T.R.
GEBZE TECHNICAL UNIVERSITY
GRADUATE SCHOOL

APPLICATION OF SINDY (SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS) ALGORITHM TO GAS TURBINE ENGINE DYNAMICS

ALPER YILMAZ

A THESIS OF MASTER OF SCIENCE
DEPARTMENT OF ENERGY TECHNOLOGIES
APPLIED PROPULSION SYSTEM DESIGN AND ENGINEERING FOR AEROSPACE TECHNOLOGIES PROGRAM

ADVISOR: PROF. DR. ALİ MURAT SOYDAN

JULY 2025

T.R.
GEBZE TECHNICAL UNIVERSITY
GRADUATE SCHOOL

APPLICATION OF SINDY (SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS) ALGORITHM TO GAS TURBINE ENGINE DYNAMICS

ALPER YILMAZ

A THESIS OF MASTER OF SCIENCE
DEPARTMENT OF ENERGY TECHNOLOGIES
APPLIED PROPULSION SYSTEM DESIGN AND ENGINEERING FOR AEROSPACE TECHNOLOGIES PROGRAM

ADVISOR: PROF. DR. ALİ MURAT SOYDAN

JULY 2025

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

SINDY (SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS) ALGORİTMASININ GAZ TÜRBİN MOTOR DİNAMİĞİNE UYGULANMASI

ALPER YILMAZ

YÜKSEK LİSANS TEZİ
ENERJİ TEKNOLOJİLERİ ANABİLİM DALI
HAVACILIK VE UZAY TEKNOLOJİLERİNDE UYGULAMALI İTKİ SİSTEMİ TASARIM MÜHENDİSLİĞİ PROGRAMI

DANIŞMAN: PROF. DR. ALİ MURAT SOYDAN

TEMMUZ 2025

MASTER of SCIENCE JURY APPROVAL FORM

A thesis submitted by Alper YILMAZ, defended on 09/07/2025 before the jury formed with the 30/06/2025 date and 2025/34 numbered decision of the GTU Graduate Administration Board, has been accepted as a MASTER of SCIENCE thesis in the Department of Energy Technologies, Applied Propulsion System Design and Engineering for Aerospace Technologies Program.

JURY

MEMBER
(THESIS ADVISOR) : Prof. Dr. Ali Murat SOYDAN

MEMBER : Prof. Dr. Ali ATA

MEMBER : Assoc. Prof. Dr. Ümran KURTAN

APPROVAL

Gebze Technical University Graduate Administration Board
…..…/…..…/..…… date and ……/…….. numbered decision.

SIGNATURE/SEAL

To my mother, father, sister and
my dear lovely wife

ABSTRACT

The Sparse Identification of Nonlinear Dynamical Systems (SINDy) is an algorithm used to discover the underlying physical equations of complex nonlinear dynamical systems. This data-driven algorithm is characterized by low computational cost and high interpretability, making it widely used in system modeling and governing equation discovery across various scientific disciplines. However, its application in the field of gas turbine engine dynamics remains limited in existing literature.

This study aims to model the dynamic behavior of gas turbine engines using the SINDy algorithm. Gas turbine engines exhibit nonlinear dynamic characteristics, which can be modeled mathematically by thermodynamic equations. SINDy employs sparse regression techniques to identify dominant terms from a library of candidate functions, deriving a differential equation. This method achieves a balance between model complexity and accuracy. SINDy offers a simple, interpretable solution that mitigates overfitting. These attributes enable SINDy to offer a novel data-driven perspective to gas turbine engine modeling.

The study examines the operational principles of gas turbine engines, focusing on dynamic modeling, investigating data analysis techniques, and data driven system modeling and the theory of SINDy and its application to gas turbine engine. Specifically, the study explores the use of SINDy in analyzing transient behavior, developing data-driven system models of the engine, estimating unmeasurable parameters, and detecting faults. The performance of SINDy-based models is compared with alternative system modeling approaches. The applicability and advantages of SINDy are discussed.

The study demonstrates that the SINDy algorithm offers a data-driven solution for addressing gas turbine engine challenges.

Keywords: SINDy, Gas Turbine Engine, Sparse regression, Data-Driven Models, Gas Turbine Modeling

ÖZET

Doğrusal Olmayan Dinamiklerin seyrek modellenmesi (SINDy), karmaşık dinamik sistemlerin temel fiziksel denklemlerini keşfetmekte kullanılan bir algoritmadır. Veriye dayalı bu algoritma, düşük hesaplama maliyeti ile başarılı sonuç alınabilmesi ve yüksek yorumlanabilirlik sunmasıyla öne çıkmaktadır. Ayrıca, birçok bilimsel alanda sistem modellemelerinde ve denklem keşiflerinde yaygın olarak kullanılmaktadır. Literatürde, gaz türbinli motor dinamiği alanında uygulaması sınırlıdır.

Bu çalışmada, gaz türbin motorlarının dinamik davranışlarının SINDy algoritması kullanılarak modellenmesi amaçlanmıştır. Gaz türbinli motorlar, doğrusal olmayan dinamik davranışlar sergilemekte olup matematiksel modellenmesi kompleks termodinamik denklemlerin çözümü ile mümkündür. SINDy, verideki davranışı oluşturulan muhtemel denklem seçeneklerinden baskın olanları seçerek seyrek regresyon teknikleri ile sistemi temsil eden en az sayıda terim içeren diferansiyel denkleminin bulunmasını sağlar. Modelin karmaşıklığı ve doğruluğu arasında bir denge kurulur. Veriye dayalı birçok algoritmaya kıyasla basit, aşırı uyumlamayı engelleyen, yorumlanabilir bir çözüm sunar. Bu özellikler, SINDy'nin gaz türbin motor modellenmesinde yenilikçi bir veri odaklı bakış açısı sunar.

Bu çalışmada, gaz türbinli motorların çalışma prensipleri incelenmiş, dinamik modellemeye ve veri analizi tekniklerine odaklanılmıştır. Regresyon yöntemleri ve yapay sinir ağlarını içeren veri tabanlı sistem modelleme teknikleri incelenmiştir. SINDy'nin altında yatan teori ve gaz türbinli motorlara uygulanışı detaylandırılarak: SINDy'nin geçici hal davranışların analizinde, motorun veri odaklı sistem modellerinin geliştirilmesinde, ölçülemeyen parametrelerin tahmininde ve arıza tespitinde kullanımı irdelenmiştir.

Netice itibariyle, SINDy algoritmasının gaz türbin motor dinamikleri uygulamalarında önemli avantajlar sunabileceği saptanmıştır.

Anahtar Kelimeler: SINDy, Gaz Türbinli Motorlar, Seyrek regrasyon, Veri Odaklı Modelleme, Gaz Türbin Modelleme

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Prof. Dr. Ali Murat SOYDAN for their guidance and support through the research. Also, I express my sincere gratitude to Mehmet Sinan UYANIK for valuable support.
I express my sincere gratitude to TUSAŞ Engine Industries Inc. for their generous supports in encouraging to academic progress.
I extend my sincere thanks to Emrah GÜLLÜ and Gökhan ARAN for their valuable contributions and feedback. Their insightful feedback and dedication to academic excellence were instrumental in shaping this thesis.
My heartfelt thanks also go to my managers, Gürhan KUZGUN, Mert ERK and my team leaders Hande ÖZCAN UZUN, Kerem SELVİ, Oktay ÇEVİK and Yücel SAYGIN for their understanding, flexibility which enabled me to balance professional responsibilities with this academic endeavor.
I am deeply indebted to my teammates for their collaboration and their supportive work environment. I would like to specially gratitude to Melih KOÇAK for his technical feedback and suggestions.
Finally, I am eternally grateful to my mother and father for their lifelong dedication, love and their sacrifices. To my sister, I extend my gratitude for her unwavering support, optimism and joyful presence.
Most profoundly, I owe an immense dept of gratitude to my dear lovely wife Beyza AYDEMİR YILMAZ for her steadfast love, endless support, boundless patience and belief in me. Your presence and love have been my inspiration.
Their belief in my potential has been a constant source of motivation. This work is as much theirs as it is mine.

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

Θ	:	Candidate Function Library
R	:	Gas Constant
T_{gas}	:	Gas Temperature
γ	:	Heat Capacity Ratio
Q	:	Heat Energy
h	:	Heat Transfer Coefficient
N_H	:	High Pressure Spool Speed
I	:	Inertia
N_L	:	Low Pressure Spool Speed
M or MN	:	Mach Number
W or m	:	Mass Flow Rate
$\bar{y}$	:	Mean Value
T_M	:	Metal Temperature
Fn	:	Net Thrust
y	:	Observed Value
Pw	:	Power
$\hat{y}$	:	Predicted Value
P	:	Pressure
e	:	Residual or error
N	:	Rotational Speed
Ξ	:	Sparse Regression Coefficient Matrix
cp	:	Specific Heat
σ	:	Standard Deviation
Ps	:	Static Pressure
Ts	:	Static Temperature
A_s	:	Surface Area
T	:	Temperature
Trq	:	Torque
Pt	:	Total Pressure
Tt	:	Total Temperature
AIC	:	Akaike Information Criterion
BIC	:	Bayesian Information Criterion
CC	:	Combustion Chamber
DMD	:	Dynamic Mode Decomposition
HPC	:	High Pressure Compressor
HPT	:	High Pressure Turbine
LPC	:	Low Pressure Compressor
LPT	:	Low Pressure Turbine
MPE	:	Mean Percentage Error
NARX	:	Nonlinear Auto Regressive with External (Exogenous) Input
RMSE	:	Root Mean Squared Error
RSE	:	Residual Sum of Error
RSS	:	Residual Sum of Square
SINDy	:	Sparse Identification of Nonlinear Dynamics

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

For over centuries, gas turbine engines have diverse applications across various industries, including power generation, marine, automotive and aviation, with both civil and military usage. Principally, gas turbine engines are classified into four categories: turbojet, turbofan, turboshaft and turboprop. Turbojet engines have the simplest structure, consisting of a compressor, combustion chamber, turbine and nozzle. The primary purpose of the engine is to accelerate the airstream through the components to generate thrust while increasing heat capacity by adding fuel into the combustion chamber. Turbofan engines are more complex versions of the turbojet engines, featuring an additional compressor, called as fan, which bypasses certain air to produce additional thrust and reduce fuel consumption. The bypass ratio of the turbofan engines varies depending on the intended use. Turboshaft and turboprop engines are optimized to produce shaft power rather than jet thrust with an additional turbine to extract heat energy and convert it into shaft power.

Brayton Cycle represents the gas turbine engine operation principle. Air comes from the atmosphere via intake. The compressor pressurizes the air, the combustor adds heat energy from the fuel to the airstream, turbines extract the power and generate energy to proceed continuous cycle. Heated and accelerated air exits from the nozzle. Thus, thrust is generated.

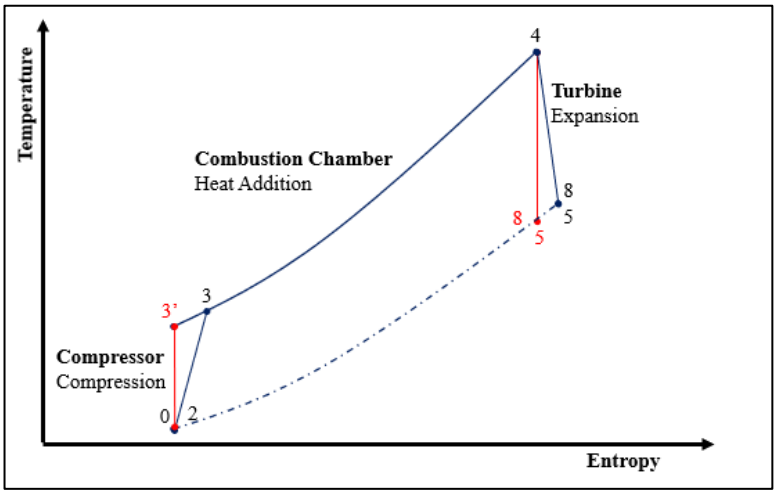


Figure 1.1 Bryton Cycle Temperature- Entropy (T-S) diagram for Turbojet Engines

Figure 1.1 illustrates the temperature – entropy diagram of simple turbojet engines. Entropy does not change during the ideal compression and expansion process which is illustrated with red lines. This hypothetical process is isentropic which is adiabatic and reversible. However, it is not the real case. Compression and expansion events occur with loss which increases the entropy. The actual compression process involves greater temperature rise and similarly, actual turbines expand the air to a higher temperature. Also, during the heat addition process and in progression of the air through the engine, there are pressure drops in real cases. Blue lines represent the real Brayton Cycle **[1, 2, 3, 4, 5]**.

Thermodynamical performance models of engines guide the design process. These models establish the fundamental requirements for aerodynamic and mechanical design. They derive the overall performance criteria and objectives of the engine. There are several techniques for engine performance modeling. Primarily, performance modeling techniques are divided into two categories: dynamic modeling and data driven modeling. The most common dynamic modeling technique is zero-dimensional component matching type thermodynamic modeling. (Hereafter it is mentioned as dynamic or thermodynamic model.) The dynamic model completely represents engine behavior during operation. If there is information about component performance, thermodynamical equations and system dynamics, engine behavior can simulate with creating dynamic models. Therewithal, gas turbine engine producers enquire about the system behavior to direct the engine design, constitute the system requirements. Hence, dynamic models become crucial.

Basically, thermodynamic models are based on mass, energy and momentum conservation laws and include component performance maps, tables and mathematical calculations. Numerical methods are employed to estimate dependent variables while varying independents to achieve model convergence **[1, 6, 7]**.

On the other hand, data-driven models may require lack of information about the system or basically to understand the test performance of the system **[8, 9]**. Data driven modeling of gas turbines is one of the main scopes of the study. Hence, before focusing on the gas turbines data driven models, general data driven modeling techniques are investigated. **Figure 1.2** illustrates the diagram of system modeling techniques which includes the scope of the thesis. Upcoming sections explain the basic theory of the data

driven dynamical system modeling techniques and their literature applications to gas turbine engines.

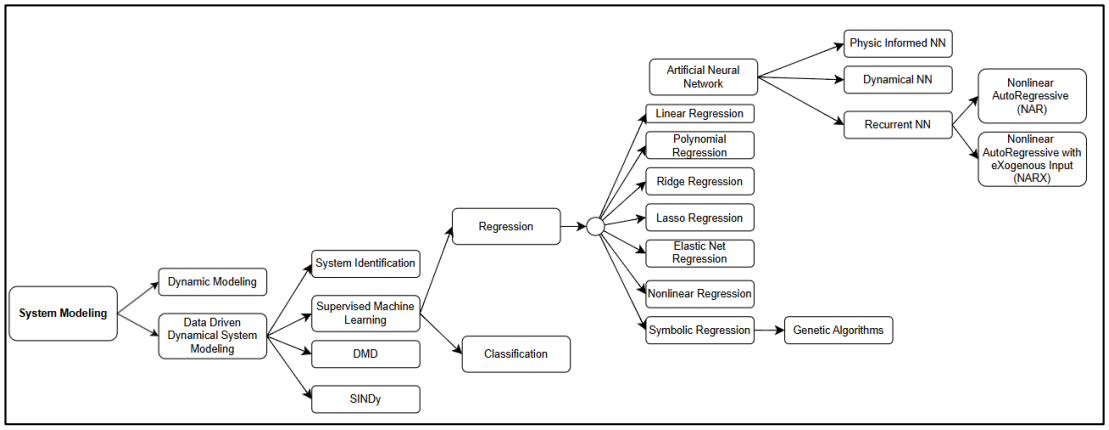


Figure 1.2 System Modeling Techniques

Discovering the governing equation of dynamical systems, modeling from data is critical part of engineering, natural science and applied mathematics problems. Many approaches are proposed to understand the physical relation, detection of anomalies, deriving the equations.

SINDy (Sparse Identification of Nonlinear Dynamics) is one of the recent interpretable algorithms to identify the governing equation of system with using fewest possible terms. It aims to avoid overfitting by promoting parsimony and maintaining a balanced level of complexity, robust to noise and capable of handling high-dimensional data **[10, 11, 12, 59]**.

Some industries including aerospace have strict regulations and certification requirements to make the use of interpretable logics and algorithms crucial. As a result, use of machine learning or artificial intelligence derived algorithms is not preferred for onboard control systems **[11]**.

SINDy has advantages to understand complex physical systems, allowing an interpretable approach and discovering the mathematical equation behind the unknown system behavior, and serving as an alternative to machine learning models, while also enabling identification of system with limited data **[12]**. **Table 1.1** demonstrates the advantages and disadvantages of SINDy.

Table 1.1 Advantages & Disadvantages of SINDy **[11,12,19]**

Advantages	Disadvantages
• Interpretability • Parsimonious modeling • Requires low amount of data compared to machine learning algorithms • Flexibility of the candidate function libraries • Applicable in many scientific areas • Avoid overfitting	• Effectiveness is highly dependent on chosen candidate library • Performance of SINDy may differ in a high dimensional system, some modifications may be required • Challenges with Stochastic Systems with random initial conditions requires multiple trajectories • Data quality, noise may affect to model

The thesis is structured into five primary chapters. Chapter 2 presents literature review, encompassing the fundamentals of gas turbine engines, dynamic engine modeling, data analysis methodologies, data-driven system modeling and their applications to gas turbine engines. This includes regression analysis, artificial neural network (ANN), Nonlinear Autoregressive Exogenous (NARX) models, Dynamic Mode Decomposition (DMD) and an in-depth examination of Sparse Identification of Nonlinear Dynamics (SINDy). Chapter 3 provides a detailed explanation of the methodology and approaches including developed engine model for data generation, introducing created SINDy model and its parameters and validation and comparison techniques. Chapter 4 demonstrates the results of the SINDy model, discussions, and proposed applications ideas to real life challenges. Finally, Chapter 5 offers the conclusion.

2. LITERATURE REVIEW

2.1. Gas Turbine Basics & Performance

Total temperature and pressure terms are used in gas turbine thermodynamic calculations. From the isentropic relations Stagnation Pressure (or Total Pressure) can be found as below formula:

$$\frac{Pt}{Ps} = \left(1 + \frac{\gamma-1}{2} M^2\right)^{\frac{\gamma}{\gamma-1}} \quad (2.1)$$

And the total temperature can be also calculated as:

$$\frac{Tt}{Ts} = 1 + \frac{\gamma-1}{2} M^2 \quad (2.2)$$

Turbomachinery component performances represented with component map which indicate the working mass flow rate, isentropic efficiency and pressure ratio of the component. Component maps of the JT9D turbofan engine are illustrated in **Figure 2.1**, **Figure 2.2** and **Figure 2.3** which are derived from the T-MATS **[6,7]**. The axis of the maps is mass flow rate and pressure ratio. Black lines represent the rotating speeds, the blue lines indicate isentropic efficiencies, and red lines indicate the compressors stall line. If two of the parameters-mass flow rates, pressure ratio, or speed-are known, the third can be determined. Additionally, Kurzke **[13]** offered auxiliary lines to facilitate the reading of maps by gas turbine performance calculation programs. In addition to this, corrected parameters are used in component maps. **Table 2.1** illustrates the corrected parameters and their calculations **[2]**.

Table 2.1 Gas Turbine Corrected Parameters

Performance Parameter	Symbol	Unit	Corrected Parameter
Temperature	T	K or °C	$\frac{T}{\sqrt{\theta}}$
Pressure	P	kPa	$\frac{P}{\delta}$
Mass Flow Rate	W, $\dot{m}$ or m	kg/s or lb/s	$\frac{W\sqrt{\theta}}{\delta}$
Rotational Speed	N	rpm	$\frac{N}{\sqrt{\theta}}$
Shaft Power	Pw	kW	$\frac{Pw}{\delta\sqrt{\theta}}$

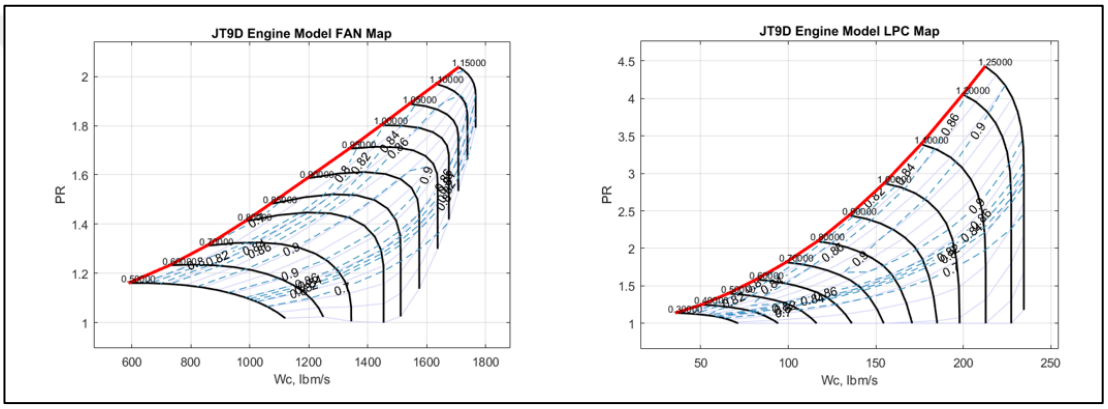


Figure 2.1 JT9D Engine Model Fan & LPC Map

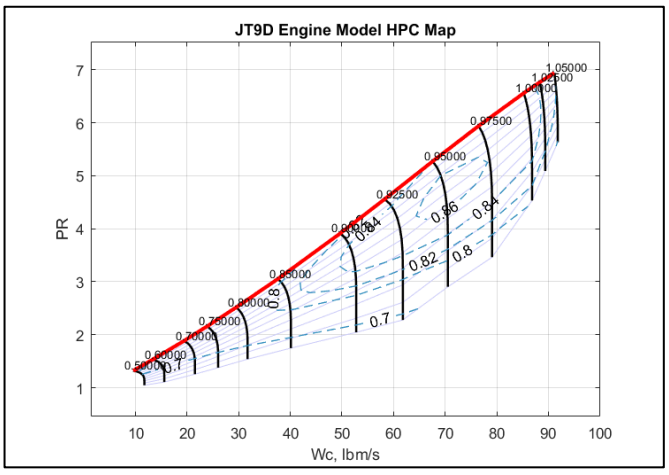


Figure 2.2 JT9D Engine Model HPC Map

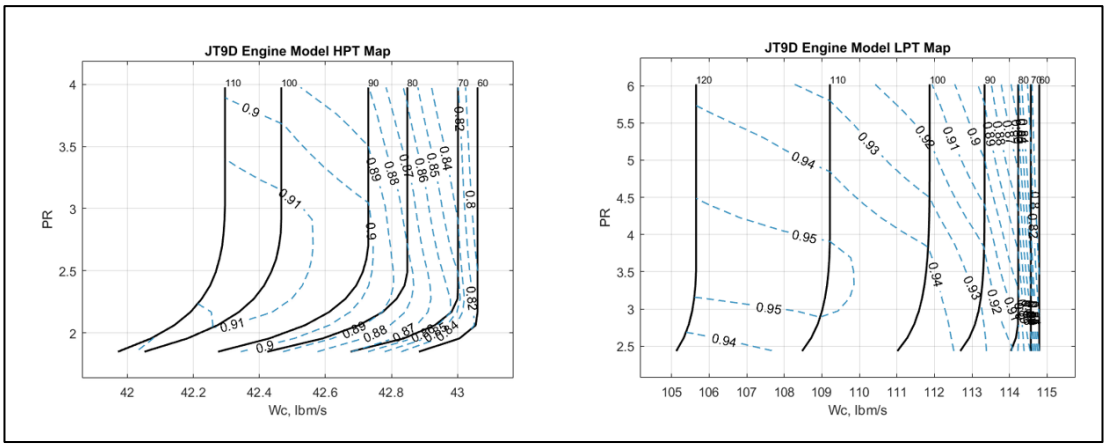


Figure 2.3 JT9D Engine Model HPT & LPT Map

Table 2.2 presents the engine stations definitions of JT9D engine which is the high bypass unmixed turbofan type engine **[6,7]**.

