



Development of charge calculation program for target steel in induction furnace

Saliu Ojo SEIDU* and Adetunji ONIGBAJUMO

Department of Metallurgical and Materials Engineering, Federal University of Technology, Akure, Ondo State. P.M.B 704

E-Mails: *seidu2@yahoo.co.uk, ebunoluwabukola@gmail.com

*Corresponding author, phone: +2347088277396

Abstract

This paper presents the development of charge calculation program for target steel in induction furnace. The simulation modelling function developed is based on mass balance analysis of the furnace production. The process engineering of the furnace follows linear algebraic mathematical function. Visual basic programming language (C#) is used in the coding and interface integration. This is used to develop a unit process based simulation program with user friendly interface for the furnace. The application could be adapted to the production of different alloy steel depending on the production standard set by the user. Also, the program is developed to calculate the mass of scrap for optimization, ferrosilicon, ferromanganese, and other additives. Iteration of scrap charge for optimization is incorporated to enable the user simulates changes and manipulates scrap charge in the furnace before ferro-alloys and carbon additives are charged depending on the foundry practice or target standard. This also helps in the decision of the furnace engineer while requesting scrap from the yard. On validation, the program was seen to give charge optimization result very close in value to standard charge rate of the integrated steel complex in which it was tested.

Keywords

Charge calculation; Induction furnace; Simulation modelling; Material balance; Process engineering; Alloy steel; Scrap; Optimization; Foundry

Introduction

Steel scrap is the most important raw-material in secondary steel making shop. It contributes between 60% and 80% of the total production costs [1]. Scrap is the major raw material for steel production in the induction furnace. With the addition of ferro-alloys different grades of alloy steel can be produced from the total scrap meltdown. However, the degree at which the scrap mix can be optimized and which melting operation can be controlled and automated to achieve the right chemistry of the melt is limited [2]. This importantly depends on the knowledge of the properties of the scrap, raw-materials in the charge mix, proper charge calculation and good scrap sorting and selection.

An optimal and constant production process in secondary steel making is only possible by controlling the metallurgical process within close limits [3]. Operational recovery value obtained from control and monitored production assist in the complex relation between the additions of slag formers. The quality of steel produced is governed by well-established metallurgical equations [4], which are a function of the process engineering (unit process and operation) of the used furnace. This therefore necessitates the usage of computers to achieve the optimal result in real time and save production downtime during melting. This optimization is especially important with regard to the constantly increasing demands on the quality of the final alloy steel [5]. Depending on the quality of the input materials and the melting practice in place at the foundry shop, it's often a difficult task to achieve [6]. Different production data requires different charge calculation effort [7]. In the Induction Furnace, the quality of alloy steel produced largely depends on the charge mix, quality of scrap and additives and effective optimization model. This work is used to develop charge calculation program for target steel in Induction Furnace. The input operating parameters, such as the amount and compositions of charged raw materials and the initial heel which are required to simulate the working operation were in line with [8]. This serves as the preliminary guide into the optimization template development for the program.

Materials and method

Methodology (Algorithm model) of the charge calculation and program took into account:

- Stoichiometric reactions of the molten steel, oxygen, carbon and alloy element addition.
- Target standard for construction steel grade [Medium Carbon-Low Alloy Steel].
- Spectrometric analysis from the Quality Control laboratory as the initial analytical basis for the scrap meltdown elemental composition which will be used for the program calculation.
- Designation of mathematical deterministic models to relate stoichiometric functions of increase and decrease in mass/weight percent of major elements; [C, Si, Mn, P, and Fe] [9, 10].
- Operational recovery as a function of the furnace shop melts practice.
- Elemental recovery of the charged elements during optimization.
- Development of mathematical linear and algebraic relationship on mass/material balance based on unit processes/operation of Induction Furnace as well as study analysis of equilibrium conditions of the metallurgical thermodynamics using standard conversion factors [11].
- Use program language (VB.net on Microsoft visual studio C#.net coding) and design simple and user friendly simulation interface to perform iterations.

