Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Manufacturing and Civil Engineering

Endre Sølvsberg

Extending Statistical Process Control using quantitative methods in Industry 4.0 approaches

Master's thesis in Sustainable Manufacturing
Supervisor: Niels Peter Østbø
June 2019

NTNU
Norwegian University of Science and Technology

SINTEF

Endre Sølvsberg

Extending Statistical Process Control using quantitative methods in Industry 4.0 approaches

Master's thesis in Sustainable Manufacturing
Supervisor: Niels Peter Østbø
June 2019

Norwegian University of Science and Technology
Faculty of Engineering
Department of Manufacturing and Civil Engineering

NTNU
Norwegian University of
Science and Technology

Abstract

This master thesis is written in cooperation with SINTEF Manufacturing AS and aims to answer the following research question; Can Statistical Process Control and be extended to function as a tool in Condition Based Maintenance for determining machine condition and Remaining Useful Life estimation, and how does this influence the usefulness of SPC in Industry 4.0? Through using methods including literature review and quantitative methods utilizing the widely used NASA C-MAPSS dataset to construct a model using SPC test data to determine machine condition and estimate RUL, the aim of the thesis is to answer this question in a satisfactory manner.

X-Bar and R control charts are constructed and analyzed in a time-series using the complete Western Electric ruleset. A condition indicator score is constructed for every time-series chart by using a scoring algorithm. The distribution of indicator scores across all engines is tested for normal distribution, allowing for calculating probabilities on remaining useful life.

The thesis concludes that the constructed model probably can be a useful manual alternative to Machine Learning for RUL estimation, and that SPC as a tool is likely to be increasingly useful in Industry 4.0. The thesis recommends further research into the viability of SPC as a tool for RUL estimation, but also focusing on the benefits of researching human-driven methods for condition monitoring and estimating remaining useful life. Creating real-life run-to-failure datasets should also be a priority.

Preface

This Master Thesis was written during the 2019 spring semester at NTNU in Gjøvik. The work has been done in cooperation with SINTEF Manufacturing AS. Research director dr. Odd Myklebust has provided a theme outline that this thesis uses as its foundation. Co-supervisor postdoc Harald Rødseth and main supervisor associate professor Niels Peter Østbø has also contributed with valuable insights and discussion on the topic during the creation of the thesis.

Gjøvik, June 11th, 2019

Endre Sølvsberg

Acknowledgements

I would like to thank SINTEF Manufacturing AS and especially dr. Odd Myklebust for providing me with a suggested topic and letting me write the thesis in cooperation with them. Working for SINTEF and dr. Myklebust with data analysis and statistical analysis has given me valuable experience and insights into Industry 4.0 and Cyber Physical Systems. To my supervisor associate professor Niels Peter Østbø for being an inspiration through seminars on Industry 4.0 and for believing I could contribute in teaching statistical methods and analysis to first year master students. To professor Aristidis Kaloudis for believing in me and rekindling my love for statistics. To my co-supervisor postdoc Harald Rødseth for going above and beyond in contributing through insights and discussion. And finally, a thank you to my mother. After my heart attack, my world fell out from under me. With your help I have been able to start rebuilding, and without your support I could not have completed this thesis. Thank you all!

Gjøvik, June 11th, 2019

Endre Sølvsberg

Table of contents

List of figures and tables

List of figures

List of tables

Keywords

Statistical Process Control, Industry 4.0, Zero-Defect Manufacturing, Quality Engineering, Condition Based Maintenance, Remaining Useful Life, Machine Learning.

1. Introduction

Beginning with a German article on an upcoming fourth industrial revolution (Kagermann, Lukas and Wahlster, 2011), and later expanded upon in a manifesto by the German National Academy of Science and Engineering (Acatech, 2013), industries in the industrialized countries are currently in the process of moving towards what we call Industry 4.0 (I4.0). I4.0 combines the Internet of Things, allowing physical devices, machines and objects to connect to the internet, and manufacturing (Hermann, Pentek and Otto, 2016).

The developments towards I4.0 systems highlights the importance of system reliability and availability, and maintenance is increasingly important. The field of maintenance is moving from utilizing reactive or time-based strategies towards predictive maintenance. By utilizing predictive maintenance strategies in I4.0 with technologies for monitoring machine condition, industrial plants will be able to identify faults before they happen (DIN, 2018). This allows for more efficient maintenance strategies where maintenance can be planned optimally and only utilized when needed.

Statistical Process Control (SPC) has been used as a tool in quality engineering for monitoring and controlling processes for nearly a century. The American statistician Walter A. Shewhart developed control charts that allowed for determining whether a process is in or out of statistical control (WEC, 1956). By analyzing process variation, finding the causes for this variation and eliminating the cause, process performance can be improved. SPC is commonly utilized in some form in industry and is a central tool within the philosophy of Zero-Defect Manufacturing (ZDM).

Considering SPC is most often used as a tool for monitoring product variables, there is a theoretical foundation for extending SPC into monitoring process variables and constructing control charts based on these (Scherkenbach, 1986, Nomikos and MacGregor, 1995, Wood, 1994, Fugate, Sohn and Farrar, 2001). An increased focus on processes is also supported by fundamental theory within the field of quality engineering (Taguchi, 1995, Scherkenbach, 1986).

A central term within Condition Based Maintenance is remaining useful life (RUL), and a central challenge is to develop tools for determining the condition of processes and estimate when they are expected to break down. This in order to utilize maintenance at an optimal time and to avoid maintaining processes more often than necessary (Si *et al.*, 2011). Most of the research into RUL estimation today is conducted using methods based on artificial intelligence and Machine Learning (ML). Although much of this research has produced positive results, ML methods does have some challenges. Developing manual, quantitative methods for estimating remaining useful life based on statistical analysis can be beneficial in many ways and help motivate a perhaps undervalued part of RUL estimation research.

Examining if SPC can be extended, not only to monitor process variables, but to be used as a tool for determining the condition of the monitored process and estimating RUL is an interesting challenge, as it can potentially be increasingly useful as a tool in Industry 4.0 environments but also provide a manual and transparent alternative to ML RUL estimation methods.

1.1 Problem statement

There are some components that have helped motivate the research in the thesis. Working as an employee at SINTEF Manufacturing and working with analyzing sensor data in part by utilizing SPC methods has affected the choice of topic. Dr. Odd Myklebust at SINTEF Manufacturing has provided a suggested topic for the thesis containing four tasks. The first task describes state of the art for Cyber-Physical Systems and digital twins and includes a discussion on how this can be used as a base for Zero Defect Manufacturing. The second task is to provide an updated framework for the CPS-Plant project, focusing on SME implementation. The third task concerns Statistical Process Control and Six-Sigma, and how these and other quantitative methods can be extended for implementation in Industry 4.0 approaches. The last task describes Taguchi methods, use of quantitative methods in Taguchi and setting up an experiment that can lead to a Zero-Defect Manufacturing solution.

The final motivation for the thesis is the cooperative work conducted with postdoc Harald Rødseth at NTNU related to a project called CPS-Plant, specifically concerning Predictive Maintenance and Remaining Useful Life estimation, and working towards publishing a paper on this topic.

Taking all this into account, a choice was made to focus on SPC more in depth, and examine the potential for extending the method into Condition Based Maintenance and RUL estimation by constructing an experimental quantitative model capable of determining the condition of a machine, and use that capability to estimate RUL. Based on these motivational influences, this thesis will focus on answering the following main research question;

Can Statistical Process Control and be extended to function as a tool in Condition Based Maintenance for determining machine condition and Remaining Useful Life estimation, and how does this influence the usefulness of SPC in Industry 4.0?

To be able to answer the first part of this question, the aim of the thesis is to construct an experimental quantitative model using SPC control charts on process variables from a simulated run-to-failure dataset and construct time-series of individual variables to test for process statistical control using the complete Western Electric ruleset (WEC, 1956). An algorithm will be constructed attempting to transform process variable test failures into a condition indicator. Indicator scores will be aggregated from all included processes and all distributions will be tested using a Kolmogorov-Smirnov test for normal distribution to determine if and how RUL can be estimated (Lilliefors, 1967).

The thesis will attempt to answer the second part of the main research question by performing a literature review on connected topics to create background and context. The review will cover theory on the concepts behind Industry 4.0 and central architectures to build a foundation for connecting the other topics. Basic theory on artificial intelligence and machine learning is presented to allow for discussion on ML RUL estimation methods versus manual methods using statistical methods. Fundamental theory on quality engineering is presented to put SPC into a broader perspective and to connect it to I4.0. Two frameworks are used to show the connection between quality engineering and I4.0. Fundamental theory on SPC and the construction of control charts is necessary to increase understanding of methods used to construct the model, and to provide theoretical background for discussing the usefulness of using SPC to monitor process variables. The last part of the literature review will present theory on Condition Based Maintenance and the concept of remaining useful life within the field of maintenance management and predictive maintenance and to connect this to I4.0. The aim is to argue for the usefulness of SPC in Industry 4.0 through discussing these concepts, how they are connected and how they influence each other.

1.2 Boundaries of the thesis

When constructing a model for determining machine condition and estimating RUL in the results section, the thesis will include data from one of the datasets included in the NASA C-MAPSS simulation data package. No other datasets will be included or reviewed. The model will utilize a sample size of 90 and subgroup size of 3. Some tests will include sample size 60 and subgroup size 5. No other combination will be considered. The model will consider X-Bar charts and R charts. All other types of control charts are outside of the thesis boundaries. The model will use the complete WE ruleset when testing the control charts, scoring will be equal for all types of test failures and all process variables are scored equally. No other combinations of these will be considered.

The literature review includes theory on topics central to increase understanding of central concepts used in constructing the model and topics related to setting I4.0 as a background for the thesis. Theory on other topics is included to provide material for discussing the main research question. Theory that is not considered relevant for this specific purpose is not included and is deemed to be outside of the thesis boundaries.

1.3 Thesis roadmap

The thesis has a section on the methodology used while developing the contents, and a literature review consisting of theory and concepts deemed necessary to describe and discuss the main research question with a systems and manufacturing perspective. The results section goes through the experimental design model in detail, where SPC and control charts are used on a simulated jet engine dataset to attempt to create an engine condition indicator and to predict remaining useful life. The discussion part contains an in-depth interpretation of the model presented in the results section, implication of the results and how SPC can be used as a tool in I4.0 based on perspectives presented in the literature review and limitations of the research done in the thesis. The thesis concludes that SPC probably can be useful as an alternative to ML methods in determining machine condition and estimating RUL. It also concludes that SPC most likely will be an increasingly important tool in I4.0. The last part of the conclusion section makes some recommendations on future work regarding the research done in the thesis.

2. Research Methodology

This section will go through the methodological view and the research method for the thesis. Work has been done on choosing a strategy for collecting relevant data, choosing a methodological perspective to interpret and analyze the data, and constructing a framework for presenting and discussing the data collected in the thesis

The chosen methodological perspective in the thesis is the systems perspective. The exception to this is the results section, where the perspective is more analytical. The reasoning behind the chosen perspective is the ability to include concepts and principles from different areas of study, and to study system interrelations and dynamics (Arbnor and Bjerke, 2009). This is relevant when interpreting and discussing the impacts of the constructed experimental model on Industry 4.0 and when discussing how this influences the usefulness of SPC in Industry 4.0. The thesis is written as a combined literature review and experimental quantitative study and is therefore multimethodological.

2.1 Research goals

The main research question of this thesis has two different parts, where the first part includes extending SPC as a tool to include machine condition determination and RUL estimation. This is a question that can be answered using a quantitative experimental study. Answering the second part on the usefulness of SPC in I4.0 is more qualitative and requires a literature review. Based on this, the main goal of the research is to provide a useful framework to maximize the ability to answer the two parts of the main research question. To help fulfill this main goal, there are certain research sub-goals:

- Obtain full run-to-failure control charts for all variables and all included engines in the C-MAPSS simulator dataset used.
- Create control chart time-series of all variables in all engines.
- Test all produced control charts using the complete Western Electric ruleset.
- Create an algorithm to transform variable test results into a condition indicator.
- Test distributions of indicator scores at different times from the point of failure.
- Calculate sigma values for the distributions to allow for failure-time estimation.
- Use a literature review to build a theoretical foundation.
- Interpret the results and discuss the implications related to both parts of the research question.

The hope is that by fulfilling these sub-goals, the ability to answer the main research question in a satisfactory manner will be strengthened.

2.2 Methods used in the thesis

Two different methodological approaches were used in this thesis. A literature review was done to gather relevant information on theory relevant to the topic of the thesis to better understand related theoretical concepts. Reading the reference lists of other articles has also helped in finding new relevant theory and additional insights for the thesis. In addition, the theory from the literature review was used to aid in interpreting results and discussing the implications of both parts of the research question. This review was instrumental in highlighting areas where the different theories included interconnect and was useful in helping create the discussion section of the thesis.

A decision was made to answer the first part of the main research question by constructing a quantitative experimental model using SPC as a central analytical tool on a dataset containing complete run-to-failure data. The decision on including run-to-failure data was influenced by the assumption that predicting and estimating failure using SPC and statistical methods would be more likely to produce meaningful results if the analysis was based on historical data.

The C-MAPSS dataset was included because of its availability and abundance of run-to-failure data. The fact that the dataset is widely used in RUL estimation research was also a contributing factor. The data is based on simulations of jet engines and is therefore not based on real-world historical failure data and not necessarily directly related to industrial processes. The lack of such datasets from more related processes contributed towards the decision of using the simulator dataset for constructing the model.

The data was extracted from the first of four datasets in the C-MAPSS data folder, and 100 excel sheets were made, each containing complete run-to failure data for one engine. All variables were identified and appropriately named in the excel sheets. External control variables and static variables were removed from each engine, and 11 dynamic internal engine variables remained. This decision was in large part influenced by an idea that the methods used in the model construction would allow for increased generalization towards industrial processes.

The Shewhart X-Bar and R control charts were initially manually constructed in Excel and testing each chart using the complete WE ruleset was a very slow process. Even learning some Excel algorithms for speeding up the process could not help with producing charts at a reasonable speed. An attempt was made to import engine datasets into MATLAB to allow for automation in making the control charts and specifying test ruleset. Although faster, the process was deemed not sustainable for constructing several thousand control charts. The solution was using Minitab. This allowed for greater automation in producing X-bar and R charts and in addition automatically provided full documentation on all test results.

Initially, tests were performed using sample sizes of 90 and 60, and in addition subgroup sizes of 3 and 5. Calculating degrees of freedom and coefficient of variation led to sample size 90 being preferred. Subgroup size 3 was chosen based on subjective preference. Control charts and full test data for the entire runtime were constructed for every variable and for all the 20 engines included to gain insight into variable behavior throughout the runtime and to consider similarities or differences in the same variable across multiple engines.

Control charts and test results were then constructed in a time series containing 90 samples with subgroup size 3 in 10 cycle increments until the point of failure for all variables and for all 20 engines. All charts and test data were imported into Excel. Different algorithms were tested for creating a scoring system that would indicate the condition of the engine across its runtime. A decision was made to utilize an algorithm that used a condition indicator score of 100 as a starting point. The algorithm gave all variables an initial score of 9, and then subtracted half a point per test failure at that point in time up to 18 test failures. A variable failing tests at 18 or more points in one chart would lead to the condition index score decreasing by 9 caused by that one variable. Aggregating scores from all 11 variables meant that maximum condition index score would be 100, and minimum score possible 1.