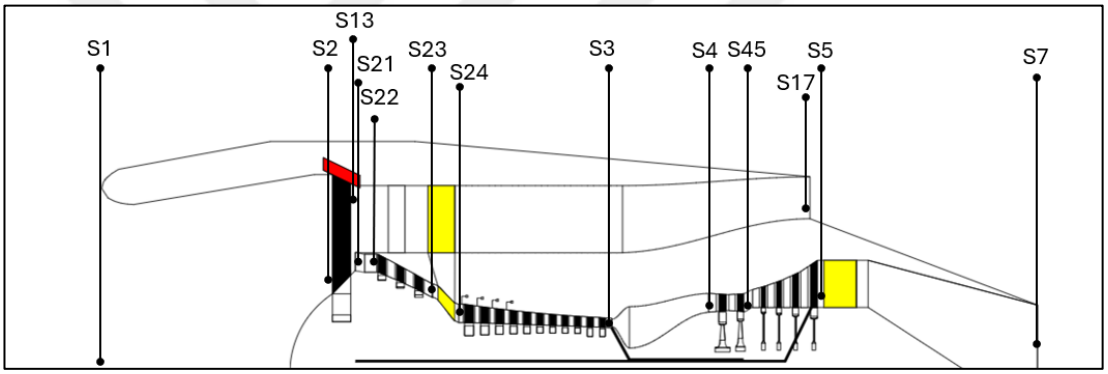


Figure 2.4 JT9D Engine Stations & Nomenclature **[6,7]**

Table 2.2 JT9D Dynamic Engine Model Station Numbering & Actions

Station Numbering	**Definition**	**Model Actions**
S1	Engine Inlet	Alt, MN and dTamb are converted to Tt & Pt
S2	Fan Inlet	Intake pressure loss is added
S13	Fan Bypass Exit	Fan component calculations are performed Flow is split according to bypass ratio. Flow enters the bypass duct
S21	Fan Core Exit	Fan component calculations are performed Flow is split according to bypass ratio. Flow enters the engine core.
S22	LPC Inlet	FAN-LPC inter-duct pressure loss is added.
S23	LPC Exit	LPC component calculations are performed.
S24	HPC Inlet	LPC-HPC inter-duct pressure loss are added.

Table 2.3 Continued: JT9D Dynamic Engine Model Station Numbering & Actions

S3	HPC Exit	HPC component calculations are performed. Cooling bleeds are extracted. HPC heat soakage is performed.
S4	Burner Exit	Burner component calculations are performed. Burner heat soakage is performed.
S45	HPT Exit	HPT component calculations are performed. HPT heat soakage is performed. HPT cooling flow is added.
S5	LPT Exit	HPT-LPT inter-duct pressure loss is added. LPT component calculations are performed. LPT heat soakage is performed. HPT cooling flow is added
S17	Bypass Nozzle Exit	Bypass pressure loss is added. Bypass flow exits from the nozzle.
S7	Core Nozzle Exit	Core flow exits from the nozzle.

2.2. Engine Dynamic Modeling

Engine dynamical modelling is based on thermodynamics and continuity, energy and momentum conservations laws. **Table 2.2** also illustrates the processes performed at each station during modeling. The basics of compressor, burner and turbine component modeling calculations are illustrated as flow diagrams respectively in **Figure 2.5**, **Figure 2.6** and **Figure 2.7 [6, 7, 1, 4]** .

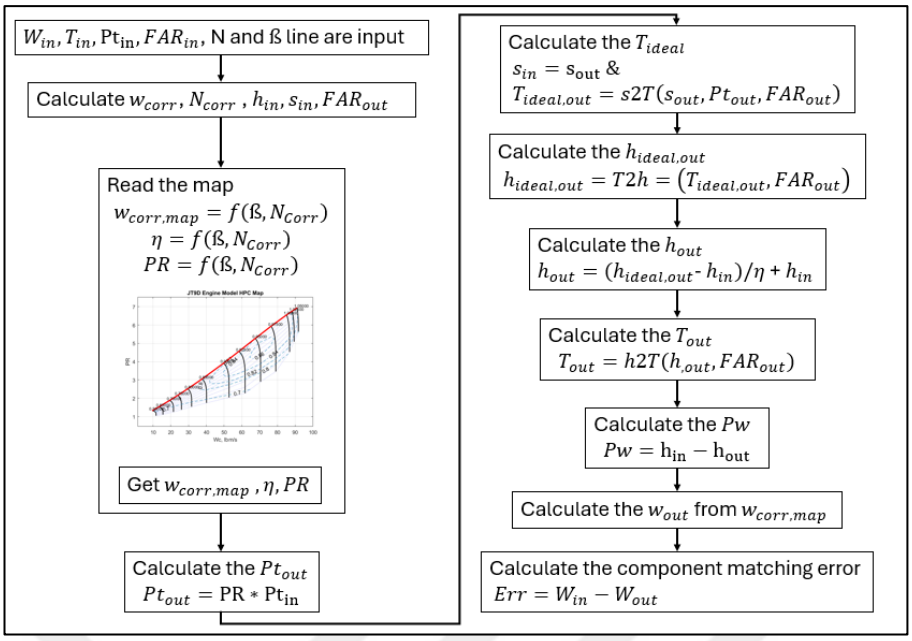


Figure 2.5 Flow Diagram of Basic Compressor Modeling Calculations

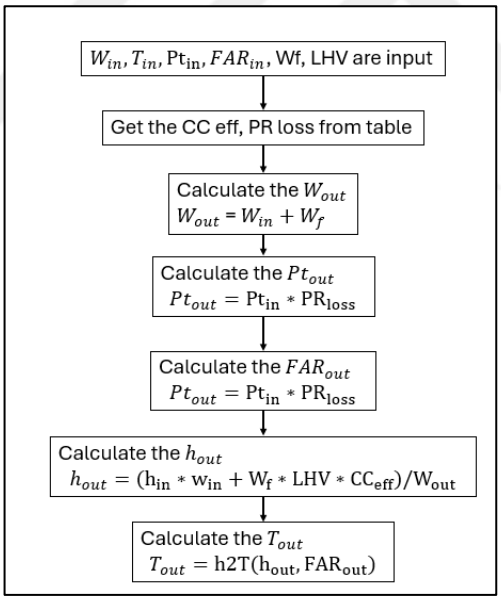


Figure 2.6 Flow Diagram of Burner Modeling Calculations

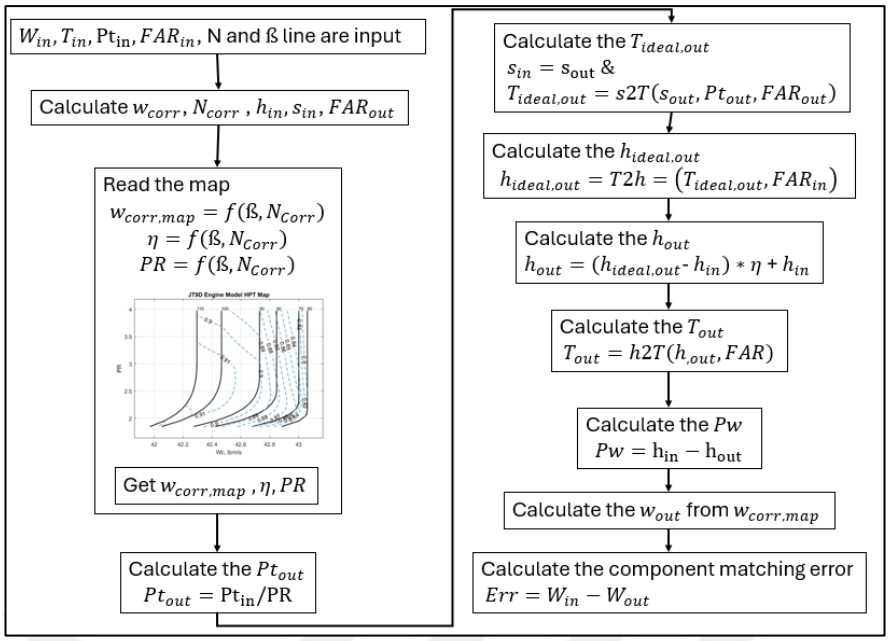


Figure 2.7 Flow Diagram of Basic Turbine Modeling Calculations

2.2.1. Transient Behaviors

The dynamics of the transient behavior of gas turbine engines are outlined in the **Figure 2.8**. Each dynamic has to be employed to represent the precise transient behavior of an engine.

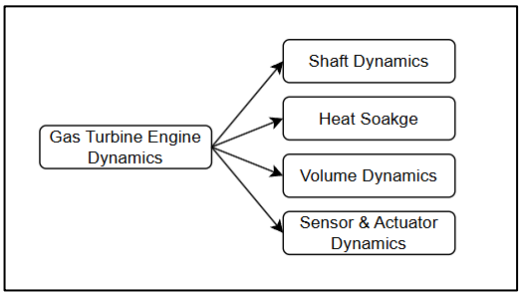


Figure 2.8 Gas Turbine Engine Transient Dynamics

2.2.1.1. Shaft Dynamics

Shaft dynamics is the main behavior of the engine. While the engine is rotating a steady-state condition, changing the fuel flow rate causes the power unbalance between the turbines and the compressors. This power imbalance results in the transfer of mechanical power between the compressor and turbine which are connected by a

shaft until equilibrium is restored. Basically, if more fuel is supplied than in the steady state, the shaft accelerates; if the fuel is reduced, the shaft decelerates **[1, 2]**.

$$\frac{dN}{dt} = \frac{T_{Net}}{I} \tag{2.3}$$

2.2.1.2. Heat Soakage

Heat soakage involves heat transfer between gas flow and metal parts during thermal disequilibrium. For the dynamic modeling aspect, heat transfer between complex geometries of the engine is simplified for practical. Generally, lumped capacitance models are preferred. **[1, 14]** Blades, disks and casings are assumed as bulk mass.

Heat transferred between the gas and metal can be calculated with following formula

$$Q = h * A_S(T_{gas} - T_m) \tag{2.4}$$

Also, it can be written as

$$m * cp * \frac{dT_m}{dt} = h * A_S(T_{gas} - T_m) \tag{2.5}$$

Thereby, the gas temperature can be calculated as

$$T_{gas} = T_m + \frac{m*cp}{h*A_s}\frac{dT_m}{dt} \tag{2.6}$$

With integrating the metal temperatures, a new state can be obtained.

2.2.1.3. Volume Dynamics

Volume Dynamics are high frequency dynamics of gas turbine engines. They represent the mass flow transfer between the volumes. Volume Dynamics become crucial when modeling of surge and stalls **[2]**.

$$W_{in} - W_{out} = Vol * \frac{\frac{dP}{dt}}{\left(1+\frac{\gamma-1}{2}M^2\right)^{\frac{1}{\gamma-1}}R*T} \tag{2.7}$$

2.2.1.4. Sensor Modeling

The measurements exhibit inherent delays and uncertainties. Sensors can be modeled as first-order lag where the time constant is the key parameter characterizing the delay. Time constant represents the time required to reach ~63.2% of its final value under step input. Additionally, uncertainties of the sensors can be effectively modeled using random Gaussian noise generation **[6,7]**.

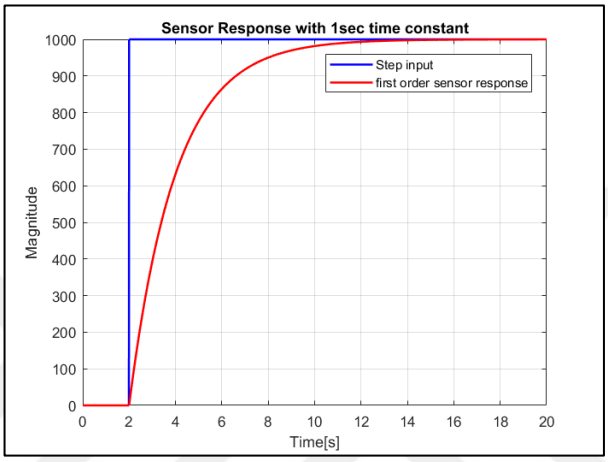


Figure 2.9 First Order Sensor Modeling with 1 second time constant value

Chapter 3.2 will describe these dynamics application into JT9D Engine Model.

2.3. Data Analysis

Some metrics and quality parameters are used to determine model performance:

The difference between the observed value from the model and the corresponding value is called as residual.

$$e_i = y_i - \hat{y}_i \tag{2.8}$$

The Least square approach is used to optimize the model to find the best fitting curve with minimizing the sum of squared residuals.

$$\text{Mimimize RSS} = \sum_{i=1}^{n} w_i(e_i)^2 = \sum_{i=1}^{n} w_i(y_i - \hat{y}_i)^2 \tag{2.9}$$

Where:

RSS is the Residual Sum of Squares or also called the sum of squared errors.

n: the number of observations.

y_i : Observed value of the variable.

$\hat{y}_i$: Predicted value of the variable.

w_i: Weight factor. It can be used for specific problems when the residuals have variance. However, for the basic approach, w_i can be taken as 1 **[10,15,16]**.

In practice, scientific programming of the regression on python, the library of "scikit-learn (sklearn)" uses least square approach to fit the regression models **[17,56,57]**.

2.3.1. Model Performance Metrics

Root Mean Squared Error (RMSE) can be used to decide the performance of the model to make predictions and comparisons between the models. Lower RMSE indicates better model performance.

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{y_i} - \hat{\mathrm{y}}_\mathrm{i})^2} \qquad (2.10)$$

Another important parameter, R^2 score**,** is a metric that indicates how well the model explains variance in the data. R^2 is a proportion explanation and ranges from 0 to 1 and higher R^2 indicates a better model. 1 corresponds to the perfect fit but also can be indicated of the overfitting into variables it may be problematic when extrapolation is needed.

$$\boldsymbol{R^2} = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \qquad (2.11)$$

Where:

$\bar{y}$: Mean of actual values.

Variance & Standard Deviation are fundamental statistical measures that indicate the dispersion of the dataset. Variance measures the average squared of deviation from the mean of the dataset. On the other hand, standard deviation is the square root of the variance **[10, 15, 16]**. Standard deviation is more comprehensive, expressed in the same unit as the original data.

Variance of population can be formulized as:

$$(\sigma)^2 = \sum_{i=1}^{n} \frac{1}{n} (y_i - \bar{y})^2 \quad (2.12)$$

For a sample variance instead of population, $n - 1$ is used instead of n.

And standard deviation can be expressed as:

$$\sigma = \sqrt{\sum_{i=1}^{n} \frac{1}{n} (y_i - \bar{y})^2} \quad (2.13)$$

In normal distribution of data set, 68% of data falls in one standard deviation and two standard deviations covers 95% of data set. Standard deviations are also used to compute confidence intervals-levels.

Residual Standard Error is squared root of RSS divided by the degrees of freedom of the model; this metric is a measure of accuracy of the regression model.

$$\mathrm{RSE} = \sqrt{\frac{1}{n-p-1}\mathrm{RSS}} \quad (2.14)$$

and

$$\mathrm{RSE} = \sqrt{\frac{1}{n-p-1}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \quad (2.15)$$

p: the number of independent variables. The term of $(n - p - 1)$ is called as degrees of freedom.

Lower RSE indicates goodness of predictions means that model's predictions are close to actual values. RSE is more direct measure of prediction error rather than $\mathbf{R^2}$ score.

Also, Mean Percentages Error (MPE) can be formulated as below:

$$\mathrm{MPE} = \frac{100\%}{n} \frac{y_i - \hat{y}_i}{y_i} \quad (2.16)$$

Akaike Information Criterion **[15, 16, 10, 18]** is one of the model selection criteria that penalized log-likelihood measure, used to find best approximated model to true model or data.

$$\mathrm{AIC} = -2\ln(\mathrm{L}) + 2\mathrm{p} \tag{2.17}$$

In other representation of AIC

$$\mathrm{AIC} = \mathrm{n}\ln\left(\frac{\mathrm{SSE_p}}{\mathrm{n}}\right) + 2\mathrm{p} \tag{2.18}$$

If the likelihood function increases score decreases therewithal number of terms increases, AIC score increases with slope of $2\mathrm{p}$ thus penalize non-parsimonious models. AIC does not depend on the sample size.

Where:

L **:** likelihood function under the fitted model

p: number of parameters

$\mathrm{SSE_p}$: sum squared error

Bayesian Information Criterion (BIC) **[10, 15, 16, 18]** is similar to AIC but additionally adding extra penalty term as the sample size. Higher the sample size increases to BIC score.

$$\mathrm{BIC} = -2\ln(\mathrm{L}) + \mathrm{p} * \ln(\mathrm{n}) \tag{2.19}$$

or

$$\mathrm{BIC} = \mathrm{n}\ln\left(\frac{\mathrm{SSE_p}}{\mathrm{n}}\right) + \mathrm{p} * \ln(\mathrm{n}) \tag{2.20}$$

Where:

n **:** number of data points or sample size

Lowest AIC/BIC scores may indicate the better model for given data.

2.4. Data Driven System Modeling

2.4.1. Supervised Machine Learning

Data-driven modelling of a system becomes apparent, when mathematical modelling is not possible due to unavailability of the system **[9]**. Machine Learning is one of the data driven techniques that allows to create models to make predictions and decisions. These methods enable the identification of patterns and relationships within data. The theoretical foundation of machine learning is based on regression techniques and categorized as supervised machine learning and unsupervised machine learning. Supervised machine learning involves two primary task types: regression and classification. Classification models predict categorical outputs and regression models predict continuous outputs **[11]**. Regression models are studies concern.

In the literature, machine learning techniques have been extensively applied to understand gas turbine engines nonlinear dynamics behavior such as the prediction of component and sensor faults and diagnosis, performance estimations, engine modelling **[8, 9, 20, 21, 22, 23, 27, 30, 32, 33, 34]**.

2.4.2. Regression Analysis and Modeling

Regression method is a fundamental statistical technique for modelling the relationship between dependent variables. Method helps to understand and predict the relationship between variables, commonly used in engineering, science & economics. **[15]** Regression models can be categorized into linear and nonlinear approaches.

2.4.2.1. Linear Regression

Linear regression model can be stated as follows:

$$Y_i = ß_0 + ß_1 X_i + \varepsilon_i \tag{2.21}$$

Y_i: Response variable

$ß_0$ and $ß_1$: regression coefficients

X_i: predictor variable

ε_i: random error component, Errors are assumed to have mean zero and variance are uncorrelated.

The relation between response and the predictor variables are linear. It is said to be "Linear in the parameters and linear in the predictor variables" **[16]**.

Let assume an experiment is done with a spring and various objects with known mass (independent variable) to observation of Hooke's Law and estimation of spring constant. The extension of a spring was measured when different masses were attached, and all data were collected. According to Hooke's Law, the extension of spring is proportional to the force applied, which is due to gravitation. In this experiment Linear Regression model will be fitted to estimate the spring constant. Another example is application of Linear Regression through measured data. Let assume certain temperature measurements were done in a field in a field and data were collected as a time variance as **Figure 2.10**.

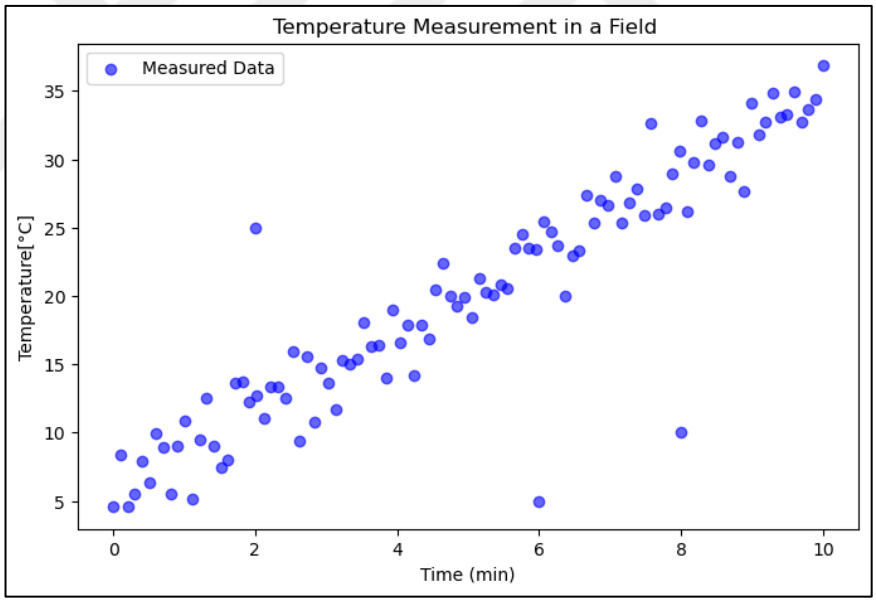


Figure 2.10 Temperature Measurement in a Field

Measurement has some random error, and one can observe that, temperature is increasing linearly with time. Regression models can be implemented to estimate temperature in present and predict future behavior.

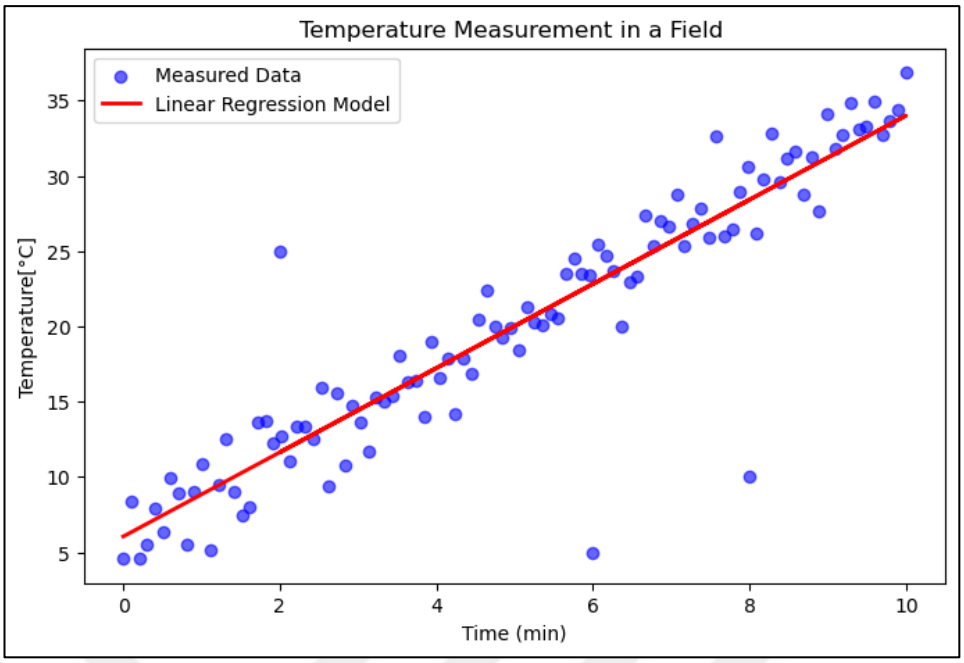


Figure 2.11 Temperature Measurement in a Field with Linear Regression Model

Regression Model with least squares approach is fitted to data and the regression equation is:

$$Y_i = 6.04 + 2.8X_i \tag{2.22}$$

Models' standard deviation of error is 3.4 and RSS value of 1193.99. Besides that, the accuracy of the model can be enhanced. One can observe that there is some suspicious data, these are not fit rationally to other measurements. It is called outlier. If the outlier data are removed from the model, the performance is improved.

