OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Machine Learning-Based Inverse Predictive Modeling of the TTT Diagram

Joseph Oyedele[1], Chung-Hyun Goh[1,*]
[1]*Department of Mechanical Engineering, University of Texas at Tyler, TX 75799 USA*
**Corresponding Author, Chung-Hyun Goh*

Date of Submission: 04-02-2020 Date of acceptance: 20-02-2020

I. INTRODUCTION

The vacuum created by the conventional search method of developing a product by selecting materials based on its properties from existing databases has necessitated recent development in material and product search methodologies. The conventional search method cannot meet up with these demands. It is widely known that within a selected alloy system, the variation in microstructure results in a large range of material properties. There is a shift in material design in present dispensation where material design and modeling are microstructurally sensitive and it is crucial to identify the very microstructure within an allow system, out of a system of large candidate that ensured desired property for the product function. Mathematical search-based approach to material design explore the hypothetical infinitely large space and presented the freedom of unknown microstructure. With the rapid growth rate of candidate space where microstructure can be represented by thousands of dimensions, the mathematical approach deteriorates. Machine learning has proven to be very efficient in material and product design. Researchers have employed several machine leaning methods to model material design and this include but not limited to Artificial Neural Network (ANN), Support Vector Machine (SVM), Bayesian Network and so on. These set-based approaches provide a satisfactory system-wide solution to complex engineering design problems and meet the increased demand for a specific product function. We explore advanced data-oriented techniques to enhance material design approach by statistical heuristics through implementation of Random Forest (RF) method with Monte Carlo simulation to model a time-temperature-transformation (TTT) curve for the design of steel within specified performance requirements. This is demonstrated as a predictive approach to modeling the TTT curve which investigate the effect of uncertainty propagation within the model structure and identify predictive capability of the machine learning approach.

This paper is categorized as follows; section one described the general knowledge of Machine Learning in relation to material design and the importance of TTT diagram in material design. Section two presents previous works by researchers and gave a gap analysis in this field of study. Section three discusses the approaching methods for this analysis. Results and discussion from our approaching methods are presented in section four while section five contains the concluding remarks.

II. LITERATURE REVIEW

The exact and suitable materials properties that satisfy the required technical requirements are often not feasible to satisfy the design need of most engineering material design. This consequently limits the performance capabilities of many systems and available products. Hence, the need for improved system performance has led to the concurrent design of material microstructure and composition alongside system level design to provide materials that meet product and performance requirements. The set-based design approach has yielded more robust solutions over several multidisplinary design optimization methods which include (Analytical Target Cascading (ATC), concurrent subspace optimization, etc.) requires an expensive iterative simulation analysis among design levels [1]. Set-based method performs better on discrete variables and discontinuous output responses that often characterize material design problems. The mechanical properties of a product have been widely known to be affected by the constituent's microstructure that evolve during the design process of steel. The variation in composition elements results in different phase formation. These variations along the process chain have been graphically presented in a continuous cooling transformation (CCT) or TTT curve diagram to better understand how microstructure varies along the process path. TTT diagrams are a practical tool for choosing and interpreting the effects of heat treatments. They are condensed graphical representations of phase transformation behavior over a wide range of cooling rates for a given steel composition and initial microstructure, containing a large amount

of information on transformation behavior, resulting microstructures and properties such as hardness. Due to the complexity of the steel transformation behavior, a measured CCT diagram is valid for the composition, thermal and deformational history for which it has been determined only; small differences in initial conditions may result in relatively large changes in the diagram. Since this is the case, CCT-diagrams have been measured for most of the common steel compositions in the last 50years. Unfortunately, CCT-diagrams are difficult to determine accurately. In most cases they are determined using a combination of dilatometry, metallography and hardness testing, where accurate measurements require good experimental skills, the interpretation of the dilatometric and metallographic data is often far from straightforward. Consequently, when CCT-diagrams are measured by different researchers, the resulting diagrams may be markedly different. Errors in the measurements are large, but rarely quantified. Up to now alternative approaches have been followed, where a database of TTT-diagrams can be queried for a specific composition, and diagrams with more or less matching compositions can be retrieved. Such systems have two limitations: they do not allow the effects of alloying elements to be investigated separately since diagrams in neatly measured, compositional series are rare, and they do not average out errors in measurement of the TTT-diagrams. An example of TTT diagram is shown in Figure 1.

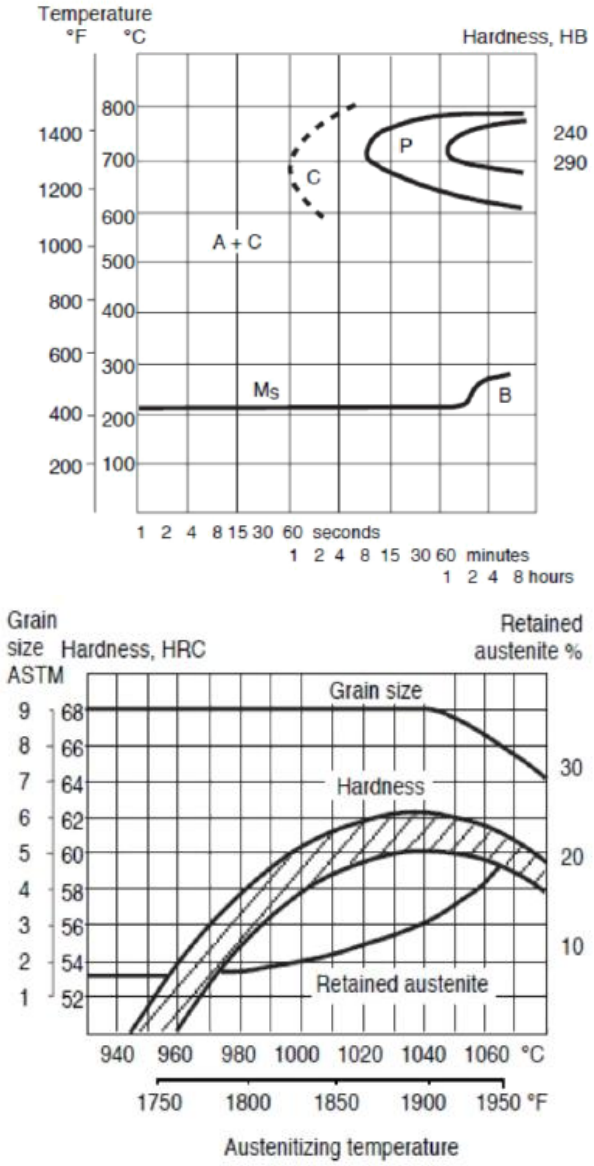

Figure 1. TTT diagram for cold working tool A8 steel (a) shows the time temperature curve with the emerging phases, and (b) relates the Austenite temperature together with hardness and grain size