Condition index scores for all 20 engines were produced using test data from all variables from the entire time series. The different engines in the data set had vastly differing runtimes, so a meaningful way of comparing index scores was to start at the point of failure and back in time. This allowed for calculating indicator mean scores and standard deviation or sigma. Testing the distributions of indicator scores from all engines using a Kolmogorov-Smirnov test for normal distribution allowed for simple calculation of the probability of an index value being inside or outside of the distribution at the points passing the normal distribution test. This then in turn allowed the model to probabilistically estimate RUL for all jet engines placed into the model. The total construction of all in all between 5500 and 6000 control charts and test results resulted in time spent constructing the model being close to 1000 hours. This meant that including data from more than 20 engines could not reasonably be done within a 20-week timeframe.

2.3 Validity and reliability

An attempt has been made to maximize the opportunity to check the reliability of this thesis in terms of transparency and description of methods utilized allowing easy access to attempt to reproduce results. In terms of reliability as in how well the results are transferable to a process population of real jet engines and even industrial machines requires extensive testing using the same or similar methods to determine (Golafshani, 2003).

The reliability of the thesis is increased through ensuring that articles are peer-reviewed, and by using commonly accepted books, articles and reports on the topics included in the thesis. An attempt has been made to avoid bias when presenting theory in the literature review section, but there is a possibility of misunderstanding or misinterpreting arguments from the original author(s). Any subjective bias in the discussion part also has a possibility of negatively influencing the reliability of the thesis (Leedy and Ormrod, 2015).

In terms of validity, an assumption is made that it seems likely that there is a correlation between SPC test failures and process condition. The model estimating RUL and determining the condition of the process is based on valid and generally accepted statistical methods. However, mistakes can occur. More testing and similar research can help determine the real validity of the thesis. It is assumed that the model probabilistically with at least some accuracy measures engine condition.

A thesis can never be fully objective, and any personal biases and subjective interpretations of data or information may negatively influence validity (Leedy and Ormrod, 2015). External validity may be limited, as the data is subjectively interpreted and discussed. This thesis' priorities and measures of relevance may differ from other researchers, and this is an element that can be criticized.

3. Literature review

3.1 Industry 4.0

Industry 4.0 or Industrie 4.0 (I4.0) is a term describing a currently ongoing and future fourth industrial revolution and associated concepts that may be difficult to clearly define in individual cases. This industrial revolution is happening as a result of an integration of the Internet of Things (IoT) with manufacturing (Hermann, Pentek and Otto, 2016). Substantial advancements in multiple technologies have been influential in enabling I4.0 development. A rapid increase in data volumes, cloud storage, the renting of computing power from external sources enabling data analysis on a level previously unattainable, increased capabilities in analytics, human-machine interactions like Augmented Reality and further innovation and developments in data transfer to physically usable objects (Lee, Bagheri and Kao, 2015).

The two main drivers of I4.0 is an application pull and a technological push. By application pull, the reasoning is that there is a need for changes through social, economic and political means in operative framework conditions (Lasi *et al.*, 2014). The time allocated for development and innovation should be shortened, and the capability for innovation should be maximized. Individualization on demand or “batch size one”, in large part caused by buyers having more market power, leads to increased individualization of products and individual products. There should be an increased focus on the flexibility of product development and production, decision making should be more efficient, and this may cause decentralization and reduced organizational hierarchies. Shortages and increased price of resources forces increased resource efficiency.

Secondly, there is an increased technological push in industrial practice (Lasi *et al.*, 2014). Technical aids that support physical work will be increasingly utilized in the future, and automatic solutions will independently control and optimize manufacturing. Increases in the digitalization of manufacturing and manufacturing support result in a substantial rise in registered actor- and sensor-data. Technical components are also increasingly networked and that leads to fully digitized environments. This in turn drives new technologies, like simulation, digital protection and augmented reality. The development also trends towards miniaturization, where computers require less space and have increased capacity. This enables new application development within production and logistics.

Industry 4.0 can be described by different concepts. Factories where sensors connected through IoT are utilized in manufacturing machines and processes, Smart Factories, and systems can communicate with both humans and other systems with different levels of autonomy. Digitized models of products and factories are developed and autonomously controlled (Lasi *et al.*, 2014). Cyber-Physical Systems (CPS) are created by the merging of the physical and the digital levels in a way that blurs the differentiation of physical and digital representations. CPS can be defined as an integration of computation, networking and physical processes (Lee, 2008). Processes parameters and component wear and tear are measured and recorded digitally. As mentioned, manufacturing systems trend towards decentralization, and traditional hierarchies decompose. This contributes to a change towards more decentralized self-organization. Distribution, procurement and development of products and services will be increasingly individualized to adapt to human needs. Lastly, there is an increased focus on sustainability and resource-efficiency that influence the design of manufacturing processes. To aid the implementation of Industry 4.0 goals, an implementation of horizontal and vertical integration, networked manufacturing systems and end-to-end digital integration of engineering across the entire value-chain is needed (Kagermann, Wahlster and Helbig, 2013).

The foundation of Industry 4.0 can be described by advances in nine main technologies, and many of them are already used in manufacturing. In Industry 4.0, these technologies will lead to integration, automation and production flow, increased efficiency and changing supplier, producer, customer and human-machine relations (Rüßmann *et al.*, 2015). These technologies are; automated robots, simulation, horizontal and vertical system integration, the Industrial Internet of Things (IIoT), cybersecurity, the cloud, Additive Manufacturing, Augmented Reality and big data and analytics. IIoT integrates machine sensors, middleware, software and backend cloud compute and storage systems to increase visibility and insight into operations and assets (Sadeghi, Wachsmann and Waidner, 2015). Advances in sensor technologies allow for precision, self-awareness and even prediction of own remaining life. Developments in miniaturization and sensor technology makes using sensors more practical, as they can more easily be embedded.

Relatively recent advances in computer technology allows for analysis of large data sets to optimize production quality, efficiency and service, so-called Big-Data. This provides historical, predictive and prescriptive analysis that can aid in revealing current functionality inside machines or processes. Through Industry 4.0, the collection of data from all sources in the value-chain will become increasingly standard, and will support the ability for real-time decision-making (Rüßmann *et al.*, 2015). More autonomous, flexible and cooperative robots lead to closer robot-robot and human-robot interactions. Simulation of products, materials and processes allow businesses to model the real world in real-time, and to test and optimize setting before implementing them in the physical world, leading to increased quality and reduced set-up time.

Going towards I4.0, IT-systems will be increasingly integrated horizontally and vertically, and this allows for more automated value-chains. Through the Internet of Things, devices are connected and can communicate with each other or to centralized controllers. This decentralizes analysis and decision-making, and enables real-time responses (Rüßmann *et al.*, 2015). The increased connectivity and communication protocols will drastically increase the need to protect critical systems and manufacturing lines through secure, reliable communications and identity and access management of machines and users. Integrated cloud-based software will allow for more data-driven services for production systems with low response times.

The development of Additive Manufacturing allows for effective prototyping and individual component manufacturing (Wong and Hernandez, 2012). In Industry 4.0, this will develop further into small-batch production of customized products, and decentralized production facilities will reduce transport distances and stock (Rüßmann *et al.*, 2015). Augmented reality can contribute toward supporting warehouse part selection, repair instructions, real-time decision-making and work procedures.

Large scale implementation of Industry 4.0 will be time-consuming, as there are scientific, technological, economic, social and political challenges involved. A smart factory will involve more artificial devices, requiring less workers, and there is still a need for more time and money to develop smart device technology (Zhou, Liu and Zhou, 2015). Building Cyber-Physical Systems must consider the collaboration of physical and computing systems and assuring access to information from physical systems is complex. Big Data analysis introduces issues within information security and privacy. To combat challenges in implementation, some argue for the development of international standards (Kagermann, Wahlster and Helbig, 2013).

When working toward the implementation of I4.0, security is one of the most important aspects. Data contained within systems need to be protected against misuse and unauthorized access (Kagermann, Wahlster and Helbig, 2013). The highly networked systems of CPS-based systems, often containing critical data, and a greatly increased number of actors involved in the value chain, reveals multiple security issues that must be addressed before implementation. As such, security must be designed into the system from the start, and security strategies, architectures and standards need to be developed and implemented.

What investments are necessary and how different I4.0 technologies are implemented will differ from organization to organization. It is necessary to obtain an overview of current state of the art in the organization concerning technologies and systems already in place. An Acatech study on I4.0 (Schuh *et al.*, 2017) proposes a Maturity Index, where an organization inputs its corporate strategy and the technologies and systems that have been implemented. An analysis on current capabilities and future desired benefits is performed. Then, the missing capabilities to achieve these benefits is identified, and a gap analysis can be utilized. This methodological analysis allows for the creation of a digital roadmap, including a step-by-step approach, reducing risk for both investments and implementation.

3.1.1 5C architecture

Cyber-Physical Production Systems (CPPS) is a subset of CPS. The main characteristics of CPPS are; intelligence as in the ability to acquire information from surroundings and act autonomously, connectedness as in the ability to set up and utilize connections to other elements of the system for collaboration and cooperation and responsiveness to internal and external changes. CPS consists of two components; real-time data acquisition from the physical world and information feedback from the cyber space through advanced connectivity and secondly, intelligent data management, analytics and computational capability that allows for building the cyber space (Lee, Bagheri and Kao, 2015). A proposed architecture, 5C, describes 5 levels of sequential workflow and describes the steps from data acquisition to value creation. The 5C architecture is shown in Figure 1.

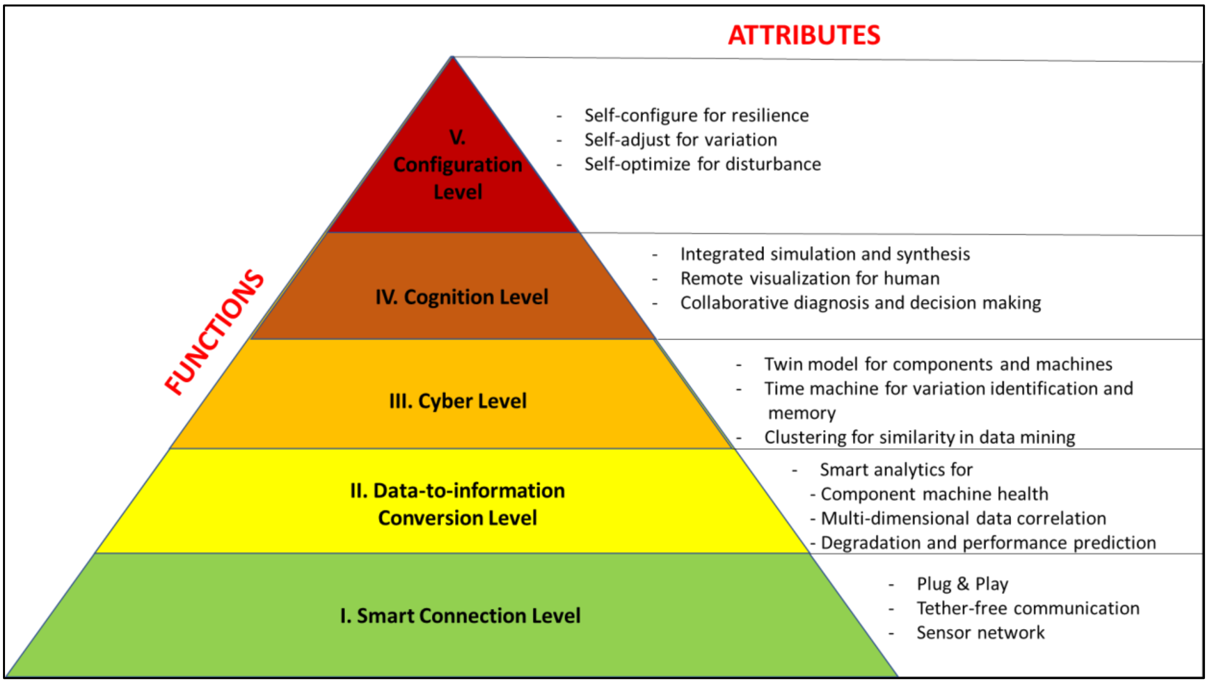


Figure 1 5C architecture for CPS implementation based on (Lee, Bagheri and Kao, 2015).

3.1.2 IIRA architecture

As a part of the IIoT, Industrial Internet Systems (IISs) will enhance business process flows and analysis. To make this possible, IISs need a standard-based architectural framework. One such framework, The Industrial Internet Reference Architecture (IIRA), is proposed by the Industrial Internet Consortium (IIC) (Lin *et al.*, 2017). The framework describes four viewpoints; business, usage, functional and implementation. Business describes system value delivery and business strategy alignment among others. Usage addresses the creation of user and system activities that deliver required outcomes. Functional relates to the functional components and how they relate and integrate internally and externally. An IIS can further be divided into five domains; Control, Operations, Information, Application and Business. A visualization of the IIRA architecture is shown in Figure 2.

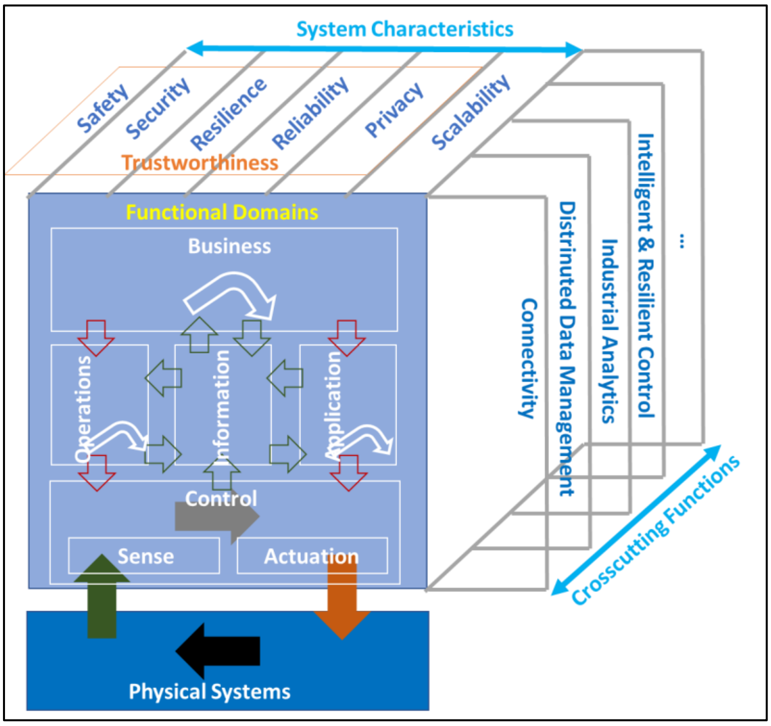


Figure 2 IIRA architecture based on (Lin et al., 2017).

3.1.3 RAMI 4.0

As much as the IIRA architecture is the American reference model for American industry, its European counterpart is called the Reference Architecture Model Industry 4.0 or RAMI 4.0. The model has been developed by BITCOM, VDMA and ZWEI, and is a three-dimensional model based in large part on the Smart Grid Architecture Model (SGAM) developed for renewable energy sources network communication (Zezulka *et al.*, 2016). The three dimensions of the RAMI 4.0 model can be described through referring to three specific axis; the vertical axis with layers, the left horizontal axis describing the life cycle and value stream of production and finally the right horizontal axis describes the hierarchy from product through enterprise to the connected world at the top level.

The bottom level of the vertical axis is the Asset Layer is a model representation of the real physical world and can describe elements like documentation, diagrams, product parts and even humans. The Integration Level transforms data from the Asset Level to be suitable for computer processing, can perform controls on data, generates events based on the assets and handles data from for example RFID chips, sensors and actuators. The Communication Level standardizes the data format and can also perform control on the previous level. The Information Layer handles event pre-processing, rule execution, ensures data integrity to provide high-quality data, information and knowledge. On the Functional Layer level, formal function descriptions are enabled and a platform for horizontal integration is created. The top level, the Business

Layer, ensures value stream function integrity, maps different business models and overall process results and contains framework conditions for laws and regulations. This top layer also links the different business processes (Zezulka *et al.*, 2016).