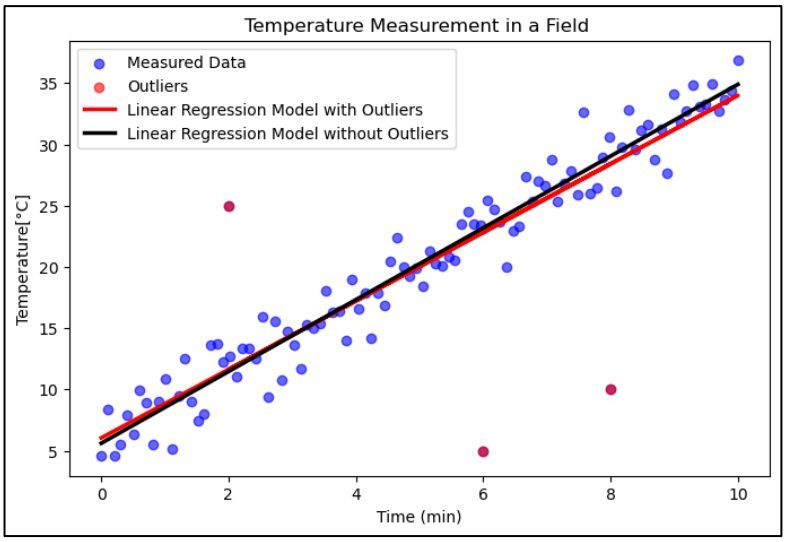


Figure 2.12 Linear Regression Model without outliers

Hence, the standard deviation of Error becomes 1.84°C and RSS value is 388.68. Here, model performance is significantly improved.

With obtained equation from the regression model, prediction of future behavior of the system can be done.

2.4.2.2. Polynomial Regression

Polynomial regression models are special cases of linear regression. Models contain higher order terms. For example, third order with two variables polynomial regression model can be shown as **[2]**:

$$Y_i = ß_0 + ß_1x_1 + ß_2x_2 + ß_3x_1^2 + ß_4x_2^2 + ß_5x_1^3 + ß_6x_2^3 + ß_7x_1^2x_2 + ß_8x_1x_2^2 + ß_9x_1x_2 + \varepsilon \quad (2.23)$$

2.4.2.3. Ridge Regression

Ridge regression, also known as L2 regularization, is one of the types of linear regression models and is a statistical regularization technique to reduce errors caused by overfitting. Rigge regression is used to correct multicollinearity when the least squares estimates are unbiassed. When the RSS score goes to 0, models perfectly fit the data however it could be indication of overfitting for training data set and tend to highly sensitive any minor variance in the new test input set, another word, model can be unstable if the coefficients are too high. Ridge regression is similar to the ordinary

least square method but there is another parameter that regulates the high value model coefficients. This term is called as regularization or penalty parameter in RSS function. Penalty term is the sum of squares of the model's coefficients **[24, 25]**.

Minimizing:

$$L2 = RSS + \lambda \sum_{i=1}^{n} (ß_i)^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{n} (ß_i)^2 \quad (2.24)$$

Where:

$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$: residual sum of squares term. It is a measure of the difference between the model prediction and the real term.

$\lambda \sum_{i=1}^{n} (ß_i)^2$: regularization term or shrinkage penalty is small when the ß model coefficients are close to zero. This term cannot be zero.

λ: Ridge parameter, penalty coefficient and should be $\lambda \geq 0$

$\sum_{i=1}^{n} (ß_i)^2$: sum of model coefficient

λ term decides the impact of the coefficient magnitude, controlling the trade-off between fitting the data and shrinking the coefficients. When the λ closes to the 0, the penalty term has no effect and ridge regression become ordinary least squares estimate. However, λ closes to the ∞, the impact of the shrinkage penalty grows then the coefficient estimations approach to zero and leading to underfitting to data.

An illustration of Ridge Regression Performance can be found in the example below.

Let be three features and dependent function as "x1, x2, x3 and y" in a random generated data to create nearly collinear features. The first and second features are almost identical but only x2 has noise and the third one is a sine function. Additionally, y value has a noise.

Let y value be:

$$y = 5 * \sin(2 * x * \pi) + 2 * x^2 + error \quad (2.25)$$

After that, data were split into two for training (60%) and test (40%). Then, trained with high order polynomial regression model (15th Degree) to prone to overfitting. RSS and Ridge Regularization terms were used. First $\boldsymbol{\lambda}$ penalty coefficient is chosen as 100.

On the **Figure 2.13** ordinary least square method (OLS) fit perfectly to training data However, test data performance of OLS is significantly lower with comparing to Ridge regularization in **Figure 2.14**.

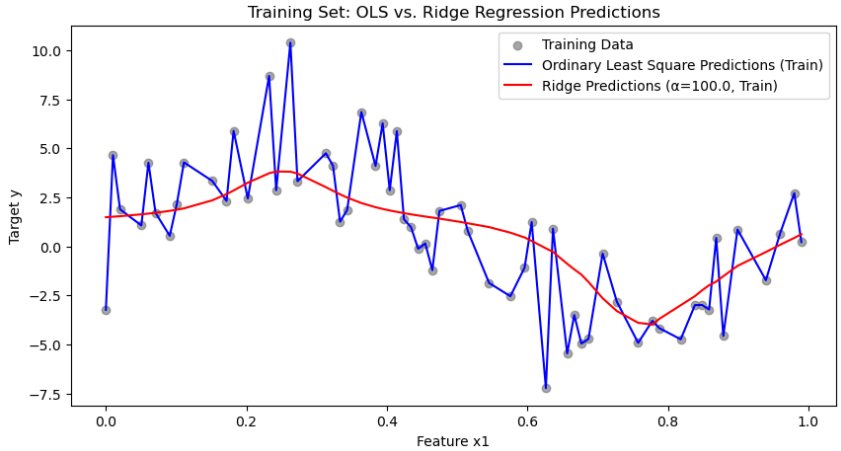


Figure 2.13 Ridge Regression Example Training Data Results

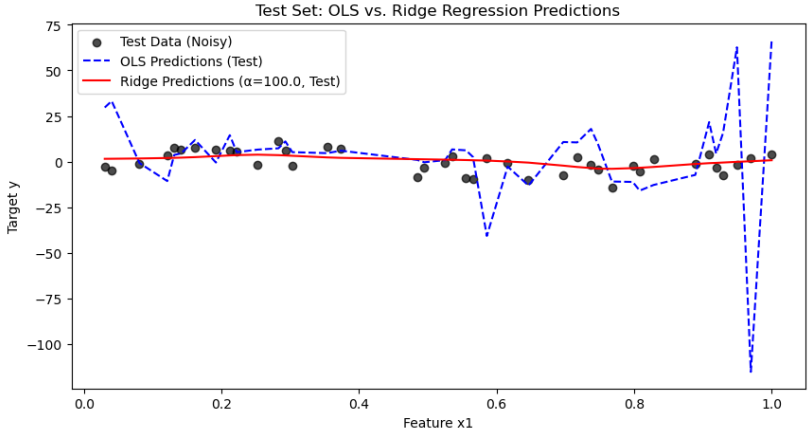


Figure 2.14 Ridge Regression Example Test Data Results

Another illustration in **Figure 2.15** shows the difference of prediction with Least Square method and Ridge method. Gray points show the true values, and red & blue points show the trained model results. While prediction with OLS has 734.87 MSE, Ridge regression has 6.1475 MSE.

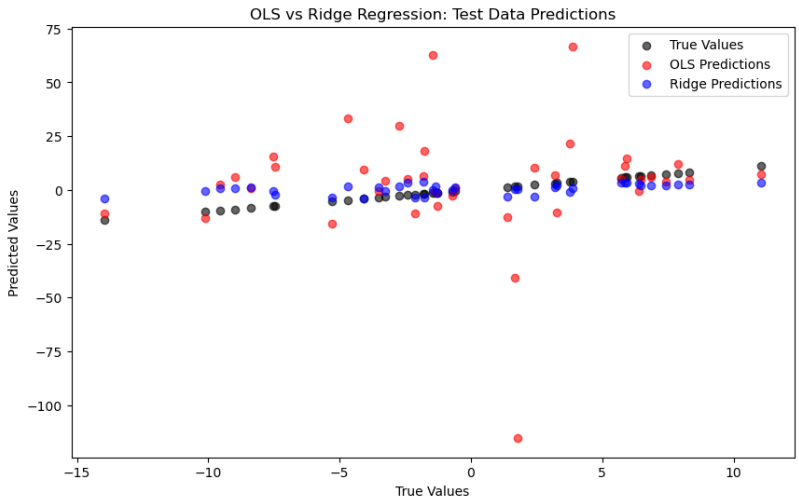


Figure 2.15 Ridge Regression Example Performance

It is understood that Regularized term overcomes the overfitting problem while decreasing the coefficient of parameters in this specific example. If the penalty coefficient is close to 0, Ridge regression closes to Least square method.

2.4.2.4. Lasso Regression

Least Absolute Shrinkage and Selection Operator (Lasso) regression is similar to Ridge regression but uses L1 penalty instead of L2. While L2 norms indicate sum of squares of coefficients, L1 norm uses sum of absolute value of coefficients. L2 penalty shrinks coefficients to zero but not able to become absolute zero, coefficients never equal zero in Ridge regression. However, Lasso regression allows to reduce coefficient to be zero. Lasso regression reduces the number of independent variables affecting the output, has advantages when feature selection is critical in high dimensions **[24, 25, 58]**.

Minimizing:

$$L1 = RSS + \lambda \sum_{i=1}^{n} |ß_i| = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^{n} |ß_i| \qquad (2.26)$$

2.4.2.5. Elastic Net Regression

Elastic Net Regression is a combination of Lasso and Ridge Regression, penalizes the L1 and L2 norm of the weights and can be formulized as below **[10,26]**.

Minimizing:

$$\text{RSS} + \lambda_1 \sum_{i=1}^{n} (ß_i)^2 + \lambda_2 \sum_{i=1}^{n} |ß_i| = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda_1 \sum_{i=1}^{n} (ß_i)^2 + \lambda_2 \sum_{i=1}^{n} |ß_i| \quad (2.27)$$

With $\lambda_1, \lambda_2 \geq 0$

Deciding the λ penalty term is a trade-off between bias and variance of the system. As bias increases, a model tends to have less accurate predictions on the training data set. Conversely, if variance increases, model predicts with lower accuracy on the test dataset. Ridge, Lasso, or Elastic Net may introduce higher bias in training dataset compared to the least square approach, which can result in a higher mean squared error (MSE) in training. However, these methods often achieve lower variance in the test dataset **[24]**.

2.4.2.6. Nonlinear Regression

Nonlinear regression is a regression technique, where the functional form is nonlinear in the parameters. The model may include exponential or logarithmic terms **[16]**.

Yazar et al. **[27]** applied various regression models to predict a compressor and turbines mass flow rates and efficiencies based on CFD data of radial compressor, high pressure and low-pressure turbines. The state parameters of regression models were speed and pressure ratios over a range of idle to maximum speed. They tried quadratic, logarithmic, exponential, linear model and compared their performance with AIC.

2.4.2.7. Symbolic Regression & Genetic Programming

The concept of genetic programming, formalized by Koza **[28]**, is a powerful paradigm for addressing complex optimization problems. Genetic programming is extensively employed across diverse scientific disciplines such as parameter estimation and optimization. Notable applications include the tuning the controller gains **[10]** and determination of optimum scaling factors for turbomachinery components maps **[29]**.

Schmidt and Lipson **[13]**, developed an automated approach to distill analytical natural laws from experiment data, creating a model without prior expert knowledge. They employed symbolic regression method, an evolutionary computational technique, for identifying nontrivial conservation laws by analyzing partial-derivative relationships in data and comparing symbolic derivatives from model to numerical partial

derivatives from data and improves their models with the find to best equation from random generated candidate symbolic functions including pendulums and harmonic oscillators.

Kim et al. **[30]** introduced an approach to prediction of gas turbine engine transient behaviors based on component map scaling factors. Also, heat soakage effects of components and thermocouples were added into their model in an attempt to match simulation results to test data. Applied scaling factors to component maps and heat transfer coefficient were optimized by using genetic algorithm (MIGA). They created a model for a low bypass F100 turbofan engine and verified them by applying their method to F404-GE-400 engine. **Figure 2.16** shows their method's flowchart **[30]** first adaptation cycle was done for steady state performance and second was for enhancing transient performance of their model. Hence, heat transfer correction factors optimization employed in second adaptation. They concluded that models normalized root-mean-square deviation was significantly reduced after employing heat transfer coefficient scaling optimization.

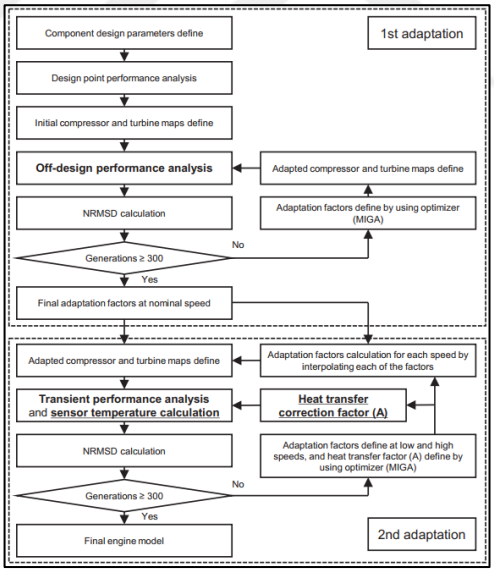


Figure 2.16 Flowchart for the engine performance adaptation using component scaling and Genetic Algorithm **[30]**

2.4.3. Artificial Neural Network (ANN or NN)

Neural Networks are characterized by their nodes and layered structure to learn patterns from data. There are various applications in gas turbine engines, some of them are mentioned below.

At the beginning of the 2000's Lazzaretto and Toffolo **[31]** implemented a two-layer feed-forward neural network (NN) topology to predict the performance of industrial 61.54 MW power plant gas turbine engine.

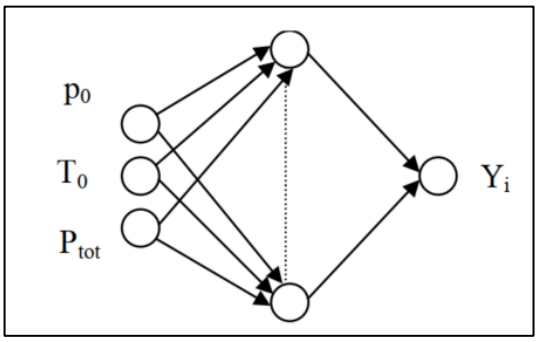


Figure 2.17 Neural Network Topology for the plant **[31]**

The model used ambient pressure and temperature and plant output power as inputs while the outputs included compressor outlet temperature, turbine outlet temperature, compressor inlet mass flow rate, fuel mass flow rate, and the compressor pressure ratio. A hyperbolic tangent sigmoid function was employed as hidden layer and output layer transfer function; however linear transfer function was also employed as output layer transfer function in some cases. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) and Levenberg-Marquardt backpropagation algorithms were used as learning algorithms. The authors reported the highest prediction error of approximately 1% across all output parameters.

Sina et al. **[20]** claims that Dynamical Neural Network (DNN) is an effective tool to fault detection and isolation of two spool jet engines. They focused on component faults such as decreasing efficiency and capacity of the turbines and compressors and created DNN models for each fault which are taking sensor measurements such as shaft speed, temperature and pressure measurements as input. Each fault has own DNN model to isolate fault from health model **Figure 2.18** shows the schematic diagrams of fault isolation.

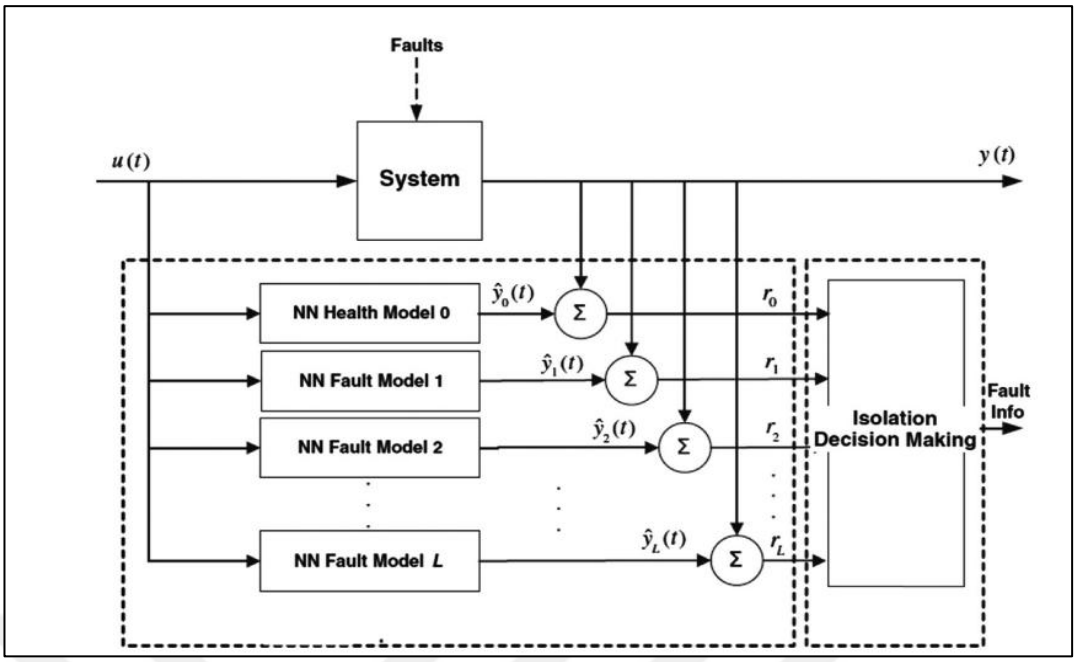


Figure 2.18 Sina et al. Proposed Fault Detection Schematic with Dynamic Neural Network Architecture **[22]**

Asgari et al. **[21]** studied black box modelling of a single shaft gas turbine engine with using Recurrent Neural Networks (RNN) as a part of Artificial Neural Networks. In RNN, each layer has a recurrent connection. RNN algorithm was adjusted to predict accurately of gas turbine engine dynamics. As selected inputs were fuel flow, load, ambient temperature and pressure so shaft speed, temperature and pressure of some points, compressor pressure ratio and efficiency of gas turbine were output parameters of RNN structure. Results claim that the percentage of RMSE is averagely smaller than 3% on test data.

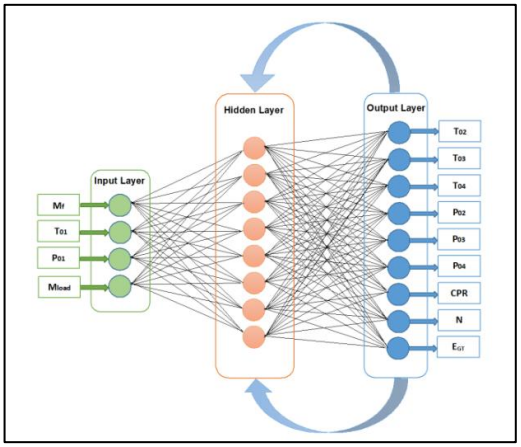


Figure 2.19 Asgari et al.'s Structure of the Recurrent Neural Network RNN 4-H-9 of a single shaft gas turbine engine **[23]**

Kim et al. **[32]** designed an approach to predict transient behavior of aircraft engines with employed ANN model with a radial basis function (RBF). They created an NPSS based model of low bypass F100-PW-100 turbofan engine from literature data called as first principle model to create data to train ANN model, and they built a simulation approach to enhance the performance of data-driven ANN model. Inputs of the model were fuel mass flow rate and ambient conditions. They claim that their proposed approach performs better than conventional ANN with RBF models, the R-squared values of the output parameters are higher than 0.98, one can show in **Figure 2.20**.

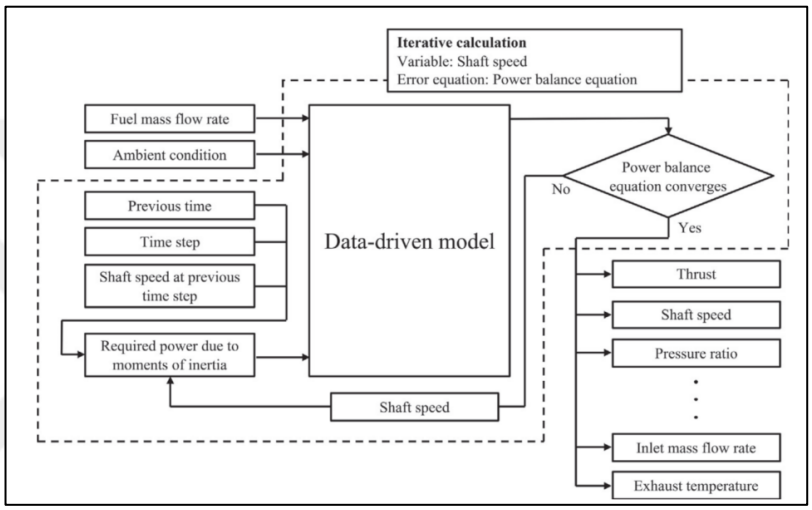


Figure 2.20 Kim et.al.' s approach to data driven ANN with RBF model for turbofan engine **[32]**

Liu and Karimi **[23]** investigated the performance prediction of heavy-duty gas turbine in a power plant by employing machine learning. They developed models to predict performance parameters, such as compressor and turbine operating characteristics and related gas temperatures based on various input variables and model structures. Specifically, they employed surrogate models including High Dimensional Model Representation (HDMR), and supervised machine learning techniques, such as ANN, trained on historical data. The ANN was configured as multi-layer perception (MLP) with one hidden layer. Their results claim that both HDMR and ANN models offer effective prediction of gas turbine characteristics. Furthermore, they developed correction curves that can serve as a basis for health monitoring and fault diagnosis of the engine.

2.4.4. Nonlinear Auto Regressive with External (Exogenous) Input Model (NARX)

NARX is a recurrent neural network structure that is characterized by a delay of input and output signals used for modeling time series data **[13]**. It predicts the current value of an output based on its past values. NARX model, as a recurrent neural network, has the capability of capturing dynamics of complicated systems such as gas turbines **[50]**.

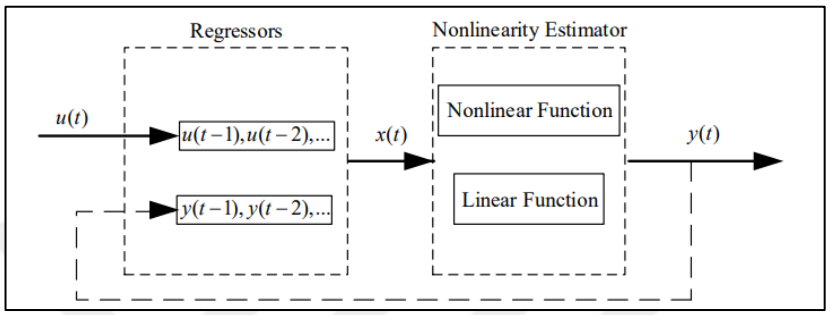


Figure 2.21 NARX model Structure **[8]**

Yu and Shu **[8]** introduced NARX identification modelling approach for high-bypass JT9D turbofan engine. Above idle engine data sets were generated by integrating the NASA T-MATS **[6, 7]** module for the prediction model. The fuel-air ratio (FAR) was used as the model input while high- and low-pressure spool speeds were predicted. They demonstrated that NARX model accurately identifies the dynamic characteristics of JT9D turbofan engine with the mean error value (Δn) of 0.008, where Δn was defined as

$$\Delta \mathrm{n} = \sqrt{\left(\frac{\mathrm{N_L}-\mathrm{N_L'}}{\mathrm{N_L}}\right)^2 + \left(\frac{\mathrm{N_H}-\mathrm{N_H'}}{\mathrm{N_H}}\right)^2} \tag{2.28}$$

Where:

N_L: low pressure shaft speed, N_H: high pressure shaft speed and N_L', N_H': predicted value.

Asgari et al. **[9]** applied NARX model to investigate the starting phase of heavy-duty-single-shaft gas turbine engine. They utilized over 1300 test data sets to train the model and validated it with test data under varying ambient and load condition. The observed RMSE values depended on training curves. The number of hidden layers and input parameters was selected to optimize performance while maintaining model simplicity.

The results demonstrated that NARX models have the potential to predict engine dynamics effectively. Furthermore, the authors noted data-driven models could provide diagnostic information for the gas turbines.