This paper makes use of cold working tool steel AISI A8 which is of air hardening grade with high content of chromium and carbon. The evolving microstructure during heat treatment of this steel is of immense importance as the final mechanical properties of this steel depend solely on the microstructural changes during the process. The transformation process is complex in nature but to a very large extent depends on temperature and time [2]. Thus, understanding of the time temperature transformation of phases, which varies for different material due to its microstructure constituents, is of relative importance. The engineering need for this material is related to its performance requirement in metal forming and cutting applications. These mechanical applications have created an increase demand especially in the manufacturing and operation industries such as automotive. The high hardenability and toughness among other mechanical properties of the A8 steel resulted in its widespread use for cold work processes.

Machine learning (ML) approach to material modeling is now been implemented due to the advantage it presents to material design approach for material and product development. R. Arroyave and co-author [3] implemented Support Vector Domain Description (SVDD) as an approximation to the infinite solution set to determine the feasibility range of data set while considering some set of thermodynamic conditions for a phase stability problem. Jordan Matthews and co-authors [4], applied the Naïve Bayesian Network classifier which is a fully disconnected Bayesian network in which no edges existed between design variables to provide mappings between multiscale levels (micro-meso model, and meso-macro model) in the design of Negative Stiffness Metamaterials. The effectiveness of Artificial Neural Network in predicting the CCT diagram for the precision forged tempering steel 50CrMo4 was shown by considering deformation CCT diagrams for several tempering steels as input data for predicting microstructural transformation [5]. Machine Learning technique in general has been seen to eliminate the challenges faced by previous approaches to material design and modeling. It is time efficient and relatively affordable compared to performing physical experiments that are cumbersome and costly. It eliminates the complexities and technicalities that arises from mathematical formulations and analysis and eliminate propagated uncertainties in the model.

For our design problem, we utilize the application of machine learning algorithm to material processing and design as a demonstration problem.

Random Forest method was implemented to successfully predict a A8 steel. The robust nature of the process helps in mitigating propagated uncertainties which tends to arise from the material models due to idealization or simplification of models.

III. APPROACHING METHOD

Machine Learning refers to the automated detection of meaningful patterns in data [5]. It is the application of artificial intelligence that enables systems to learn and substantiate future decisions from experience without being explicitly programmed. Machine learning algorithms can be divided into three broad categories — supervised learning, unsupervised learning, and reinforcement learning. Supervised learning is useful in cases where a property (*label*) is available for a certain dataset (*training set*) but is missing and needs to be predicted for other instances. That is, the computer is presented with ranged set of input and output dataset. The objective of supervised learning is to learn the rule that maps input space to an output space. Unsupervised learning is useful in cases where the challenge is to discover implicit relationships in a given *unlabeled* dataset (items are not pre-assigned). Reinforcement learning falls between these two extremes. There is some form of feedback available for each predictive step or action, but no precise label or error message.

Supervised learning is categorized based on the expected output of the algorithm. These are broadly in two categories; classification and regression model. In classification model, new predictions are identified to a set of categories that are subset of the same class. Classification model is considered on the instances of supervised learning. When the focus of simulation is on the relationship between dependent variable and sets of independent variables or predictors, a regression model is used. This analysis presents the response of a dependent variable when there is a change in any one of the independent variables while other variables are held constant.

There are several machine learning algorithms in use to carry out training and testing of data set for a required output. The random forest (RF) method, classified under supervised learning as proposed by [6] is made up of nearest neighbor whereas “Ni–Mn” indicates an alloy of some composition Ni_xMn_{1-x}. algorithm. It is an ensemble of trees that easily captures nonlinear relationships between an input dataset and a target set by using ensemble of regression trees. This predictive model is composed of a weighted combination of multiple regression trees. These models are useful in several prediction tasks where the features are more or less self-evident. In general, the model combines multiple regression trees to increase predictive performance. ML predictive modelling was used in [7] to predict the microstructure of a machined titanium alloy. Experimental results were used as an input data set for the predictive modelling of micro hardness and grain size. Sixteen (16) samples were generated by varying cutting conditions. These were machined, hardness and microstructure analysis of grain size evolution and volume fraction was conducted by using SEM scanning Electron Microscope imaging together with

a proprietary image processing program written in MATLAB. The temperature field was simulated by using 3D finite element simulations of machining the alloy. This temperature set serves as input dataset to predict hardness using the machine learning. The problem structure for our analysis consists of continuous variable that are more than two with several observations. A regression analysis was suitable for this problem because of the continuous nature of the design problem. ANOVA method implementation into random forest is the metrics on which feasibility check was performed. ANOVA, known as analysis of variance method, basically is a statistical methodology that determines the variation data points from the mean response of the desired variable. Mathematically, ANOVA is used to analyze the importance of one or more factors by comparing the mean of the response variable at the different factor levels. Analysis of variance requires a continuous response variable and at least a categorical factor with two or more levels. ANOVA require data from approximately normally distributed populations with equal variances between factor levels. However, ANOVA procedures work quite well even if the normality assumption has been violated, unless one or more of the distributions are highly skewed or if the variances are quite different. Transformations of the original dataset may correct these violations. ANOVA was applied to random forest algorithm to determine at what point the individual tree splits to classify each variable into feasible points or infeasible points. Figure 2 below shows a general regression tree structure for the analysis considered. The target variable is the dependent variable. Each branch of the tree is created at the point when the decision is made according to the splitting criteria at each branch node. The response variable for this analysis is continuous and does not belong to any class, thus a regression model is fit to each of the independent variables by isolating these variables as nodes to decrease error.