The left horizontal axis is split into two parts; type and instance. Type describes a product, machine, software or hardware from an idea through development, design and testing up to production prototype and eventual validation. Instance describes the production of unique, individual products that are sold to customers. These products are then used and may need maintenance (Zezulka *et al.*, 2016). A visual representation of the RAMI 4.0 model is found in Figure 3.

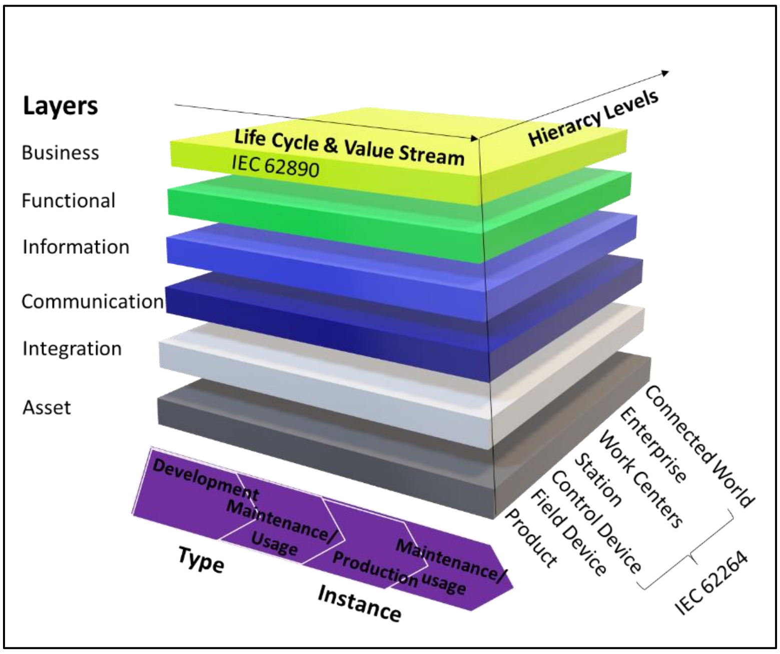


Figure 3 RAMI 4.0 architecture based on (Adolphs et al., 2015).

3.2 Machine Learning and Artificial Intelligence

Modern Artificial Intelligence (AI) traces back to early 20th century inventions in electronics and the rise of modern computers post-WWII. In its infancy, AI was influenced by a number of disciplines like engineering, biology, experimental psychology, communication theory, game theory, mathematics and statistics, logic and philosophy and linguistics (Buchanan, 2005).

Machine Learning (ML) involves the construction of sets of algorithms that can learn from and make predictions based on data. ML can be either supervised, learning by example, or unsupervised. Unsupervised learning is feeding data to the learning system without any labels, and letting the algorithms try to identify distributions, structure, clusters, probabilities and so on by statistical inference (Dunjko and Briegel, 2018).

In ML, there is a distinction between artificial neural networks (NN) and support vector machines. NNs are inspired by biology, where a set of multiple artificial neurons interact with inputs and other neurons to produce an output, and each neuron is a function or an algorithm. The set of neurons can be layered in multiple levels, and signals can be sent back and forth from one layer to another, depending on the type of NN. The NN can be feed-forward with no loops, or recurrent where the output is fed back into the network. Support vector machines are generally supervised systems that analyze and classify data using mostly non-probabilistic methods and for kernel methods which is a form of pattern recognition (Dunjko and Briegel, 2018). An example of a simplistic NN is shown in Figure 4.

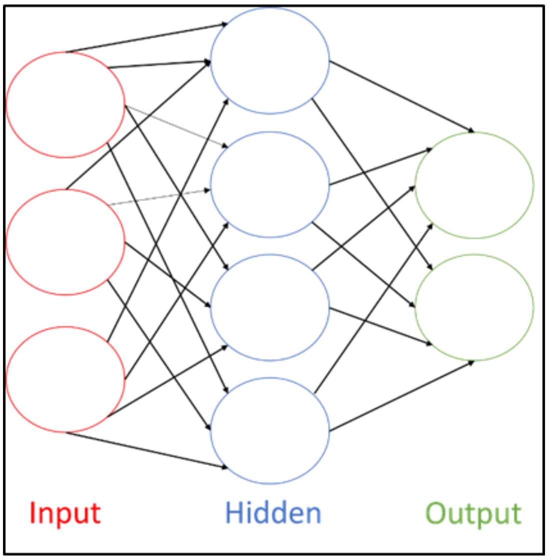


Figure 4 Artificial neural network of interconnected neurons through inputs, hidden layers and outputs.

The Machine Learning sub-topic of Deep Learning (DL) functions by using so called representation learning, which can be explained as feeding raw data to a system, which then automatically discovers the representations needed for detection and classification (LeCun, Bengio and Hinton, 2015). This process then goes through multiple levels, transforming the data into ever higher and more abstract levels. DL is thought to have a very high potential, especially because of the low amount of engineering the system requires to function.

Reinforcement learning (RL) is a form of ML where the system is not fed static data but is connected to an interactive environment of tasks. The system learns through interaction with the environment. The outcome of the learning algorithms can be modified through four modes of system conditioning; positive reinforcement where the system is rewarded when correct, negative reinforcement where a negative state is removed when correct, positive punishment as in introducing a negative reward when incorrect and negative punishment where a reward is removed when the outcome is incorrect (Dunjko and Briegel, 2018). This process of learning emulates similar ways of biological conditioning.

Using ML software solutions, with one example being Microsoft Azure ML, raw data is collected and run through an iterative series of data pre-processing modules that gradually prepares the data. The learning algorithms are then applied to the data, and again through an iterative process gradually builds a candidate model. The model that is chosen, is then deployed and can be applied to various purposes (Chappell, 2015). In most cases a series of candidate models is produced and improved upon, before choosing a final model.

As a result of the often-probabilistic nature of NN algorithms outputs will seldom be of a binary yes or no nature, but will often return a probability between 0 and 1. Depending on the type of problem the algorithms are set to solve, choosing a probability that is acceptable is subjective and dependent on the individual case (Chappell, 2015).

Multi-layered NNs, often called convoluted neural networks (CNN) and deep NNs, have some challenges. The complexity of the functions realized by the NN can be difficult or impossible to interpret. Even if output results often are very successful, there is a lack of understanding of why they are successful. The ability to successfully interpret the results from NN algorithms is a central issue when decision-making is increasingly delegated to these systems. The result of this general lack of understanding and interpretability, NNs are generally not used for decision-making in critical systems where the consequence of a failure might be catastrophic (Dunjko and Briegel, 2018).

The possibilities of future developments in AI raises some ethical issues. If machine learning algorithms are based on complicated neural networks, it may be extremely difficult or even impossible to determine how the algorithm functions. ML software does in most cases include modules that are so-called “Black boxes”, where some or most of the inner workings of the software is hidden (Wikipedia, 2019), making it impossible to get a complete view of how the software algorithms works. If an AI produces recommended actions as outputs, and statistical analysis shows that these recommendations prefer solution X and not Y, and choosing X over Y is ethically questionable, discriminatory or morally ambiguous, how do we determine why this is happening (Bostrom and Yudkowsky, 2014)?

3.3 Quality Engineering

Traditional definitions of quality tend to be qualitative and based on attributes. A precise quantitative definition is absent, and they tend to be binary, as in they describe if a product or service is within or outside set specifications. This motivates improving quality until it is within acceptable levels. There has also been a historical focus on the result of the manufacturing process, and less focus on the design stage. This entails an application of the concepts of quality at the end of the manufacturing process instead of at the beginning. Definitions of quality should connect functionality with the engineering design process and promote never-ending or continuous improvement. In addition, any definitions of quality should increase the focus on process instead of product, to prevent instead of contain lack of quality(Wang, 2013, Scherkenbach, 1986). Quality should be measured as a function of customer losses instead of producer losses.

3.3.1 Deming

The Deming method is heavily influenced by statistics in both philosophy and methods. Deming lists 14 points that are beneficial when working towards improving quality. Management should be able to adopt a long-term strategic perspective on research and education and be able to adopt the idea that quality is a never-ending process of improvement. The focus should be moved from defect detection to defect prevention. Controlling the process is more beneficial in identifying root causes than controlling the product (Scherkenbach, 1986). The difference between product control and process control for quality is shown in Figure 5 and Figure 6.

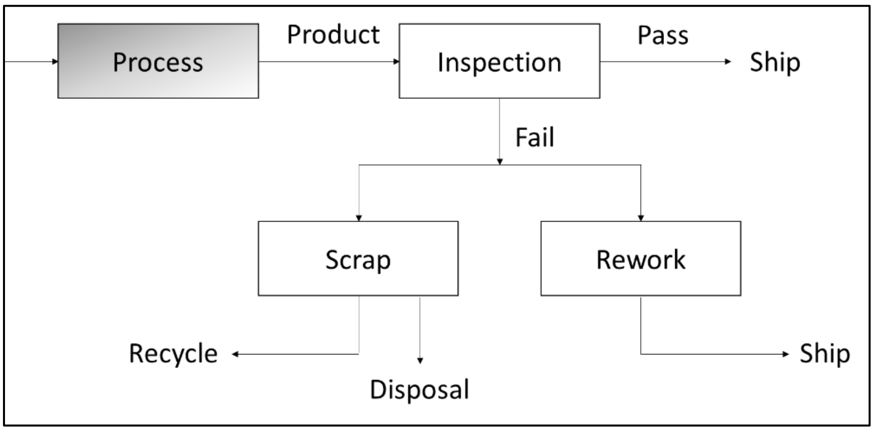


Figure 5 Product-oriented quality model.

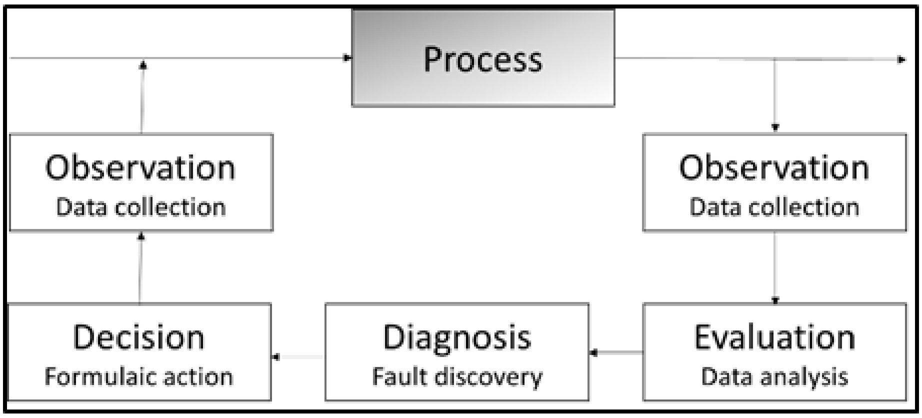


Figure 6 Process-oriented quality model.

Continuous improvement of production, service, quality and productivity is paramount in the Deming approach, as it will lead to a constant decrease of costs. This is the basis of the Deming cycle, a four-step iterative procedure; recognize the existence of an opportunity, test a theory to be able to achieve this opportunity, observe the results of this test and act on the opportunity. All these can be placed in a feedback loop, and gives insight into how Deming defines continuous improvement (Scherkenbach, 1986).

3.3.2 Taguchi

The Taguchi method is not only concerned about building a quantitative formulation to the design of experiments (DOE), but perhaps more importantly on building an understanding of a philosophy. This foundation of this philosophy is based on three basic ideas; quality should be part of the design of a product, not by inspection of the product, quality is achieved by minimizing variance from an optimal target by design and the cost of quality should be measured by how product variables deviate from this optimum. These losses should be measured across the entire system (Roy, 2010).

Traditional perspectives on product quality describes loss when a product deviates from an optimal target in such a way that it is no longer inside a set of given specifications for the product, causing rework or discarding the product. Taguchi identifies that two samples of a product both can be within specifications, but still be very different in how product variables deviate from an optimum. When quantifying this perspective, the result is a continuous function of loss as product variables deviate from a perfect or optimal product (Roy, 2010). An illustration of the difference between a traditional within/outside of spec loss function and a Taguchi loss function is found in Figure 7. Taguchi methods can be beneficial when applied at the earliest stage of product or process development and tend to become costlier in later development stages. The Taguchi quality loss function can be described by $QL \equiv C_{fv} + C_o + C_{ee}$

, where QL is quality loss, C_{fv} is cost due to functional variation, C_o is operating cost and C_{ee} is cost due to environmental effects (Taguchi, 1995).

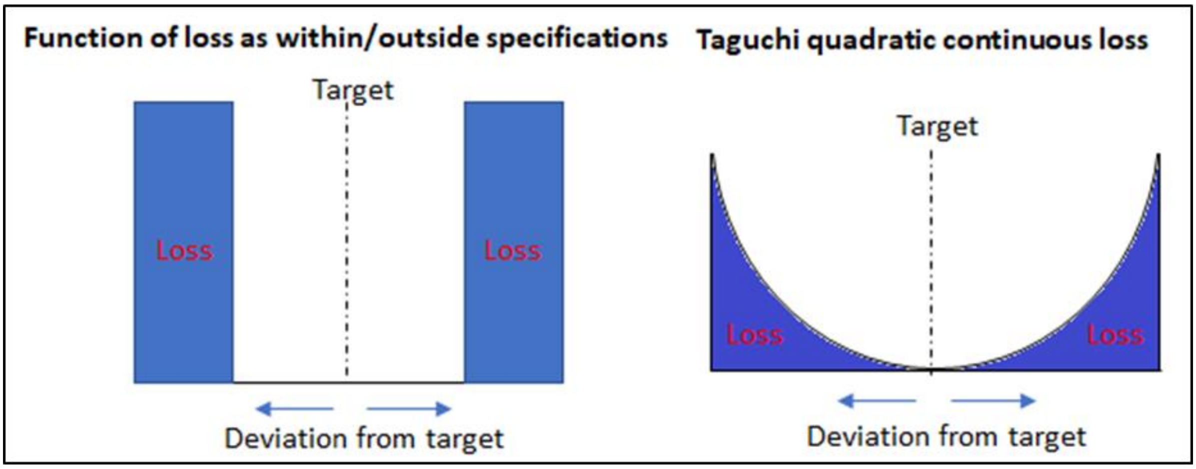


Figure 7 Example of within/outside of spec loss function and Taguchi continuous loss function.

When quantifying loss of quality by monetary means with Taguchi loss functions, an interesting question is the cost of achieving quality by reducing variance. By interpreting the loss function philosophy literally, one would assume an optimal point of level of quality and cost of achieving that quality. This would in effect be contrary to a philosophy that promotes continuous improvement. But when costs related to maintaining a system for quality control, maintaining a system for quality assurance, losses, scrap and rework in manufacturing and warranty, repair and service, any logical assumption on optimal quality would lead to quality within acceptable limits. By adhering to the Taguchi perspective on loss, a central question is the cost of not having quality or not continually working on quality when this work could lead to improvements in production management, increased flexibility and adaptability, creating and enhancing opportunities for innovation and breakthroughs, encourage employees by transparency in the search for improvements and an increased ability to identify flaws in the processes(Lofthouse, 1999).

3.3.3 Zero Defect Manufacturing

The concept of zero-defect was introduced in the US Army during the 1960s. It generally discusses manufacturing practices that seek to minimize the number of process defects and errors and to instill a practice of doing things right the first time. The utilization of Zero-Defect Manufacturing (ZDM) practices in a manufacturing environment can help improve quality and minimize cost (Wang, 2013).