Pogorelov et al. **[13]** developed a dynamic model for gas turbines by using recurrent neural networks (RNN) with NARX. The model was trained using start-up, ground and flight test data from the engine and implemented in a hardware-in-the-loop (HIL) system with a Full Authority Digital Engine Control (FADEC) closed-loop-control. Consequently, the NARX model performed successfully in HIL test bed with a FADEC. However, the authors also noted that the neural network model exhibited reduced performance during sharp transition point such as starter shutdown and end of the acceleration.

Giorgi and Quarta **[33]** investigated the application of machine learning and artificial neural network techniques to predict Exhaust Gas Temperature (EGT) using their Viper 632-43 engine model, comparing the performance of these methods. Various input combinations, including atmospheric conditions, shaft speed, turbine inlet temperature, fuel mass flow rate, were tested using MultiGene Genetic Programming (MGGP) as Machine Learning techniques to explore mathematical relationship between input parameters and EGT. The results indicated that MGGP is an effective technique for EGT estimation. Additionally, the Nonlinear Autoregressive with Exogenous Inputs (NARX) model, employed as an ANN, was used to predict EGT for the next time step. The authors employed Bayesian Regularization, Scaled conjugate gradient and Levenberg-Marquardt as training algorithm, with They reported Bayesian Regularization has higher R-squared performance. However, they noted that NARX networks require the output parameter from the previous time step. In the case of output parameters that are unavailable, NARX models cannot be used.

2.4.5. Dynamic Mode Decomposition (DMD)

Dynamic Mode Decomposition is used to dimensional reduction of high dimensional systems and create a model how the system evolves in time. DMD is an equation-free, data driven method to obtain linear reduced order models allows estimate the spatial temporal modes without requiring any pre-knowledge about the system **[10, 34]**.

Basic application of the DMD start with collecting number of pairs of snapshots of the systems states.

$$\mathbf{X} = \begin{bmatrix} | & | & \dots & | \\ x_1 & x_2 & \dots & x_{n-1} \\ | & | & \dots & | \end{bmatrix} \tag{2.29}$$

And

$$\mathbf{X}' = \begin{bmatrix} | & | & \dots & | \\ x_2 & x_3 & \dots & x_n \\ | & | & \dots & | \end{bmatrix} \tag{2.30}$$

Where *n* represent snapshots, thereby each column are snapshots along to dynamics of the system. $\mathbf{X}$ matrices represent of each spatial measurement snapshot per $\Delta \boldsymbol{t}$, however $\mathbf{X}'$ is the matrices of shifted $\Delta \boldsymbol{t}$ time step into future.

DMD algorithm's objective is to find the best fit linear operator "A". Essentially, DMD approximates the leading eigen decomposition of the eigenvalues, computes the dominant eigenvalues and eigenvectors of A without computing totally A matrices.

$$\mathbf{X}' \approx \mathbf{AX} \tag{2.31}$$

Basically, on theory of DMD, $\mathbf{X}$ is recomposed with using singular value decomposition to find dominant coherent structure. Then, some matrix operations are computed with using reduced matrix to find eigenvalues and eigenvectors (called as modes). Finally, obtained low-rank structure represents the systems' future state behaviors **[10,35]**.

Krishnan and Sever **[34]** investigated the vibration response of an aero engine using Input-Output Dynamic Mode Decomposition (ioDMD) method. They found that ioDMD method requires full access to state of the systems, which is challenging for aero engines due to their limited measurement capabilities. To address this, the authors proposed a multi-resolution ioDMD approach, which enables the approximation of the states of the system. This approach improves capturing of dynamical behavior of measured vibration data **[34]**.

2.4.6. Sparse Identification of Nonlinear Dynamics (SINDy)

2.4.6.1. Lasso Regression

A matrix with most of its elements are zero defined as sparse matrix. For instance, A matrix is sparse matrix.

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0.96 \\ 0 & 1.21 & 0 \end{bmatrix} \tag{2.32}$$

LASSO regression can be used in parameter selection problems. $\boldsymbol{l_1}$ regularization manages the sparsity and weights the parameters relation and effectiveness to the system dynamic.

$$\xi = \underset{\xi'}{\operatorname{argmin}} ||\xi'\Theta - y||_2 + \lambda ||\xi'||_1 \tag{2.33}$$

Sparsity balances the complexity and accuracy of the model which can be in optimal position in a Pareto front. **[11, 36]** λ parameter weights the sparsity constraints. SINDy algorithm uses l_1 regularization or sequential thresholded least square (STLSQ) **[11, 12, 36]**. l_1 regularization has more sparse representation however STLSQ is most robust with noisy data **[19]**.

Pseudo logistic diagram of sequential thresholded least square (STLSQ) can be illustrated as follows.

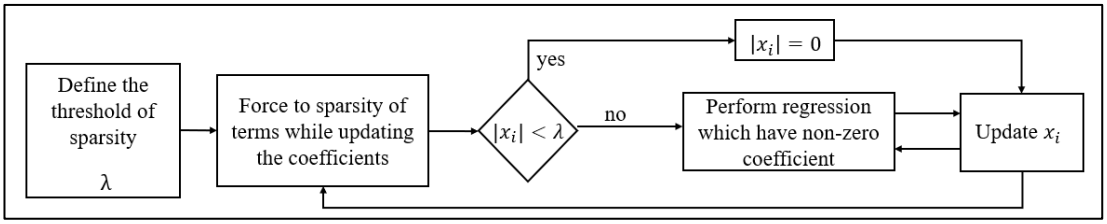


Figure 2.22 Logistic Diagram of Sequential Thresholded Least Square, based on Brunton et al. **[12]**

2.4.6.2. Theory

SINDy algorithm is explained in detail on the original paper. **[12]** The underlying theory and practical solution proposal of finding dynamical system governing equations have inspired this thesis. This section summarizes the theory of SINDy based on original paper **[10,12]**.

Below notation description is stated as follows. Scalar quantities represent in lower case letters (e. g. , x), vector quantities represent in bolt lower case letters ($e.g.,\boldsymbol{x}$), and bolt capital letters indicate matrices.

Let $\boldsymbol{x(t)} \in \mathbb{R}^{\boldsymbol{nx1}}$ is the $\boldsymbol{n}$-dimensional state vector at time $\boldsymbol{t}$.

$$\dot{\boldsymbol{x}}(\boldsymbol{t}) = \boldsymbol{f}\big(\boldsymbol{x}(\boldsymbol{t})\big) \tag{2.34}$$

Where,

$\dot{\boldsymbol{x}}(\boldsymbol{t}) = [\boldsymbol{x_1(t)} \quad \boldsymbol{x_2(t)} \quad \cdots \quad \boldsymbol{x_n(t)}]^T$ and $\boldsymbol{f}\big(\boldsymbol{x}(\boldsymbol{t})\big)$ is nonlinear function that define the system.

The object is that estimate the function $\boldsymbol{f}$ from collected time-based data where $\boldsymbol{f}$ consist of a few elements making it sparse in the space of possible functions.

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{x}^{\boldsymbol{T}}(\boldsymbol{t_1}) \\ \mathbf{x}^{\boldsymbol{T}}(\boldsymbol{t_2}) \\ \vdots \\ \mathbf{x}^{\boldsymbol{T}}(\boldsymbol{m}) \end{bmatrix} = \begin{bmatrix} | & | & \dots & | \\ \boldsymbol{x_1(t_i)} & \boldsymbol{x_2(t_i)} & \dots & \boldsymbol{x_n(t_i)} \\ | & | & \dots & | \end{bmatrix} \tag{2.35}$$

Where, $\boldsymbol{X} \in \mathbb{R}^{\boldsymbol{nxm}}$ is matrices collection of the data which is composed $\boldsymbol{n}$ - dimensional state and $\boldsymbol{m}$- dimensional time snapshots. $\boldsymbol{t_i}$ indicates the i-th time step.

And the derivative of $\boldsymbol{X}$ matrix can be represented as below. Derivatives can be measured or numerically calculated. Besides, filtering may be required due to high noise level.

$$\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{\mathbf{x}}^{\boldsymbol{T}}(\boldsymbol{t_1}) \\ \dot{\mathbf{x}}^{\boldsymbol{T}}(\boldsymbol{t_2}) \\ \vdots \\ \dot{\mathbf{x}}^{\boldsymbol{T}}(\boldsymbol{m}) \end{bmatrix} = \begin{bmatrix} | & | & \dots & | \\ \dot{\boldsymbol{x}}_{\boldsymbol{1}}(\boldsymbol{t_i}) & \dot{\boldsymbol{x}}_{\boldsymbol{2}}(\boldsymbol{t_i}) & \dots & \dot{\boldsymbol{x}}_{\boldsymbol{n}}(\boldsymbol{t_i}) \\ | & | & \dots & | \end{bmatrix} \tag{2.36}$$

Candidate nonlinear function library of $\boldsymbol{\Theta(X)}$ is constructed. Selection of the functions are arbitrary based on operator decision. Candidate functions can consist of polynomial, trigonometric or any other terms.

$$\boldsymbol{\Theta(X)} = \begin{bmatrix} | & | & | & | & \dots & | & | & | & | & | & \dots \\ \mathbf{1} & \boldsymbol{X} & \boldsymbol{X^{P_2}} & \boldsymbol{X^{P_3}} & \dots & \boldsymbol{X^{P_m}} & \mathbf{sin}\,(\boldsymbol{X}) & \mathbf{cos}\,(\boldsymbol{X}) & \mathbf{sin}\,(\mathbf{2}\boldsymbol{X}) & \mathbf{cos}\,(\mathbf{2}\boldsymbol{X}) & \cdots \\ | & | & | & | & \dots & | & | & | & | & | & \dots \end{bmatrix} \quad (2.37)$$

$$\boldsymbol{\Theta(X)} = \begin{bmatrix} | & | & | & | & \dots & | & \dots \\ \mathbf{1} & \boldsymbol{X} & \boldsymbol{X^{P_2}} & \boldsymbol{X^{P_3}} & \dots & \boldsymbol{X^{P_m}} & \cdots \\ | & | & | & | & \dots & | & \dots \end{bmatrix} \quad (2.38)$$

To illustrate, the above candidate function library consists of first degree ($\boldsymbol{X}$), quadratic ($\boldsymbol{X^{P_2}}$), cubic ($\boldsymbol{X^{P_3}}$) and nth degree polynomial terms. Also, trigonometric functions are stacked into matrix. Quadratic term from the candidate library can be expanded as below.

$$\mathbf{X^{P_2}} = \begin{bmatrix} \boldsymbol{x_1^2(t_1)} & \boldsymbol{x_1(t_1)x_2(t_1)} & \dots & \boldsymbol{x_2^2(t_1)} & \boldsymbol{x_2(t_1)x_3(t_1)} & \vdots & \boldsymbol{x_n^2(t_1)} \\ \boldsymbol{x_1^2(t_2)} & \boldsymbol{x_1(t_2)x_2(t_2)} & \dots & \boldsymbol{x_2^2(t_1)} & \boldsymbol{x_2(t_2)x_3(t_2)} & \vdots & \boldsymbol{x_n^2(t_1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{x_1^2(t_m)} & \boldsymbol{x_1(t_m)x_2(m)} & \dots & \boldsymbol{x_2^2(t_1)} & \boldsymbol{x_2(t_m)x_3(m)} & \vdots & \boldsymbol{x_n^2(t_1)} \end{bmatrix} \quad (2.39)$$

Likewise cubic polynomials or trigonometric functions can be expanded. Operator is free to choose any nonlinear function while creating of candidate function library.

Thereby, general equation of nonlinear system can be written as below.

$$\dot{\mathbf{X}} = \boldsymbol{\Theta}(\mathbf{X})\boldsymbol{\Xi} \quad (2.40)$$

Where $\boldsymbol{\Xi}$ is sparse regression coefficient. Matrix can be illustrated as below.

$$\boldsymbol{\Xi} = \begin{bmatrix} | & | & \dots & | \\ \boldsymbol{\xi_1} & \boldsymbol{\xi_2} & \dots & \boldsymbol{\xi_n} \\ | & | & \dots & | \end{bmatrix} \quad (2.41)$$

The aim of the algorithm is that which arbitrary function is active in each row and how the system behavior is represented in optimum and simple way with a few terms. Hence, sparse regression problem is solved to obtain sparse regression coefficient $\boldsymbol{\Xi}$ to

determine the active terms. On the Ξ matrix, only a few terms should be active while many elements should be zero otherwise, solution cannot be sparse.

For some cases derivate of the function have noise. Then the equation becomes

$$\dot{\mathbf{X}} = \boldsymbol{\Theta}(\mathbf{X})\boldsymbol{\Xi} + \boldsymbol{\eta}\boldsymbol{Z} \tag{2.42}$$

Where $\boldsymbol{Z}$ is a matrix of independent Gaussian noise with zero means. $\boldsymbol{\eta}$ is the noise magnitude. Measurement noise in the data can decrease the performance of SINDy. Hsin et al. **[19]** suggested that smoothing the noisy measurement with Gaussian process regression improves the effectiveness of SINDy, called as GPSINDy.

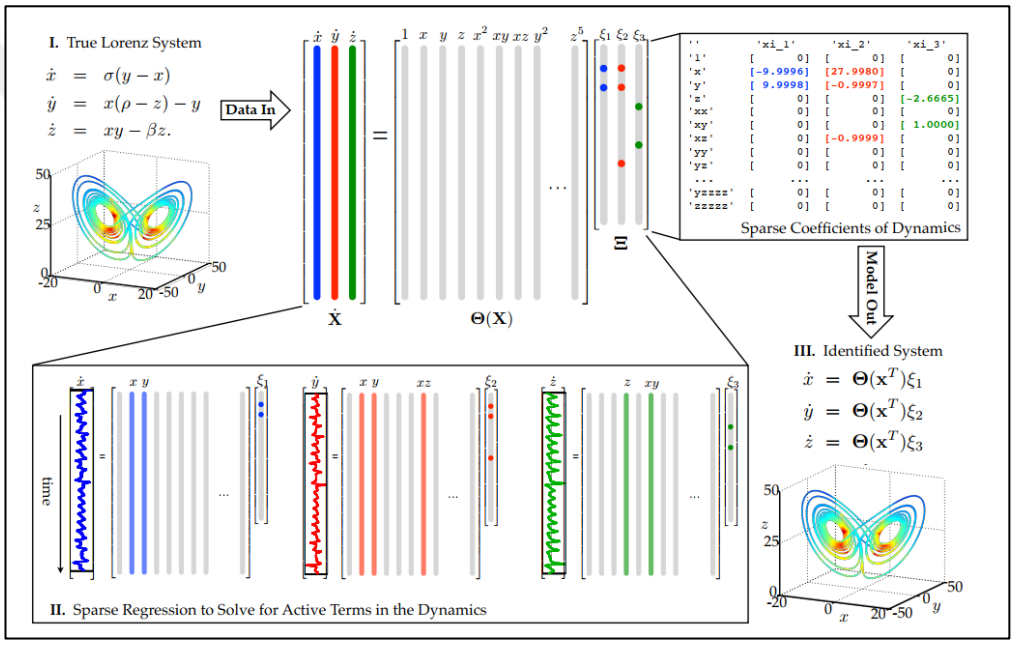


Figure 2.23 Schematic of the SINDy algorithm, demonstrated on the Lorenz Equation, Reprinted from Brunton et al. **[12]**

Figure 2.23 is the demonstration of SINDy algorithm under Lorenz system. Measurements are collected in time based on the states of $\mathbf{X}$ and derivatives $\dot{\mathbf{X}}$. $\mathbf{X}$ Matrix is consisted of $\mathbf{x}, \mathbf{y}\,\mathbf{z}$ states. Then, arbitrary candidate function $\boldsymbol{\Theta}(\mathbf{X})$ are constructed as consist of polynomial functions including first to fifth degree terms. Sparse regression problem is solved to find coefficient matrix $\boldsymbol{\Xi} = [\boldsymbol{\xi_1}, \boldsymbol{\xi_2}, \boldsymbol{\xi_3}]$. Colourful dots in the $\boldsymbol{\Xi}$ matrix represent the active terms in function. Identified system representation is demonstrated at the end of figure **[59]**.

SINDy performs well in system with low state dimensions and is applicable to various types of systems. However, Corberta **[11]** mentioned that SINDy may be less capable in higher-dimensional system, where determining the candidate library can be challenging. For stochastic system with random initial conditions, multiple trajectories are required to accurately determine dynamics, which may require more training data and that increase computational costs.

PySINDy is an open-source python package to apply the SINDy algorithm to scientific model discovery **[12, 37, 38]**. The package involves many customizable features for numerical differentiation, candidate function libraries (polynomial, Fourier), sparse regression techniques. Also, it includes tutorials, guidelines and many application examples. **Figure 2.24**, shows the basics of usage of PySINDy based on cited packages.

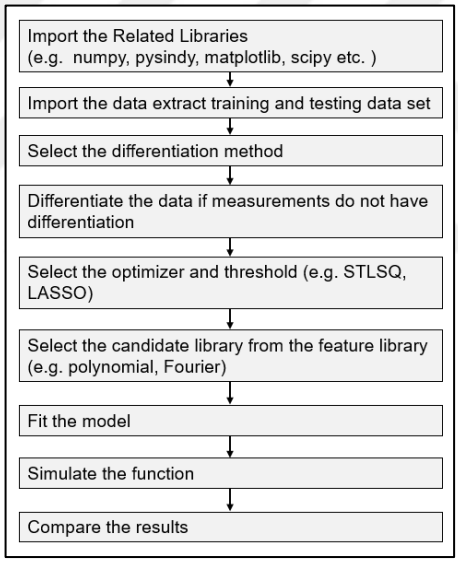


Figure 2.24 Flow Diagram of Discovering the model with PySINDy package **[37,38]**

2.4.6.3. Applications

Machado and Jones **[39]** proposed SINDy-SI, an enhancement of the Sparse Identification of Nonlinear Dynamics (SINDy) method, designed to model nonlinear dynamical systems using sparse, noisy data by incorporating Side Information (SI)-prior knowledge and constraints about the considered system. The authors highlight that many existing system identification methods produce unreliable models, violate

physical laws and exhibit unreliable behavior outside of the training data. Therefore, they propose SINDy-SI, integrates SINDy with Sum-of-Squares (SOS) programming, enforcing sparsity and physical consistency through iterative optimization while adhering to side information constraints.

The paper demonstrates SINDy-SI's performance through numerical experiment on the generated Lorenz system data with different noise scenarios and Single-Machine-Infinite-Bus (SMIB) systems. Results show that SINDy-SI outperforms methods like SINDy and Ordinary-Least-Squares, achieving a better balance in sparsity, accuracy, adherence to system properties.

The authors of **[40,60]** employed SINDY during the system identification phase to develop a nonlinear state space model of angular velocity dynamics for small-scale turbojet engines enabling to estimate thrust. They highlight the difficulty of direct measurement of thrust due to limited instrumentation of small turbojet engines. Hence accurate thrust estimations become challenging. To address this, authors propose a combination data driven method to estimate thrust: Grey-box modelling of the jet engine to identify the system with using SINDy method and EKF (Extended Kalman Filter) refines the model parameters and thrust estimation from angular speed measurement. They conclude to use second order candidate functions library based on angular speed, first and second derivative of angular speed and input signal, additionally to enhance accuracy they add steady state function derived from experimental data. They identify the system as below **[40]**:

$$\dot{\omega} = f(\omega, \dot{\omega}, u) = K_{ss}\left(\omega - a_1 u^{b_1} - c_1\right) + K_d\dot{\omega} + K_{\omega d}\omega\dot{\omega} + K_{\omega\omega d}\omega^2\dot{\omega} \quad (2.43)$$

Where K_{ss}, K_d, $K_{\omega d}$, $K_{\omega\omega d}$ are model parameters, $\boldsymbol{a_1}$, $\boldsymbol{b_1}$ are identified from regression on steady state data and $\boldsymbol{c_1}$ is idle angular speed.

Then, they use EKF to estimate thrust. Validation on a test bench with JetCat P160 and P220 turbojets shows an absolute mean error in thrust estimation below 2% of max thrust, even during engine failures **[40]**.

Numerical techniques allow accurate simulations of gas-turbine engine behavior nevertheless require significant computational power during real- time application. A turboshaft examples covers to prediction of delivered torque based on various flight data of the AW189 Twin Engine Leonardo's helicopter **[41]**. Authors employ

supervised data driven techniques architectures, which are Feed-Forward Neural Network (FFNN) and Long-Short-Term-Memory (LSTM) as a Multi Input Single Output (MISO) Model and SINDy to derive the relationship between torque and delivered fuel flow as low dimensional dynamical modelling. They conclude that FFNN is not able to predict the torque properly and LSTM has better predictions. On the other hand, SINDy allows interpretable, more accurate results without knowing underlying dynamics especially using second order model to train SINDy model **[41]**.

L'Erario et. al. **[42]** investigated data driven modelling for small turbojet engine's thrust. They applied second-order SINDy, Extended Kalman Filter (EKF) based identification and iterative Least Square methods to analyze thrust behavior of the engine in response to input and compared the method results with measured data. Their findings suggest that EKF based identification outperforms SINDy and iterative least squared method.

3. METHODOLOGY

From the scope of thesis, SINDy algorithm is employed to the gas turbine engines as a data-driven model. Identification and validation of data-driven modelling should be based on real engine ground or flight test data. Since accessing such a big flight and ground test data about real engines are company restricted, engine dynamic models can be employed to generate time dependent data as other data driven examples on literature. Concurrently, one of the fundamental principles of thesis is to conduct an open-source study, ensuring transparency and accessibility.

NASA T-MATS tool is employed to generate dynamic engine data **[6, 7]**. T-MATS tool was generated in MATLAB/Simulink environment. On the tool there is an example of Pratt & Whitney unmixed high bypass turbofan JT9D engine course model. Building upon the T-MATS structure and engine of this example, additional enhancements have been incorporated while maintaining fidelity to the original design. Detailed information about the modelling of JT9D engine will be provided in upcoming sections.

The PySINDy tool, a python package for SINDy **[37, 38]**, is used to generate SINDy model upon the JT9D engine. PySINDy tool provides systematic and basic approaches to application of SINDy algorithm. The research has focused on utilizing SINDy to propose solutions for the challenges encountered in gas turbine engines. Thrust estimation, turbine temperature prediction, fault diagnosis, and development of SINDy-based dynamic engine model are the main aspect of the research. **Figure 3.1** shows the methodology diagram of creating SINDy-based model.

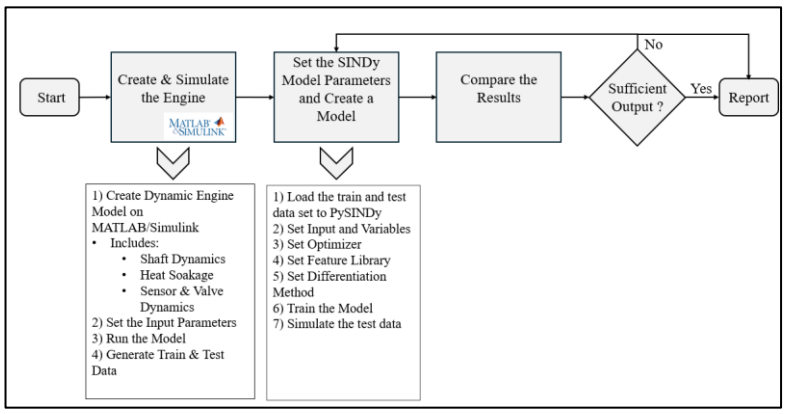


Figure 3.1 Methodology Diagram of Creating SINDy based Model

3.1. The JT9D Engine

The JT9D was Pratt & Whitney's (P&W) commercial high bypass unmixed turbofan engine. It was the first high-bypass ratio jet engine to power a wide-body aircraft. The program was launched in 1965, and the first engine tested in 1966. The engine powered the Boeing 747, Boeing 767, Airbus A300, Airbus A310, McDonnell Duglas DC-10 **[43,44]**.