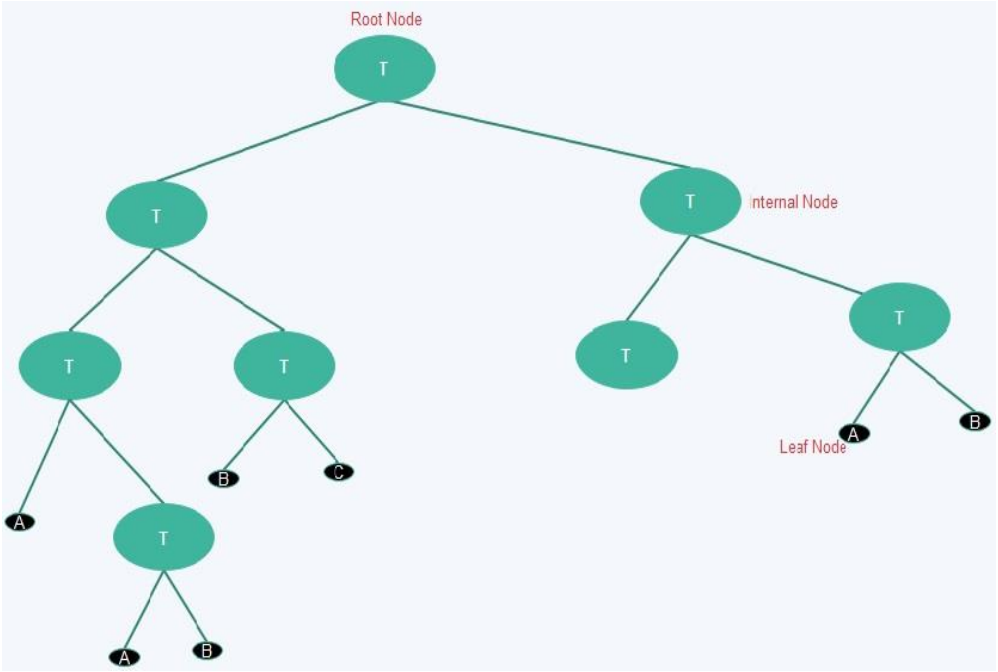

Figure2. A schematic of the regression model

Random forest forms a family of methods that consist in building an ensemble of decision trees that are grown from a randomized variant of the tree induction algorithm. This predictor ensemble of decision tree grows in a randomly selected space [6]. Various empirical studies have shown that random forests are fast and easy to implement, generate high prediction accuracy and can handle large number of input data set without the possibility of over fitting. An interesting feature is the built—in possibility of using left out sample to form estimates of important statistics (out of bag estimate) [6]. Random forest maintains high level of accuracy even with inconsistent data and gives a correlation plot to indicate important variables for the analysis. This algorithm can be used for either classification or regression model. Random Forest has been classified as a combination of decision trees (classification or regression) with great noise elimination ability that combines weak classifier of the sample type to produce a strong classifier. It provides an accurate and robust tool for solving many machine learning tasks ranging from regression, classification, density estimation, manifold learning [7]. Breiman's Random Forest was based on the early work of Amit and Greman [8] based on the geometric feature selection, combined with the random space method of Ho [9] and the random split selection method of Dietterich [10]. Random Forest method is fast and easy to implement with high prediction accuracy and it is built with capacity to handle high volumes of input variables as well as small sample sizes and high-dimensional feature spaces

without overfitting. It can be easily parallelized, thus having the capacity to deal with real-life systems. These attributes make it one of the preferred among other machine learning methods [11, 12, 13, 14, 15]. Random Forest operates according to a "divide and conquer" principle by sampling a fraction of input data, grow a tree predictor on each small fraction of data randomly, then aggregate each predictor.

Random Forest as a robust model can deal with both supervised classification and regression analysis. However, we focus on the regression analysis for our study. Categorizing Random Forest into classification or regression analysis is based on the desired output for the specified analysis. If the desired output is categorical, Random Forest classification method is used, and the classifier is given by the expression $\Psi: X \longmapsto Y$, where Y is a finite set of classes denoted by $\{c_1, c_2, c_3, \ldots\ldots, c_k\}$.

While Random Forest regression model is used when the response variable is numerical and the regressor is a function

$$\psi: X \to Y \qquad \text{where} \quad Y = \mathbb{R} \tag{1}$$

Equation (1) indicate the relation between Ψ *and* X (independent variable) and Y (predicted variable), where Y belong to class of real numbers. The general framework for the regression model is nonparametric regression estimation in which an input random vector $P\epsilon\ [0,1]^k$ is observed. For Random Forest analysis, a training data set is of the form

$$T_n = \{(X_1, Y_1), \ldots\ldots\ldots (X_n, Y_n)\} \tag{2}$$

For a regression analysis, random forest is a predictor consisting of a collection of M regression trees. For the i-th tree in the forest, the predicted value at the split point **x** is denotated by $m_{M,n}(\mathbf{x}; \Theta_1, \ldots\ldots, \Theta_M, T_n)$ where $\Theta_1, \ldots\ldots, \Theta_M$ are independent random variables, $m_{M,n}$ is the regression error estimate, and T_n is the training samples. Thus, the total tree estimate for the forest is the sum of individual tree estimate that makes the forest. This is denoted by equation (3)

$$m_{M,n}(\mathbf{x}; \Theta_1, \ldots\ldots, \Theta_M, T_n) = \frac{1}{M}\sum_{i=1}^{M} m_n(\mathbf{x}; \Theta_i, T_n) \tag{3}$$

The random Forest in R has a default value of M (number of trees in the forest) to be 500 (that is, $ntree = 500$. Though this value can be adjusted to any arbitrary value higher than the default value. In a classification domain, the random forest classifier is obtained by a majority vote among various classification trees and the decisions for classifying the classes is given by equation (4) and equation (5)

$$m_{M,n}(\mathbf{x}; \Theta_1, .., \Theta_M, T_n) = 1, \qquad if\ \frac{1}{M}\sum_{i=1}^{M} m_n(\mathbf{x}; \Theta_i, T_n) > \frac{1}{2} \tag{4}$$

$$m_{M,n}(\mathbf{x}; \Theta_1, .., \Theta_M, T_n) = 0, \qquad if\ \frac{1}{M}\sum_{i=1}^{M} m_n(\mathbf{x}; \Theta_i, T_n) < \frac{1}{2} \tag{5}$$

From equation (4) and equation (5), 0 and 1 denote two different classes into which data samples are classified upon satisfying the indicated criteria for the case of binary classification problem. However, random forest can intrinsically handle multi-class problem. Random Forest has however been modified based on its splitting method and criteria to generate a variance from the basic principles of Breiman Random Forest [16]. This include the purely random forest; under which the centered forest is classified as a method of selecting a variable among other variables and performing a split at the center of the variable selected. The tree growing process stops when a full binary tree is reached to give a 2^k leaves for each tree as the process is repeated *k* times, where k $\epsilon\mathbb{N}$. K is a smoothing parameter of the algorithm. Another modified model is the uniform random forest which varies slightly from the centered random forest as split are performed uniformly at random over the selected variable among all variables.