In order to achieve zero defects, quality monitoring and optimization tools must be used. A ZDM system has some requirements; automatic capture, cleaning and formatting of data using a sensor system, automatic signal processing, filtering and extraction of features, data mining and knowledge discovery for the purpose of diagnosis and prognosis, provide clear and concise information and advice on defects and the ability to self-adapt and optimize (Wang, 2013).

A ZDM framework named intelligent fault diagnosis and prognosis system (IFDAPS) has been developed at the Knowledge Discovery Laboratory at the Norwegian University of Science and Technology (NTNU) (Wang and Wang, 2012). The functions of this framework are; continuous collection of data from sensors, continuous processing of sensor data and online evaluation of equipment and processes, identification of conditions or faults and inform operators and managers if and where components, machines or processes have degraded to a degree that actions should be taken. As a result of condition identification, remaining useful life or possible future faults can be predicted, operation and plans can be optimized and produced performance indicators can be used for self-adjustment to correct or compensate for faults (Wang, 2013). An EU-project called Intelligent Fault Correction and self-Optimizing Manufacturing Systems or IFaCOM has worked on building a methodology and a framework for connecting ZDM philosophies to Industry 4.0 and Cyber-Physical Systems (Eleftheriadis and Myklebust, 2016). Figures 8 and 9 visualizes the IFDAPS and IFaCOM models.

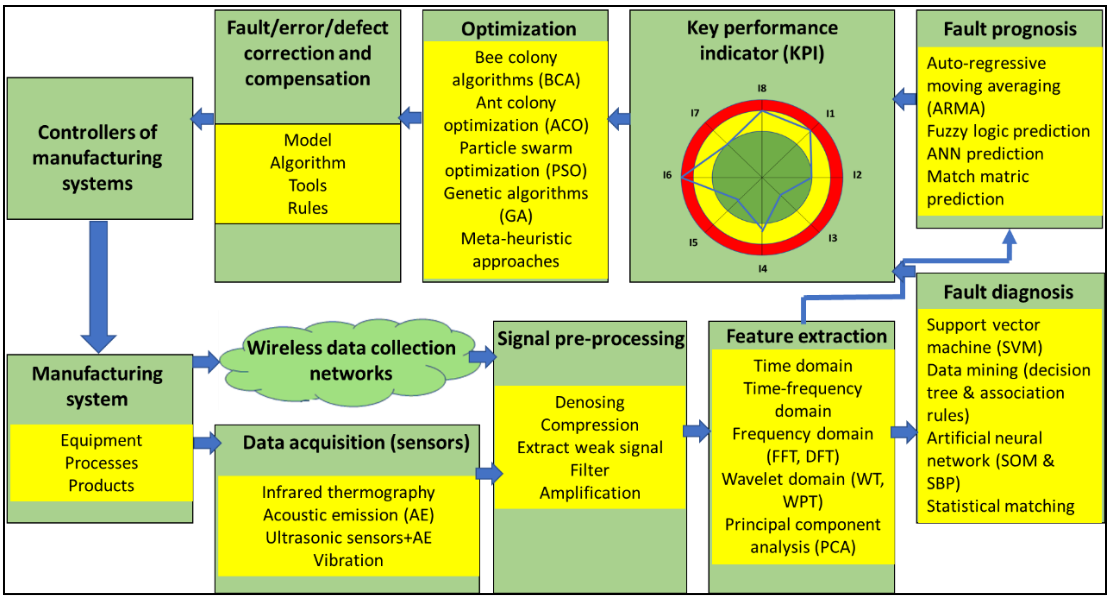


Figure 8 Intelligent fault diagnosis and prognosis system (IFDAPS) based on (Wang, 2013).

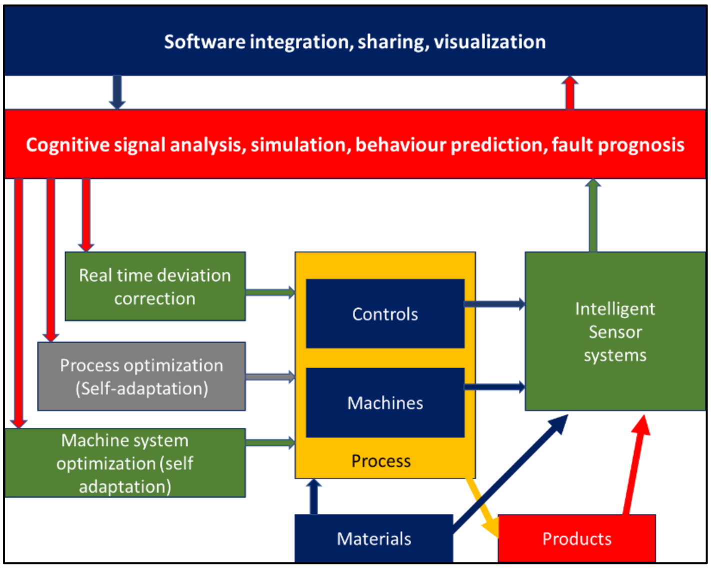


Figure 9 IFaCOM framework based on (Eleftheriadis and Myklebust, 2016).

3.3.4 Statistical Process Control

Statistical Quality Control analyzes data using scientific methods, and applies this data to solve problems within engineering, management, maintenance and more or less any other activity that can be expressed using numbers (WEC, 1956). Statistical refers to methods drawing conclusions from numbers, quality has to do with the characteristics of what is being studied and control is related to keeping what is being observed within specific boundaries. In Statistical Process Control or SPC, the object being observed and analyzed is a process. SPC is closely related to ideas from both Taguchi and Six Sigma in that it seeks to identify, monitor and minimize process variation (Montgomery, 2007).

A process can be defined as a set of conditions or causes, that work together to produce a result (WEC, 1956). Another definition is a set of interrelated or interacting activities that use inputs to deliver a result (EN17007, 2017). This includes single and multiple machines, single or multiple human actions, measuring or assembly methods, anything else that can be expressed as a series of numbers and any combination of these. It can refer to a single operation or a complicated combination of multiple operations.

Process data of a multitude of processes are collected and charted by most industrial enterprises, but often only used in reports or summaries before they are archived. How these data are connected to the historical performance of processes is seldom properly analyzed. Shewhart control charts uses process data to ascertain if a process is in or out of statistical control. A process is controlled if, using historical data, it is possible to predict, within limits, how the process will vary in the future (WEC, 1956).

A process out of statistical control may influence productivity and economic efficiency and is in effect wasteful and inefficient. A process may produce products within the set quality parameters, but still be out of statistical control. Producing within specifications meets customer expectations, while producing within specifications and while the process is in statistical control meets customer expectations economically (WEC, 1956).

Multiple types of data can be used to detect causes of process disturbance. In order of sensitivity, ranges, averages, percentages or individual numbers like temperature or pressure can be useful in identifying problems in a process (WEC, 1956).

The X-Bar and R charts measure how process variable means and variance behave over time and uses upper and lower control limits to analyze whether the variance within the charts is common cause, variance within the control limits, or special cause, outside the limits. This can also be referred to as normal or abnormal variance (WEC, 1956). The Shewhart control charts can serve as a tool to identify faults in a process, help identify root causes for process behavior and form a statistical basis for any actions to improve the process. In addition, the charts can inform decisions on whether a process needs adjusting or maintenance or can be left alone. Depending on the patterns that are formed within these charts, these can aid in identifying cycles, trends, shifts, instability or interactions between two or more variables (WEC, 1956).

Shewhart control charts are largely based on analysis of a normal distribution, and the H0 hypothesis is that the X-Bar and the range mean is equal to the process means. By applying control limits of +/- 3 sigma to the y-axis, where the α is 0,3% and the significance level is 99,7%, points outside the control limits will lead to rejection of the H0. In addition, the x-axis represents time, and adds another dimension to the hypothesis testing. This introduces a more complex ruleset for assessing whether the H0 should be rejected or not(WEC, 1956).

The aim of SPC is to use historical data to improve future performance, and traditionally the analysis has tended to focus on the output rather than the process or processes. An output in the form of a product can be a result of multiple process variables and using control charts to identify root causes to why there are problems with the process might be difficult or impossible. By analyzing individual process variables, the chances of identifying root causes rises, and therefore is a logical step. This perspective also makes control charts viable within service systems and other systems that have no tangible output to measure (Wood, 1994).

Today, most processes are monitored by computers, and a single process can have 50 or more variables that are monitored every second or multiple times per second (Nomikos and MacGregor, 1995). As mentioned, a process can be partly defined as anything that can be represented by a series of numbers (WEC, 1956). Monitored process variables have this trait, and when the data from multiple variables are synchronized, SPC can be performed across all monitored process variables (Nomikos and MacGregor, 1995).

Experiments have been done, using SPC to construct control charts of vibration variables from bridge columns. In this case, the method is used to detect anomalous vibration measurements from the construction. In all cases, the control charts were successful in detecting some system anomaly. Testing on method robustness also confirmed that there was no indications of producing false-positive warnings when using the charts for anomaly detection. Outliers in the control charts did not necessarily indicate structural damage but did indicate statistically significant changes to the vibration signature of the bridge columns. Damage to the columns that does not change the vibration data significantly cannot be detected by the control charts (Fugate, Sohn and Farrar, 2001).

3.3.5 Constructing the X-Bar and R charts

The variable or variables to measure must be chosen, number of datapoints included in the sample and the size of the subgroup. Degrees of freedom is a measure of uncertainty in estimated sigma. Related to sample size, degrees of freedom will rise as sample size rises, and the better the estimate of moving range and sigma. The coefficient of variation (COV) measures uncertainty in sigma and is defined as $COV = \frac{sigma}{mean}$. When constructing X-Bar and R control charts, degrees of freedom can be calculated using $df = 0,9k(n-1)$ for n<7, and where k is number of subgroups and n is subgroup size. COV is calculated using $COV = \frac{1}{\sqrt{(2df)}}$. Using

these formulas, it is possible to calculate good sample sizes for the control charts (Wheeler, 1995). The relation between sample size and degrees of freedom is shown in Figure 10. An example of the relation between degrees of freedom and coefficient of variation can be seen in Figure 11.

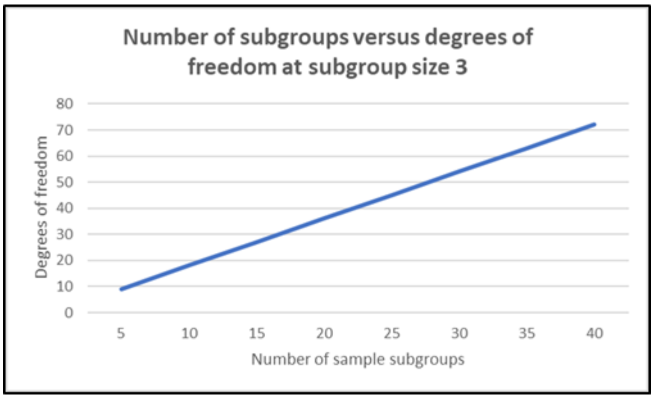


Figure 10 Number of subgroups vs degrees of freedom at subgroup size 3.

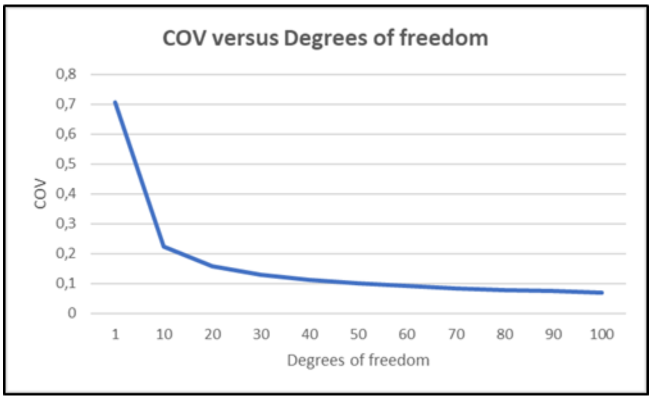


Figure 11 COV vs degrees of freedom.

Means are calculated for every subgroup in the sample using $\overline{X} = \dfrac{X_1 + X_2 ... X_n}{n}$ where $\overline{X}$ is the average of a series of X's, X is an individual observation and n is the number of observations in a group. N should be 2,3,4 or more, but should not exceed 10(WEC, 1956). Then the mean of all subgroup means is calculated. This provides the points in the X-Bar chart and the center line in the chart. The standard deviation of the sample or sigma is calculated using $s = \sqrt{\dfrac{1}{N-1}\sum_{i=1}^{n}(X_i - \overline{X})^2}$, and +/- 3 sigma marks the upper and lower control lines. Ranges are calculated for every subgroup using $R = N_{\max} - N_{\min}$. The mean of R is calculated to find the center line of the R chart. The control lines for the R chart is found using constants, where the formulas are $UCL = d_4\overline{R}$ and $LCL = d_3\overline{R}$. A table showing these constants are shown in Table 1.

Subgroup n	d3	d4
2	0	3,267
3	0	2,575
4	0	2,282
5	0	2,115
6	0	2.004
7	0,076	1,924
8	0,136	1,864
9	0,184	1,816
10	0,223	1,777

Table 1 Constants for calculating R chart UCL and LCL.

To analyze the results from the control charts, 8 tests are performed. (1) Any points outside the control limits of both X-Bar and R charts signals a process out of control. (2) 2 out of 3 points > 2 sigma in the X-Bar chart signals an out of control process. (3) 4 out of 5 points > 1 sigma in the X-Bar chart signifies an out of control process. (4) 8+ successive points above or below the center line of both X-Bar and R charts indicates that the process has shifted. (5) 6+ points in either chart continually increasing or decreasing indicates a systemic trend. (6) 14+ points oscillating up and down in either chart also indicates a systemic trend. (7) 8+ points in a row on either side outside of +/- 1 sigma in the X-Bar chart indicates an out of control process. (8) If 15+ points in a row in the X-Bar chart are within the +/- 1 sigma area, this indicates that the process is out of control (WEC, 1956).

3.4 Maintenance management and predictive maintenance

Maintenance can be defined as a set of activities that are used to restore an item to a state where it can perform the functions it was designed for (Ahmad and Kamaruddin, 2012). Another definition is "*a combination of all technical, administrative and managerial actions during the life cycle of an item intended to retain it in, or restore it to, a state in which it can perform the required function*" (EN13306, 2017).

As technologies evolve over time, so has maintenance functions. Strategies for maintenance can be divided into Corrective Maintenance and Preventive Maintenance. Corrective Maintenance is used to restore equipment to a required functional state after failure. This generally leads to high machine downtime and maintenance costs (Ahmad and Kamaruddin, 2012). Preventive maintenance involves performing maintenance prior to failure, where the goal is to reduce failure rate or frequency. This leads to lower cost of failures and less machine downtime. Time-based maintenance is a form of preventive maintenance, where analysis of previous failure times is used to predict expected lifetime of a machine.

3.4.1 Condition Based Maintenance

With modern technology evolving rapidly, products increasing in complexity and with higher reliability requirements, the cost of preventive maintenance has increased substantially. Condition-based maintenance (CBM) is an attempt to manage this situation. A CBM approach acquires information from a machine that is relevant to the machine's condition or health. This information is then analyzed to increase system understanding. Based on the analysis, a decision is made as to which maintenance policies are most efficient (Jardine, Lin and Banjevic, 2006).

