GAUSSIAN PROCESS REGRESSION BASED GAS TURBINE PERFORMANCE DECK GENERATION

Emrah Güllü
TUSAŞ Engine Industries Inc.
Hacettepe Üniversitesi Teknoloji Geliştirme Bölgesi
Üniversiteler Mah. 1596. Cadde Safir F Blok
Bina No:6 Kat No:4 Ofis No:5
06800 Beytepe Çankaya / Ankara
emrah.gullu@tei.com.tr

Volkan Aran
TUSAŞ Engine Industries Inc.
İstanbul Teknoloji Geliştirme Bölgesi
Sanayii Mah. Teknopark Bulvarı
No:1/7A 7301
34906 Pendik / İstanbul
volkan.aran@tei.com.tr

Abstract—**A gas turbine "performance deck" is an engine model in the form of a computer program which can simulate engine performance for varying input parameters. A performance team is usually held responsible to produce such a computer program in a gas turbine project and it is transferred to other project teams and customers for their needs. This computer program may include complicated component models and an iterative matching algorithm or it may include bunch of tables to interpolate/extrapolate data from. Although the first approach is way to go for the use of a performance specialist, the latter is better suited to transfer such a computer program outside the performance team. This is due to the fact that iterative matching algorithms can have convergence problems and produce no result. Tabulation of the performance data can be done with a grid based approach where each input parameter is divided into grids and an n-dimensional table is created, where n is the number of input parameters. Although it is quite straightforward to obtain such a table, its size increases exponentially with the number of input parameters. Considering that number of input parameters of a performance deck vary between 5-10, even with relatively coarse grids for each input parameter (5-10), number of performance points required to be solved can reach millions easily. This type of a table can take days to produce. We propose using Gaussian Process Regression (GPR) in order to tackle this problem. GPR is a non-parametric regression algorithm whose computational complexity does not depend on the number of the input dimensions. GPR is also the state of the art technique for the internal combustion engine modeling for the automotive industry. GPR can be evaluated as a gray-box modeling since the model preserves physical data and estimates distribution between them in place of total parametric abstraction. In this study, a GPR based performance deck is created for NASA T-MATS JT9D engine model and quite promising results are obtained.**

TABLE OF CONTENTS

1. INTRODUCTION 1
2. GAUSSIAN PROCESS REGRESSION 2
3. GPR ESTIMATION OF JT9D TMATS MODEL 3
4. RESULTS 3
5. CONCLUSIONS 4
REFERENCES 5
BIOGRAPHY 6

1. INTRODUCTION

Gas turbine engine performance computer programs, commonly known as "performance decks", have been used to help design engines themselves, their control, condition monitoring and diagnosis systems and platforms these engines will be mounted to since the early 50s. The first computer program that is able to simulate performance of a gas turbine was developed in the early 50s at NACA Lewis Laboratory. They used component performance maps, constant gas properties and an iterative procedure, called as the "component-matching", to make sure that flow and work compatibilities are achieved. In the mid-60s, the efficiency of the numerical procedure used for the component-matching is improved by the Air Force Aero Propulsion Laboratory using the well-known Newton-Raphson root finding method [1] The component-matching using Newton-Raphson iterations has been the industry standard since then. All the gas turbine companies all around the world rely on either a commercial software, such as GasTurb [2] and NPSS [3], or their in-house codes which are built using the same component-matching algorithm today. Details of this algorithm can be found in books of Saravanamuttoo et al. [4], Walsh and Fletcher [5] and Kurzke and Halliwell [6].

Although the component-matching is a reliable and accurate tool for gas turbine engine performance simulations, it has two main drawbacks; convergence and computational time. Since the component matching is an iterative procedure which tries to solve a highly non-linear problem, it is hard to guarantee that it will converge to a solution in a pre-defined range of input parameters. Computational time is an issue especially for control applications since they require real-time models. Linearization or a simplified form of the component-matching is usually employed to achieve real-time engine models compromising some accuracy [7]. An n-dimensional table, where n is the number of input parameters, created with performance data obtained from a component-matching model can solve both of these problems. It can guarantee a solution in a pre-defined range of input parameters and it can do it in real-time with no-problem. However, size of this table increases exponentially with the number of input parameters, limiting the capability and the accuracy of the model. Instead of a grid-based tabulation approach, a non-parametric regression algorithm can be used to fit a black-box model which are quite popular in machine learning literature.