The premise of our investigation is fairly simple. Given some set of factors $(a_1, a_2, a_3, \ldots\ldots, a_n)$ in the domain A, we want to predict the outcome of interest, P. Figure 2 imply the domain of all factors associated with output interest P in descending order of importance. In traditional regression models, a single equation or model is developed to represent the entire data set. Our method provides an alternative approach to this, in which the data space is partitioned into smaller sections where variable interactions are better understood. This analysis uses this recursive partitioning to create a tree where each node T in Figure 2 represents a cell of the recursive partition. To each cell, a simplified model is attached. This process is similar to conditional modeling along the tree structure where conditions are met on a particular variable as we move down the nodes. The final split or node is often

referred to as the leaf node. In Figure 2, A, B, and C are each terminal nodes (leaves). The leaf node implies that after this split, further splitting of the data does not give enough explanation of the variance to be considered relevant in describing the output outcome describing P.
Mathematically, say we need to find a function *d(a)* to map our domain A to our response variable P, we need to assume of the existence of a sample of *n* observations,

$$\beta = [(a_1, p_1), \dots\dots, (a_n, p_n)] \qquad (6)$$

According to regression equations standard, our criterion for choosing *d(a)* will be the mean squared prediction error given by the expression, $E[d(a) - E(p|a)]^2$. For each of the individual leaf-node *l* and training samples *k* in the regression tree, then, our model is given by,

$$\hat{p} = \frac{l}{k}\sum_{k=1}^{k} p_1 \qquad (7)$$

Equation (7) represents the sample mean of the response variable considered for each cell which creates a piecewise constant model. This feasibility criteria maximizes the decrease in impurity, and it is the principal driving statistical mantra behind regression analysis using regression tree. Materials discovery process can be significantly expedited and simplified if we can learn effectively from available knowledge and data [17]. This is the main principle on which machine Learning techniques operate. However, one of the challenges faced with machine learning approach is the limited data available for modeling. The accuracy of a machine learning technique relied solely on the robustness of the model. The more the dataset, the more robust the analysis and the more accurate the prediction. To deal with this staggering challenge, several methods have been implemented to improve the accuracy of a machine learning techniques. For our analysis, Monte Carlo simulation was implemented to deal with uncertainty propagation that arise from model parameter uncertainty. Model parameter uncertainty are associated with design variables or control factors as a result of incomplete knowledge of model parameter/input. Monte Carlo simulation performs a random sampling and conduct large numbers of simulation on computer in order to observe the statistical characteristics of the model output. For an experiment, the possible input random variables $X = (X_1, X_2, \dots., X_n)$ are sampled according to their distribution function. Then the values of the output variables Y are calculated through the performance function $Y = g(X)$ at the samples of input random variables. With several experiments performed in this manner, set of output variable samples are obtained for the statistical analysis used to estimates the characteristics of the output variable Y. The simulation process of Monte Carlo is depicted in Figure 3. Three steps are required in the process: Step 1 involves sampling on random input variables X, step 2 involves evaluating the model output Y, and step 3 is performing statistical analysis on model output.

Problem Analysis

In this paper, we consider the task of predicting temperature transformation and continuous cooling transformation diagrams as an approximation problem by obtaining a model with known data. For this reason, we describe each curve, which indicates the start or end of a microstructural transformation with the function

$$T = f_i(time), \quad i = \overline{1, n} \qquad (8)$$

where T is the transformation temperature in (degree Celsius), $time$ is the time of initiation or completion of a microstructural transformation, and n is the number of phase transformation. This model is developed by using Random Forest algorithms incorporated with Monte Carlo simulations. The chemical composition of A8 steel is shown in Table 1 below.

Table 1. Chemical composition of A8 steel

Material	Carbon	Manganese	Silicon	Chromium	Molybdenum	Tungsten
% Composition	0.55	0.30	0.95	5.0	1.25	1.25

The constituent materials in their percent compositions are the factors that determines its mechanical properties for its application as dies and punches, drift pins, pneumatic tools, hammers, chuck jaws, hot rolls, and so on. It has a very good toughness, wear resistance, and excellent dimensional stability in heat treatment. These attributes make A8 steel an excellent choice for applications that requires higher toughness than that of high-carbon or high- Chromium steels and better wear resistance than that of shock-resistance steels.