Deterministic model for charge in Induction Furnace

An algorithm was implemented (TROS), by using the followings:

- The chemical constitution of scrap, lime and Ferro-alloy
- The resultant effect of such addition in the overall chemistry of the steel
- Lime in = lime out

Constant functions

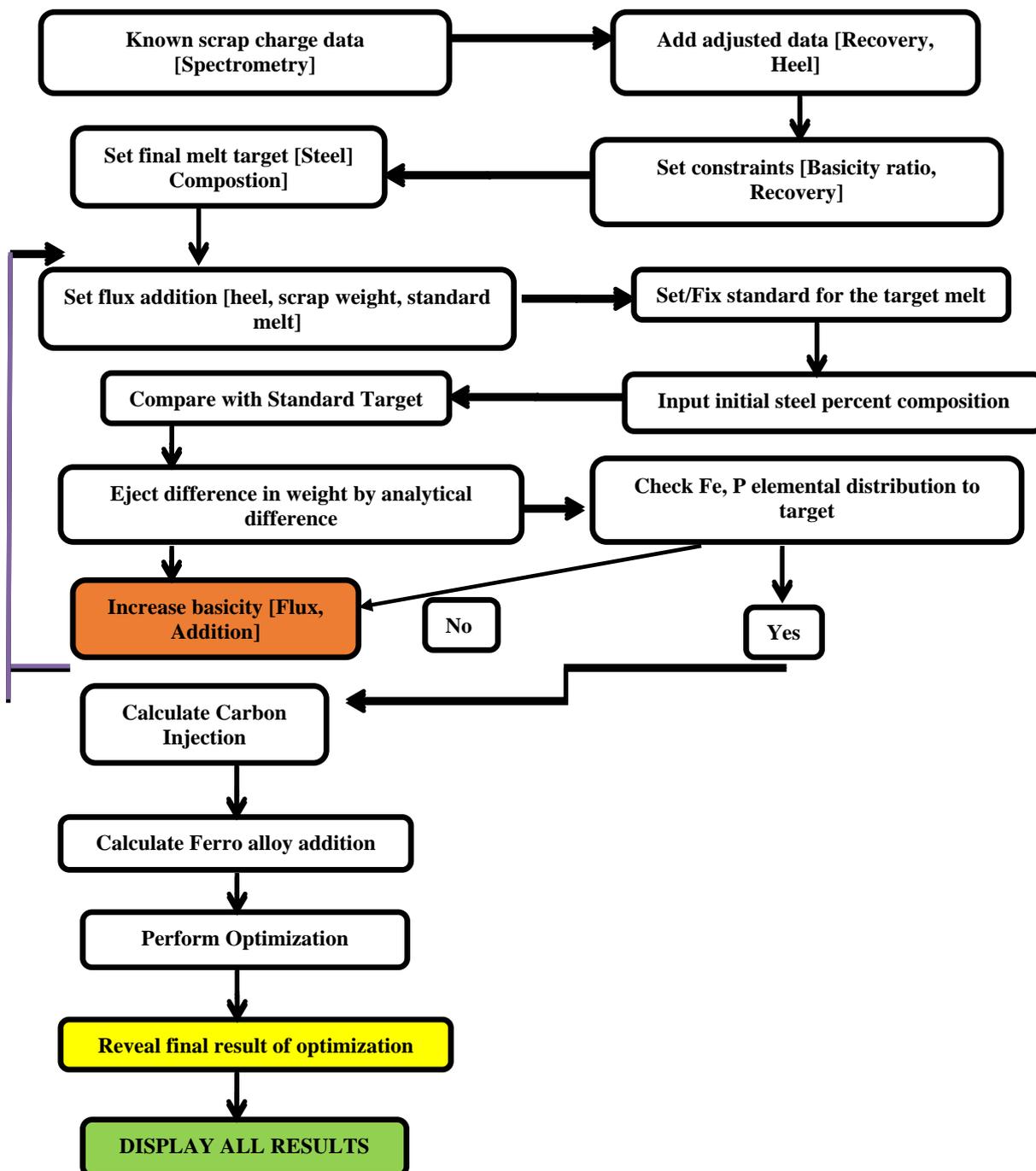
- Size/capacity of furnace
- Elemental standard composition e.g. (Standard Organization of Nigeria - SON, British Standard – BS) for Alloy (low, medium or high) steel.