Application of machine learning algorithms to obtain a gas turbine engine performance model is investigated by many researchers. Chiras et al. [8] employed artificial neural network (ANN) to model the relationship between fuel flow and shaft speed of a Rolls Royce Spey engine. They compared the obtained model with a previously created linearized model and concluded that their model is superior. Asgari et al. [9], Lazzaretto and Toffolo [10] and Badihi et al. [11] also applied ANN to obtain gas turbine performance models. They all used performance data created with a component-matching model and concluded that the ANN-based models are in a good agreement with the component-matching models. Bruner [12], who was working in Raytheon Aircraft Company at the time, fitted an ANN to a flight data and successfully estimated engine performance parameters within 3% error margin. He reduced the computational time of their data analysis from weeks to hours. All of these studies show the capability of machine learning algorithms to accurately capture the complex non-linear nature of gas turbine engines.

Gaussian Process Regression (GPR) or Kriging (named after Daniel Krige a famous mining engineer) is a major engineering approach for geostatistics since 1951 [13]. Its application is initiated as a spatial interpolation tool and found wider applications with increasing theoretical development in other engineering disciplines as well. Gaussian process regression (GPR) models are utilized for online inverse estimation of the robotic systems [14]. Inner loop dynamics of the throttle valve is modelled by nonlinear autoregressive with exogenous inputs (NARX) model whose nonlinear part is a GPR [15]. Diesel engine fuel system dynamics are predicted with local GPR models in [16] for offline model based calibration. Further applications on diesel engine turbocharger and exhaust gas recirculation valve control is presented in [17] and [18]. Recently, an ECU supplier has introduced an advanced modeling unit in its ECU and online simulation of GPR models are now practical for the automotive industry. This paper aims to initiate GPR based gas turbine modeling studies.

A short introduction to Gaussian Process Regression is given in the following section. Utilized modeling process is explained in section 3 and modeling results are presented in section 4. Finally, conclusions are presented.

2. GAUSSIAN PROCESS REGRESSION

Gaussian probability distribution is a specific form of a general Gaussian process. A probability distribution defines random variables of scalars or vectors but the properties of functions are characterized with a stochastic process. Roughly, a function can be considered as an infinitely long vector with input x to output f(x) couples. If one samples a finite number of points and searches for the properties of the function, then the same answer will be found with Gaussian process inference as if all points were considered [19].

Formally, each y observation is assumed to be a sample of underlying function f(x) with a Gaussian noise model as

$$y = f(x) + N(0, \sigma_n^2) \tag{1}$$

where σ_n is the standard deviation of the noise. Prior covariance on the noisy observations y_1 and y_2 is defined as follows

$$cov(y_i, y_j) = k(x_i, x_j) + \sigma^2 \delta_{ij} \tag{2}$$

Covariance function $k(x_i, x_j)$, also called kernels, is defined over input samples x_i and x_j, and δ_{ij} is the Kronecker delta function. Covariance function can be estimated with various representations and squared exponential is the most popular one. Definition of $k(x_i, x_j)$ for the squared exponential covariance, which is used in this study, terms is given as

$$k(x_i, x_j) = \sigma_d e^{-0.5 r^T r} \tag{3}$$

where σ_d so-called horizontal scale parameter and r is a scaled input sample given as

$$r = \left[\frac{x_{i1} - x_{j1}}{l_1} \; \frac{x_{i2} - x_{j2}}{l_2} \ldots \frac{x_{in} - x_{jn}}{l_n}\right]^T \tag{4}$$

where so-called length scale parameters l_j determine weights ,therefore relevancies, of input channels.

