992, but raised in Durango, Durango. He received his B. S. in Chemical Engineering with Honorific Mention from the Instituto Tecnológico de Durango in June 2015. Currently, he is pursuing his MSc. In Energy Engineering at the Instituto Tecnológico y de Estudios Superiores de Monterrey under the sponsorship of SENER-CONACYT.

This document was typed in using LaTeX 2_ε[a] by Armando Torres Vázquez.

[a]The style file `phdThesisFormat.sty` used to set up this thesis was prepared by the Center of Intelligent Systems of the Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey Campus