Variables

- Mass of scrap charge
- Chemical composition of scrap charge (Fe, Mn, Si, P, S, C)
- Mass of slag

- Mass of Ferro-alloy
- Composition of Ferro-alloy

Algorithm for the program development (TROS)



Implemented algorithm (TROS is the application code name for the developed program)

To determine the excess/deficiency of an element ‘x’ relative to the target/aim



standard is expected elemental ladle composition,

Let the mass of the charged scrap = M_{st}

The recovery function in line with operational practice of scrap melt = R^i

Let the % composition by weight of element x from spectrometer analysis = x^i

Let the % composition by weight of element x from standard composition/target = x^o

Let the % composition by weight of element x from additional charge (alloy) = x^n

Let the mass of additional Scrap = M_{sc}

Let the mass of additional Ferro alloy = M_{fa}

The % composition of element X from alloy addition = x^m

- the following were derived mathematically, the expected/resultant % composition of element x when additional scrap is added (where R^o represents the recovery of element 'x' in the melt down):

$$\% x^o = \frac{(M_{st} X x^i) + (M_{sc} X x^n X R^o)}{(M_{st} + sc) X 100\%}$$

- the resultant % composition of element X when Ferro-alloy is added (*provided that the Ferro-alloy contains element x):

$$\% x^o = \frac{(R^i \cdot M_{st} X x^i) + (M_{fa} X x^m X R^o)}{(M_{st} + M_{fa}) X 100\%}$$

- if the two additions were made i.e. (i and ii), overall resultant composition will be:

$$\% x^o = \frac{(R^i \cdot M_{st} X x^i) + (M_{sc} X x^n X R^o) + (M_{fa} X x^m X R^o)}{(M_{st} + M_{fa} + M_{sc}) X 100\%}$$

The mass of scrap to be added to achieve the target aim composition was obtained from the equation:

$$\begin{aligned} \% x^o &= \frac{(M_{st} X x^i) + (M_{sc} X x^n X R^o)}{(M_{st} + M_{sc}) X 100\%} \\ x^o (M_{st} + M_{sc}) &= (R^i M_{st} X x^i) + (M_{sc} X x^n X R^o) \\ (M_{st} x^o + M_{sc} x^o) &= (R^i M_{st} X x^i) + (M_{sc} X x^n X R^o) \\ (M_{sc} x^n X R^o + M_{sc} x^o) &= (R^i M_{st} x^i - M_{st} x^o) \end{aligned}$$

therefore,

$$M_{sc} = \frac{M_{st} (x^i R^i - x^o)}{(x^o - x^n R^o)}$$

where M_{sc} represents mass of scrap that must be added.

Therefore the mass of Ferro-alloy to be added could also follow the equation:

$$M_{fa} = \frac{R^i M_{st} (x^i - x^o)}{(x^o - x^m R^o)}$$

- From the above equation, the mass of Ferro-alloy when scrap has been added and its resultant effect upon the whole melt was deduced.

By manipulating equations given above, the result becomes

$$M_{fa} = \frac{M_{st} (R^i x^i - x^o) + M_{sc} (x^n + R^o - x^o)}{(x^o - x^m R^o)}$$

Developed Simulation Interface (DSI)

- (WTI – Work Tool Interface)

Figure 1. TROS 1st WTI – (standard input, scrap weight input, heel, basicity, recovery and A - Sample spectrometric analysis input)

The 1st WTI is the input interface on the charge calculation program where initial charge data are input by the user.

- **SON/Standard Setter:** this is the upper portion of the interface where the user will set the Program up according to the prevailing Target Standard of the final melt of the major elements in the produced steel
- **i and j Scrap:** are scrap input tabs which allow the user input the initial mass of scrap – i and the final mass of scrap – j into the program. The i-scrap module is calculated according to the available operational recovery practice which will be set in the Recovery

tab before the j-scrap.

- Melt:** this portion of the interface is also set by the user according to the initial spectrometric test result (percentage) obtained from the metal analysis when the whole scrap has fully turn into melt. The program automatically converts this into weight composition of the constituent element.

The 2nd WTI reveals the difference between the initial composition of the constituent element in the furnace melt and the Actual weight composition by calculating based on the Target standard input initially by the User on the 1st WTI.