For an experiment of m samples, following covariance matrix can be constructed.

$$K(X, X) = \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_m) \\ \vdots & \ddots & \vdots \\ k(x_m, x_1) & \cdots & k(x_m, x_m) \end{bmatrix} \tag{5}$$

The length scale l and the horizontal scale σ_d are main free variables of the model and they are called hyperparameters. These parameters are calculated by maximum likelihood estimation. Unlike parametric methods, training data are used for finding hyper-parameters and they are also embedded into the model through Kmatrix. The test values are denoted by x_*. The covariance vector between the test point and the training points is given as

$$k_* = [\mathrm{k}(\mathrm{x}_*, \mathrm{x}_1) \mathrm{k}(\mathrm{x}_*, x_2) \ldots k(x_*, x_m)]^T \tag{6}$$

Predicted output y_* is then calculated by

$$y_* = k_*^T (K + I\, \sigma_n^2)^{-1} y \tag{7}$$

Since the term $(K + I\sigma_n^2)^{-1}$ is fixed, an efficient form of (7) is presented as

$$y_* = k_*^T \alpha \tag{8}$$

where $\alpha \in R^m$and $\alpha = (K + I\sigma_n^2)^{-1} y$.

Maximum likelihood optimization cost for the subsequent optimization is defined as

$$\log(\,p(y|X)) = -0.5\,y^T\alpha - trace(\log(L)) - \frac{nlog(\pi)}{2} \quad (9)$$

where L is retrieved through the cholesky decomposition; i.e.

$$L = cholesky(K + I\,\sigma_n^2) \quad (10)$$

Overall training process of the model can be summarized as follows. For given measurements of inputs X, output y and measurement noise σ_n, first, select a covariance function (e.g. squared exponential). Minimize expected variance (or maximize likelihood) on training points by varying the hyper-parameters (parameters of the kernel or covariance function). After reaching the optimum hyper-parameters, one can estimate the output any given input x_* via (7).

3. GPR Estimation of JT9D TMATS Model

In this study, we constructed model of a model in order to show feasibility of the method. But same procedure is applicable to the real world data as well. Uniform random numbers are generated for the model inputs, namely deviation from standard ambient temperature (dTamb), inlet Mach number (MN), ambient pressure (Pamb) and fuel to air ratio (FAR) as shown in Figure 1. Gross Thrust, By-pass Ratio (BPR), High pressure compressor (HPC) flowrate, High Pressure Compressor (HPC) speed are the selected outputs for this paper. Modeling of additional outputs are also studied but not included in this paper.

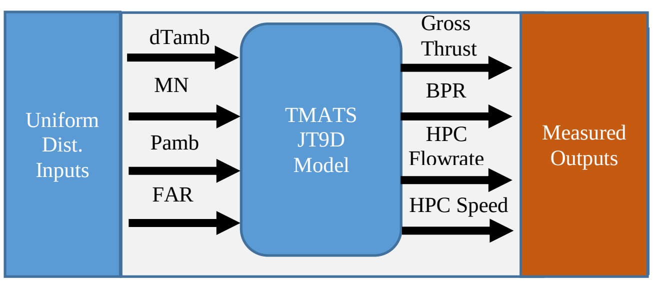

Figure 1. Model Inputs and Outputs

Standart inputs of the JT9D_Model_SS model are inlet total pressure (Pt), inlet total temperature (Tt), ambient pressure (Pamb) and fuel air ratio (FAR) [20]. Total pressure and total temperature values are calculated from dTamb and MN values via ambient atmosphere model of the TMATS [21]. Identically distributed independent input variables with uniform random distribution are generated for each input channel. The random numbers are resized to predetermined maximum and minimum values of the respective inputs. Simulations are run for each input sets on the TMATS model. Results of the converged simulations are used for the subsequent analysis.

Half of the converged simulation results are selected for training and remaining data is used for the validation. GPR is a nonparametric model and it embeds the training data and its outputs but still the (hyper) parameters of the correlation function (i.e. length scale l and the horizontal scale σ_d) should be determined. Optimum hyper-parameters are found using training data as depicted in Figure 2.