Figure 3.2 Pratt& Whitney JT9D Engine Cutaways **[43, 44]**

The engine has a single stage fan, 3-stage low pressure and 11-stage high pressure compressors, annular type combustor and 2 stage high pressure and 4- stage low pressure turbine. The engine's weight is approximately 4150 kg. According to P&W website **[43]** engine characteristics can be tabulated as below. Since the engine has been in service for an extended period, its specifications may vary from one model to another.

Table 3.1 JT9D Engine Specification

JT9D Engine Specification	**British Unit**	**SI Unit**
Fan Tip Diameter	93.4 inches	2372.4 mm
Length, flange to flange	132.7 inches	3370.6 mm
Take-off thrust	48000-56000 lbf	231.5-249 kN
Flat rated temperature	86 °F	30 °C
Bypass Ratio	4.8	
Overall Pressure Ratio	26.7	
Fan Pressure Ratio	1.67	

From the literature review some useful specifications are collected and deduced. It should be noted that, the provided data depends on the variation model of JT9D engine **[6, 7, 45,46,47]**.

- Engine controlled with Electronic Engine Control (EEC) unit.
- Engine modulates the thrust with Engine Pressure Ratio
- HPC has variable stator vanes.
- JT9D engine uses handling bleed while acceleration to compensate surge margin of HPC.
- Inlet total temperature (Tt2), Inlet Total Pressure (Pt2), Low Pressure Spool Speed (NL), High Pressure Spool Speed (NH), Low Pressure Turbine Exit total temperature (Tt7, Exhaust Gas Temperature, EGT), Low Pressure Turbine Exit total pressure (Pt7), Compressor Exit Static Pressure (Ps3, however it called as Ps4 on literature but remaining faithful to research station numbering, considering as Ps3).
- Electronic Engine Control limits the engine thrust according to maximum allowable EGT and max spool speeds based on the ratings.
- Engine is cruising at ~0.85 Mach
- 100% Low pressure spool speed is 3750 rpm.
- 100% High pressure spool speed is 8000 rpm.

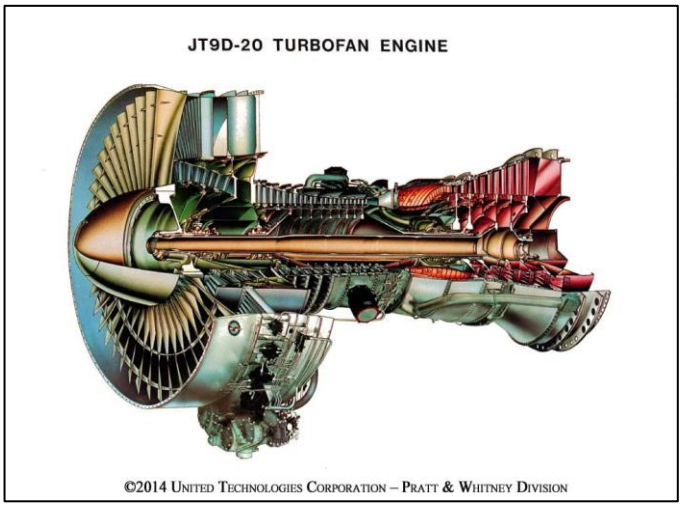


Figure 3.3 Pratt &Whitney JT9D Engine Cutaway **[43]**

3.2. JT9D Engine Model

JT9D dynamical engine model is developed in MATLAB/Simulink using NASA's Toolbox for the Modeling and Analysis of Thermodynamic Systems (T-MATS) **[6, 7]**. T-MATS provides a comprehensive library of gas turbine components, enabling modular design within Simulink. The model generates time-dependent solutions based on specific inputs. The overall comprises the JT9D engine component, sensor and actuator models.

The model's inputs include fuel flow rate (Wf) and atmosphere information such as pressure altitude (Alt), flight Mach number (MN) and delta from ISA condition (dTISA). These inputs are delivered to dynamic engine model which incorporates thermodynamic calculations to produce outputs, including temperature, pressure, mass flow rate and fuel air ratio at each station. Additional outputs include shaft speeds, metal temperatures and performance outputs such as thrust, specific fuel consumption (SFC). Selected outputs of the engine model are then transferred to sensor models which are sensed. **Figure 3.4** illustrates a simplified schematic of the model structure.

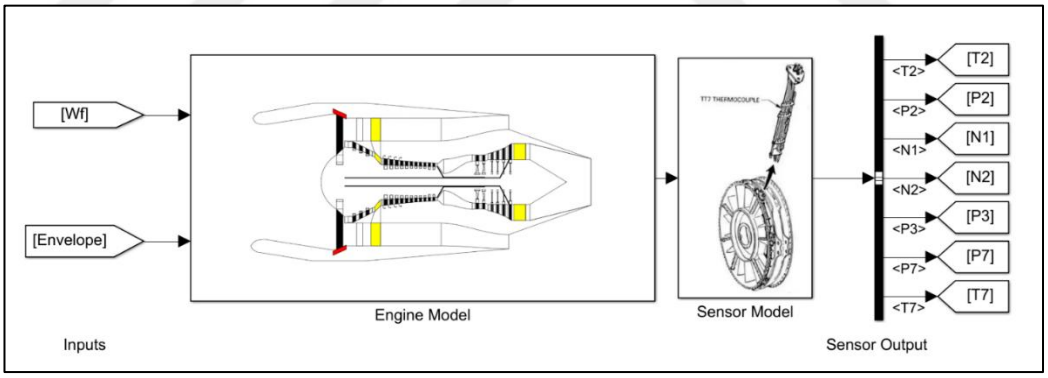


Figure 3.4 JT9D Engine Simplified Schematic of Model Structure

The engine model employs a multi-loop architecture. The outer loop iterates over time t with integrator blocks generating the system's next time-step solutions by integrating the derivatives of shaft speeds and metal temperatures. These solutions are fed into the inner loop, which consists of turbomachinery components in thermodynamical imbalance. The systems imbalanced terms, dependents variables, transferred to iterative solver. The iterative solver utilizes the Newton-Raphson (NR) method to advance the plant model toward a solution by generating Jacobian matrices. The

dependent variables, representing the inner loop's component errors, are iterated using a while-loop iterator until the converged tolerance is achieved. Upon convergence of the step, the solver produces thermodynamically balanced independent variables, which are unknows of the inner loop. The independents variables may include the input mass flow rate (W2), bypass ratio (BPR), compressors R-Lines (or Beta Lines), turbines pressure rations. Consequently, the system's state solution is obtained, allowing the model to progress to the next time step until solutions for all time steps within the specified duration are achieved **[6,7]**.

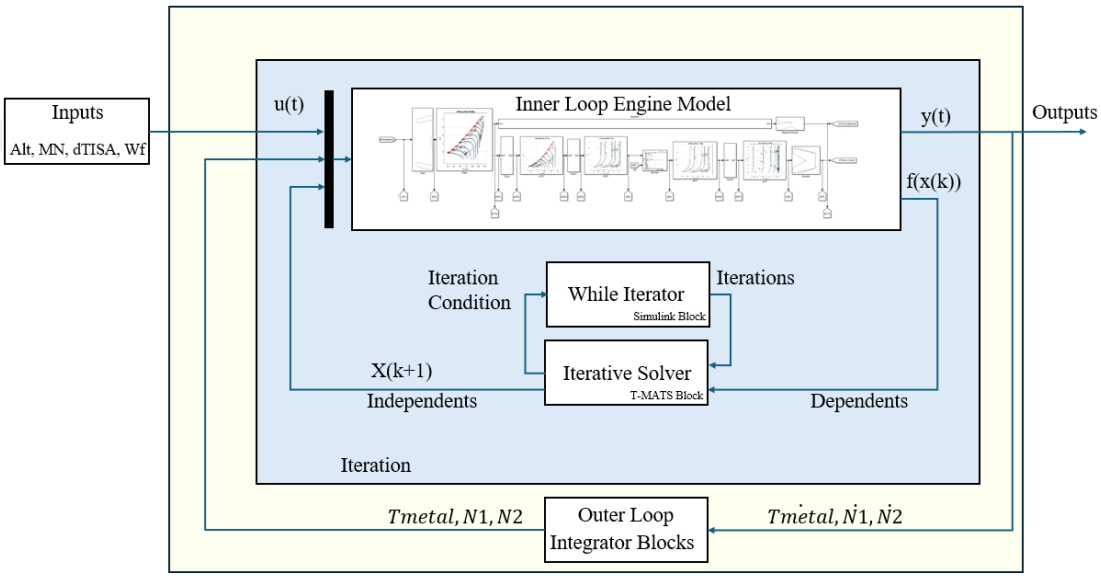


Figure 3.5 T-MATS Based Model Architecture **[6, 7]**

Figure 3.6 illustrates the engine modular modelling via Simulink. Component models are arranged to solve step solutions and make thermodynamic calculations. The calculations behind the component are based on **Section 2.1.** Employed components maps FAN, LPC, HPC, HPT and LPT turbomachinery were illustrated in **Figure 2.1**, **Figure 2.2** and **Figure 2.3**. Each component block input/outputs are defined as stations, to identify designated points based on ARP 755 standards **[48]**. **Table 2.2** represents the JT9D engine station nomenclature.

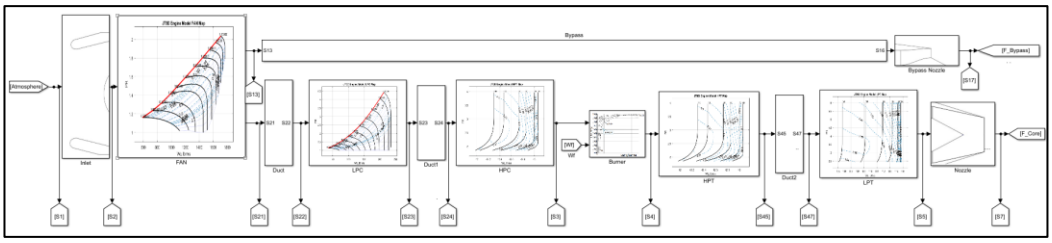


Figure 3.6 JT9D Representative Schematic of Engine Modular Component Structure

The steady state performance of the JT9D engine model at 100% LP Spool Speed in sea level static condition is presented in **Table 3.2.** The engine model has relatively sufficient output performance with comparing literature information. The steady-state outputs conform to the model described in the T-MATS example, ensuring consistency with established benchmarks **[7]**.

Table 3.2 JT9D Engine Model Steady State SLS Output

Alt [ft]	**Flight MN**	**dTamb**	**Pamb [kPa]**	**Tamb [K]**
0	0	0	101.325	288.15
Fuel Flow [kg/s]	**N1 [rpm]**	**N2 [rpm]**	**BPR**	**EPR**
2.364	3750.0	7887.5	5.1554	1.6029
Net Thrust [kN]	**Fg [kN]**	**Fbypass [kN]**	**Fcore [kN]**	**OPR**
233.33	233.33	179.04	54.29	21.53
Station	**W [kg/s]**	**Tt [K]**	**Pt [kPa]**	**FAR**
S1	732.433	288.15	101.325	0
S2	732.433	288.15	100.514	0
S21	118.989	336.39	164.0930	0
S13	613.443	336.39	164.0930	0
S22	118.989	336.39	163.682	0
S23	118.989	440.84	378.829	0
S24	118.989	440.84	377.8823	0
S3	108.280	752.84	2163.751	0
S4	110.645	1480.82	2044.745	0.02184
S45	121.354	1153.79	758.044	0.01987
S5	121.354	813.31	161.116	0.01987
S7	121.354	813.31	159.505	0.01987
S17	613.443	336.39	162.862	0
FAN				
Rel. Corr. Speed	**Wc [kg/s]**	**Pressure Ratio**	**Efficiency**	**Power [kW]**
0.9511	738.34	1.6325	0.8961	-35508.2
LPC				
Rel. Corr. Speed	**Wc [kg/s]**	**Pressure Ratio**	**Efficiency**	**Power [kW]**
0.9478	79.59	2.314	0.8742	-12576.5
HPC				
Rel. Corr. Speed	**Wc [kg/s]**	**Pressure Ratio**	**Efficiency**	**Power [kW]**
1.003	39.46	5.726	0.8214	-39284.6
HPT				
Rel. Corr. Speed	**Wc [kg/s]**	**Pressure Ratio**	**Efficiency**	**Power [kW]**
0.9978	19.26	4.994	0.9027	39284.4
LPT				
Rel. Corr. Speed	**Wc [kg/s]**	**Pressure Ratio**	**Efficiency**	**Power [kW]**
1.0159	50.55	6.135	0.8988	48084.7

From the perspective of dynamic modeling, the JT9D engine model incorporates shaft, heat, sensor and actuator dynamics. These dynamics are sufficient to represent the engine's transient behavior above idle conditions. However, advanced dynamics, such as stall and surge behavior, require volume dynamic modeling to simulate the momentum and transition of the gas flow and energy between the volumes. As this is beyond the scope of the thesis, volume dynamics were not modeled.

Shaft dynamic is the main dynamic of a transient behavior of the engine. The input of the dynamic is shaft inertias, which have been taken as they are in the example: LP shaft inertia is 135.6 $kg.m^2$ and HP shaft inertia 27.1 $kg.m^2$.

Heat dynamics called as heat soakage, are employed for HPC, CC, HPT and LPT components. However, according to low temperature behavior, heat dynamics are not implemented to FAN and LPC. T-MATS has its own heat soakage calculation library block. Lumped heat transfer models are employed. Thereby, each component is modeled as a representative bulk mass. Inputs of each heat transfer block are tabulated as **Table 3.3**. Input parameters are derived from literature **[6 ,7 ,14]**:

Table 3.3 JT9D Heat Soakage Model Parameters

Component	Wdesign [kg/s]	Tdesign [K]	Surface Area [m^2]	Heat Transfer Coefficient [W/m^2K]	Mass [kg]	cp [J/kgK]
HPC	108.9	600	9.3	3271	608	520
CC	111.1	1111	0.9	3148	192	520
HPT	122.5	1278	4.6	3148	816	520
LPT	117.9	983	4.6	1022	704	520

The order of magnitude of heat transfer between the gas and metal is deemed approximately appropriate and can be represented as heat transferred energy divided by component power while accelerating the engine from idle to maximum thrust level based on the following figure **[1]**.

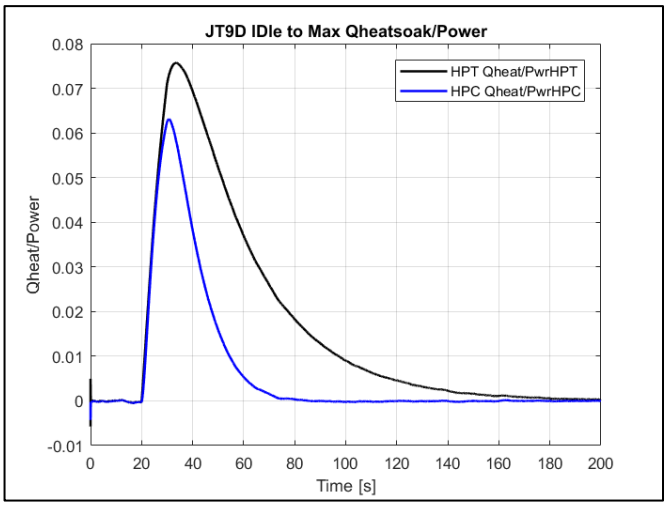


Figure 3.7 JT9D Heat Transfer Model Behavior

Measurements exhibit inherent delays and uncertainties in their natural behavior. To simulate realistic behavior, these dynamics were implemented into the engine model. Sensors are modeled as first order lag systems with specific uncertainty values derived from random noise generation. **Table 3.4** specifies sensor modeling input parameters for each sensor. Time constants represent the delay of the sensor under response to a step input.

Table 3.4 JT9D Engine Model Sensor Specification

Sensor	Time Constant [s]	Max Range	Uncertainty
N1- LP Shaft Speed [rpm]	N/A	4500 rpm	0.20%
N2 – HP Shaft Speed [rpm]	N/A	10000 rpm	0.20%
Tt2 -Fan Inlet Temperature[°C]	2	130 °C	Max 1.5°C
Pt2 - Fan Inlet Pressure [kPa]	0.05	300 kPa	0.10%
Pt3- HPC Exit Pressure [kPa]	0.05	3000 kPa	0.10%
Pt7- Core Nozzle Exit Pressure [kPa]	0.05	300 kPa	0.10%
Tt7 - Core Nozzle Exit Temperature [°C]	3	725°C	Max 5°C

Fuel flow is injected into the engine via metering unit, which is modeled as a first-order lag system with a time constant of 0.7 seconds.

The model is generated in MATLAB/Simulink. Time step of the simulations is taken as 0.01 seconds and Runge-Kutta (ode4) is used as solver **[49, 50]**.

The dynamics described above enable the model to adequately represent the transient dynamics of JT9D engine.

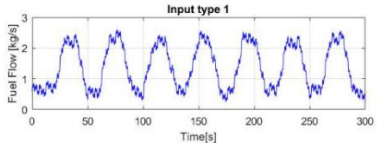
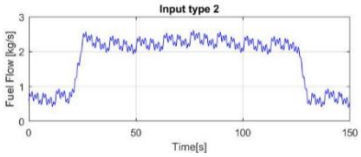
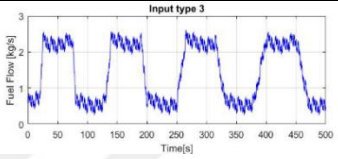
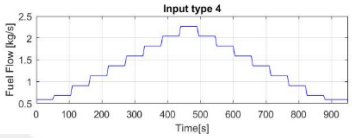
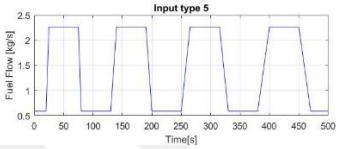
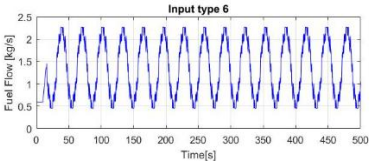
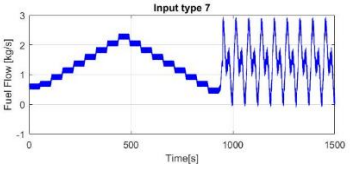
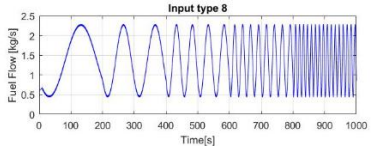
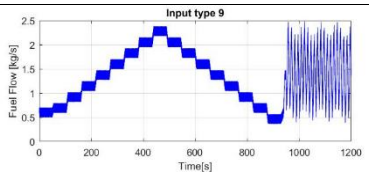
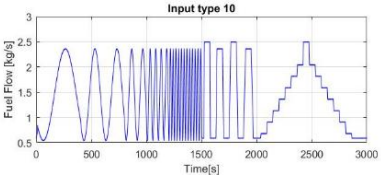
3.3. Inputs to Generating Train and Test Data Set

The general input variables for a gas turbine engine include fuel flow rate, variable vane actuator position, nozzle actuator positions (for engines with variable nozzle area) and atmospheric conditions. Within the scope of this study, only fuel flow rate is considered to model single-input-multiple-output system, simplifying the methodology by excluding variable vanes. Besides, corrected parameters represent the atmosphere effects quite effectively, thus sea level static condition performance of the engine is of interest for generating inputs for the SINDy model **[2]**.

The ability of data-driven models, such as the Sparse Identification of Nonlinear Dynamics (SINDy) model, to generate successful predictions depends on the scope and quality of the input data. The data quality must encompass the engine's entire operational domain and all relevant dynamics. Consequently, the input data represents one of the most critical factors. Hence, various fuel flow inputs are generated to cover a wide range from idle to max speed, the range of the fuel flow is from 0.43 kg/s to 2.5 kg/s.

Ten data set were generated, including sinusoidal, ramp, square, stairs maneuvers to involve the response of the system dynamic. **Table 3.5** provides details and graphical representation of the input data sets.

Table 3.5 System Fuel Inputs Options

Input Name	Duration [s]	Definition	Input Graph
Fuel Input 1	300	Sinusoidal	
Fuel Input 2	150	Square with sinusoidal	
Fuel Input 3	500	Narrower square with sinusoidal	
Fuel Input 4	900	Stairs	
Fuel Input 5	500	Square	
Fuel Input 6	500	Higher frequence sinusoidal	
Fuel Input 7	1500	Stairs + Sinusoidal with random gaussian noise	
Fuel Input 8	1000	sinusoidal with varying frequencies	
Fuel Input 9	1200	Stairs + sinusoidal with high random gaussian noise	
Fuel Input 10	3000	sinusoidal with varying frequencies, square + stairs	

3.4. SINDy Model

Generated measured data sets are used to train SINDy model and create governing. Flow diagram at **Figure 3.8** demonstrates the steps of the selection of SINDy model inputs.

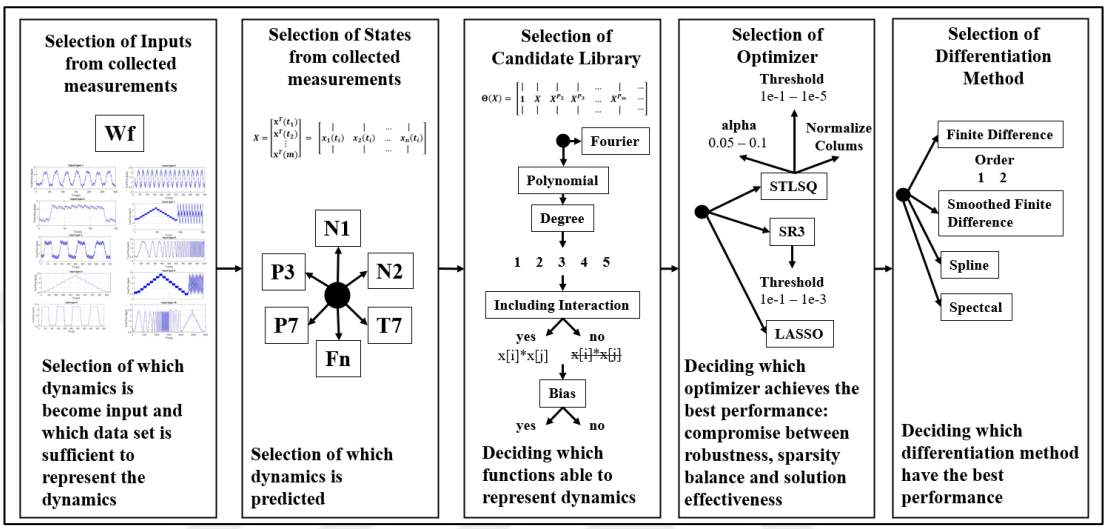


Figure 3.8 The Diagram to Selection of SINDy Model Parameters

There are various model parameters that can directly affect the results. Different combinations of parameters yield better results depending on the specific objective. Consequently, this gives rise to multiple possibilities that need to be evaluated. Although gas turbine engines exhibit complex dynamics, the selection of parameters that better represent their behavior should not be determined randomly. Major model parameters, such as inputs, state variables, and the selection of the candidate library must align with both mathematical approaches and the physics underlying the systems dynamics **[10, 12]**.

Estimating unmeasurable parameters, such as thrust and exhaust gas temperature (EGT), and developing a dynamic model based on SINDy are among the objectives considered. The objective of the model determines which parameters employ as inputs and which are used as state variables. Estimated parameters are chosen as states, other states and inputs are attempted which one is performing better. An increase in the number of inputs and state variables may result in a more complex model, thereby elevating the risk of overfitting. In response to this, if the model's sparsity threshold is

raised to reduce the number of terms, this may lead to the failure to capture essential dynamics, potentially compromising accuracy. On the other hand, if fewer inputs and state variables are used, the model may struggle to represent the dynamics with a limited number of terms. Consequently, balance between accuracy and complexity is required while selection of input and state variables.

A similar step involves the selection of the candidate library. Typically, polynomial and Fourier libraries are employed. Polynomial libraries are more feasible to turbomachinery dynamics. Therese, unless a specific case requires otherwise, the use of polynomial libraries is appropriate. When determining the degree of polynomial features and the interaction between parameters, a balance is achieved between complexity and accuracy.