Figure 2. Tros 2nd WTI - (difference in theoretical target and A-Sample result from 1st WTI, flux charge input calculation)

The target portion of the interface gives the weight composition of the Standard while the difference section reveals the Loss or Gain as a result of the difference between the two weight compositions.

Flux additions (Lime & Dolomite) is calculated based on the relation [8]:

$$Q_{sl} = (Q_{sl}^r + Q_{oxyd}^{ch}) / (1 - (Fe)_{sl})$$

where:

Q_{sl} = quantity of slag, Q_{sl}^r = quantity of slag heel, Q_{oxyd}^{ch} = quantity of slag from oxidation, $(Fe)_{sl}$ = Fe content in the slag.

Note that is Q_{sl} a function of the liquid slag composition Q_{oxyd}^{ch} ($Q_{oxyd}^{ch} = Q_{ch}^m - Q^{st}$, Q_{ch}^m = Quantity of slag element in meltdown, Q^{st} = Quantity of element in standard composition).

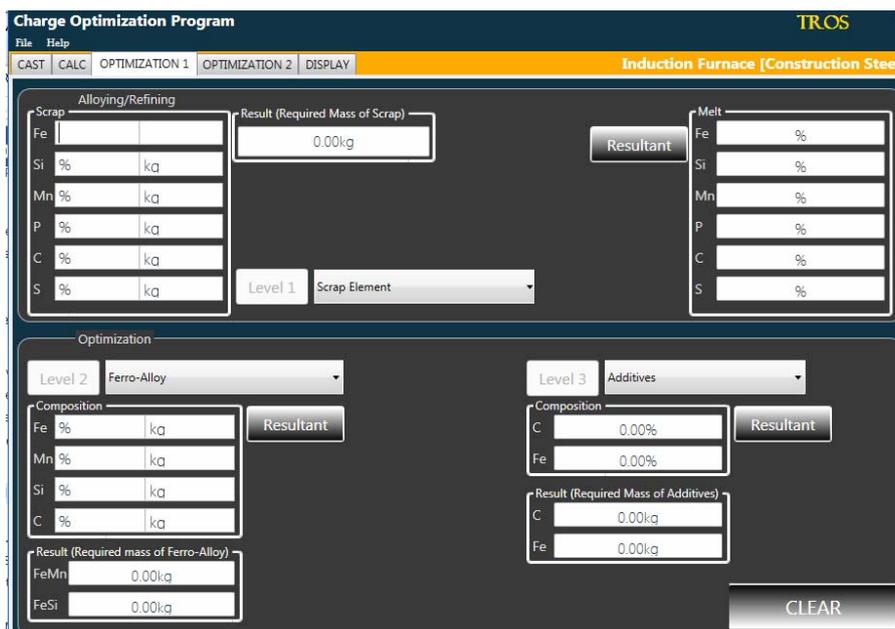


Figure 3. Tros 3rd WTI (scrap optimization, Ferro-alloy addition and other additives resultant effects of addition)

The 3rd WTI allows the User to perform optimization by calculating the amount of Scrap, ferro-silicon, ferro-manganese, carbon and mill scale (optional). It enable the user to input necessary value of scrap, ferro-alloy elemental percentage compositions derived from scrap analysis and manufacturers quote respectively.

This Interface works according to the following relations:

$$M_{sc} = \frac{M_{st} (x^l R^l - x^o)}{(x^o - x^n R^o)}, \quad M_{fa} = \frac{M_{st} (R^l x^l - x^o) + M_{sc} (x^n + R^o - x^o)}{(x^o - x^m R^o)}$$

where M_{fa} is the Mass of ferro-alloy addition and M_{sc} represents mass of scrap that must be added.

The 4th WTI allows the User perform scrap iteration by inputing scrap percentage compositions of different scrap available or selected from the scrap yard. The program calculates the resultant effect of such addition in the overall melt and the new melt composition is revealed. The user can either alter the percentage compositions or the Mass of scrap intended to charge and such additions are calculated as the program reveals such iteration according to the relation:

$$\% x^o \sum ni - n3 = \frac{(M_{st} X x^l) + (M_{sc} X x^n X R^o)}{(M_{st} + M_{sc}) X 100\%}$$

where each symbols have their usual meaning.