Being able to detect faults in the observed system and identify the nature of these faults is an important part of CBM. In addition, system analysis should provide knowledge that help estimate when and how a fault is likely to occur. Prediction is potentially a very effective tool in reducing system downtime, but the ability to diagnose faults is essential when these predictions fail (Jardine, Lin and Banjevic, 2006).

Using statistical methods when analyzing signal data from a system is relatively common in CBM. A connection between CBM fault detection using SPC has been discussed, where control charts are used to detect anomalies in monitored vibration variables for a concrete bridge column (Fugate, Sohn and Farrar, 2001).

A motivation for CBM is that in 99 percent of all cases there will be signs, conditions or indications that a failure is going to occur before the failure affects the machine observed. If the critical variables are monitored, detection of the failure indicated is possible (Lee *et al.*, 2014). Monitoring can be performed on-line, while the machine is running, or off-line. In addition, monitoring can be periodical or continuous. Continuous monitoring is expensive and can cause signal noise, while periodical monitoring runs the risk of missing critical information between intervals.

3.4.2 Remaining Useful Life (RUL)

Remaining useful life is an estimation of the useful life left of a machine or asset at a given time. This estimation is central to CBM and health management. Typically, RUL for any given machine or process is random and unknown. As such, an estimation is done based on existing information from the observed machine. There are several different approaches concerning RUL estimation methods, but there is not necessarily any best method, at least not a method that can be generally used universally (Si *et al.*, 2011).

A definition of useful life can be the period where an asset or property can be used for the purpose it was intended for. Another definition can be the time where a depreciating asset will be productive (Si *et al.*, 2011). Being able to estimate RUL accurately is critically important in CBM, as it influences maintenance planning, acquisition of spare parts, performance of operations and profitability. RUL is also an important part of product reuse and recycling and is therefore connected to sustainable strategies.

RUL of any given asset is random and depends in large part on the age of the asset observed, operation environment and condition monitoring. An estimation of RUL can be calculated using $f(x_t \mid Y_t) - f(x_t) - \frac{f(t + x_t)}{R(t)}$, where x_t is the random variable RUL, $f(x_t \mid Y_t)$ is the probability density function and Y_t is the history of operational profiles and condition monitoring up to time t(Si *et al.*, 2011).

Applying statistical data driven approaches to estimate RUL, relies on available historical data and statistical models. The data used can be event data or condition data. Event data is synonymous with historical failure data. Depending on the process being analyzed, failure data may be difficult or impossible to acquire. The process in question may not be allowed to ever run to failure. Any statistical approach to RUL estimation is dependent on data availability and nature (Si *et al.*, 2011).

Data obtained from monitoring a process can be classified as direct or indirect condition monitoring (CM) data. Direct CM data can describe the state of the system directly, and threshold levels of that variable can be used for prediction purposes. Indirect CM data can only indirectly or partially indicate the condition of the system observed, and there may be a need for additional failure data to be able to estimate RUL (Si *et al.*, 2011). A visualization of common statistical approaches with direct and indirect DM data can be found in Figure 12.

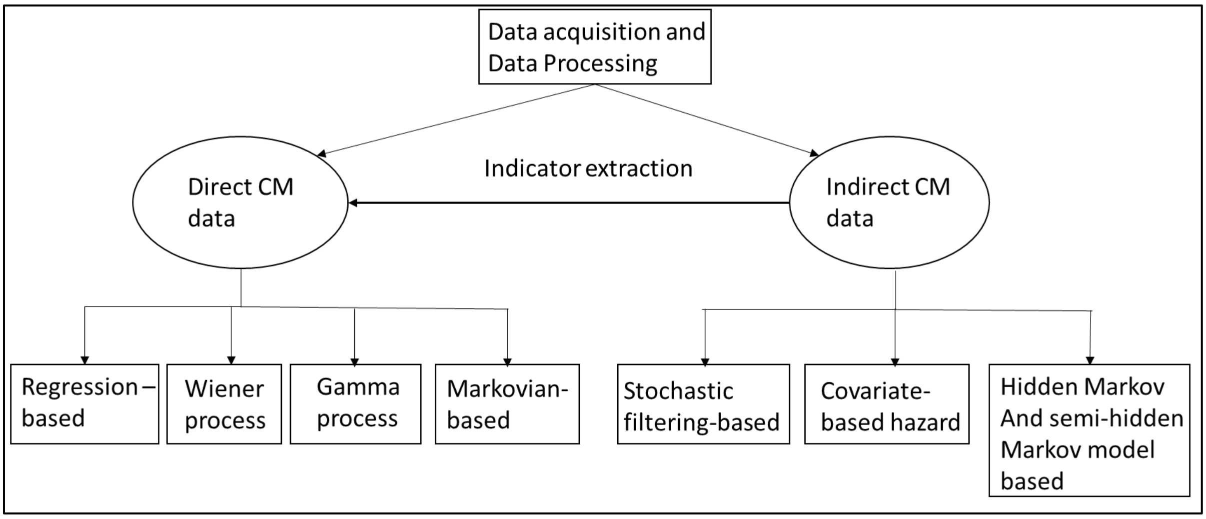


Figure 12 Overview of statistical methods used for RUL estimation.

Generally, failure cases are caused by several correlated processes degrading, and the severity of this degradation can be close to impossible to detect without using advanced equipment. However, data from variables that reflect the underlying degradation are substantially easier to obtain given appropriate sensor use. If critical variables in a process is identified and sensored appropriately, close to continuous variable data can be collected at a relatively low cost (Liao, Zhao and Guo, 2006). The data from these variables can provide insight into the state, condition or health of the process being observed. A challenge is that the same degradation can be cause by another combination of process variables, and therefore the failure boundaries can be difficult to define. Root causes to degradation or failure may be unidentified or not measured properly, and degradation may happen without being observed until process failure.

The ability to estimate RUL accurately can be crucial in many areas of industry. Sectors like aircraft industry, medical equipment and powerplants are examples where inaccurate estimation of RUL can lead to catastrophic results. Being able to schedule maintenance well in advance order parts in time helps create more efficient strategies for replacement and maintenance, and also minimizes costs by avoiding machine maintenance when unnecessary (Zheng *et al.*, 2017).

With the ongoing technological developments within the field of information and communications technology (ICT), there is considerable work being done on using artificial intelligence (AI) and specifically Machine Learning and neural networks to build prediction models for estimating RUL. An example uses Echo State Networks (ESN), a type of recurring neural network (RNN), which inputs signal data into a set of internal units consisting of hidden layers and dynamical reservoirs. Machine Learning algorithms treat the input data through this set of algorithms, and finally produces outputs. The named echo state property, specific to ESN, has the effect that initial conditions gradually disappear as time passes, and ESN is therefore theoretically an interesting application for RUL estimation experiments (Rigamonti, Baraldi and Zio, 2016).

A proposed approach to using ESN for RUL estimation is to use sensor data and perform pre-processing procedures, which includes normalization of data, filtering of signal noise and selecting prognostic signals. The pre-processing output is then normalized and filtered prognostic signals that are significant. The point where signal degradation occurs, the elbow point, is detected using Z-tests. The ESN algorithms then uses this data to predict RUL for the component being observed through a learning process (Rigamonti, Baraldi and Zio, 2016).

Using NNs for RUL estimation can be very data intensive, especially when training the NN. Using 70-80% of datasets for training purposes is not unusual (Zheng *et al.*, 2017). This means that a vastly superior percentage of data is used for training, leaving a lesser percentage of data for testing the NN models.

4. Results

The object of the experiment performed in this thesis is to attempt to model sensor data from a machine process using Statistical Process Control (SPC) and use this model to indicate process condition and Remaining Useful Life (RUL). When attempting to model predictive process failure, one important element to consider is previous failure data (Goode, Moore and Roylance, 2000). Having a set of run-to-failure data from the process being modelled is beneficial regarding quality of analysis. Collecting real-world run-to-failure data is often problematic, as the datasets tend to be incomplete (Saxena *et al.*, 2008), or organizations may be unwilling to provide the datasets for various reasons.

To combat the lack of availability of such real-world datasets, the experiment will use a dataset from the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) at NASA (NASA, 2008). For the purpose of this experiment, the dataset contained in the text file test_FD001.txt will be used.

When importing the dataset into an Excel spreadsheet, the variables are identified and appropriately named. The variables can be split into 3 different categories; identifier variables, control variables and system response variables. The variables are shown in Table 2. The data is then ordered by engine, copying each engine's data into a separate spreadsheet. The experiment will focus on the 11 dynamic system response variables, and all control variables and static system response variables are excluded from analysis and modelling.

Variable	Unit
Engine number	
Cycle number	
Variable bleed valve deviation	
Variable stator vanes deviation	
Fuel flow	
Total temperature at fan inlet	°R
Total temperature at LPC outlet	°R
Total temperature at HPC outlet	°R
Total temperature at LPT outlet	°R
Pressure at fan inlet	psia
Total pressure in bypass-duct	psia
Total pressure at HPC outlet	psia
Physical fan speed	rpm
Physical core speed	rpm
Engine pressure ratio (P50/P2)	
Static pressure at HPC outlet	psia
Ratio of fuel flow to Ps30	pps/psi
Corrected fan speed	rpm
Corrected core speed	rpm
Bypass Ratio	
Burner fuel-air ratio	
Bleed Enthalpy	

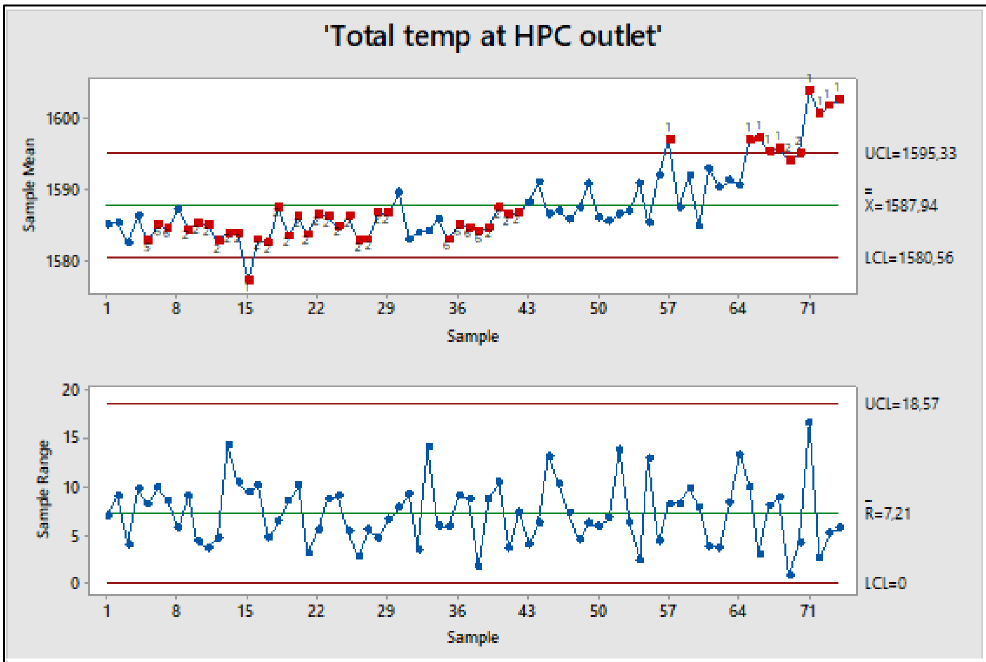
Table 2 C-MAPSS variable list.

The per-engine data is then imported into Minitab for analysis. Testing 60 and 90 samples with subgroup sizes 3 and 5, calculations can be made to determine a good combination for producing the control charts (Wheeler, 1995). Estimating degrees of freedom by using the formula $df = 0{,}9k(n-1)$, where df is degrees of freedom, k number of subgroups and n is subgroup size. Coefficient of variance (COV) or $\frac{sigma}{mean}$ is calculated using the formula $COV = \frac{1}{\sqrt{(2df)}}$. Using 90 samples in 3*30 subgroups, the degrees of freedom is 54 and the COV is about 9,6%. This is deemed as satisfactory for the control charts as increasing df further will only slightly lower COV. Subgroup size 3 is preferred as opposed to 5 based on personal preference. The control charts are constructed using X-Bar and R charts. The ruleset used is shown in Table 3.

Ruleset
1 point > 3 sigma from center line
9 points on the same side of the center line
6 points in a row, all decreasing or increasing
14 points in a row, alternating up and down
•2 out of 3 points > 2 sigma from the center line (same side)
•4 out of 5 points > 1 sigma from the center line (same side)
•15 points in a row within 1 sigma of the center line (either side)
•8 points in a row > 1 sigma from the center line (either side)

Table 3 Ruleset for testing X-Bar and R charts.

For the experiment, 20 engines are included and all 11 dynamic variables in each engine are analyzed. To get an overview of how the different variables look and behave over time, charts are made for each variable for the entire runtime for all 20 engines. An example of this is shown in Figure 13. These charts generally indicate a more stable period, a period of degradation and a point of failure.



Test Results for Xbar Chart of Total temp at HPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 15; 57; 65; 66; 67; 68; 71; 72; 73; 74
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; 25; 26; 27; 28; 29; 39; 40; 41; 42; 69; 70; 71; 72; 73; 74
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL).
Test Failed at points: 5; 16; 17; 66; 67; 68; 69; 70; 71; 72; 73; 74
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 5; 6; 7; 9; 10; 11; 12; 13; 14; 15; 16; 17; 19; 35; 36; 37; 38; 39; 65; 66; 67; 68; 69; 70; 71; 72; 73; 74
TEST 8. 8 points in a row more than 1 standard deviation from center line (above and below CL).
Test Failed at points: 16; 17; 70; 71; 72; 73; 74
* WARNING * If graph is updated with new data, the results above may no longer be correct.

Figure 13 Example of X-Bar and R charts of engine variable full run-to-failure.

To be able to analyze and test how the variables perform over time, 90 samples with subgroup size 3 are used for every 10 cycles in a time-series until failure. This provides a better view of when the process variable is under statistical control or not and increases the probability of understanding the timeframes of variable degradation until failure. These control charts are then constructed for every variable and for all 20 engines. In addition, complete test results are provided for every chart. An example of process variable control charts in the stable part and before point of failure is shown in Figure 14. An example of test results for the variable is shown in Table 4.

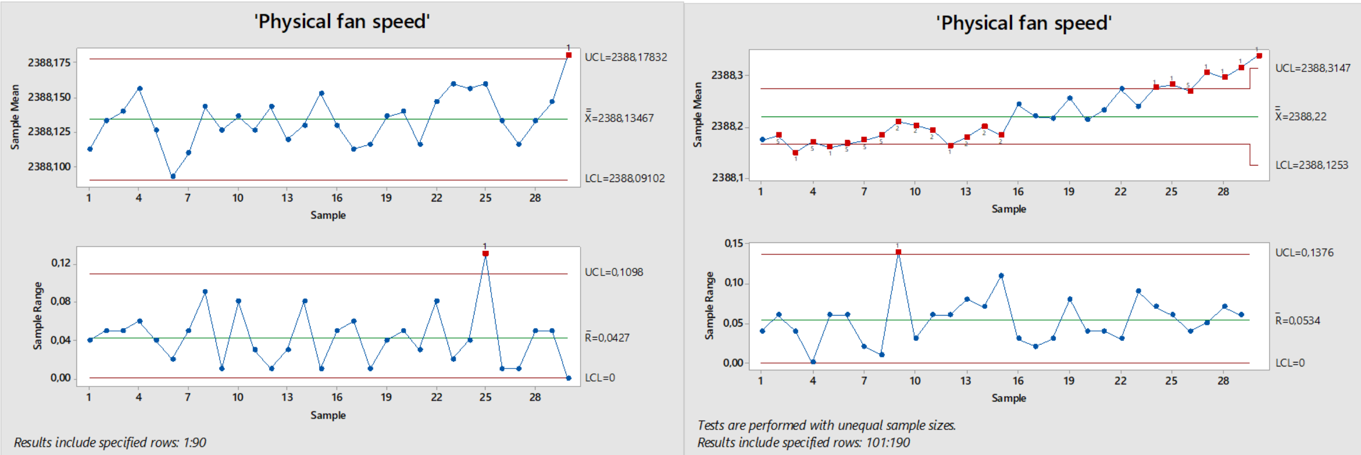


Figure 14 SPC time-series of engine variable in stable phase and close to failure.