Generally, Sequential Thresholded Least Squares (STLSQ) algorithms perform well for the engine dynamics, additionally which is more robust to noise than LASSO. **[12]** An optimization between threshold of STLSQ and maximum degree of the polynomial features is employed. As the final parameter, derivative methods were tested to identify the best solution.

3.5. Model Validation and Comparison

Model performance metrics facilitate comparison of results across models with varying SINDy model parameter configurations. Given the time-dependent nature of the dataset and the regression-based problem, regression metrics are suitable for evaluating performance. Mean Percentage Error (MPE), Root Mean Squared Error (RMSE), and the R-squared coefficient are employed as performance metrics. The one exhibiting a higher R-squared and lower MPE and RMSE values is selected. Additionally, the SINDy model response and training dataset were visualized in the time domain, with their dynamics' similarity and physical consistency observed and evaluated across different responses. Subsequently, the selected model performance was tested with different dataset.

To compare the SINDy's performance with different data driven models is another validation technique of the study. Nonlinear Auto-Regressive with External (Exogenous) Input Model (NARX) is employed as a neural network time domain data

driven model. There are several gas turbine modeling applications of NARX in the literature **[8, 9, 13, 33]**.

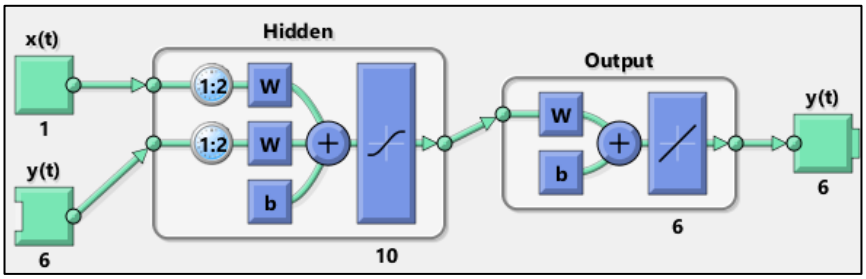


Figure 3.9 Structure of NARX model **[51]**

The NARX model was trained for each output at the same time by using 10 hidden layers and 2 number of delays. Levenberg-Marquardt was configured as training algorithm. Wf was provided as input, while N1, N2, P3, P7, T7 are as output. The Fuel Input 10 dataset was utilized, randomly divided into 70% for training 15% for validation and 15% for testing **[51]**. Training was terminated after 125 epochs. The model was validated with different Fuel Input dataset and compared to the SINDy model.

4. RESULT & DISCUSSION

4.1. Estimating Thrust

The primary function of the jet engines is to provide thrust to aircraft. Despite the existence of few methods, the measurement of thrust during flight is highly challenging **[56]**. Extensive instrumentation is required. Due to this reason, it is impractical particularly for military applications during operation. Fortunately, Thrust is highly correlated with engine mass flow rate, spool speeds Engine Pressure Ratio (EPR) or turbine temperature. However, transient behavior, envelope effects, engine deterioration, bleed and power off taken and engine anomalies affect the thrust level.

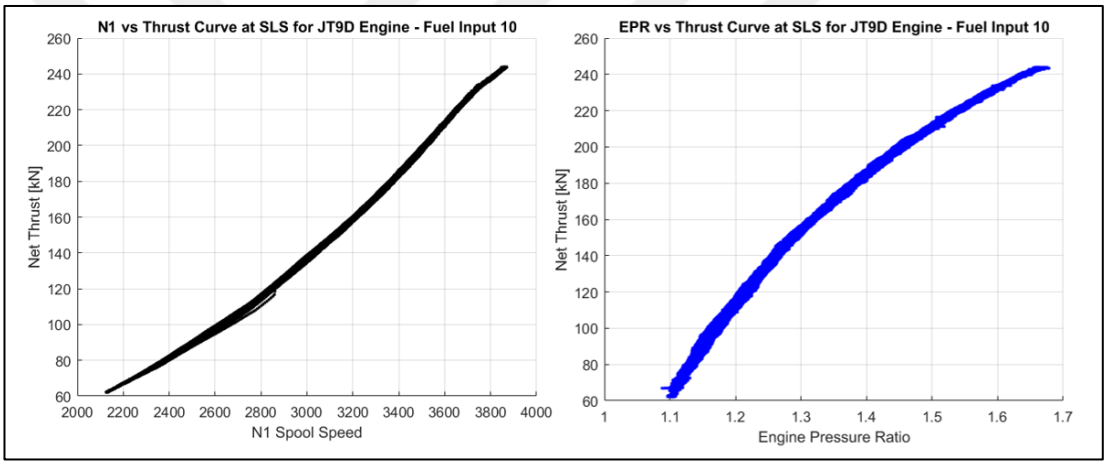


Figure 4.1 N1 vs Thrust & EPR vs Thrust Curves of JT9D Engine

Knowledge of thrust during flight offers essential insights for mission profile planning, maneuverability and the operational conditions of engine. Model-based and data-driven methods provide solution proposals for the requirement of thrust estimation **[14,57,52,53]**.

SINDy offers interpretable readily applicable practical solutions to prediction of thrust. To overcome this challenge, the SINDy model was trained with various fuel inputs and SINDy model configuration parameters. The comparison of multiple options reveals that **Figure 4.1** demonstrates the best model configuration with the best prediction performance. Due to its comprehensive profile, Fuel Input 10 contributed to an improvement in the solution's performance. Thrust (Fn) is the state of the model

and as an input variable, fuel flow (Wf), N1 Spool Speed and EPR combination achieved the best performance as expected.

Table 4.1 SINDy Model Parameters for Estimating Thrust

Training Data	Variables	Library	Optimizer	Differentiation Method
Fuel Input 10	X = Fn u = Wf, N1, EPR	Polynomial 2nd order without interaction	STLSQ 0.01 Threshold 0.05 alpha	Finite Difference 2nd order

The governing equation, as below, has polynomial second order variables without interaction between variables and the coefficients of the parameters are listed in **Table** 4.2.

$$\frac{dFn}{dt} = c_1 + c_2 Fn + c_3 N1 + c_4 EPR + c_5 Wf + c_6 EPR^2 + c_7 Wf^2 \qquad (4.1)$$

Table 4.2 Thrust Estimation Model Coefficients

Coefficient	c1	c2	c3	c4	c5	c6	c7
Fn [kN]	-2607.7	-5.038	0.15	3232.61	65.534	-815.1	-5.158

The model prediction response and its residual can be visualized as shown in **Figure 4.2**.

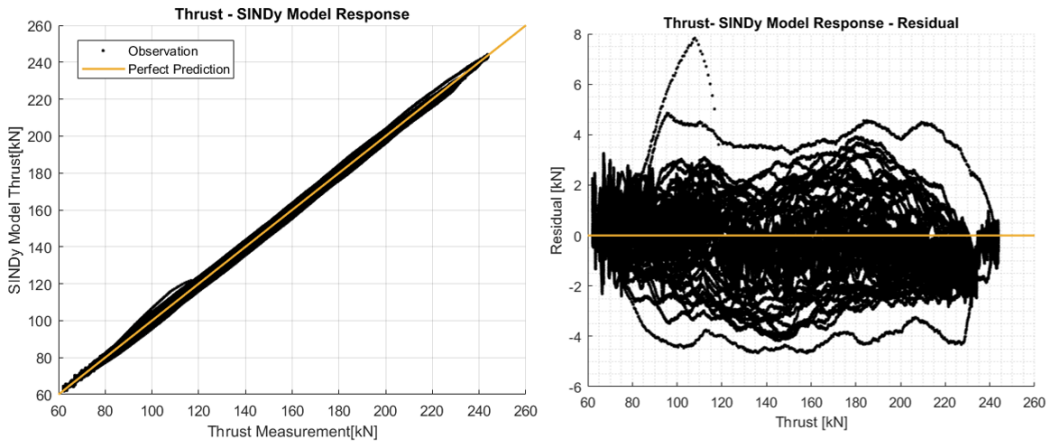


Figure 4.2 Thrust Estimation SINDy Model Response & Residual

Additionally, using the same input profile the NARX model was employed for comparison, and the performance metrics of both models are listed in **Table 4.3** Concurrently, **Figure 4.3** presents the time-dependent graph of the data and the models. The training model performs well with a higher R-squared value and a low error rate. Error rates are lower than 1%. Even NARX model provides ten times better performance than SINDy, with parsimonious architecture, SINDy model delivers adequate performance.

Table 4.3 Thrust Estimation Model Training Performance Metrics

Model	R-squared	RMSE [kN]	Mean Percentage Error [%]
SINDy	0.999657	1.134	0.7139
NARX	0.999994	0.148	0.0927

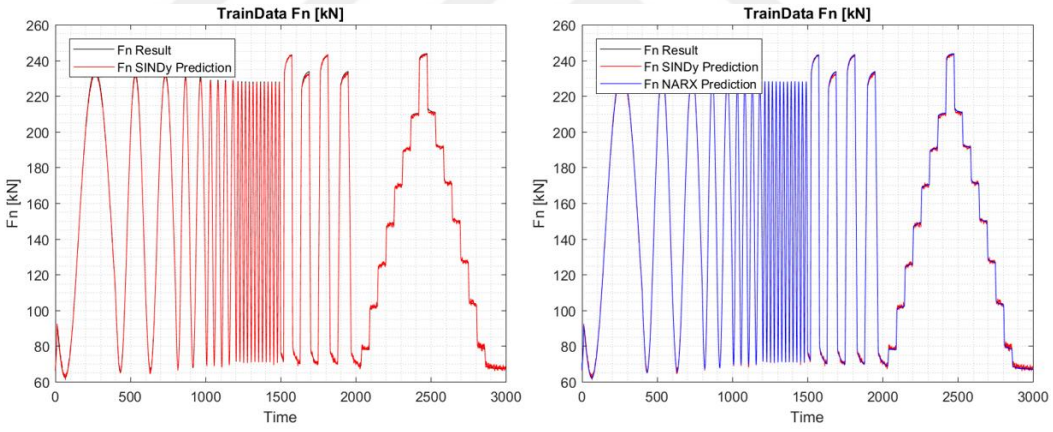


Figure 4.3 Thrust Comparison Between Training Dataset and SINDy Model

The trained model was validated with two different input profiles. The first test is acceleration and deceleration test and Fuel Input 4. The test results demonstrate that the model can predict thrust with 1% error without overfitting. However, it was observed that the model yields noisy outputs particularly at low velocity levels.

Table 4.4 Thrust Estimation Model's Test Data Performance

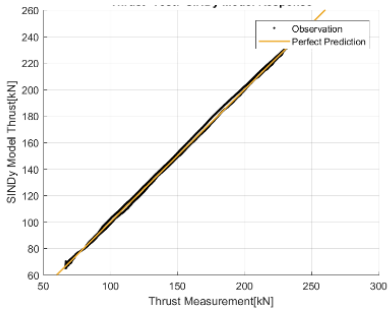
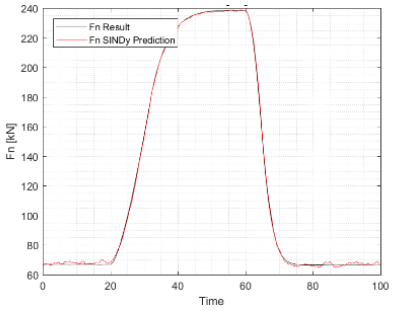
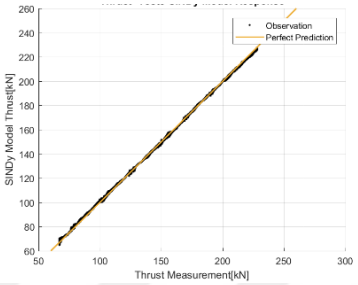
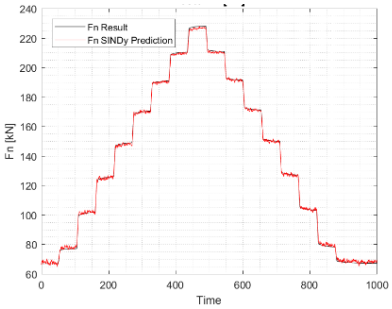
Test Data	Response	SINDy Model Prediction vs Thrust Measurement
Accel Decel Test Response		
Fuel Input 4		

Table 4.5 Thrust Estimation Model Test Performance Metrics

Input	Model	RMSE	Mean Percentage Error [%]
Acceleration-Deceleration	SINDy	1.161	1.055
	NARX	0.169	0.110
Fuel Input 4	SINDy	0.964	0.713
	NARX	0.107	0.073

The SINDy-based thrust model can be implemented into Simulink, utilizing four real-time sensor feeds and an integrator block for continuous thrust estimation.

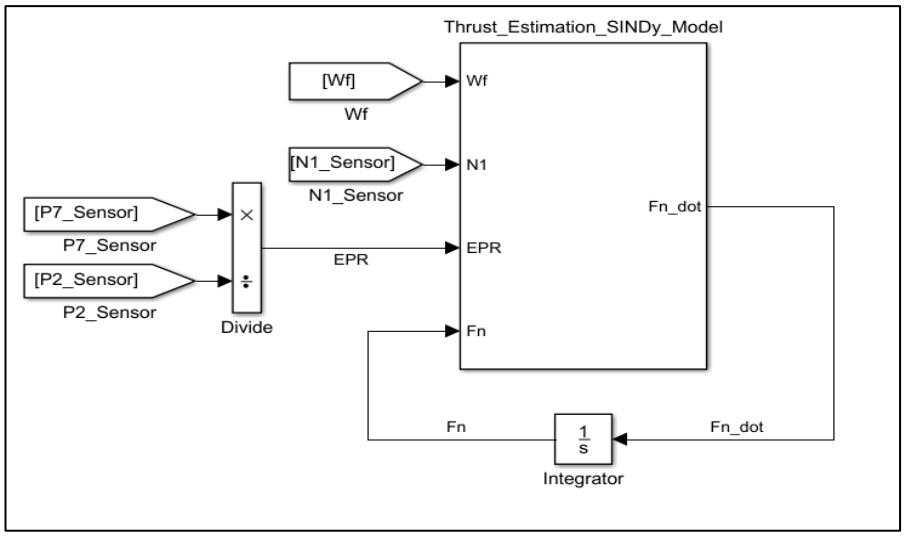


Figure 4.4 Simulink Diagram of SINDy Model for Thrust Estimation

4.2. Estimating Exhaust Gas Temperature (EGT)

Gas turbine engines operate at high temperatures, burner exit temperatures reach up to 1800°C. Particularly in military applications, where they are required to operate at the maximum temperature limits permitted by material. As turbine blades are subjected to extreme temperatures and mechanical stress, their service life is inherently limited. Consequently, regulating turbine temperatures to balance extended engine life with the ability to meet high-performance demands represents one of the most critical functions of the engine control unit. Since measuring the blade temperature requires extensive instrumentation, exhaust gas temperature or turbine exit temperature indirectly provides insights into turbine blade metal temperatures. Consequently, exhaust gas temperature (EGT) serves as a critical indicator of engine health, enabling fault detection and monitoring of performance degradation **[3, 4, 5, 51, 54]**.

Typically, EGT are measured by thermocouples. To avoid the inherent delays associated with thermocouples and enhance measurement reliability, the need for a temperature estimation model may arise **[54, 55]**. In this study, the applicability of SINDy as a data-driven solution has been investigated.

Similar to the thrust estimation, the SINDy model was trained in various configurations. **Table 4.6** demonstrates the best training model configuration. T7 (EGT) is the state of the model, the combination of fuel flow and engine pressure ratio

exhibited the best prediction via using second order polynomial functions with interaction of parameters.

Table 4.6 SINDy Model Parameters for Estimating EGT

Training Data	Variables	Library	Optimizer	Differentiation Method
Fuel Input 10	X = T7 u = Wf, EPR	Polynomial 2nd order with interaction	STLSQ 0.05 Threshold 0.05 alpha	Smoothed Finite Difference with Savitzky Golay filter 2nd order

The governing equation of the specified model and coefficient of the variables are as follows.

$$\frac{dT7}{dt} = c_1 + c_2 Wf + c_3 EPR + c_4 T7 * EPR + c_5 Wf^2 + c_6 Wf * EPR + c_7 EPR^2 \quad (4.2)$$

Table 4.7 EGT Estimation Model Coefficients

Model	c1	c2	c3	c4	c5	c6	c7
T7	694.508	387.348	-1361.1	-0.16	49.676	-380.38	756.525

The prediction performance metrics indicate that the model fits well. Although, its R-squared value is slightly lower than thrust estimation model performance, 0.98 R-squared value with 1.7% error rate it remains noteworthy due to SINDy's sparse and parsimonious modeling approach.

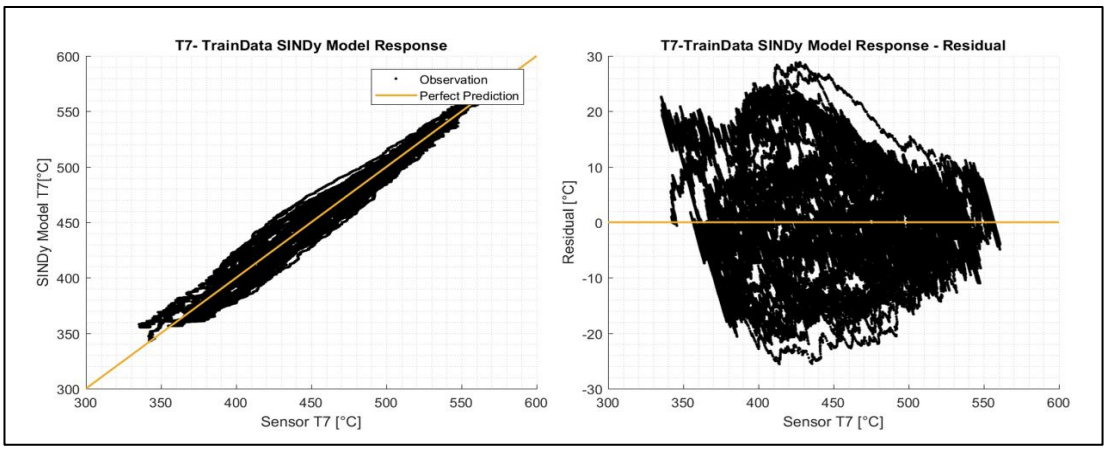


Figure 4.5 EGT Estimation SINDy Model Response & Residual

Table 4.8 EGT Estimation Model Training Performance Metrics

Model	R-squared	RMSE [°C]	Mean Percentage Error [%]
SINDy	0.9815	8.453	1.68
NARX	0.99998	0.26	0.045

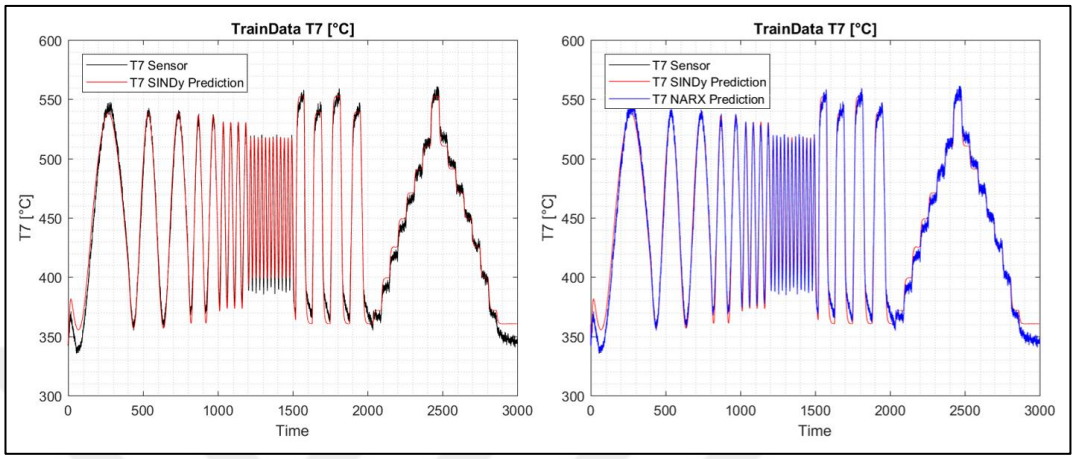
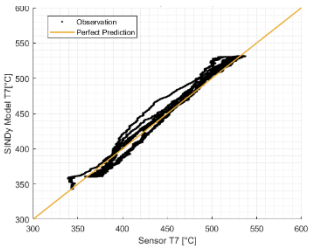
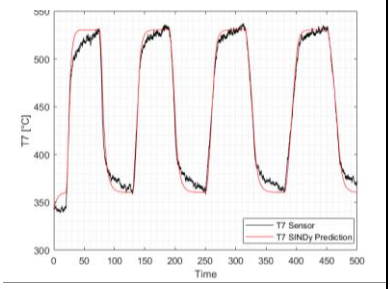
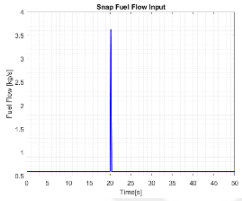
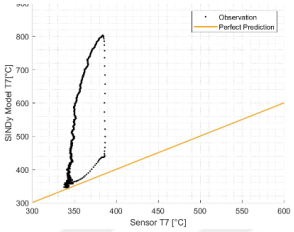
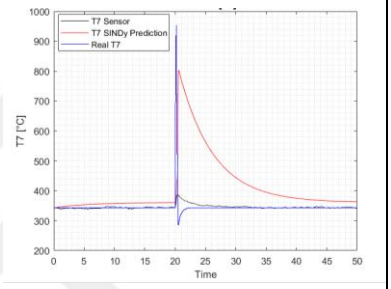


Figure 4.6 EGT Comparison Between Training Dataset and SINDy Model

Figure 4.6 shows the illustration of model outputs with sensor data, NARX and SINDy prediction. SINDy effectively follow sensor dynamic behavior and suppressed existing noise. However, its prediction became more challenging during sharp maneuvers.

The estimation model was tested with various inputs and two notable results are presented in tables below. Fuel Input 5 represents one of the model's reported responses, achieving an RMSE of approximately 9°C. Additionally, the model was evaluated using a dataset involving snap fuel injection, a highly extreme maneuver in practice, potentially resulting from controller or fuel metering faults. In this case study, the SINDy model demonstrated an acceptable response to the snap fuel injection. Although sensor dynamics could not accurately capture real T7 behavior due to their response delays, the SINDy model's results were closer to real T7 value.

Table 4.9 EGT Estimation Model's Test Data Performance

Test Data	Response	SINDy Model Prediction vs EGT Measurement
Fuel Input 5		
Snap Fuel Injection 		

The NARX model EGT estimation performs better than SINDy due to the complex NN algorithms behind. However, it can be stated that SINDy's performance criteria are also satisfactory.

Table 4.10 EGT Estimation Model Test Performance Metrics

Input	Model	RMSE [°C]	Mean Percentage Error [%]
Fuel input 5	SINDy	8.89	1.73
	NARX	0.301	0.048
Snap Fuel Injection	SINDy	110.6	18.2
	NARX	4.018	0.148

The SINDy-based EGT estimation model can be implemented into Simulink, utilizing three real-time sensor feeds and an integrator block for continuous EGT estimation as **Figure 4.7.**

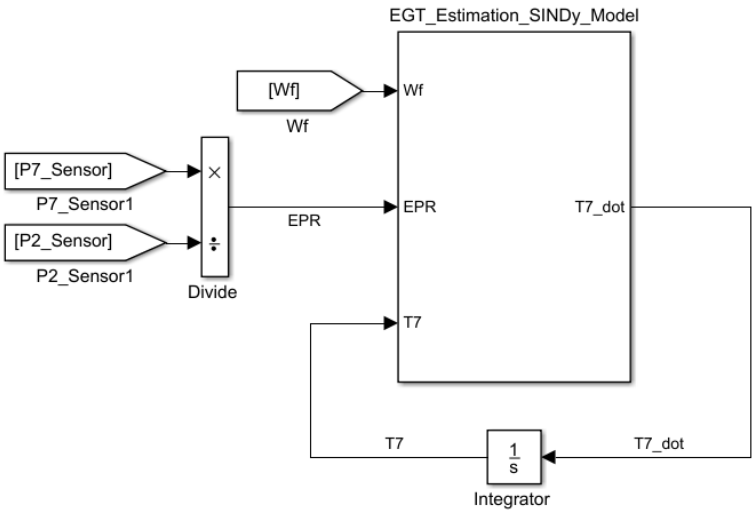


Figure 4.7 Simulink Diagram of SINDy Model for EGT Estimation

4.3. SINDy Based Data Driven Model

SINDy allow to capture dynamics of the system, generate governing equations and can be used for system identification **[12]**. In this chapter, SINDy's data driven system modeling capabilities will be examined using the JT9D turbofan engine, and the performance of the developed model will be discussed.