$ni - n3 =$ is the sum of the 3 simultaneous iterations on the WTI.

Scrap (Kg)	Resultant	Scrap (Kg)	Resultant	Scrap (Kg)	Resultant
0.00kg		0.00kg		0.00kg	

Scrap 1	Scrap 2	Scrap 3
Fe	Fe % kg	Fe % kg
Si % kg	Si % kg	Si % kg
Mn % kg	Mn % kg	Mn % kg
P % kg	P % kg	P % kg
C % kg	C % kg	C % kg
S % kg	S % kg	S % kg

Melt
Fe %
Si %
Mn %
P %
C %
S %

Figure 4. Tros 4th WTI (scrap charge iteration and resultant effect of addition)

Initial Scrap	Optimization 1	Optimization 2
Fe % kg	Fe	Fe
Si % kg	Si	Si
Mn % kg	Mn	Mn
P % kg	P	P
C % kg	C	C
S % kg	S	S

Required Mass of Scrap in optimization1: _____

Mass of scrap in optimization 2: _____

Ferro-Manganese FeMn: _____

Mass of Lime: _____

Ferro-Silicon FeSi: _____

Mass of flux: _____

Required Carbon Additive: _____

Mass of Dolomite: _____

Required Mill scale Additive: _____

Total Mass of Scrap (Optimization 1): _____

Mass of Heel: _____

Total Mass of Scrap (Optimization 2): _____

Figure 5. Tros 5th WTI (display of overall results from 1st - 4th WTI)

The 5th WTI displays the overall results of all the various results obtained from flux addition, Optimization, ferro-alloy addition, scrap iterations and total mass of scrap charged to obtain the Target melt.

Results and discussion

Comparison of additive charge by Thumb Rule (Guess) and the developed simulation program

Table 1. Comparison of the developed simulation with the integrated steel complex additive charge (Thumb Rule/Guess) – Production Data Source: Universal Steel Ltd. Lagos, Nigeria

Day	FeMn*	FeMn**	FeSi*	FeSi**	Coke*	Coke**
28/08	Not released	357.323	Not released	100.08	Not released	129.9
30/08	Not released	327.44	Not released	91.972	Not released	116.2
31/08	Not released	335.75	Not released	93.98	Not released	119.1
31/08	Not released	323.46	Not released	90.54	Not released	120.12
Total (Kg)	1130	1343.973	295	376.572	301.58	485.32
Total Weight of Melt	96.04	96.04	96.04	96.04	96.04	96.04
Rate (Kg/T)	11.77	13.99	3.07	3.99	3.14	5.05
Standard (Kg/T)	14		4.15		5	

*[Guess/Thumb rule]; ** [Program Result]

The table above shows the result of the use of the model in an integrated steel complex in Lagos, South-West, Nigeria. The comparison was made between the charge calculation program and the steel melting shop “thumb rule/guess practice. The basis of functionality is dependent upon the charge rate stipulated by the company’s standard practice in line with Standard Organization of Nigeria (SON). Although the Steel Melting shop did not release the daily additive charge data (ferro-silicon, ferro-manganese, carbon), the weekly production charge data was released for analysis and comparison with this program.

From the analysis of the Standard Charge Rate, the program was shown to have the closest charge input result for the target stel composition.

Error function correction

It is noted from the validation study of the program that the optimization result of Silicon in the final melt is higher than the target composition, usually having values between 0.26 – 0.264 as against 0.25Max. This increment is hypothesized to have come from the optimization value of Ferro-Silicon which was generated by the program in the Ferro-Additive charge interface on the 5th WTI and the resultant optimization result reflects the overall expected Silicon percentage composition in the final melt.