Xbar-R Chart of Physical fan speed	Xbar-R Chart of Physical fan speed
Test Results for Xbar Chart of Physical fan speed TEST 1. One point more than 3,00 standard deviations from center line. Test Failed at points: 30 **Test Results for R Chart of Physical fan speed** TEST 1. One point more than 3,00 standard deviations from center line. Test Failed at points: 25 Results include specified rows: 1:90 98 rows are excluded. * WARNING * If graph is updated with new data, the results above may no longer be correct.	**Test Results for Xbar Chart of Physical fan speed** TEST 1. One point more than 3,00 standard deviations from center line. Test Failed at points: 3; 5; 12; 24; 25; 27; 28; 29; 30 TEST 2. 9 points in a row on same side of center line. Test Failed at points: 9; 10; 11; 12; 13; 14; 15; 29; 30 TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL). Test Failed at points: 2; 3; 4; 5; 6; 7; 8; 13; 15; 24; 25; 26; 27; 28; 29; 30 TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL). Test Failed at points: 4; 5; 6; 7; 8; 14; 15; 25; 26; 27; 28; 29; 30 TEST 8. 8 points in a row more than 1 standard deviation from center line (above and below CL). Test Failed at points: 8; 29; 30 **Test Results for R Chart of Physical fan speed** TEST 1. One point more than 3,00 standard deviations from center line. Test Failed at points: 9 Results include specified rows: 101:190 100 rows are excluded. * NOTE * 2 empty rows ignored. * WARNING * If graph is updated with new data, the results above may no longer be correct.

Table 4 Test results from SPC time series of engine variable in stable phase and close to failure.

The number of test failures will generally increase over time as the process variable moves closer to the point of failure. In order to use this information to create an indicator for engine condition, some assumptions have been made. The first assumption is that a correlation exists between engine condition and SPC test results. The second assumption is that it is feasible to construct an algorithm that transforms SPC test data into a meaningful engine condition indicator value.

After testing different algorithms, the following formula is used to create a condition indicator value: $CI = 100 - (\sum_{n=1}^{11} 9 - \frac{1}{2}(VTF_n))$, where CI is the engine condition indicator value, VTF is the number of test failures for the variable and n is the numbered variable calculated. This provides an indicator value starting from 100 and subtracts half a point per test failure per variable up to 18 failures or 9 points per variable, meaning a minimum indicator value of 1. The variables are weighted equally, and scoring is equal for all variables and for failures in both X-

Bar and R charts. This translates to an engine with process variables with a low amount of test failures receiving higher indicator scores, and as the sum of test failures rises, indicator values will be lower. An example of engine condition index scoring is shown in Table 5 and graphically represented in Figure 15. This algorithm is then used on all 20 engines to construct the model.

Samples	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	3	0	4	1	8	1	1	3	3	2	87
11-100	1	1	0	2	0	18	2	2	12	0	0	81
21-110	1	0	0	4	0	18	0	3	18	0	0	78
31-120	1	11	0	2	0	18	1	2	18	1	3	71,5
41-130	0	1	2	2	0	18	1	3	18	0	0	77,5
51-140	3	2	1	3	0	18	0	0	18	1	1	76,5
61-150	12	7	3	1	0	18	4	2	18	1	0	67
71-160	2	2	7	2	0	18	8	1	18	6	0	68
81-170	2	5	11	6	0	18	7	0	18	8	8	58,5
91-180	12	16	18	6	6	18	18	5	18	15	18	25
101-190	16	18	18	18	15	18	18	18	18	18	18	3,5
111.200	18	18	18	18	18	18	18	18	18	18	18	1

Table 5 Engine Health Indicator scoring chart.

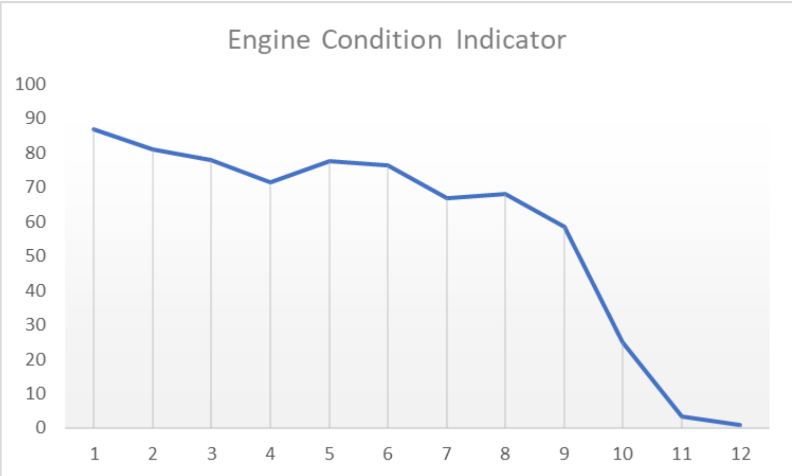


Figure 15 Engine Health Indicator Graph.

The simulated engines have vastly differing runtimes, and to be able to calculate means, distributions and sigma, the engines are compared from the point of failure and -10(n) cycles. Indicator values from all 20 engines are then tested using a Kolmogorov-Smirnov test for normal distribution. The test is performed as the number of engines included in the model is low enough so that the principles of the central limit theorem may not apply, and therefore testing for normal distribution are is reasonable. The formula for the Kolmogorov-Smirnov test can be defined as $D = \max_x |F^*(X) - S_N(X)|$, where $S_N(X)$ is the sample cumulative distribution function and $F^*(X)$ is the cumulative normal distribution function. The value of D is then compared to a D-table of critical values, where the H0 hypothesis claims that the distribution is normal, and a Ha claiming the distribution is not normal (Lilliefors, 1967). A D value exceeding the critical value in the table will lead to H0 being rejected and the claim that the distribution is not normal.

Plotting the indicator values for all 20 engines into a chart is useful to observe behavior and distribution. This is shown in Figure 16. For the purpose of the Kolmogorov-Smirnov test, H0 is that the data follow a normal distribution and Ha is that the data do not follow a normal distribution. Using the formula $D = \max_{1<i<N}(F(Y_i) - \frac{i-1}{N}, \frac{i}{N} - F(Y_i))$, where F is the cumulative distribution of the distribution tested and with α=0,05, we can test all 20 engines. The point of failure is expected to fail the test, but most other points is expected to pass. The test is performed using Minitab.

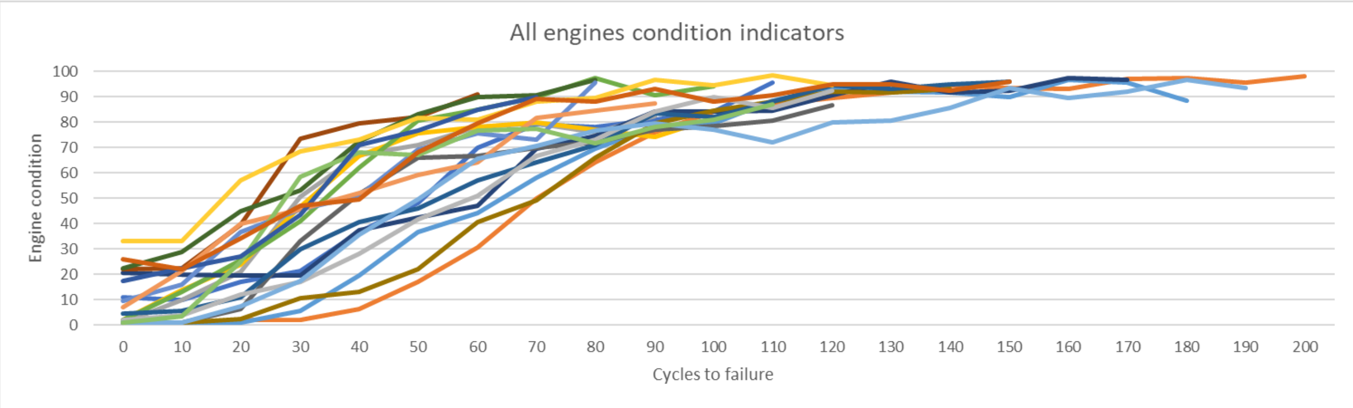


Figure 16 Condition indicators for all 20 engines from point of failure and -10(n) cycles.

The point of failure does indeed fail the test and is not normal distributed. Points 2-13 and 15-16 passes the test and can be assumed normal distributed. The remaining points fail the test, most probably caused by few remaining engines. This means that when means and standard deviation is calculated, there is a strong assumption that for the points that passed the Kolmogorov-Smirnov test, point of failure -10 cycles to -120 cycles and -140 cycles to -150 cycles, +/- 1,2 and 3 sigma and so on can be used to ascertain the probability of all tested engines health indicator values in the future with probabilities 68,3%, 95,5% and 99,7% and so on.

Plotting the mean and sigma values into a chart indicates probabilities of engine health indicators in that range for all points passing the Kolmogorov-Smirnov. This plot is shown in Figure 17. Using 3 sigma as an example cutoff point at an indicator value of 43,5 at point 2 or failure minus 10 cycles, $P(X > 43,5) = 1 - P(X < 43,5)$ when $\mu = 12,58$ and $\sigma = 10,29$ is 0,0014, and about 1 out of 1000 engines will be 10 cycles from failure and average RUL will be between 30 and 40 cycles. Using an indicator value of 45 as a cutoff, the P-value is 0,08%, and at 50 the

P value is 0,014% or about 1 out of 10.000 engines. Mean RUL at an indicator value of 50 is just over 40 cycles. For all points that have passed the Kolmogorov-Smirnov test for normal distribution, probabilities can be calculated for any engine falling inside or outside of the distribution, based on sigma value, and in effect calculate RUL based on observed current engine condition index score.

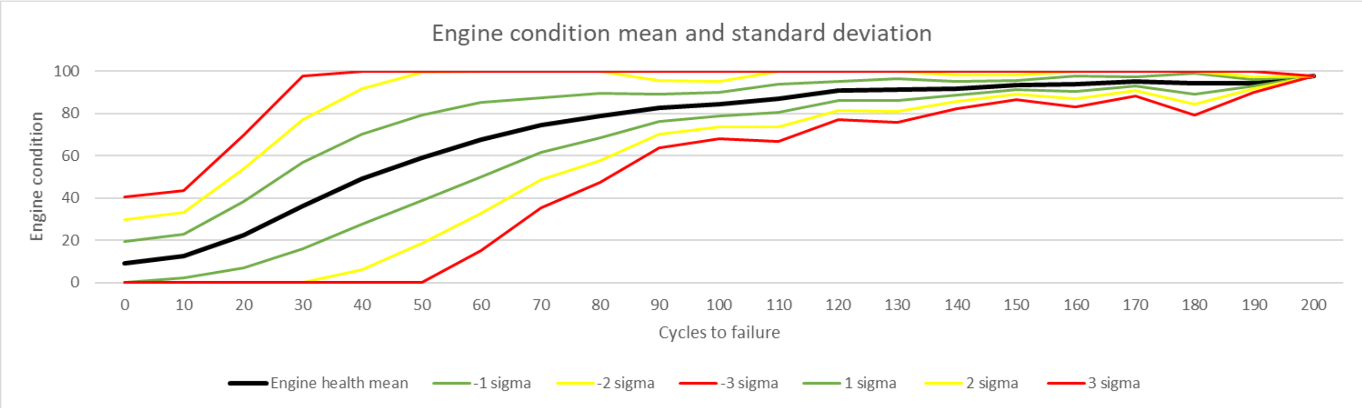


Figure 17 Engine condition means and +/- 1,2 and 3 sigma ranges.

Testing the model, an engine was randomly selected from the dataset and a condition index was constructed for every point in the time series. This data is then put into the model. This is visualized in Figure 18. Using a condition index value of 40 as a cutoff point, this engine will be stopped for maintenance 30 cycles before failure. Using 45 as the cutoff, the engine will be stopped for maintenance 40 cycles before failure, and at 50 as the cutoff, 50 cycles before failure.

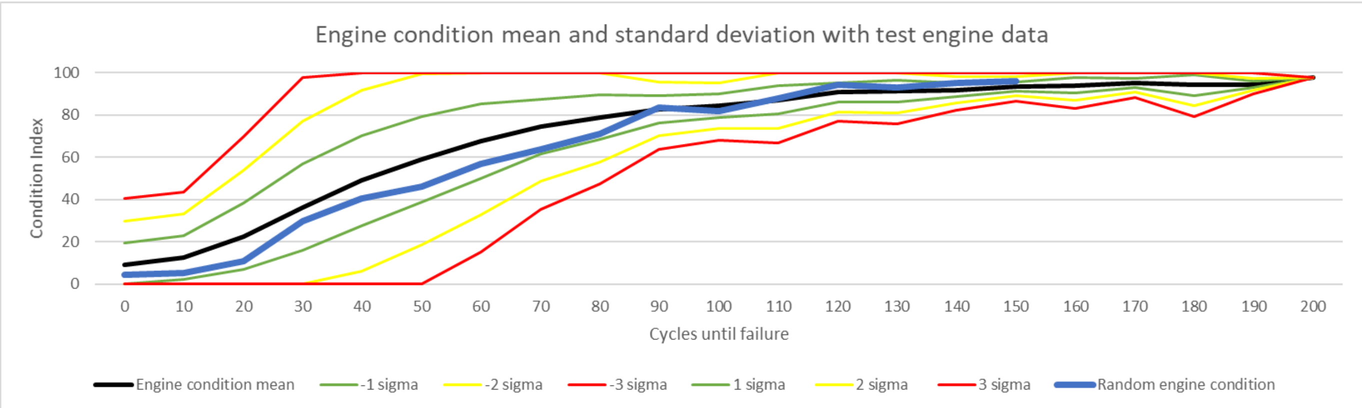


Figure 18 Engine condition distribution with test engine example graphed.

5. Discussion

Generally, the control chart time-series show a more stable period with no or very few test failures, evolving towards a period of increasing test failures until the engine fails. This is generally the case for all variables in all 20 engines. The assumption that there is a correlation between engine condition and the control chart test failures is strengthened. By using the transformation algorithm, the new engine condition indicator shows values between 100 and 1, where start values are closer to 100 and values closer to or at the point of failure is closer to 1. This is true for all 20 engines. The Kolmogorov-Smirnov test shows that most indicator distribution points are normally distributed, and this allows for simple calculations on probabilities of less than 10 cycles remaining until failure, where the probability is dependent on what indicator value is set as a critical value. This suggests that by using SPC on dynamic engine variables in a time series, scoring the test failures per X-Bar and R chart, creating an algorithm for scoring each engine and including scores for every time-series for all 20 engines into a model, there is a strong assumption that the individual condition scores are linked to engine condition and a high probability that the model can predict if any engine at a given condition index score has less than or more than X cycles of run-time remaining until the point of failure.