Figure 4.8 illustrates the diagram of the model. Two real-time sensor measurements are required to use as input which are fuel flow and P2 measurement for calculating Engine Pressure Ratio. The output includes N1, N2, P3, T7 and T7, which represent the sensors of JT9D engine and system's state variables. Each state has their own model and interacts with others when one state uses another variable as an input. Integrator blocks compute the next step of the predictions and the outputs feed back to the model. The desired model is simpler, parsimonious, independent from complex thermodynamic calculations and does not require any numerical iteration methods.

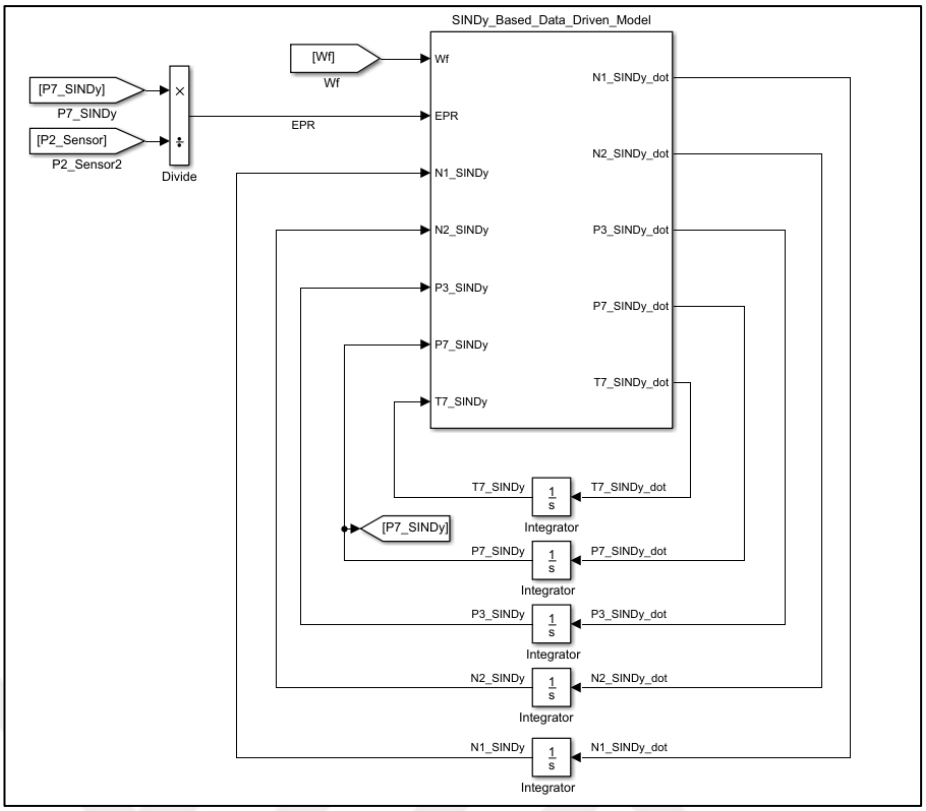


Figure 4.8 Simulink Diagram of Data Driven JT9D Engine SINDy Based Model

Several combinations of model parameters were examined to provide the best performance of their own and interaction with each other. **Table 4.11** highlight the selected models.

Table 4.11 SINDy Based Model Selected Parameters

Training Data	**Variables**	**Library**	**Optimizer**	**Differentiation Method**
Fuel Input 10	X = N1, N2 u = Wf	Polynomial 2nd order with interaction	STLSQ 0.1 Threshold 0.05 alpha	Finite Difference 2nd order
Fuel Input 10	X = P3 u = Wf, N1, N2	Polynomial 3rd order without interaction	STLSQ 0.05 Threshold 0.05 alpha	Smoothed Finite Difference with Savitzky Golay filter 2nd order
Fuel Input 10	X = P7 u = Wf	Polynomial 2nd order with interaction	STLSQ 0.1 Threshold 0.05 alpha	Smoothed Finite Difference with Savitzky Golay filter 2nd order
Fuel Input 10	X = T7 u = Wf, EPR	Polynomial 2nd order with interaction	STLSQ 0.05 Threshold 0.05 alpha	Smoothed Finite Difference with Savitzky Golay filter 2nd order

Fuel flow serves as a consistent input for each model. The results indicate that N1, N2 or EPR become an input for the models as anticipated from SINDy's sparse parameter selection algorithm. The reason of that N1 and N2 represent the actual state of the turbofan engine system while EPR effectively captures the system dynamics. N1 and N2 are solved concurrently during the training phase, and T7's governing equation was established in the previous section. Polynomial libraries outperform Fourier libraries in representing dynamics, as is known. The STLSQ optimizer performs effectively due to its noise performance and robust structure, with a specified threshold value. In certain cases, smoothed finite difference is preferred over the standard finite difference method for differentiation. The governing equations of each model are presented below.

$$\frac{dN1}{dt} = c_1 + c_2 N1 + c_3 N2 + c_4 Wf + c_5 N1 * Wf + c_6 N2 * Wf + c_7 Wf^2 \quad (4.3)$$

$$\frac{dN2}{dt} = c_1 + c_2 N1 + c_3 N2 + c_4 Wf + c_5 N1 * Wf + c_6 N2 * Wf + c_7 Wf^2 \quad (4.4)$$

$$\frac{dP3}{dt} = c_1 + c_2 P3 + c_3 Wf + c_4 N1 + c_5 N2 + c_6 Wf^2 + c_7 Wf^3 \quad (4.5)$$

$$\frac{dP7}{dt} = c_1 + c_2 P7 + c_3 Wf + c_4 Wf^2 \quad (4.6)$$

Table 4.12 SINDy Based Data Driven Model Coefficients

Parameter	c1	c2	c3	c4	c5	c6	c7
Fn	-2607.7	-5.038	0.15	3232.61	65.534	-815.1	-5.158
N1	17419.2	1.68	-3.29	4650.61	-1.026	-0.084	340.466
N2	14523.4	1.441	-2.745	3087.98	-0.847	0.027	256.806
P3	8427.89	-2.25	2214.95	1.102	-1.508	-232.02	19.355
P7	102.787	-1.004	13.443	4.817			
T7	694.508	387.348	-1361.1	-0.16	49.676	-380.38	756.525

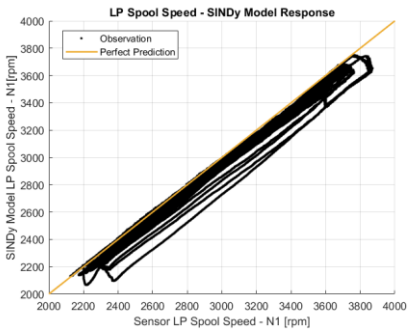
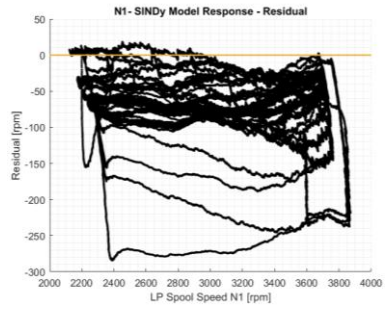
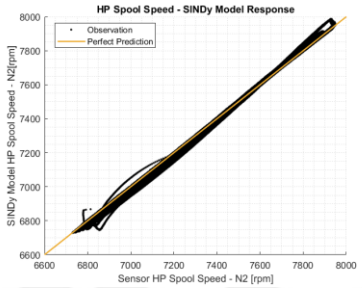
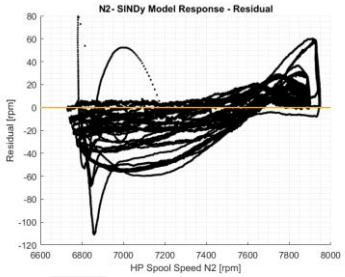
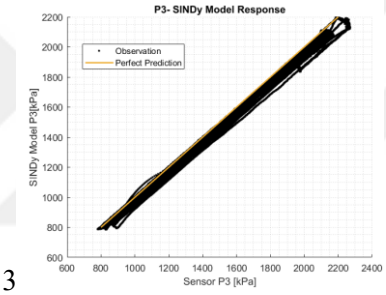
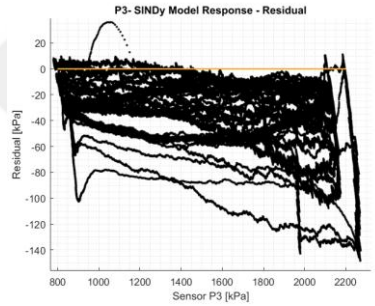
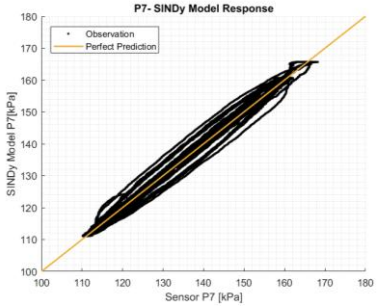
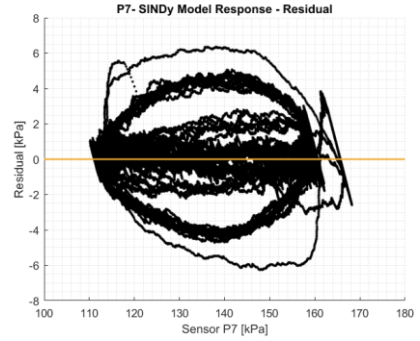
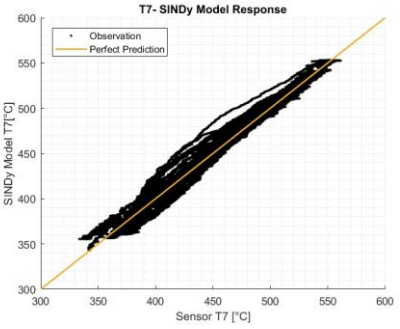
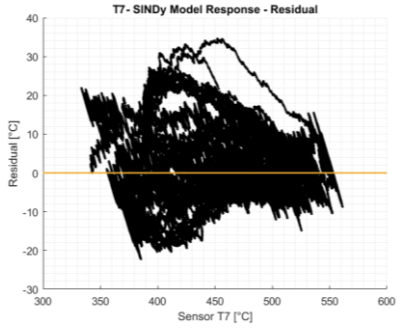
Table 4.13, **Table 4.14** and **Table 4.15** demonstrate the training performance of the model. The model exhibited a performance with an error rate below 2% for all variables and an R-squared value above 0.98, which is readily acceptable for such a parsimonious data-driven model.

N1 and P3 exhibit relatively high error rates and fail to adequately capture the system behavior at high power settings, which could not be improved within the scope of this study. However, it can be stated that the prediction performance for the other variables is preferable.

Table 4.13 SINDy Based Engine Model Performance Metrics

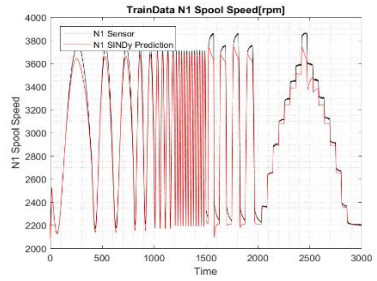
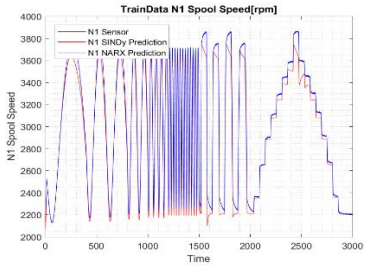
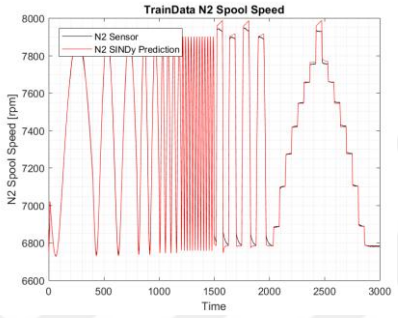
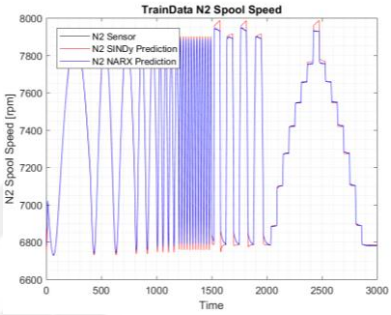
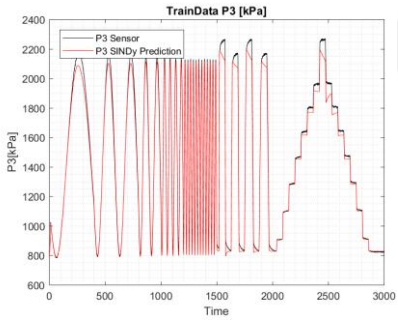
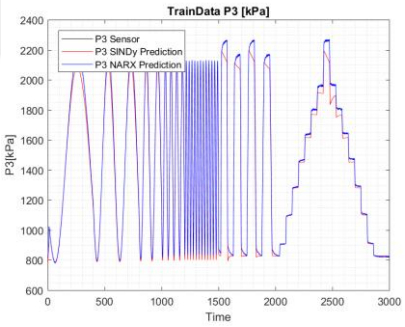
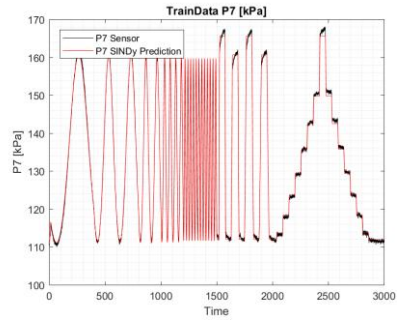
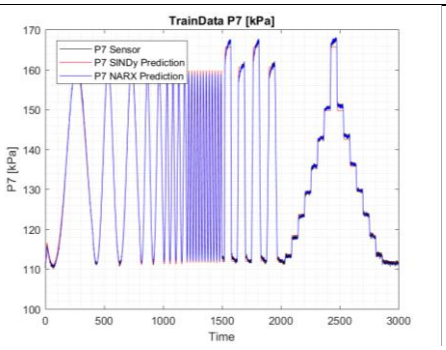
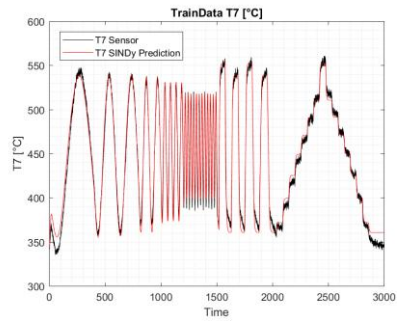
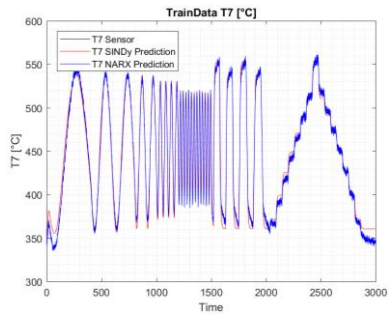
Training Data	R-squared	RMSE	Mean Percentage Error [%]	AIC
N1	0.9802	79.6 rpm	1.9997	26264
N2	0.998	18.4 rpm	0.1697	17464
P3	0.9924	43.4 kPa	1.9647	22629
P7	0.9955	1.23 kPa	0.6086	12580
T7	0.983	8.4 °C	1.582	12774

Table 4.14 SINDy Based Model Response and Residual of each Variable

Parameter	Response	Residual
N1		
N2		
P3		
P7		
T7		

3

Table 4.15 Output and Comparison Between Training Dataset and SINDy Based Model

Parameter	Training Dataset and SINDy Model Comparison	SINDy and NARX Model Comparison
N1		
N2		
P3		
P7		
T7		

Finally, the model was tested with several data sets and selected two results (Fuel Input1 and Fuel Input 6) are presented in below tables.

Table 4.16 SINDy Based Model Test Results for Fuel Input 1

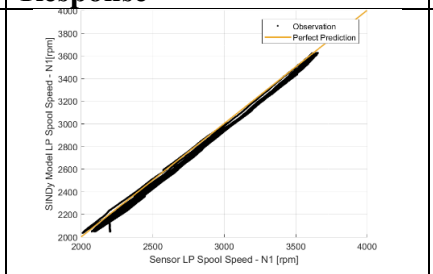
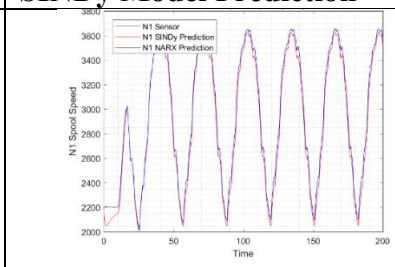
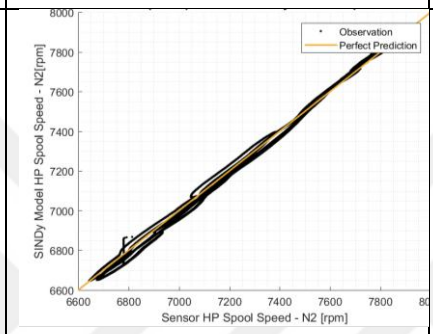
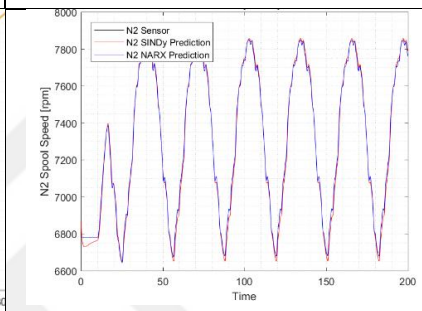
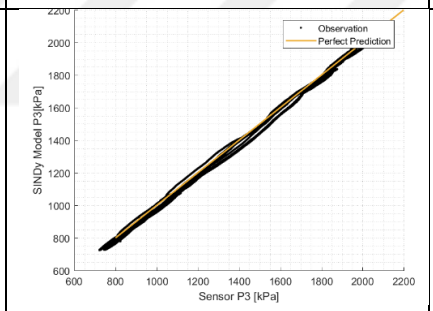
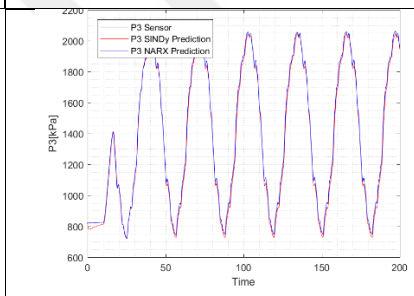
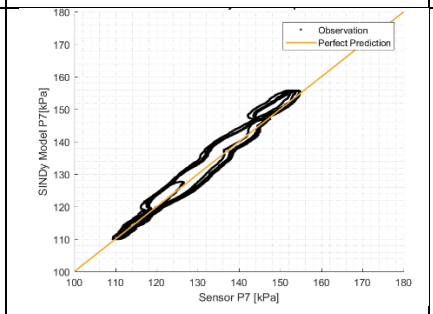
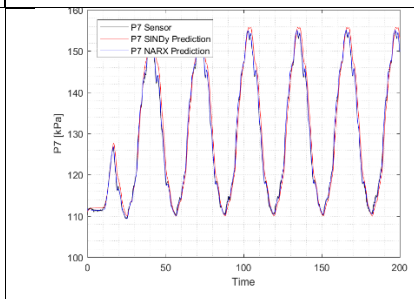
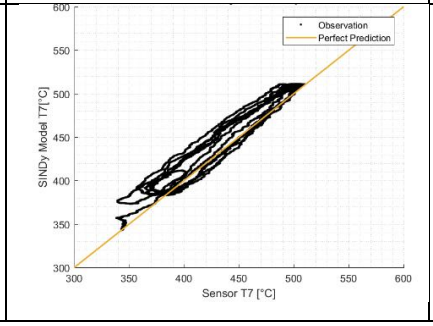
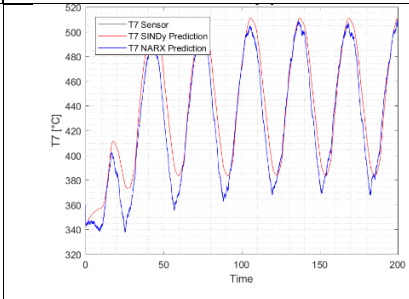
Parameter	Response	SINDy Model Prediction
N1		
N2		
P3		
P7		
T7		

Table 4.17 SINDy Based Model Test Results for Fuel Input 6

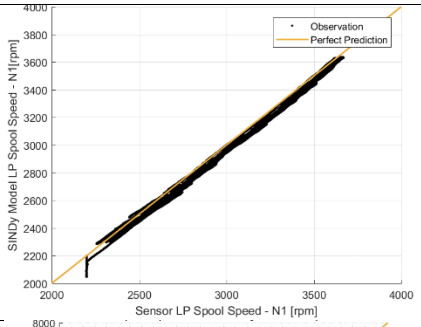
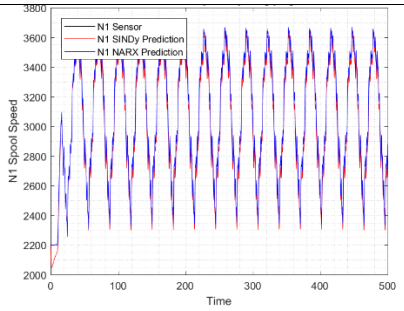
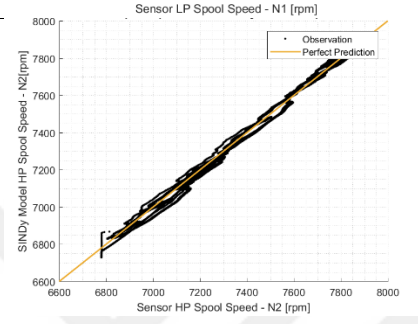
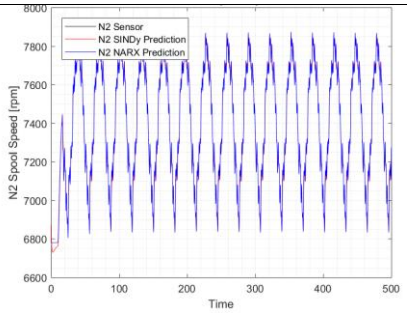
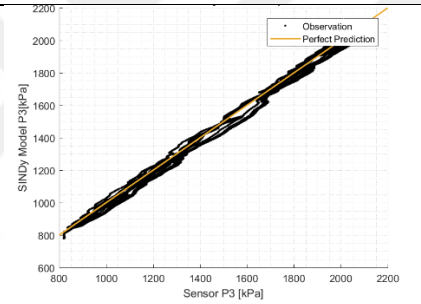
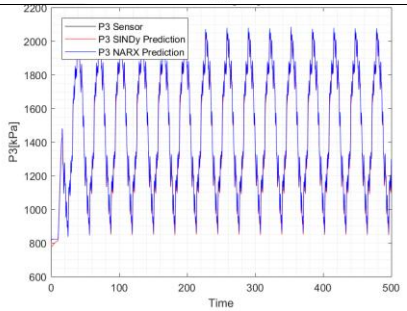
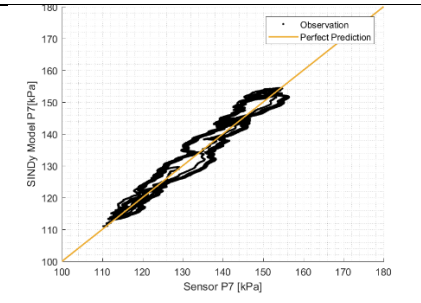
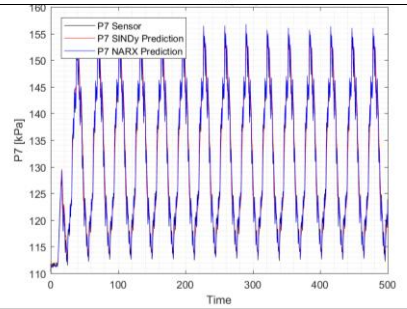
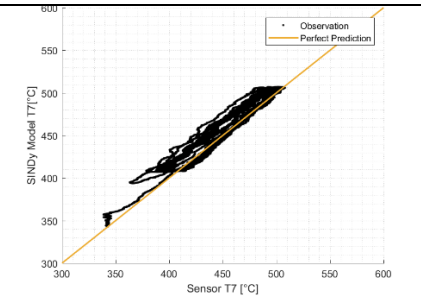
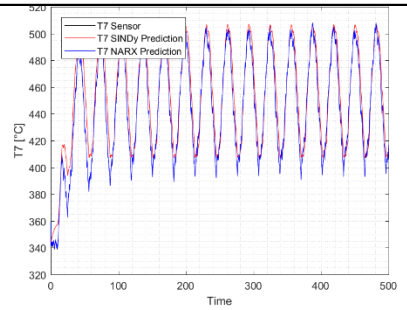
Parameter	Response	SINDy Model Prediction
N1		
N2		
P3		
P7		
T7		

Table 4.18 SINDy Based Model Test Performance Metrics for Fuel Input 1

Parameter	Model	RMSE	Mean Percentage Error [%]
N1 [rpm]	SINDy	56.243	1.825
	NARX	0.648	0.017
N2 [rpm]	SINDy	22.957	0.255
	NARX	0.931	0.006
P3 [kPa]	SINDy	25.074	1.794
	NARX	0.513	0.031
P7 [kPa]	SINDy	2.301	1.371
	NARX	0.434	0.278
T7 [°C]	SINDy	16.480	3.241
	NARX	0.405	0.064

Table 4.19 SINDy Based Model Test Performance Metrics for Fuel Input 6

Parameter	Model	RMSE	Mean Percentage Error [%]
N1 [rpm]	SINDy	52.216	1.635
	NARX	1.114	0.025
N2 [rpm]	SINDy	21.767	0.236
	NARX	1.009	0.008
P3 [kPa]	SINDy	30.098	1.775
	NARX	0.918	0.052
P7 [kPa]	SINDy	2.344	1.391
	NARX	0.660	0.405
T7 [°C]	SINDy	11.313	2.072
	NARX	0.379	0.064

The results and performance metrics indicates that, SINDy Based Model exhibits acceptable performance. The error rates can be further reduced through improvements, enabling the model to serve as a representative system model for various applications. Although the NARX model achieved significantly better predictions compared to SINDy model, SINDy's inherent interpretability, structural simplicity, physical consistent formulation, and acceptable error margins make it a suitable candidate within the scope of this research on gas turbine engine applications.