This could be corrected by determining the extent of deviation from the Standard Specification and:



1. Apply this correction into the original source code i.e mathematical model for Ferro-Silicon charge.
 2. Apply the error function correction into the target standard so as to end up with silicon optimization value that will fall within the target composition (from 0.26 – 0.25Max to 0.265 – 0.25Max): Error Range = 0.01 – 0.015
- Percentage Error Function = $(0.01/0.25 \times 100 = 4\%)$ to $(0.015/0.25 \times 100) = 6\%$
 - Percentage Error Function Range = 4 – 6 %

Therefore a correctional value of a minimum of 0.01 is applied to the Ferro-Silicon source code:

```
DualValueDisplay.ExactValue(Fe_Result.Text) +
    DualValueDisplay.ExactValue(C_Result.Text)) *          (Si_Melt2.GetPc -
My.Settings.SON_Si)) /          (My.Settings.SON_Si - Level1_Si1.GetPc
* 0.95)
```

We could also set the target composition to a minimum value for Silicon composition in the final optimization melt. Giving a possible lower limit of 10% Percentage error function, therefore the lower limit of the target Silicon will be by:

$$(100 - 10)\% \times 0.25 = 0.225$$

Hence a Silicon target composition set at 0.225 will ensure a final optimization value of $(0.225 + 0.01) = 0.235$ in the final melt.

Every other element were agreed to fall between the target composition and the model taken to be valid.

Conclusion

The closeness in the optimization data (above) with operational standard charge rate of the Integrated Steel Complex (from the Standard Body) under study validates the Simulation Application of the Developed program. It also reveals the much deviation in the plant existing charge procedure for meeting target standard.

Acknowledgements

The success of this research is due to the contribution of the following individual; Mr Augustine Okafor (Universal Steel Ltd, Ogba, Ikeja, Lagos), Dr. I.O Otunniyi & Dr. K.K Alaneme (Dept. of Metallurgicals & Material Engineering, Federal University of Technology, Akure, Ondo State, Nigeria), Banjo Mofesola Paul (Dept. of Computer Science, Federal University of Technology, Akure, Ondo State, Nigeria).

The funding for this research is made available by the authors and had worked together in the study design, data collection and analysis, model development, simulation application design, decision to publish, and preparation of the manuscript without any conflict of interest in any way.

References

1. Melting of scrap – A worldwide phenomenon (online). Available at <http://www.steelworld.com> 2010 (accessed at 10/09/2012).
2. Hiroyuki M., Christopher P. M., Ralmundo A. F., Richard J. F., *Development of a decarburization and Slag Fomation Model for Electric Arc Furnace*, ISIJ international Journal, 2008, 48(9), p. 1197-1205.
3. Basic oxygen steelmaking simulation guide, version 2 user guide, 2010, (online) available at <http://www.steeluniversity.org> (accessed at 12/10/2012).
4. Mills K., *A short course on Estimation of Slag Properties, Southern African Pyrometallurgy*, 2011, Department of Materials, Imperial College, London, UK.
5. Yasar Y., Unal C., Ismail E., *Optimum charging materials for electric arc furnace [EAF] and Ladle Furnace [LF] system: a simple case*. International Iron & Steel Symposium, Turkey, 2012, 14(5), p. 1-6.
6. Ekmekci I., Yetisken Y. and Camdali U., *Mass Balance Modeling for Electric Arc Furnace and Ladle Furnace System in Steelmaking Facility*, Iron and Steel Res, Int., 2008, 14(5), 1-6, 55.
7. Bock M., Louis A.K., Müller R., *Computer Supported Calculation and Evaluation of the Correct Composition of BOF Converter Slag*, Steel Research, 2008, 9, p. 63-68.
8. Abubakre O. K., Muriana R. A., *Mathematical Model for Optimizing Charge and Heel*



- Levels in Steel Re-melting Induction Furnace for Foundry Shop*, Journal of Minerals & Materials Characterization & Engineering, 2009, 8(6), p. 417-425.
9. Sandberg, Lennox, Undvall, *Multivariate predictions of end-conditions for electric arc furnace*, 2nd International Conference On Process Development In Iron And Steelmaking, 2004, 2, p. 447-456.
 10. Sandberg E., *Energy and scrap optimisation of electric arc furnaces by statistical analysis of process data*, Licentiate Thesis Luleå University of Technology Department of Chemical Engineering and Geosciences, Division of Process Metallurgy, 2005, 1, p. 435-444.
 - 11 Rao Y. K., *Stoichiometry and Thermodynamics of Metallurgical Processes*, Cambridge University Press, New York, 1985.

Appendix Program Coding (TROS)

The program coding below is used in the development of the TROS program application.