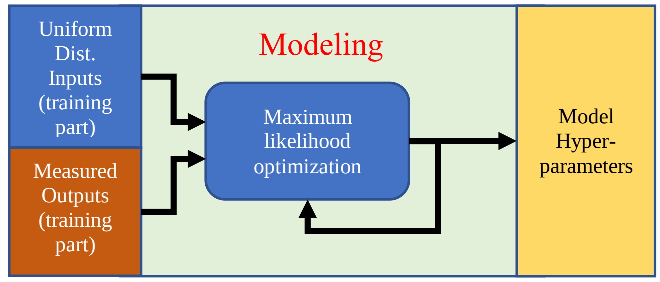

Figure 2. Model Training

Simulation with a GPR model is based on finding correlations between training data and the test inputs. Thus, model shall remember all the training data and hyper-parameters of the correlation function as in Figure 3. Weighted sum of test input to the training data is calculated as in (8).

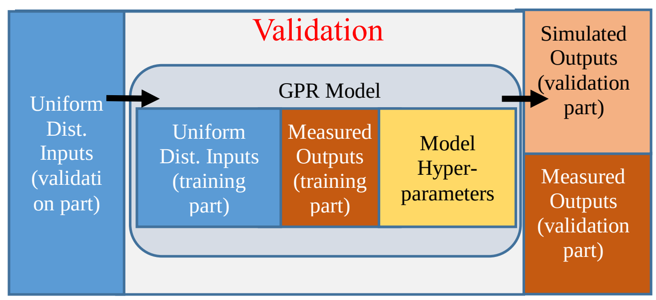

Figure 3. Model Validation

Additional training data should be created with the described procedure when validation results are not satisfactory. Random input experiment design allows adding new data without having any additional concern about the input distribution. Details of the respective fits and related arguments are explained in the results section.

4. Results

A total of 5000 performance points are simulated with T-MATS JT9D model and they are used as training and validation data. Half of the points are used for training and the remaining half are used for validation. It is decided by the authors that a variation between the predicted data and the testing data within 1% is acceptable. This is justified by the fact that even a change of gas model used in the component-matching model can cause its results to vary within 1%. It is presumed that higher than acceptable errors are resulted from inadequate training data density in the respective region of the input space.

Comparisons for gross thrust, bypass ratio (BPR), high pressure (HP) spool speed and HP compressor corrected mass flow rate (Wc) are shown in Figure 4. All validation results stay within 1% error interval but Gross Thrust. These