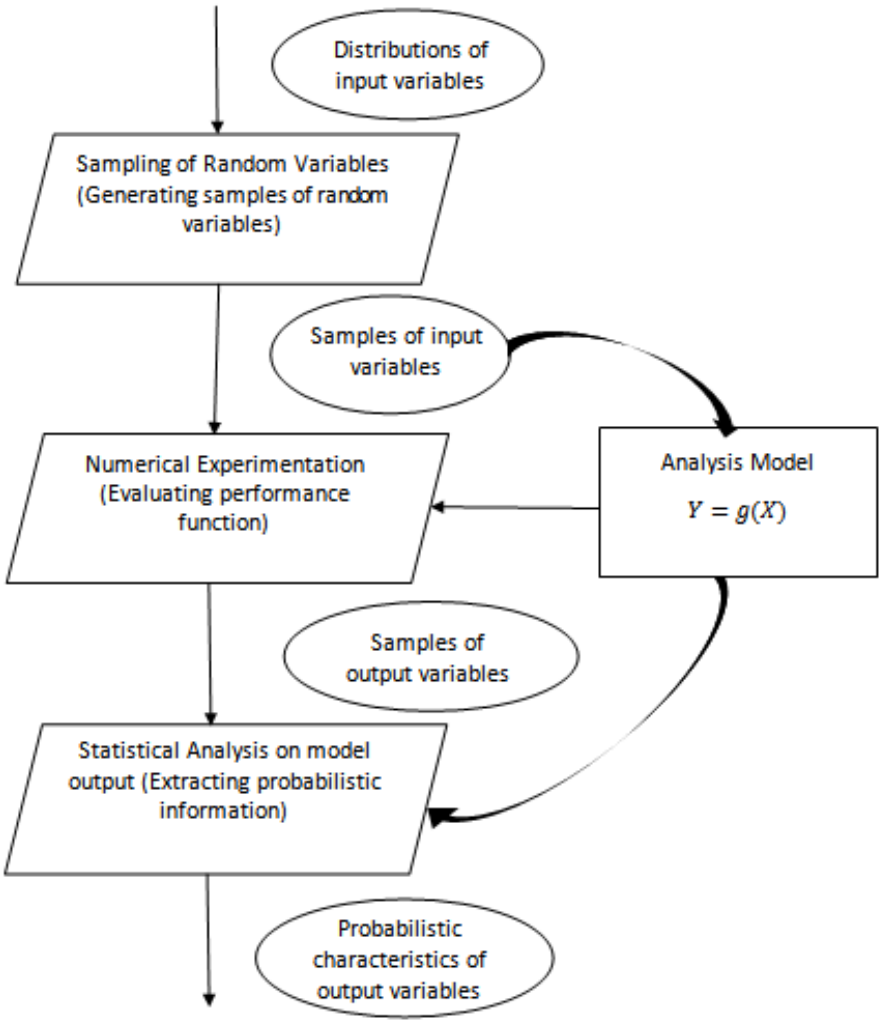

Figure 3. Monte Carlo algorithm

The carbon content in this steel significantly helps in austinite and martensite phase formation during microstructural transformation which helps to improve the mechanical strength and hardness respectively. In general, alloying elements have been found to influence the transformation kinetics and mechanism in which carbide forming elements such as Molybdenum and Chromium produce tremendous changes in the isothermal transformation process. The formation of austinite structure and the ferrite structure has been generally seen to be the major transformation phases effected by alloying constituents [18].Austinite grain size, alloying elements which tends to slow the transformation reaction and carbon content has been found to be the major contributors to the hardenability of the steel. The steel is hardened when it is transformed from austinite phase to martensite on quenching. As the steel is cooled, microstructural transformation from austinite to martensite occur. The martensite phase stops at a temperature where pearlite and eventually ferrite microstructure begins in fine grain steel. This causes a reduction in grain size and thus, there is a shift in the TTT diagram towards the left. Increase in percentage of carbon lowers the martensite start temperature and thus a shift in the curve towards the right is observed [19]. The complexity of steel transformation behavior follows directly from even the simplest physical analysis of the process. Steel consists of iron and carbon, and usually also contains additional alloying elements. Pure iron in equilibrium has a body centered cubic (BCC) crystal structure called ferrite at room temperature and pressure, and a face centered cubic (FCC) crystal structure called austenite at temperatures between 912 ºC and 1394 ºC, and again a BCC crystal structure at temperatures between 1394 ◦C and 1538 ◦C, the temperature at which pure iron melts. Most low alloy steels have the same equilibrium phases, yet at different temperatures, which depend on composition, since equilibrium phase transformations are controlled by thermodynamics: changes in free energy of the phases involved with temperature determine transformation temperatures. Furthermore, the thermodynamics determines the reaction heat of a phase transformation. The free energies of the

FCC and BCC phase change significantly when alloying elements are added, depending on the amount, type, and combination of alloying elements. In some cases, the effect of an element is even reversed when another element is added. Furthermore, these elements may form other phases in combination with iron, carbon, or other elements. Measuring the effect of a single alloying element on the free energy is already difficult; quantifying the effect of a number of alloying elements is progressively harder. These thermodynamics characterize the equilibrium structure of a steel, yet phase transformations do not proceed according to equilibrium. To be able to describe the structure, the thermodynamics determine the driving force for the phase transformation, and the kinetics of the phase transformations determine which phases actually form. The kinetic aspects of steel transformations thus influence the phase transformations and the resulting structure and properties greatly. During cooling, diffusion of alloying elements is required for the formation of equilibrium phases in most cases, notably the diffusion of carbon, since the solubility of alloying elements is limited in equilibrium phases. This diffusion will result in local compositional variations. The diffusion rate of alloying elements depends most strongly on temperature and the free energies of the phases occurring, yet it is also greatly affected by the presence of other alloying elements, i.e. the local composition. If during cooling insufficient time for diffusion is available for the equilibrium phase transformations to complete, then one or more other phase transformations occur resulting in metastable (non-equilibrium) phases, such as bainite and Martensite; strictly speaking cementite and pearlite too are metastable. These phases each have their own free energies and transformation kinetics, again depending on composition. During non-equilibrium phase transformations alloying elements may remain in place in a supersaturated solution in the new phase, may form small particles of yet another phase (precipitates), and may affect the diffusion rate of other elements. An impressive number of these mechanisms is already known, yet this knowledge is far from conclusive. Steel is a polycrystalline material, i.e. on a microscopic level a steel specimen consists of number of crystals with different orientations and in many cases with12different crystal structures. The size, shape and distribution also have a large effect on the kinetics of the phase transformations of steels. To be able to fully describe the kinetics of steel transformations, the thermodynamics of the non-equilibrium phases needs to be known, as well as the diffusion coefficients of all alloying elements as a function of temperature, local composition and structure, which are all very difficult to measure.

For this analysis, we collected TTT diagram for A8 steel in a graphical format. The statistical approach to modeling the TTT curve is only able to process numerical information. Therefore, TTT diagram need to be converted into a numerical format. Numerous conversion methods can be utilized but the one used for this model is an intercept method. In the intercept method, a number of fixed test lines is drawn over the diagram, where the intercepts of these lines with the phase boundaries in the CCT-diagrams determine coordinates describing the diagram. An advantage of intercept methods above grid methods is that the diagram can be represented by a relatively small number of data, since all data point at the relevant phase boundaries instead of describing the entire phase regions. These data were imported into MATLAB for Monte Carlo simulation. The Monte Carlo simulation uses probability distributed function (pdf) to analyze and generate random samples as depicted in Figure 3. 80% of the random output variables were then used to train our Random Forest model while the remaining 20% was used to test our model.