5.1 Interpretation

Including the dataset from the C-MAPSS simulator gives the experiment access to complete run-to-failure data. This was done for several reasons. The access to industry-related run-to-failure datasets is in most cases difficult or impossible to acquire (Goode, Moore and Roylance, 2000, Saxena *et al.*, 2008). The access to run-to-failure data from a multitude of machines, in this case jet engines, would provide data useful for building the experimental prediction model that available industrial datasets could not. The assumption was that building a model predicting failure using statistical methods and SPC control charts should be based on historical failure data, and that access to such data would increase the precision of failure estimation. In addition, there was an assumption that it would be possible to generalize from the model towards industrial machines and processes. The dataset is also widely used as a benchmark in RUL research (Zheng *et al.*, 2017).

The decision to exclude control variables is based on the view that they are external to the machine and do not describe what is happening inside the engine, but rather variables that can influence engine behavior from the outside. The SPC control charts used in this experiment should represent internal machine variables, not external. Exclusion of the static variables was done because they do not in this case provide any useful data.

The reasoning for weighting the variables equally is based on lack of expertise and knowledge on jet engine physics and mechanics. There was also an assumption that this could cause the model to be more specific towards jet engines and less general in nature. A decision was also made to score all test failures from the WE ruleset equally, even though test failures in test 4,5 and 6 signify shifts or trends and not out of control processes. This was done because the goal of the control charts was not to indicate a process in or out of statistical control, but to provide information for the purpose of constructing a condition indicator.

Constructing control charts for the entire runtime of each variable in all 20 engines generally indicated a stable period, and then a sudden shift up or down in value until the point of failure. This indicates that a correlation between control chart test failures and engine condition is a plausible assumption. This assumption is strengthened when looking at the time-series control charts. The sum of all variable control charts for any of the 20 engines included show a stable period with no or very few test failures, until a point where test failures occur increasingly until engine failure. This happens in every engine and is generally true for individual variables as well.

Because the jet engine dataset simulates engines with differing amounts of wear, their runtimes vary. Because of this, looking at the indicator scores across all engines from the point of failure and back in time, a decreasing number of engines will be included. However, the most critical information is believed to be gathered from the period of indicator value decline until the point of failure, and the Kolmogorov-Smirnov tests lead to not being able to reject the hypothesis that the distributions are normal distributed. On this basis, estimating probabilities of remaining useful life based on indicator value is believed to be statistically sound. Using sigma values in a normal distribution, the model can predict the probability of more than or less than X cycles remaining until failure. This indicates that RUL can be probabilistically estimated by the model.

5.2 Implications

Continuous improvement is a central term within most philosophies in quality engineering. Deming was a proponent in increasing focus on processes and less on the products themselves. Focusing too much on the product as a measure of quality can lead to improvements until the product is inside of specifications, and it can be argued that this is directly contradictory to the idea of continuous improvement. Taguchi also shares this view of process importance. Focusing on the process today includes focusing on process variables. Even if the vast amount of data obtained from process sensors today is under-utilized, there is an increasing number of proponents for using SPC as a tool for monitoring these variables (Scherkenbach, 1986, Wood, 1994, Nomikos and MacGregor, 1995, Fugate, Sohn and Farrar, 2001).

A product can be within set specifications, when the process is out of statistical control. Using SPC on the product will most likely not provide optimal insight into if there are problems with a process and how optimized the process is. Using control charts on process variables provides superior insight into process behavior and increases the ability to detect faults, shifts and trends in the process. This can aid process optimization and root cause detection.

Working towards Industry 4.0, industrial systems will increase in complexity. Information will be sent from machine to machine, from machine to humans and across the value chain, both vertically and horizontally. Some of this information will be sensor data. Assuming the usefulness of using SPC as a tool to monitor sensor data, control charts can be constructed both on-line, off-line and if needed in real-time. This provides the ability to monitor individual process variables, which in turn provides insight into process stability over time. This can be a tool for building process knowledge and understanding. The data produced from SPC analysis can be useful for a multitude of purposes; simulations, digital twins, process condition monitoring and RUL estimation. The ability to monitor processes in real-time and provide up-to-date feedback on process condition can lead to new insights that help further optimization and improvement work.

Assuming increasingly complex industrial systems going towards I4.0, making informed decisions, optimizing strategies and planning across the value chain can be difficult. Using SPC as a tool and having real-time access to control chart data can help inform these activities. Knowledge on the current condition of all connected and monitored processes can greatly help with making more optimal decisions, and the ability to predict machine RUL not only optimizes maintenance planning but can be a tool for other planning activities and overall strategy.

The advent of Big Data has seen a substantial increase in process data. Many, if not most, businesses only use a fraction of this data to optimize their processes. Tools used to analyze this data must be able to provide meaningful information that businesses can utilize. SPC can be one of these tools. Using control charts on the datasets can help increase the use of Big Data as a resource in optimizing processes.

The two main components of Cyber Physical Systems, acquiring information from the physical world and providing valuable feedback from the cyber to the physical, assumes the existence of a toolset for managing and analyzing data to provide optimal information. Use of existing tools can in many cases be extended, and assuming the usefulness of SPC in such cyber-physical environments, the ability to provide process information both on-line, off-line and real-time can be beneficial in CPS. Historical analysis of process variables in and out of statistical control, control chart time-series to give insight into how the process variables develop over time, monitoring current, historical and time-series machine condition and the ability to estimate remaining useful life for all monitored processes should help increase the value of information from the cyber realm.

Looking at different architectures for I4.0, specifically 5C, IIRA and RAMI 4.0, SPC can be useful for a multitude of activities. In the 2nd layer of 5C, SPC can be used for monitoring machine health, degradation and predicted performance. In the 3rd layer, SPC can be helpful when building digital twin model and for identifying and monitoring process variation. From an IIRA and RAMI 4.0 perspective, utilizing process control can improve control in that it can provide accurate, up-to-date and if needed real-time information on all monitored processes. Knowledge gained from control chart analysis can improve information in the system, which can help system resilience and reliability. Increasing the quality and detail of the information sent between the different layers of these architectures can provide additional insights that can be useful for optimization and continuous improvement.

The field of artificial intelligence and machine learning is complex. Experiments using ML for specific tasks has yielded impressive results but understanding of why these systems are so good at some tasks is generally limited. Most research being done on estimating RUL is done using ML methods, and much of this research has provided valuable insights into RUL estimation. However, when most ML software uses black boxes for their algorithms and the understanding of why a ML system produces the outputs it does in most cases is limited, the estimation does come with some challenges. Utilizing information systems and tools for analysis that are mostly or completely transparent is beneficial for several reasons. A transparent system can help build understanding and knowledge. Understanding how the tool works can help optimize not only the processes but also the tool itself. The tool can be tweaked to better fit specific processes to provide more precise estimations. There is a reluctance in using AI and ML for decision-making in critical processes today. Perhaps utilizing a manual, transparent tool based on statistical methods can be a useful addition to RUL estimation based on ML?

Both the IFDAPS and the IFaCOM ZDM frameworks describe the usefulness of evaluating equipment and processes to be able to identify conditions and faults. In addition, there are benefits in acquiring the ability to estimate RUL and predict future faults to optimize planning and operation. Using SPC as a tool for monitoring process variables can be useful for fault diagnosis and prognosis. SPC can also be helpful for predicting future production system behavior.

The ability to precisely predict process failures can be greatly beneficial in reducing downtimes and minimizing losses by avoiding unnecessary maintenance and maximizing uptime. But when probabilistic predictions and estimates fail, tools should be employed to diagnose failure causes. Using process control as a tool for monitoring process variables can increase the probability of identifying root causes to failures and help optimize process variables so that failures occur less frequently. Assuming the usefulness of the model, SPC can additionally provide an alternative for monitoring process condition and predict remaining useful life for processes.

Estimating RUL relies on historical failure data and the availability of such data. ML methods for RUL estimation can be very data intensive, and often require a large percentage of large datasets to train the algorithms. Examples of neural networks using 70-80% of the dataset for training purposes is not uncommon. An assumption is that the usefulness of ML RUL estimation relies on the availability of run-to-failure datasets that are large enough to allow for extensive training and then testing. The SPC-based model presented in this thesis uses 20% of the dataset for building a statistically sound model, which is considerably less that using ML methods. In addition, future testing of the model does not require run-to-failure data but can predict based on real-time process condition index score.

Process degradation and failure can be caused by multiple combinations of variables, and this can increase the difficulty of detecting and identifying root causes to problems or failures. In 99 percent of all failure cases, failures will be preceded by signs or indications that failures will occur (Lee *et al.*, 2014). Using SPC to monitor critical process variables both real-time, on-line and off-line increases available information on the state of the process and process variables. Combining the ability to efficiently monitor the process and all monitored variables with using SPC to estimate current process condition and RUL, shows that SPC can probably be a valuable tool for condition-based maintenance work and RUL estimation.

5.3 Limitations

The thesis uses a dataset from the NASA C-MAPSS simulator. This dataset is based on jet engine simulation and is not related to industrial process data. Although this dataset is widely used for RUL estimation purposes, the external validity of the model may be limited when applied to industrial processes. The lack of available run-to-failure data from industry meant that using the C-MAPSS dataset was the only option to allow for model construction. However, this model is not constructed specifically for jet engines, and the same concepts that allow for model construction based on engine variable data are assumed to apply to industrial machines as well. This assumption must be tested.

Constructing an experimental model for estimating RUL based on SPC, is a mostly manual method. Using SPC control charts for constructing engine condition indicators and estimating RUL meant going beyond state of the art for SPC, and this required extensive testing. In the 20-week span of this thesis, around 900-1000 hours was spent testing different sample sizes, subgroup sizes, creating full run-to-failure X-Bar and R charts for 11 variables in 20 engines, creating control charts in time series for all variables in all engines and producing test results from all variables and engines both for the complete run-to-failure charts and the time series. In addition, control charts were produced for both sample size 60 and 90 for around half of the engines. This produced around 5500-6000 X-Bar and R charts with full test results for all charts. The time spent constructing the model has been a limiting factor in the rest of the thesis and may have contributed to mistakes and errors. It may also have limited the depth of the literature review. Given the results of the model produced, showing a transparent, statistically driven manual method for calculating machine condition and estimating RUL, this is believed to be a reasonable trade-off, as most research into RUL estimation today is based on ML methods and not based on manual statistical analysis.

The model constructed is relatively simplistic in methods applied. The goal was to be able to test the viability of using SPC and control charts on process variables to construct condition indicators and to estimate RUL. When building a proof-of-concept model, using relatively simple statistical methods is believed to be a reasonable approach, also considering the timeframe. However, this implies that there is a potential for more extensive and advanced statistical method to be applied to the model, and this could help improve model precision and general usefulness.

6. Summary and conclusion

The main question this thesis aimed to answer was the following; Can Statistical Process Control be extended to function as a tool in Condition Based Maintenance for determining machine condition and Remaining Useful Life estimation, and how does this influence the usefulness of SPC in Industry 4.0?

6.1 Can SPC be extended to function as a tool in Condition Based Maintenance for determining machine condition and Remaining Useful Life estimation?

By using X-bar and R charts to monitor all dynamic variables across all 20 engines included in the model through a time-series, using 90 samples in 10 cycle increments, the results from applying the complete WE ruleset indicate a stable period with no or very few test failures. During the time-series analysis, at a certain point the number of test failures will start to increase, and this acceleration of test failures will generally happen in most or all variables in all engines until the engines fail. By using an algorithm, the number of test failures in each variable at each point in the time-series is counted and subtracted from a theoretical starting condition indicator score of 100. This number is divided by the number of variables and rounded to the closest whole number. The algorithm subtracts half a point for every test failure up to 18 per time per variable. This translates to a minimum indicator score of one. The results show that applying this method gives the engines a starting value closer to 100. This will vary somewhat but is expected due to the simulation of different levels of wear in the different engines. This number will stay somewhat stable or decline slowly. At a certain point, the scores will decrease more rapidly and tend to be close to 1 at the point of failure.

Using the Kolmogorov-Smirnov test for normal distribution reveals that most condition indicator mean points must be assumed to be normal distributed. By calculating sigma, it is elementary to calculate the probability of an engine with a specific indicator value being inside or outside of the distribution, thereby calculating remaining useful life of that engine probabilistically.

Based on these findings it is probable that SPC can be used as an alternative to ML methods to determine machine condition and estimate RUL. Assuming this is the case, this demonstrates the usefulness of monitoring process variables as part of SPC, and this allows us to answer the second part of the research question; how does this influence the usefulness of SPC in Industry 4.0?

6.2 How does this influence the usefulness of SPC in Industry 4.0?

Theory within the field of quality engineering underlines the importance of continuous improvement. Both Deming and Taguchi states that a focus on products can be contradictory to the philosophy of never-ending work towards improving processes. Assuming the importance of focusing on the processes and their variables, SPC can be a valuable tool for monitoring processes, detecting problems and identifying root causes. The complexity of industrial systems going towards Industry 4.0 is likely to increase and this includes information. Using SPC as a tool can help create more detailed insight into processes and individual process variables. In today's industrial environment control chart time-series can be constructed on-line and real-time, and the ability to monitor the processes and process variables using control charts, inform on process condition and remaining useful life can greatly improve decision-making, strategies and planning activities.

Meaningful utilization of most sensor data, ideally all, can help build knowledge and understanding on processes, how to improve them and how to optimize them. The increasing amount of sensor data collected today is greatly under-utilized, and Big Data is often used as a tool for storing data, not for analyzing. Tools are needed for improving the use of sensor data for analytic purposes, and SPC can be one such tool. Increasing utilization of sensor data can help improve the quality of information in Cyber Physical Systems, where feedback and analytic results from the cyber realm can inform decisions for all stakeholders in the physical world. Improving the quality of information is an important part of CPS and the resilience and reliability of the systems, and SPC can be an important part of that improvement work. Access to detailed and precise information and analysis on processes and process variables can improve the ability to construct digital twins and conducting simulated experiments. CPS ZDM frameworks like IFDAPS and IFaCOM highlight the importance of evaluating equipment and processes, identifying conditions, faults and fault diagnosis, prognosis and prediction. SPC can be helpful as a tool for all these purposes.

Research on AI and machine learning is very much in fashion. Reading articles on determining machine condition and estimating remaining useful life using ML and neural networks reveals that almost all focus is on this field. Very few articles focus on extending and developing manual, statistical tools for handling the increasing complexities of CPS and Industry 4.0. Although results from ML systems can be promising regarding RUL estimation, the lack of understanding as to exactly why ML is good at predicting can be problematic. Black boxes in these ML systems also prevents effective learning and changing settings or attempting to improve the ML system can be challenging or impossible. Having a manual, statistically driven method for determining current machine condition and predicting useful life might be a useful alternative to ML methods, and the transparency of a manual method may help increase knowledge on the processes and optimization. Transparency also makes it possible to improve on the method itself, by changing or adding statistical methods, improving scoring and more detailed and specific weighting of the different process variables. ML RUL estimation can also be very data intensive, whereas a manual method requires far less data to allow for model construction. This makes it less difficult to obtain real world failure data compared to ML estimation models.

Given the ability to monitor processes and process variables, and assuming the ability to determine process condition and RUL, it seems very likely that SPC will be increasingly important as a tool in Industry 4.0.

6.2 Recommendations

The results, discussion, limitations and conclusion of this thesis leads to several suggestions concerning further work. The model was constructed within a limited timeframe and uses a set of basic statistical methods to produce a proof-of-concept model. Developing a quantitative statistical model for determining process condition and estimating remaining useful life is a very useful alternative to solely focusing on ML methods. The model constructed in this thesis needs to be extensively tested for real-world industrial process viability. The methods applied in the model construction can be improved upon to make the model more precise in its estimation capabilities, specifically critical variable identification, process knowledge, variable weighting and scoring. Alternatively, the model constructed here can serve as inspiration for constructing similar statistical tools for determining process condition and RUL estimation based on different approaches. Much of this work will require industrial partners, academic learning factories or both. This can potentially be included in future project work related to I4.0 and can be both interesting and suitable for master level students, PhD. candidates, postdocs and others.