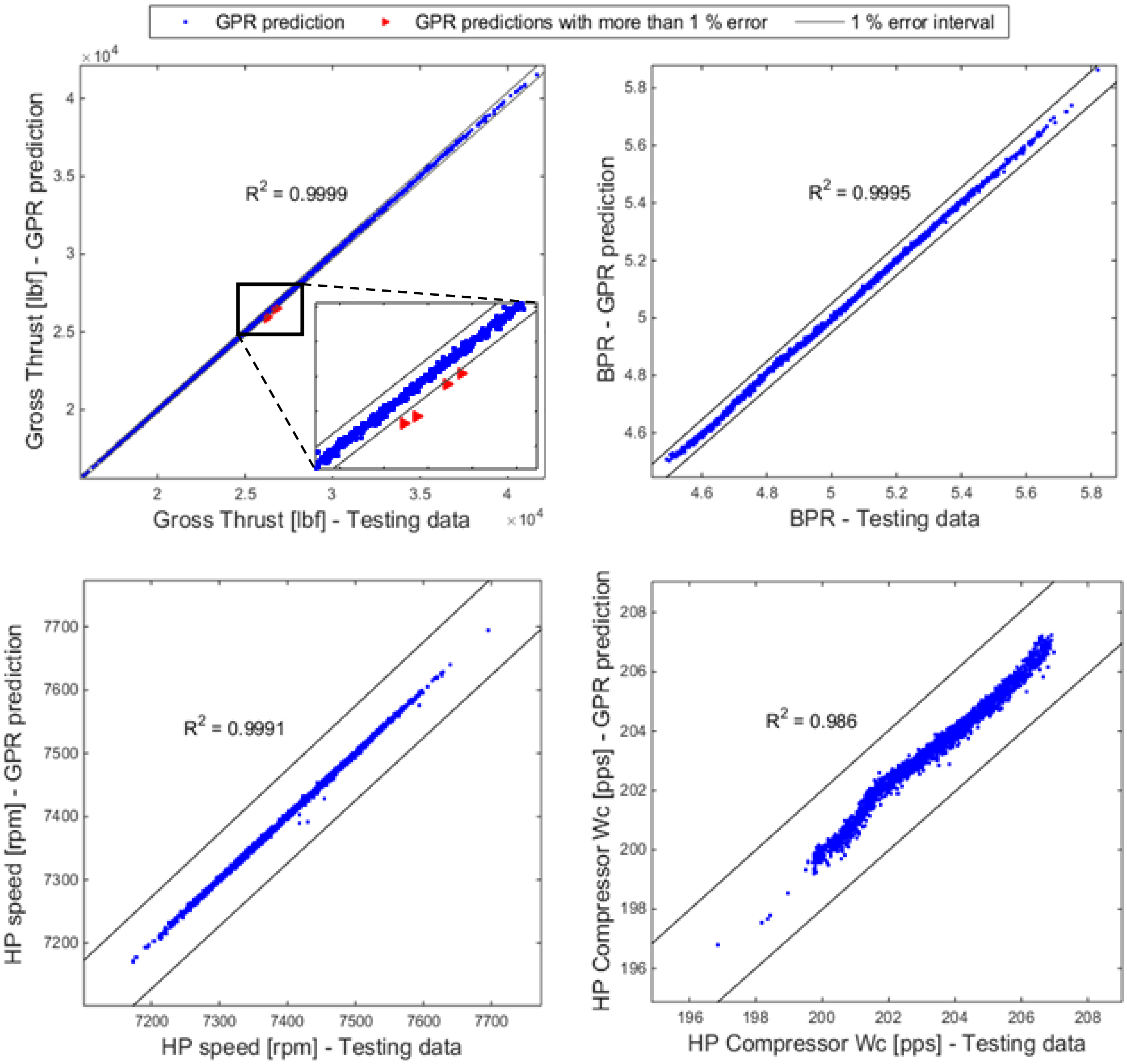

Figure 4. Comparisons of the predicted data with the testing data for Gross Thrust, BPR, HP spool speed and HP compressor Wc.

excessive error points are marked with red and further investigated. A GPR model has two main components embedded training data and hyper-parameters. Since most of the validation predictions of the output are well agreed with the measurement results, GPR hyper-parameters are assumed to be acceptable for the model. However, training data may not cover neighborhood of the specific outputs enough. The input space of the training data is depicted in Figure 5. This graph matrix shows all the input data and validation data with the excessive error values. It can be seen that high prediction error points are in the edges of multi-dimensional training input data space. Since GPR is an input correlation based modeling approach, this result is reasonable and should be taken into account in the model utilization.

Number of training points are increased from 500 to 2500 at intervals of 500 and change of root-mean-square (RMS), mean and maximum percent errors are listed for gross thrust predictions as shown in Table 1. This is done to see if adding more training points would be beneficial or not. Although it seems that increasing the number of training points a little bit more could be beneficial, even 500 training points seem enough for the purpose of this study.

5. CONCLUSIONS

In this study, it is aimed to investigate the potential of GPR to create black-box gas turbine performance decks from the data obtained with a component-matching model.