Class MainWindow

```
#Region "Window Movement"
```

```
Private isDown As Boolean
```

```
Private offset As Point
```

```
Private Sub Grid_MouseLeftButtonDown_1
```

```
(sender As Object, e As MouseButtonEventArgs)
```

```
    offset = e.GetPosition(Me)
```

```
    isDown = True
```

```
End Sub
```

```
Private Sub Grid_MouseMove_1(sender As Object, e As MouseEventArgs)
```

```
    If isDown Then
```

```
        Dim pos = PointToScreen(e.GetPosition(Me) - offset)
```

```
        Me.Left = pos.X
```

```
        Me.Top = pos.Y
```

```
    End If
```

```
End Sub
```

```
Private Sub Grid_MouseLeftButtonUp_1
```

```
(sender As Object, e As MouseButtonEventArgs)
```

```
    isDown = False
```

```
End Sub
```

```
#End Region
```

```

Private SON As New SON
Public Shared Main As MainWindow
Private HasOptimized As Boolean = False
Private HasOptimized2 As Boolean = False
Event Reset(target As String)

Sub LoadData(Optional ByVal for_expectation As Boolean = False)
    With My.Settings If Not for_expectation Then
        Fe.Text = .SON_Fe
        Si.Text = .SON_Si
        Mn.Text = .SON_Mn
        P.Text = .SON_P
        S.Text = .SON_S
        C.Text = .SON_C
    Else
        Fe_expect.pc.Text = .SON_Fe & "%"
        Si_expect.pc.Text = .SON_Si & "%"
        Mn_expect.pc.Text = .SON_Mn & "%"
        P_expect.pc.Text = .SON_P & "%"
        S_expect.pc.Text = .SON_S & "%"
        C_expect.pc.Text = .SON_C & "%"
    End If End With
    Dim Mfa As Double = Math.Abs(((Mst * (X_prime - X_not) +
    (DualValueDisplay.ExactValue(RefineScrap.Text) * (Xn - X_not))) * R_prime) /
    (X_not - (Xm * R_not)))
    Hidden_StepMum_For_Level2.Text = Mfa & "kg"
    CType(FindName(String.Format("Fe{0}_Result", element), TextBox).Text = Mfa &
    "kg"
    CType(FindName("Level2_C_" & element), DualValueDisplay).pc.Text =
    Level2_C.pc.Text
    CType(FindName("Level2_C_" & element), DualValueDisplay).kg.Text =
    Level2_C.kg.Text
    CType(FindName("Level2_Fe_" & element), DualValueDisplay).pc.Text =
    Level2_Fe.pc.Text
    CType(FindName("Level2_Fe_" & element), DualValueDisplay).kg.Text =
    Level2_Fe.kg.Text
    Select Case element
        Case "Mn"
            Level2_Si.pc.Text = "0.00%"
        Case Else
            Level2_Mn.pc.Text = "0.00%"
    End Select
    RaiseEvent UpdateDataFields(False)
    HasOptimized = True
End Sub

Private Sub Level3_Btn_Click
(sender As Object, e As RoutedEventArgs)

```



Handles Level3_Btn.Click

```
Dim element As String = Level3_Element.SelectedItem.Content.ToString.Split("
")(0)
Dim Mst As Double = DualValueDisplay.ExactValue(jScrap.Text) +
DualValueDisplay.ExactValue(Heel_kg.Text) + FindName("Level2_" & element &
"_Mn").GetKg + FindName("Level2_" & element & "_Si").GetKg +
DualValueDisplay.ExactValue(RefineScrap.Text)
Dim R_not As Double = Element_Recovery(element)
Dim X_not As Double = Element_SON(element)
Dim Xj As Double = DualValueDisplay.ExactValue(CType(FindName("Level3_" &
element), TextBox).Text)
Dim X_star As Double = DualValueDisplay.ExactValue(FindName(element &
"_Melt").Text)
Dim M_Ad As Double = ((X_not - X_star) * Mst) / (Xj * R_not)
(CType(FindName(element & "_Result"), TextBox).Text = M_Ad & "kg"
HasOptimized = True
```

End Sub

Private Sub Resultant_Click

(sender As Object, e As RoutedEventArgs)