IV. RESULTS AND DISCUSSION

From the generated datasets for the TTT curve, the measured time and corresponding temperature values were imported into MATLAB for the Monte Carlo simulation. To carry out Monte Carlo simulation, constitutive equation is required for numerical experimentation. This is the analysis model in which the performance function is evaluated to generate samples of output variables. Monte Carlo simulation was to generate 100 random data samples as shown in figure 4 before applying the analysis model.

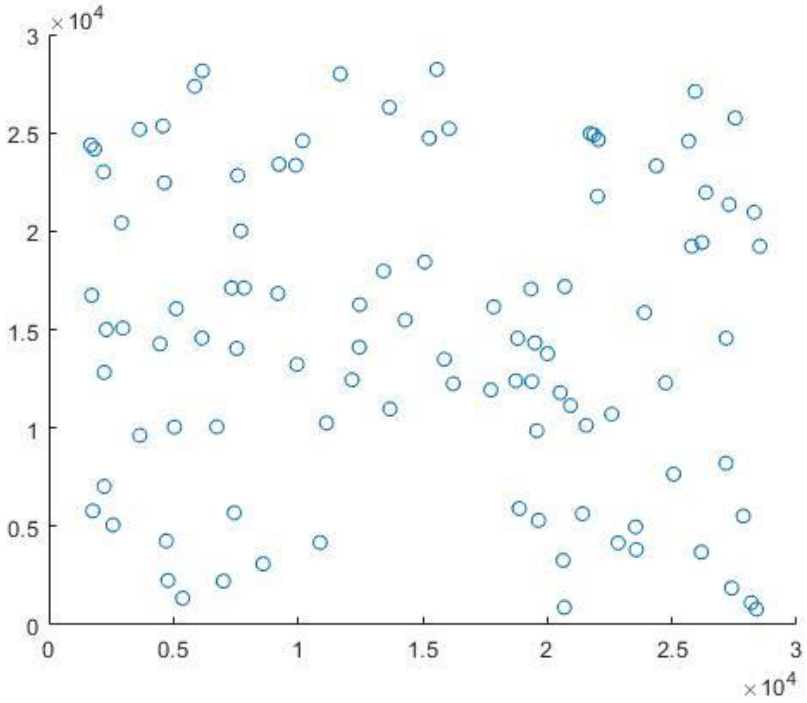

Figure 4. Random data sample generated with Monte Carlo simulation

The applied constitutive model together with the uniform distribution function in MATLAB was used to extract probabilistic information from output data samples. In order to determine the constitutive equation for the TTT plot, several analyses were implemented in MATLAB to generate best curve fit from the measured datasets. The best method was achieved by dividing the TTT dataset into two, the upper bound and the lower bound as shown in Figure 5 and Figure 6 respectively.