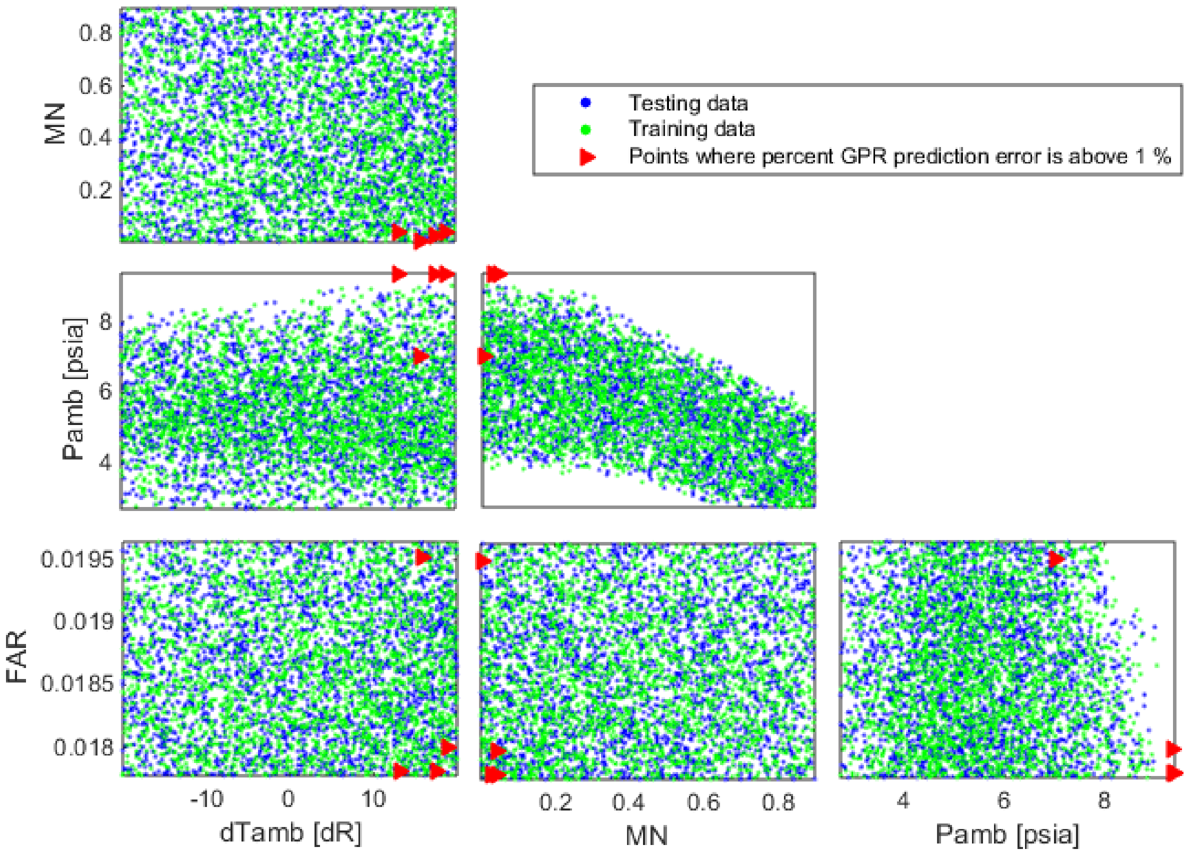

Figure 5. Scatter-plot of input space variables (dTamb, MN, Pamb and FAR). Points where percent GPR prediction error for gross thrust is above 1 % are marked with red.

Table 1 Change of percent errors with the number of training points for Gross Thrust prediction

# of training points	RMS % Error	Mean % |Error|	Max. % |Error|
500	0.29	0.19	3.87
1000	0.20	0.15	1.82
1500	0.18	0.13	1.81
2000	0.17	0.12	1.68
2500	0.16	0.11	1.66

Four selected outputs of TMATS JT9D steady state model is estimated with GPR method. Gross thrust is the most challenging output and its models reach error (RMS) under 0.3% using relatively small datasets. 500 training points are enough to obtain such a model for the problem. The most of the validation data points with more than 1% error values are observed in the boundaries of the input space. This type of GPR model should be used in a safe "distance" from the input space boundaries of the training data.

Computation of a single performance point takes 0.4 milliseconds using the GPR model constructed with 2500 training points while it takes 0.8 seconds using the TMATS component-matching model. A PC with 3.4 GHz i5-4670 CPU is used for the computations. Therefore, computational time of a GPR prediction is pretty insignificant after the model is trained; it can predict thousands of performance points in less than a second while the component-matching models could take hours to produce the same data. Training itself takes couple of minutes to optimize hyper-parameters; however, it is only done once.

This study will be extended to transient performance modeling and control.

REFERENCES

[1] L.C. Jaw and S. Garg. Propulsion Control Technology Development in the United States, A Historical Perspective. 2005. NASA/TM-2005-213978.

[2] J. Kurzke. Advanced User-Friendly Gas Turbine Performance Calculations on a Personal Computer. 1995. ASME 95-GT-147.

[3] J.K. Lytle. The Numerical Propulsion System Simulations. A Multidisciplinary Design System for Aerospace Vehicles. 1999. NASA/TM-1999-209194.