Handles Resultant.Click

```
Try
    Dim elements = sender.Tag.ToString.Split(",")
    For Each element In elements
        Dim Mst As Double = (DualValueDisplay.ExactValue(jScrap.Text))
            + DualValueDisplay.ExactValue(Heel_kg.Text)
        Dim R_not As Double = Element_Recovery(element)
        Dim X_not As Double = Element_SON(element)
        Dim X_prime As Double = Pcent_Element_Comp_In_Melt(element)
        Dim Xn As Double = Pcent_Element_Comp_In_Incoming_Scrap(element)
        Dim Msc As Double = DualValueDisplay.ExactValue(RefineScrap.Text)
        Dim Pcentage_Element_Composition As Double =
            ((Mst * X_prime) + (Msc * Xn * R_not)) / (Mst + Msc)
        Dispatcher.BeginInvoke(Sub() CType(FindName(element & "_Melt"),
            TextBox).Text = Pcentage_Element_Composition & "%" End Sub)
    Next
Catch ex As Exception
End Try
```

End Sub

Private Sub Level2_Resultant_Click

(sender As Object, e As RoutedEventArgs)

Handles Level2_Resultant.Click

```
Try
    Dim elements = sender.Tag.ToString.Split(",")
    For Each element In elements
        Dim Mst As Double =
            (DualValueDisplay.ExactValue(jScrap.Text)) +
```

```

        DualValueDisplay.ExactValue(Heel_kg.Text)
    Dim R_not As Double = Element_Recovery(element)
    Dim X_not As Double = Element_SON(element)
    Dim X_prime As Double =
    Pcent_Element_Comp_In_Melt(element)
    Dim Xn As Double =
    Pcent_Element_Comp_In_Incoming_Scrap(element)
    Dim Mfa As Double =
    Mass_Of_Element_In_Incoming_FerroAlloy(element,
Level2_Element.SelectedItem.Content.ToString.Split(" ")(0).Replace("Fe", ""))
    Dim Xm As Double =
    Pcent_Element_Comp_In_Incoming_FerroAlloy(element,
Level2_Element.SelectedItem.Content.ToString.Split(" ")(0).Replace("Fe", ""))
    Dim Msc As Double =
        DualValueDisplay.ExactValue(RefineScrap.Text)
    Dim Pcentage_Element_Composition As Double =
    ((Mst * X_prime) + (Msc * Xn * R_not)
+ (Mfa * (Xm * R_not))) / (Mst + Msc + Mfa)
    Dispatcher.BeginInvoke(Sub()
    CType(FindName(element & "_Melt"), TextBox).Text =
    Pcentage_Element_Composition & "%" End Sub)
Next
Catch ex As Exception
End Try
End Sub

```

```

Private Sub Level3_Resultant_Click
(sender As Object, e As RoutedEventArgs)
Handles Level3_Resultant.Click

```

```

    Try
        Dim elements = sender.Tag.ToString.Split(",")
    For Each element In elements
        Dim Mst As Double = (DualValueDisplay.ExactValue(jScrap.Text))
+ DualValueDisplay.ExactValue(Heel_kg.Text)
        Dim R_not As Double = Element_Recovery(element)
        Dim X_not As Double = Element_SON(element)
        Dim X_prime As Double = Pcent_Element_Comp_In_Melt(element)
        Dim Xn As Double =
            Pcent_Element_Comp_In_Incoming_Scrap(element)
        Dim Mfa1 As Double =
            DualValueDisplay.ExactValue(FeMn_Result.Text)
        Dim Mfa2 As Double =
            DualValueDisplay.ExactValue(FeSi_Result.Text)
        Dim Xm1 As Double =
            Pcent_Element_Comp_In_Incoming_FerroAlloy(element, "Mn")
        Dim Xm2 As Double =
            Pcent_Element_Comp_In_Incoming_FerroAlloy(element, "Si")
        Dim Mad As Double =

```



```
        DualValueDisplay.ExactValue(CType(FindName(element &
            "_Result"), TextBox).Text)
    Dim Xj As Double =
        Pcent_Element_Comp_In_Incoming_Additive(element)
    Dim Msc As Double =
        DualValueDisplay.ExactValue(RefineScrap.Text)
    M_Xn = 0
Else
    M_Xn = R_not
Next
End Try
End Sub
```