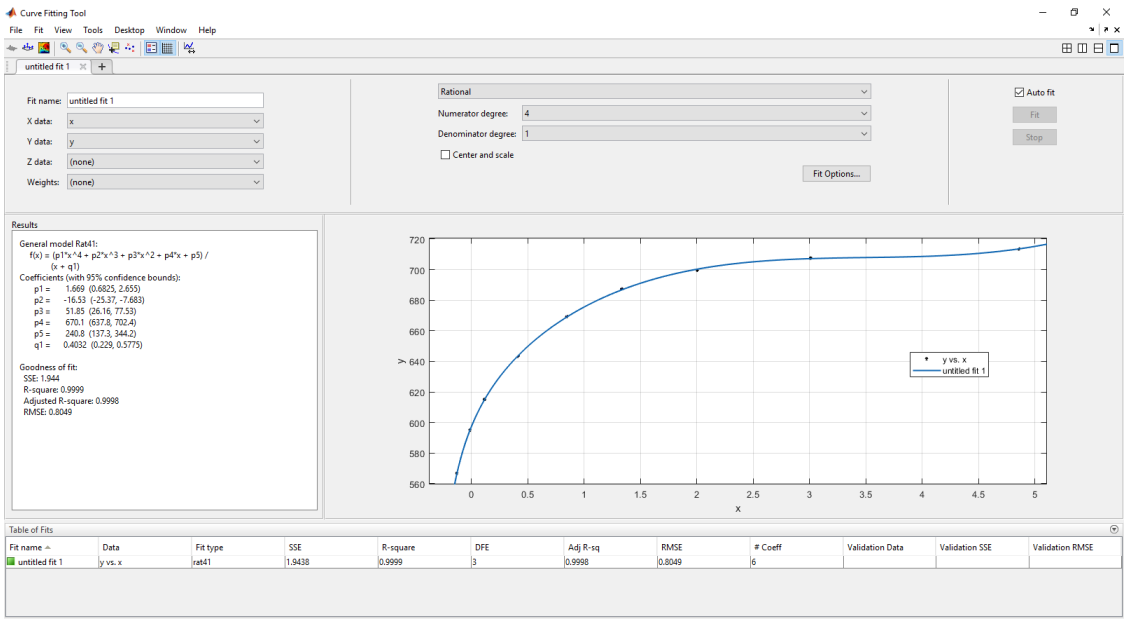

Figure 5. Upper bound plot with Monte Carlo method

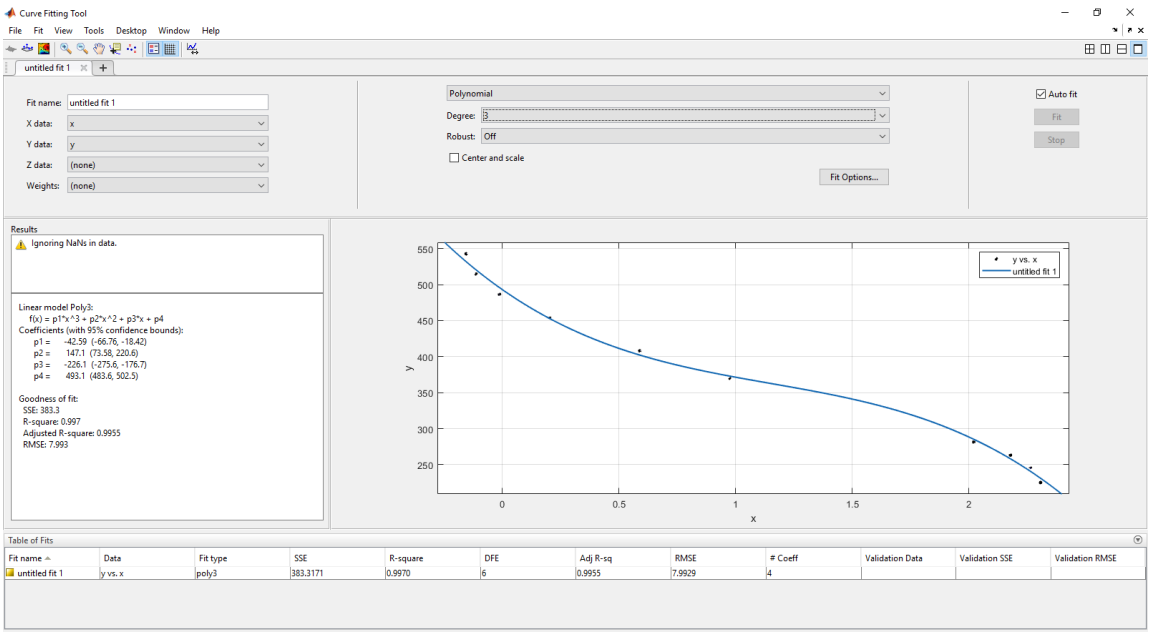

Figure 6. Lower bound plot with Monte Carlo method

The final CCT curve was achieved by super-imposing the two halves. The best fit for the upper bound after several manipulations is Power function with the number of terms as 2. For the lower bound of the CCT curve, power function with 2 number of terms was found to give the best fit with an R-square value of 0.999 and a 95% confidence interval as shown in the figure below. The above-mentioned process analysis helps in determining the constitutive equation to model the CCT diagram. For our case, two different equations were derived for the Monte Carlo simulations. Monte Carlo algorithm was implemented at this point after modifying the open source code to suit our analysis. To test the correctness of the constitutive equations, 100 simulations was performed, and the plot below shows the generated datasets and the TTT curve. This plot clearly shows the superposition of the upper and lower bound. This plot is similar to the one generated with measured data.

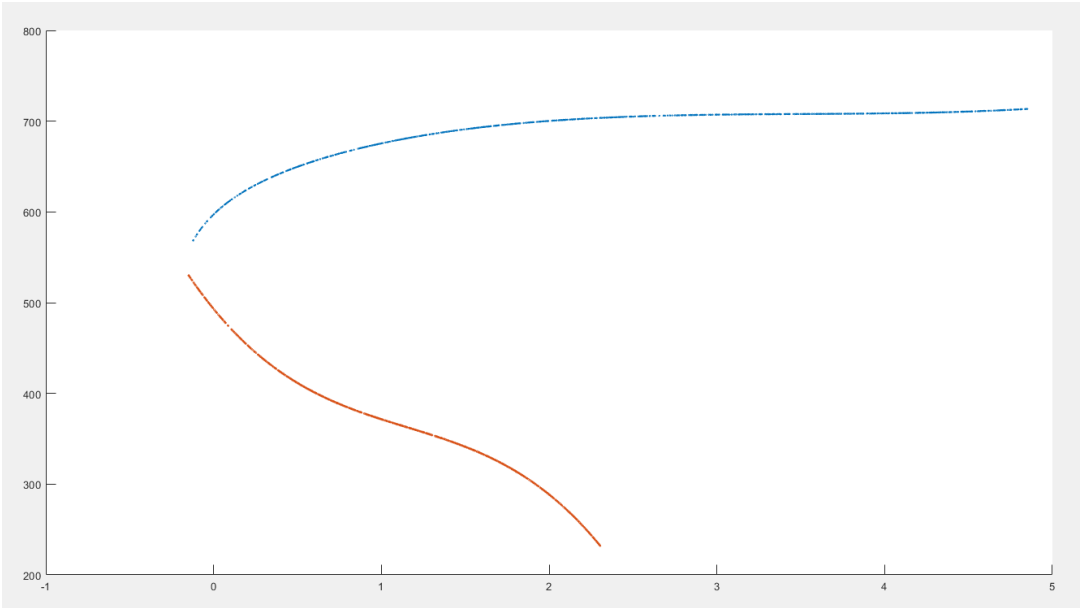

Figure 7. Super-position of the upper and lower bound plot

The output of Monte Carlo analysis was imported into R-statistical environment for regression analysis. The Random Forest model was trained with 90% input sample and a 10% test data samples was used to analyze how well our model performed. A mean square error (MSE)value of 87.5% was derived from the model. The model gave a Root Mean Square Error (RMSE) of 90.5%. these values indicated that our model predicted values

are very close to the actual data set. To validate our model, data points was digitized from [2] and used as input data for Monte Carlo simulation. The output data from Monte Carlo was imported into R. the input data was fitted to the Random Forest model and further statistical analysis was carried out to determine the performance accuracy of our regression model

Figure 8 below shows the plot of the TTT curve. The model gave a very good prediction of the TTT curve with an accuracy matrix of 92.7%, this is approximately 100% and it indicate that the model accurately predicts the TTT curve. Residual standard error and multiple regression error of the model are shown in Table 2.