4.4. Improvements and Applications to Real Life Problems

4.4.1. Envelope Model

Within the scope of this study, the SINDy models are created as single-input models, only fuel flow effects are considered. The models represent sea level performance and estimations. Atmospheric effects can be partially accounted by using corrected parameters **[2]**. However, comprehensive envelope performance model requires direct atmospheric information as input. It can be pressure and temperature measurements at the engine inlet or flight Mach number and altitude provided by the aircraft. Flight condition measurements can be implemented to SINDy models as an input. The coefficients of the governing equations for each state can be trimmed with envelope information, In the cases of JT9D engine, it would be P2 and T2 measurements. To achieve this, a time-dependent dataset should be generated or tested from various points on the envelope, and the SINDy model should be trained for each point. Subsequently, the coefficients matrix is generated with dimensions of P2 and T2 measurements. SINDy based engine models provide acceptable predictions of sensor outputs with low computational cost making them suitable for controller design as an alternative to complex thermodynamic models.

4.4.2. Single Sensor Failures

SINDy based models can provide to detect single sensor failures. The logic of fault detection algorithm over EGT faults is illustrated in **Figure 4.9** via Simulink. Generally, real time EGT's are measured with multiple thermocouples and the multiple measurements are averaged, transmitted to engine controller for temperature limiting function. In the case of, one or several thermocouples are broken physically or electrically, the failure detection algorithm can detect and isolate the failure based on SINDy EGT estimation model. If the difference between real time sensor value and model exceed the threshold value, fault signal is transmitted to controller to take an action.

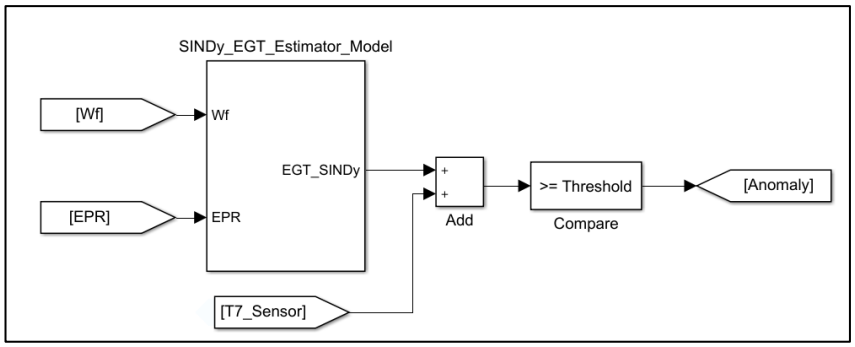


Figure 4.9 Single Sensor Failure Detection with SINDy Model

4.4.3. The Whole Engine Model for Deterioration Calculation and Anomaly Detection

Gas turbine engines, which operate under high temperatures and stress, experience performance degradation over time. This deterioration leads to increased fuel consumption and a reduction in thrust **[51]**. Additionally, the components used in these engines undergo wear and deformation, eventually requiring maintenance. Detection of maintenance requirements is essential for minimizing operational costs and ensuring sustained engine performance. To achieve this, real-time implementation within the engine's control and/or monitoring unit is necessary. Various remaining useful life estimations and predictive maintenance techniques have been proposed in the literature to address this challenge **[47, 20]**.

Degradation effects to drop compressors and turbines efficiencies varies the capacities **[51]**. If there is certain information about of the component's deterioration behavior and whole engine measurement signals test or validated thermodynamic model data, SINDy can offer practical solution to deterioration estimation of the engine. **Figure 4.10** shows the logic for deterioration detection, as a similar diagram to Tayarani-Bathaie et. al. **[20]** studies.

New and component deteriorated SINDy models are inserted to logic. The algorithm takes the SINDy models outputs and real-time temperature measurements, compare and weight the models and isolate components deterioration rates and send the signal to monitoring unit to inform pilot and adjust the controller if there is any trim is required due to degradation. The critical aspect in this concept is having a certain pre-knowledge about the component degradation behavior and impacts to the engine's performance measures.

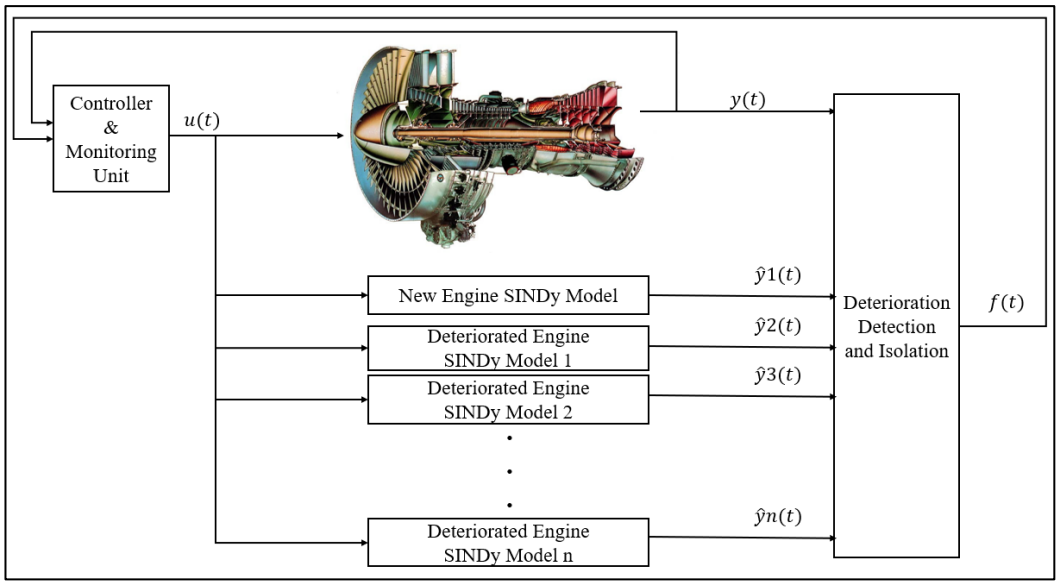


Figure 4.10 Whole Engine Model Logic for Deterioration Calculation

Similar to the deterioration model, whole engine failure can be detected with using SINDy Faulted Engine Models.

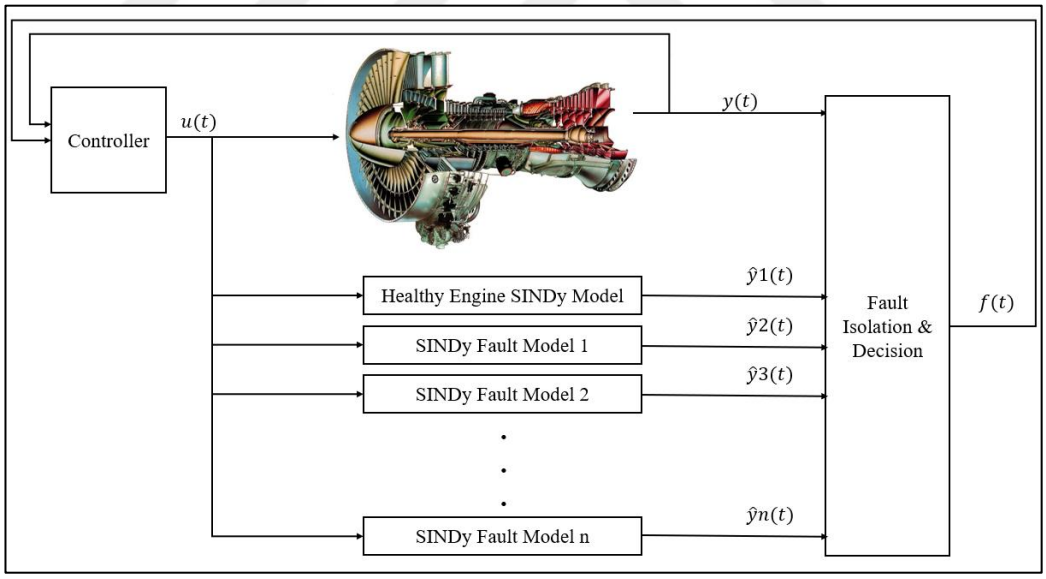


Figure 4.11 Whole Engine Model Logic for Anomaly Detection

5. CONCLUSION

Data driven modeling and exploring governing equations are crucial to understanding existing system behaviors. Sparse Identification of Nonlinear Dynamics (SINDy) allow to analyze system in a parsimonious, interpretable way with comparing the other data driven approaches. In this thesis SINDy algorithm is applied to gas turbine engine dynamics focusing on exploring governing equations and models to represent the system simplest and adequate way. JT9D high bypass turbofan engine dynamic engine model is employed to generate data sets for SINDy model. SINDy model parameters are analyzed, and the best models are selected. Furthermore, SINDy models are validated and compared with NARX models.

SINDy models are created to propose an approach to real life gas turbine challenges. Instant estimation of thrust and EGT while engine is operating are some of concerns. The results indicate that SINDy models perform adequate to prediction of parameters. The other scope of the study is generating SINDy based data driven models. May of black box data driven models are far from interpretability, hereby especially in certifications required industries such as aviation and aerospace these models may not be inserted to the real time systems. SINDy based models propose an explainable solution besides complex thermodynamic iterative modeling, simple governing equations are employed to identify system behaviors.

The models can be used to detect single sensor failures, anomaly detection and deterioration calculations for gas turbine engines **Chapter 4.4** Improvements and Applications to Real Life Problems comprise the approaches to these challenges.

In conclusion, the study highlights that the SINDy algorithm offers a novel data-driven approach to gas turbine engine challenges.

REFERENCES

[1] Kurzke J., Halliwell I., (2018), “Propulsion and Power An Exploration of Gas Turbine Performance Modeling”, 1st Edition, Springer,

[2] Walsh, P.P., Fletcher, P., (2004), “Gas Turbine Performance”, 2nd edition, Blackwell Science Ltd,

[3] Mattingly, J.D., (2010), “Elements of Gas Turbine Propulsion”, 1st Edition, TATA McGraw-Hill

[4] Mattingly, J.D., (2002), “Aircraft Engine Design”, 1st Edition, American Institute of Aeronautics and Astronautics, Reston.

[5] Cumpsty, N., Heyes, A., (2015), “Jet Propulsion: A Simple Guide to the Aerodynamics and Thermodynamic Design and Performance of Jet Engines”, 3rd edition, Cambridge: Cambridge University Press

[6] Chapman, J.W., Lavelle, T.M., May, R. D., Litt, J.S., and Guo, T. H., (2014), “Propulsion System Simulation Using the Toolbox for the Modeling and Analysis of Thermodynamic Systems (T-MATS)”, NASA/TM—2014-218410

[7] Chapman, J., W., May, R., D., Lavelle, T., M., et al., (2014), “Toolbox for the Modeling and Analysis of Thermodynamic Systems (T-MATS) User's Guide”, NASA

[8] Yu B., Shu W., (2017), “Research on Turbofan Engine Model above Idle State Based on NARX Modeling Approach”, IOP Conf. Ser.: Mater. Sci. Eng. 187 012002

[9] Asgari, H., Chen, X., Morini, M., Pinelli, M., Sainudiin, R., Spina, PR., Venturini, M., (2016), “NARX models for simulation of the start-up operation of a single-shaft gas turbine”, Appl Therm Eng 2016;93:368e76

[10] Brunton, S., L., Kutz, J. N., (2022), “Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control”, Second Edition, Cambridge University Press

[11] Corbertta, M., (2020), “Application of sparse identification of nonlinear dynamics for physics-informed learning”, SGT Inc., NASA Ames Research Center Moffett Field, CA 94035

[12] Brunton, S. L., Proctor, J. L., Kutz, J. N., (2016), “Discovering governing equations from data: Sparse identification of nonlinear dynamical systems” Proceedings of the National Academy of Sciences, 113(15), 3932–3937

[13] Kurzke, J., (1996), “How to get component maps for aircraft gas turbine performance calculations”, ASME 96-GT-164

[14] Naylor, H., P., (2004), “Gas Turbine Transient Performance Heat Soakage Modelling”, EngD Thesis, Cranfield University

[15] Montgomery, D., C., Peck E., A., Vining, G. G., (2012), “Introduction to Linear Regression Analysis”, 5th Edition, John Wiley & Sons,

[16] Kutner, M. H., Nachtsheim, C.J., Neter, J., Li, W., (2005), “Applied Linear Statistical Models”, 5th Edition, McGraw-Hill Irwin

[17] Pedregosa, et al., (2011), “Scikit-learn: Machine Learning in Python”, JMLR 12, pp. 2825-2830,

[18] Acquah, H., G., (2010), “Comparison of Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) in Selection of an Asymmetric Price Relationship”, Journal of Development and Agricultural Economics Vol. 2(1) pp. 001-006, ISSN 2006- 9774

[19] Hsin, J., Agarwal, S., Thorpe, A., Fridovich-Keil, D., (2023), “GPSINDy: Data-driven discovery of equations of motion”, University of Texas at Austin,

[20] Tayarani-Bathaie, S., S., Vanini, Z., N., Sadough, Khorasani, K., (2014), “Dynamic neural network-based fault diagnosis of gas turbine engines”, Neurocomputing, Volume 125, Pages 153-165, ISSN 0925-2312,

[21] Asgari, H., Ory, E., Lappalainen, J, (2021), “Recurrent Neural Network Based Simulation of a Single Shaft Gas Turbine”, VTT Technical Research Centre of Finland Ltd.

[22] Xingyun, J., Dengji, Z., Jiarui, H., Yushan, M., Zhike, P., (2023), “Dynamic simulation based on feature transfer learning with source domain adaptive optimization: Application of data-driven model for aero-engines”, Measurement, Volume 223,113786, ISSN 0263-2241,

[23] Zuming, L., Karimi, I., A., (2020), “Gas turbine performance prediction via machine learning”, Energy, Volume 192, 116627, ISSN 0360-5442,

[24] Kavlakoglu, E., Murel, J., https://www.ibm.com/think/topics/ridge-regression , Accessed Date: 2.03.2025

[25] James, G., Witten, D., Hastie, T., Tibshirani, R., (2017), “An Introduction to Statistical Learning with Applications in R”, Springer

[26] Zou, H., Hastie, T., (2005), “Regularization and variable selection via the elastic net”, Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(2), 301–320

[27] Yazar, I., Yavuz, H., S., Yavuz, A., A., (2017), “Comparison of various regression models for predicting compressor and turbine performance parameters”, Energy, Volume 140, Part 2, 2017, Pages 1398-1406, ISSN 0360-5442,

[28] Koza, J., R., (1992), “Genetic Programming: On the Programming of Computers by Means of Natural Selection”, MIT Press

[29] Li, Y., G., Ghafir, M., A., Wang, L., Singh, R., Huang, K., Feng, X., Zhang, W., (2012), "Improved multiple point nonlinear genetic algorithm based performance adaptation using least square method", J Eng Gas Turbines Power, 134(3):031701

[30] Sangjo, K., Kuisoon, K., Changmin S., (2020), "A new transient performance adaptation method for an aero gas turbine engine", Energy, Volume 193, 116752, ISSN 0360-5442,

[31] Lazzaretto, A., Toffolo, A., (2001), "Analytical and Neural Network Models for Gas Turbine Design and Off-Design Simulation", International Journal of Thermodynamics, 4(4), 173-182.

[32] Sangjo, K., Kuisoon K., Changmin S., (2020), "Transient system simulation for an aircraft engine using a data-driven model", Energy, Volume 196, 117046, ISSN 0360-5442,

[33] Giorgi, M., G., D., Quarta, M., (2020), "Hybrid MultiGene Genetic Programming - Artificial neural networks approach for dynamic performance prediction of an aeroengine", Aerospace Science and Technology, Volume 103, 105902, ISSN 1270-9638,

[34] Krishnan, M., Gugercin, S., Sever, İ., Tarazaga, P., (2020), "Dynamic Data Driven Modeling of Aero Engine Response", The Society for Experimental Mechanics

[35] Schmid., P., J., (2010), "Dynamic mode decomposition for numerical and experimental data", Journal of Fluid Mechanics, 656:5–28

[36] O'Brien, A., Weber, R., Kim, E., (2022), "Investigating SINDy as a tool for causal discovery in time series signals", Drexel University

[37] Silva, B., M., Champion, K., Quade, M., Loiseau, J., C., Kutz, J., N., Brunton, S., L., (2020), "PySINDy: A Python package for the sparse identification of nonlinear dynamical systems from data", Journal of Open Source Software, 5(49), 2104

[38] Kaptanoglu, A., A., et al., (2022), "PySINDy: A comprehensive Python package for robust sparse system identification" Journal of Open-Source Software, 7(69), 3994,

[39] Machado, G., F., Jones, M., (2024), "Sparse Identification of Nonlinear Dynamics with Side Information (SINDy-SI)" American Control Conference (ACC), Toronto, ON, Canada, 2024, pp. 2879-2884

[40] Momin, A., J., A., et al., (2022), "Nonlinear Model Identification and Observer Design for Thrust Estimation of Small-scale Turbojet Engines", 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, pp. 5879-5885

[41] Paniccia, D., Tucci, F., A., Guerrero, J., Capone, L., Sanguini, N., Benacchio, T., Bottasso, L., (2025), "A Supervised Machine-Learning Approach for Turboshaft Engine Dynamic Modeling Under Real Flight Conditions", Cornell University

[42] L'Erario, G., et al., (2020), "Modeling, Identification and Control of Model Jet Engines for Jet Powered Robotics", IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2070-2077

[43] Pratt & Whitney, https://www.prattwhitney.com/en/products/commercial-engines/jt9d, Accessed Date: 01.06.2025

[44] National Air and Space Museum, https://airandspace.si.edu/collection-objects/pratt-whitney-jt9d-1gt2-turbofan-engine-cutaway/nasm_A19820085000, Accessed Date: 01.06.25

[45] Jay, A., Todd, E. S., (1978), "Effect of Steady Flight Loads on JT9D-7 Performance Deterioration", NASA-CR-13407, NASA Lewis Research Center,

[46] Kafka, P., G., Skibo, M., A., White, J., L., (1977), "JT9D Engine Diagnostics: Task 2: Feasibility Study of Measuring In-service Flight Loads", NASA CR-135395, Nasa Levis Research Center

[47] Ziemianski, J., A., Mehalic, C., M., (1980), "Investigation of Performance Deterioration of the CF6/JT9D High-Bypass Ratio Turbofan Engines", NASA-TM-81552, Nasa Lewis Research Center

[48] SAE ARP 755, (1991), "Aircraft Propulsion System Performance Station Designation and Nomenclature", Aerospace Recommended Practice

[49] The MathWorks, Inc., (2017), MATLAB version: 9.2.0, R2017a.

[50] The MathWorks, Inc., (2017), Simulink version: 8.9., R2017a

[51] The MathWorks, Inc., (2017), Statistics and Machine Learning Toolbox version: 11.1, R2017a

[52] Conners, T., R., Sims, R., L., (1998), "Full flight envelope direct thrust measurement on a supersonic aircraft", NASA Technical Memorandum NASA/TM 206560, H 2266, NASA Dryden Flight Research Center, Edwards, CA, USA

[53] Wen, H., T., Wu, H., Y., Liao, K., C., Chen, W., C., (2023), "JT9D engine thrust estimation and model sensitivity analysis using gradient boosting regression method", Aerospace, vol. 10, no. 7, article 639

[54] Paniagua, G., Denos, R., Oropesa, M., (2002), "Thermocouple probes for accurate temperature measurements in short duration facilities", ASME Turbo Expo 2002: power for land, sea, and air. American Society of Mechanical Engineers; p. 209e17

[55] Gaylord, A., M., Compton, W., A., Furgurson R., G., (1969), "High temperature sensors for gas turbines", ASME gas turbine conference and products show. American Society of Mechanical Engineers, pp. V001T01A030- V001T01A030

[56] Jones, E., Oliphant, T., Peterson, P., et al., (2001), "SciPy: Open-source scientific tools for Python"

[57] Hunter, J., D., (2007), “Matplotlib: A 2d graphics environment,” Computing in Science & Engineering, vol. 9, no. 3, pp. 90–95

[58] Tibshirani, R., (1996), “Regression shrinkage and selection via the lasso”, Journal of the Royal Statistical Society: Series B (Methodological), vol. 58, no. 1, pp. 267–288

[59] Lingmont H., Alijani, F., Bessa, M., A., (2020), “Data-Driven Techniques for Finding Governing Equations Of Noisy Nonlinear Dynamical Systems”, Master’s Thesis, Delft University of Technology

[60] Many, M., Weber, M., et al., (2022), “System Identification of a Turbojet Engine using Multi-Sine Inputs in Ground Testing”, Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal-Oberth e.V.

BIOGRAPHY

The author, Alper YILMAZ, received his B.Sc. degree from Automotive Engineering Program at Hacettepe University in 2021. He is currently employed as a Performance & Operability Engineer in the aviation industry.