[4] H. Saravanamuttoo, H. Cohen and G.F.C. Rogers. Gas Turbine Theory. 5th Ed. 2001. Pearson Education, Ltd.

[5] P.P Walsh and P. Fletcher. Gas Turbine Peformance. 2nd Ed. 2004. Blackwell Science, Ltd.

[6] J. Kurzke and I. Halliwell. Propulsion and Power. 2018. Springer International Publishing.

[7] F. Lu, Y. Ye and J. Huang. Gas Turbine Engine Identification Based on a Bulk of Self-Tuning Wiener Models Using Fast Kernel Extreme Learning Machine. 2017. Energies.

[8] N. Chiras, C. Evans and D. Rees. Nonlinear Gas Turbine Modeling Using Feedforward Neural Networks. 2002. Proceedings of ASME TURBO EXPO.

[9] H. Asgari, X. Chen, M.B. Menhaj and R. Sainudiin. Artificial Neural Network-Based System Identification for a Single-Shaft Gas Turbine. 2013. Journal of Engineering for Gas Turbines and Power. ASME.

[10] A. Lazzaretto and A. Toffolo. Analytical and Neural Network Models for Gas Turbine Design and Off-Design Simulation. 2001. Int. J. Applied Thermodynamics. Vol 4.

[11] H. Badihi, A. Shahriari and A. Naghsh. Artificial Neural Network Application to Fuel Flow Function for Demanded Jet Engine Performance. 2009. IEEE Aerospace Conference.

[12] H.S. Bruner. The Analysis of Performance Flight Test Data Using a Neural Network. 2002. AIAA 2002-0989.

[13] Assibey-Bonsu, W.. The basic tenets of evaluating the Mineral Resource assets of mining companies, as observed through Professor Daniel Krige's pioneering work over half a century. J. S. Afr. Inst. Min. Metall., Johannesburg ,v. 116, n. 7, p. 635-643, July 2016

[14] Schreiter J., Nguyen-Tuong D., and Toussaint M. Efficient sparsification for Gaussian process regression, Neurocomputing, vol. 192, 2016.

[15] Bischoff B., Nguyen-Tuong D., Koller T., Markert H., and Knoll A., Learning throttle valve control using policy search, Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8188 LNAI,no. Part 1, pp. 4964, 2013

[16] Tietze N., Model-based Calibration of Engine Control Units Using Gaussian Process Regression, Ph.D. Thesis, Vom Fachbereich Elektrotechnik und Informationstechnik der Technischen Universitat Darmstadt. 2015

[17] V.Aran, M. Unel, Gaussian Process Regression Feedforward Controller for Diesel Engine Airpath, International Journal of Automotive Technology, Vol. 19, 2018

[18] V.Aran, Flexible and robust control of heavy duty diesel engine airpath using data driven disturbance observers and GPR models, Ph.D. Thesis, Sabancı University. 2019

[19] Rasmussen C. E. and Williams C., Gaussian Processes for Machine Learning, the MIT Press online version, 2016.

[20] Chapman J.W., Lavalle T.M., May R.D., Litt J.S. and Guo T., Toolbox for the Modeling and Analysis of Thermodynamic Systems (T-MATS) User's Guide, NASA/TM—2014-216638.

[21] Toolbox for the Modeling and Analysis of Thermodynamic Systems, TMATS v1.2.1, GitHub repository, https://github.com/nasa/T-MATS

BIOGRAPHY

Emrah Güllü *received a B.S. in Mechanical Engineering from Middle East Technical University, Ankara in 2013. He received his M.Sc. in Mechanical Engineering from Middle East Technical University in 2015. He is currently working as a performance engineer at TUSAŞ Engine Industries Inc., Ankara.*

Volkan Aran *received a B.S. in Mechanical Engineering from Middle East Technical University, Ankara in 2009. He received his M.Sc. in Industrial Engineering from Sabanci University in 2011. He has been with Ford-OTOSAN engine controls team between 2012-2019. He has received his Ph.D. in Mechatronics Engineering from Sabanci University in 2019. He is currently working at Tusaş Engine Industries, İstanbul.*