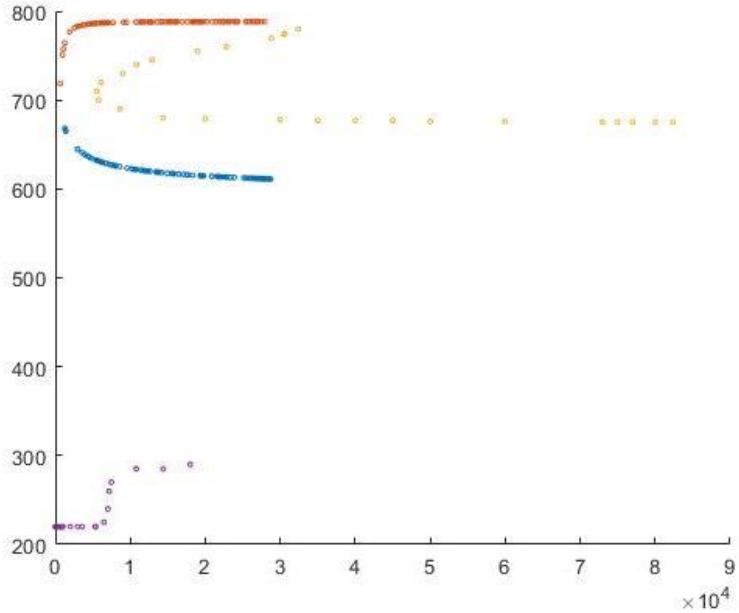

Figure 8. Plot of a TTT diagram with Random Forest model

Table 2. Statistical components of the Random Forest algorithm

```
Residuals:
   Min     1Q Median     3Q    Max
-6.265 -3.922 -1.401  2.277 30.627

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.601e+02  6.277e+01  10.517   <2e-16 ***
x1          -1.133e-03  7.185e-05 -15.775   <2e-16 ***
x            1.072e-04  8.673e-05   1.236    0.219
y           -2.818e-02  8.040e-02  -0.350    0.727
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.207 on 96 degrees of freedom
Multiple R-squared:  0.7244,   Adjusted R-squared:  0.7157
F-statistic: 84.09 on 3 and 96 DF, p-value: < 2.2e-16
```

V. CONCLUDING REMARKS

A TTT diagram of cold working steel A8 steel that contains the microstructural changes during heat treatment was observed in this paper. Digitized diagrams were used as input data for Monte Carlo simulation. Monte Carlo simulation provides a framework for model robustness and deals with uncertainty propagation due to limited data set. The Random Forest method was utilized to predict the TTT curve. From the results, our regression model gave a good prediction of the curves. However, our model prediction accuracy is limited to the type of steel to which it was trained. Use of different steel types could give a poor prediction due to the different percentage of constituent elements which affect the mechanism and kinetics of steel transformation during heat treatment.

NOMENCLATURE

k	Smoothing Parameter
M	Number of trees in the forest
$m_{M,n}$	Regression Error Estimate
$\mathbb{N}$	Set of Natural Numbers
T	Temperature
T_n	Training Samples
Y	Predicted (output) Variable
$\alpha_1, \alpha_2, \cdots, \alpha_n$	Set of factors
$\Theta_1, \cdots, \Theta_M, \psi, X$	Independent Variables
β	Sample of n Observations

REFERENCES

[1]. W. Chen, X. Yin, S. Lee, and W.K. Liu. "A multiscale Design Methodology for Hierarchical Systems with Random Field Uncertainty," *Journal of Mechanical Design*, 132 (4), 041006, pp. 1-11, (2010).

[2]. N. Pillai and R. Karthikeyan. "Prediction of TTT Curves of Cold Working Tool Steels using Support Vector Machine Model", 2018 *IOP Conference Series: Materials Science and Engineering*, 346, 012067 (2018).

[3]. R. Arroyave, S.L Gibbons, E. Galvan, and R.J Malak, "The Inverse Phase Stability Problem as a Constraint Satisfaction Problem: Application to Material Design", *Journal of the Minerals, Metals & Materials Society (TMS) JOM*, 68 (5), pp. 1385-1395 (2016).

[4]. J. Matthews, T. Klatt, C. Morris, C.C. Seepersad, and M. Haberman, "Hierarchical Design of Negative Stiffness Metamaterials using a Bayesian Network Classifier", *Journal of Mechanical Design*, 138 (4): 041404 (2014).

[5]. S. Shalev-Shwartz and S. Ben-David, "Understanding Machine Learning: From Theory to Algorithms" by Cambridge University Press, (2014).

[6]. G. Biau,"Analysis of a Random Forest Model", *Journal of Machine Learning Research*, 13, pp. 1063-1095, (2012).

[7]. Y.M. Arisoy and T. Özel, "Machine Learning Based Predictive Modeling of Machining Induced Microhardness and Grain Size in Ti–6Al–4V Alloy", *Materials and Manufacturing Processes*, 30 (4), pp. 425-433, (2015).

[8]. Y. Amit and D. Geman, "Shape Quantization and Recognition with Randomized Trees", *Neural Communication*, 9, pp. 1545-1588, (1997).

[9]. T.K. Ho, "The Random Subspace Method for Constructing Decision Forests", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20 (8), pp. 832-844 (1998).

[10]. T. Dietterich, "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting and Randomization, *Machine Learning*, pp. 1-22, (1998).

[11]. R. Diaz-Uriarte and S.A. de Andres, "Gene Selection and Classification of Microarray Data using Random Forest", *BMC Bioinformatics*, 7, pp. 1471–2105, (2006).

[12]. L. Breiman, "Random forests", *Machine Learning*, 45, pp. 5–32, (2001).

[13]. R. Genuer, J.-M. Poggi, and C. Tuleau, "Random Forests: Some Methodological Insights", Research Report RR-6729 INRIA, arXiv:0811.3619, (2008).

[14]. R. Genuer, J.-M. Poggi, and C. Tuleau-Malot, "Variable Selection using Random Forests", *Pattern Recognition Letters*, 31, pp. 2225–2236, (2010).

[15]. ErwanScornet, "Learning with Random Forests", *Statistics [math.ST]*. Université Pierre et Marie Curie, (2015).

[16]. W.Y. Loh, "Classification and Regression Tree Methods". *Encyclopedia of Statistics in Quality and Reliability*, (2008).

[17]. G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, and R. Ramprasad, "Accelerating Materials Preperty Predictions using Machine Learning", *Scientific Reports*, 3 (2810), (2013).

[18]. M. Maalekian, "The Effects of Alloying Elements on Steels", Technische Universitat Graz, October 2007.

[19]. H.P. Hougardy, "Transformation of Steels during Cooling", In *Theory and Technology of Quenching* (edited by B. Liscic, H.M. Tensi, and W. Luty), Springer (1992).

Joseph Oyedele and Chung-Hyun Goh, "Machine Learning-Based Inverse Predictive Modeling of the TTT Diagram." *International Journal of Modern Engineering Research (IJMER)*, vol. 10 (02), 2020, pp 07-18.