The C-MAPSS dataset utilized in this thesis is widely used by researchers in subjects concerning both ML and RUL estimation. There is a need for real-world industrial run-to-failure datasets. Creating precise methods for predicting RUL and many other types of statistical analysis is potentially of great value not only for academia, but for industrial companies as well. Prioritizing partnerships between academia and industry to allow for producing real-world run-to-failure datasets would be beneficial for all parties. Academic learning factories can also contribute towards constructing these datasets in their laboratories.

Further research is needed in determining the usefulness and value of monitoring process variables. Allowing SPC to analyze the state of processes by utilizing sensored variable data can potentially provide a wealth of information that can help improve process knowledge and ability to optimize them. Perhaps an increased academic focus on SPC state of the art can help motivate industry towards using more of the data that their process sensors provide, to better prepare them for I4.0 environments and to aid in continuous improvement work?

7. References

Acatech, F. (2013) Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0.

Adolphs, P. *et al.* (2015) Reference architecture model industrie 4.0 (rami4. 0), *ZVEI and VDI, Status Report*.

Ahmad, R. and Kamaruddin, S. (2012) An overview of time-based and condition-based maintenance in industrial application, *Computers & industrial engineering*, 63(1), pp. 135-149.

Arbnor, I. and Bjerke, B. (2009) *Methodology for creating business knowledge*. 3rd edn. London: Sage.

Bostrom, N. and Yudkowsky, E. (2014) The ethics of artificial intelligence, *The Cambridge handbook of artificial intelligence*, pp. 316-334.

Buchanan, B. G. (2005) A (very) brief history of artificial intelligence, *Ai Magazine*, 26(4), pp. 53.

Chappell, D. (2015) Introducing azure machine learning, *A guide for technical professionals, sponsored by microsoft corporation*.

DIN (2018) *German standardization roadmap - Industrie 4.0. Version 3*. German Institute for Standardization. Available at: https://www.din.de/blob/65354/57218767bd6da1927b181b9f2a0d5b39/roadmap-i4-0-e-data.pdf (Accessed: 15.05.2019).

Dunjko, V. and Briegel, H. J. (2018) Machine learning & artificial intelligence in the quantum domain: a review of recent progress, *Reports on Progress in Physics*, 81(7), pp. 074001.

Eleftheriadis, M. S. R. J. and Myklebust, O. (2016) A guideline of quality steps towards zero defect manufacturing in industry, *Proc. Int. Conf. Ind. Eng. Oper. Manag*. pp. 332-340.

EN13306 (2017) NS-EN 13306:2017 Vedlikehold Vedlikeholdsterminologi: Standard Norge. (Accessed: 18.5.2019).

EN17007 (2017) NS-EN 17007:2017 Vedlikeholdsprosess og tilhørende indikatorer: Standard Norge. (Accessed: 15.05.2019).

Fugate, M. L., Sohn, H. and Farrar, C. R. (2001) Vibration-based damage detection using statistical process control, *Mechanical systems and signal processing*, 15(4), pp. 707-721.

Golafshani, N. (2003) Understanding reliability and validity in qualitative research, *The qualitative report*, 8(4), pp. 597-606.

Goode, K., Moore, J. and Roylance, B. (2000) Plant machinery working life prediction method utilizing reliability and condition-monitoring data, *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 214(2), pp. 109-122.

Hermann, M., Pentek, T. and Otto, B. (2016) Design principles for industrie 4.0 scenarios, *2016 49th Hawaii international conference on system sciences (HICSS)*. IEEE, pp. 3928-3937.

Jardine, A. K., Lin, D. and Banjevic, D. (2006) A review on machinery diagnostics and prognostics implementing condition-based maintenance, *Mechanical systems and signal processing*, 20(7), pp. 1483-1510.

Kagermann, H., Lukas, W.-D. and Wahlster, W. (2011) Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution, *VDI nachrichten*, 13(1).

Kagermann, H., Wahlster, W. and Helbig, J. (2013) *Securing the future of German manufacturing industry*. (Recommendations for implementing the strategic initiative INDUSTRIE 4.0). acatech.de: Forschungsunion and acatech. Available at: http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_Website/Acatech/root/de/Material_fuer_Sonderseiten/Industrie_4.0/Final_report__Industrie_4.0_accessible.pdf.

Lasi, H. *et al.* (2014) Industry 4.0, *Business & Information Systems Engineering*, 6(4), pp. 239-242.

LeCun, Y., Bengio, Y. and Hinton, G. (2015) Deep learning, *nature*, 521(7553), pp. 436.

Lee, E. A. (2008) Cyber physical systems: Design challenges, *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. IEEE, pp. 363-369.

Lee, J. *et al.* (2014) Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications, *Mechanical systems and signal processing*, 42(1-2), pp. 314-334.

Lee, J., Bagheri, B. and Kao, H.-A. (2015) A cyber-physical systems architecture for industry 4.0-based manufacturing systems, *Manufacturing letters*, 3, pp. 18-23.

Leedy, P. D. and Ormrod, J. E. (2015) *Practical research*. 11th edn. Pearson Custom.

Liao, H., Zhao, W. and Guo, H. (2006) Predicting remaining useful life of an individual unit using proportional hazards model and logistic regression model, *RAMS'06. Annual Reliability and Maintainability Symposium, 2006.* IEEE, pp. 127-132.

Lilliefors, H. W. (1967) On the Kolmogorov-Smirnov test for normality with mean and variance unknown, *Journal of the American statistical Association*, 62(318), pp. 399-402.

Lin, S.-W. *et al.* (2017) Architecture Alignment and Interoperability: An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper *White Paper, Industrial Internet Consortium.*

Lofthouse, T. (1999) The Taguchi loss function, *Work Study*, 48(6), pp. 218-223.

Montgomery, D. C. (2007) *Introduction to statistical quality control*. John Wiley & Sons.

NASA (2008) Turbofan Engine Degradation Simulation Data Set, i NASA (ed.). NASA prognostic data repository: NASA. Available at: https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/ (Accessed: 20.01.2019).

Nomikos, P. and MacGregor, J. F. (1995) Multivariate SPC charts for monitoring batch processes, *Technometrics*, 37(1), pp. 41-59.

Rigamonti, M. M., Baraldi, P. and Zio, E. (2016) Echo state network for the remaining useful life prediction of a turbofan engine, *annual conference of the prognostics and health management society 2015*. pp. 255-270.

Roy, R. K. (2010) *A primer on the Taguchi method*. Society of Manufacturing Engineers.

Rüßmann, M. *et al.* (2015) Industry 4.0: The future of productivity and growth in manufacturing industries, *Boston Consulting Group*, 9.

Sadeghi, A.-R., Wachsmann, C. and Waidner, M. (2015) Security and privacy challenges in industrial internet of things, *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE, pp. 1-6.

Saxena, A. *et al.* (2008) Damage propagation modeling for aircraft engine run-to-failure simulation, *2008 international conference on prognostics and health management*. IEEE, pp. 1-9.

Scherkenbach, W. W. (1986) *The Deming route to quality and productivity: Road maps and roadblocks*. George Washington Univ Department of.

Schuh, G. *et al.* (2017) *Industrie 4.0 Maturity Index: Managing the Digital Transformation of Companies*. Utz, Herbert.

Si, X.-S. *et al.* (2011) Remaining useful life estimation–a review on the statistical data driven approaches, *European journal of operational research*, 213(1), pp. 1-14.

Taguchi, G. (1995) Quality engineering (Taguchi methods) for the development of electronic circuit technology, *IEEE Transactions on Reliability*, 44(2), pp. 225-229.

Wang, K.-S. (2013) Towards zero-defect manufacturing (ZDM)—a data mining approach, *Advances in Manufacturing*, 1(1), pp. 62-74.

Wang, K. and Wang, Y. (2012) Data mining for zero–defect manufacturing: Tpir Academic Press, London.

WEC (1956) *Statistical quality control handbook*. Western Electric Company.

Wheeler, D. J. (1995) *Advanced topics in statistical process control*. SPC press Knoxville, TN.

Wikipedia (2019) Black box. wikipedia.org. Available at: https://en.wikipedia.org/wiki/Black_box (Accessed: 08.04.2019).

Wong, K. V. and Hernandez, A. (2012) A review of additive manufacturing, *ISRN Mechanical Engineering*, 2012.

Wood, M. (1994) Statistical methods for monitoring service processes, *International Journal of Service Industry Management*, 5(4), pp. 53-68.

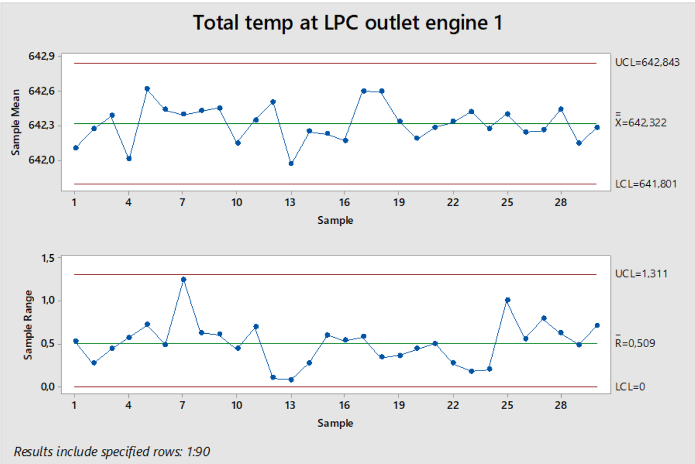
Zezulka, F. *et al.* (2016) Industry 4.0–An Introduction in the phenomenon, *IFAC-PapersOnLine*, 49(25), pp. 8-12.

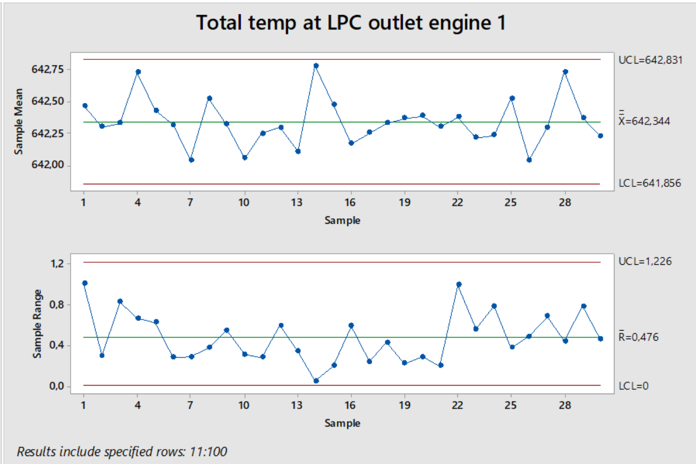
Zheng, S. *et al.* (2017) Long short-term memory network for remaining useful life estimation, *2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE, pp. 88-95.

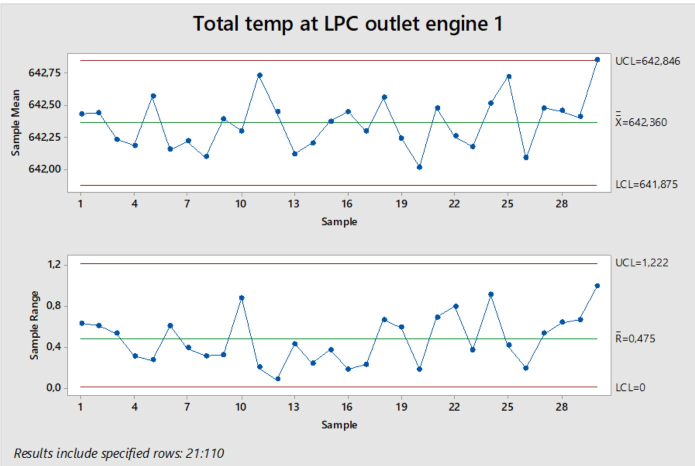
Zhou, K., Liu, T. and Zhou, L. (2015) Industry 4.0: Towards future industrial opportunities and challenges, *Fuzzy Systems and Knowledge Discovery (FSKD), 2015 12th International Conference on*. IEEE, pp. 2147-2152.

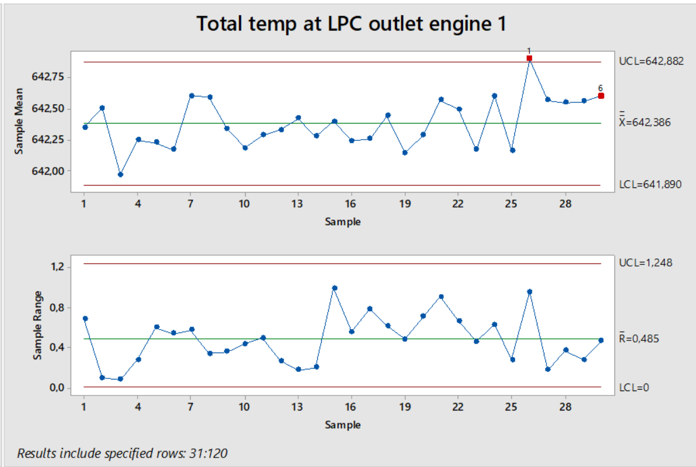
Appendices

Appendix I – Complete set of X-Bar and R control charts with test results from example variable in simulated engine 1.



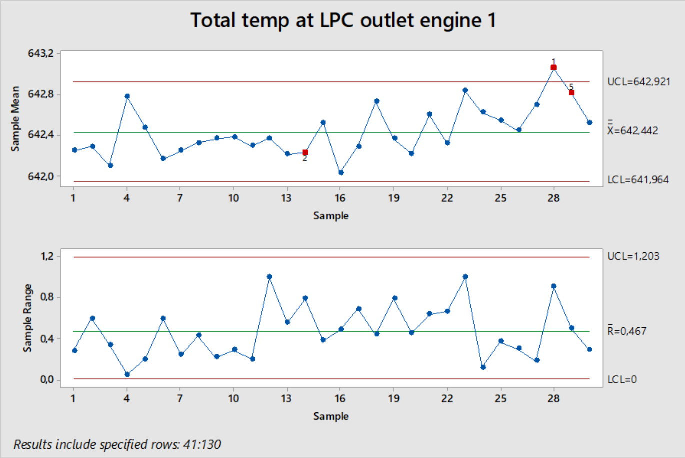






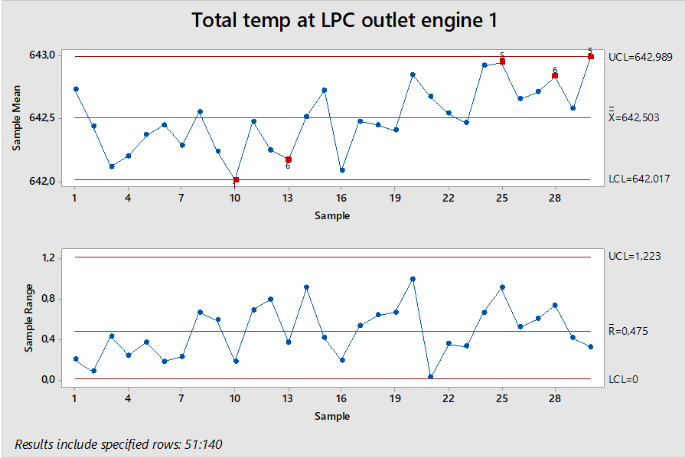
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 26
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 30
Results include specified rows: 31:120
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



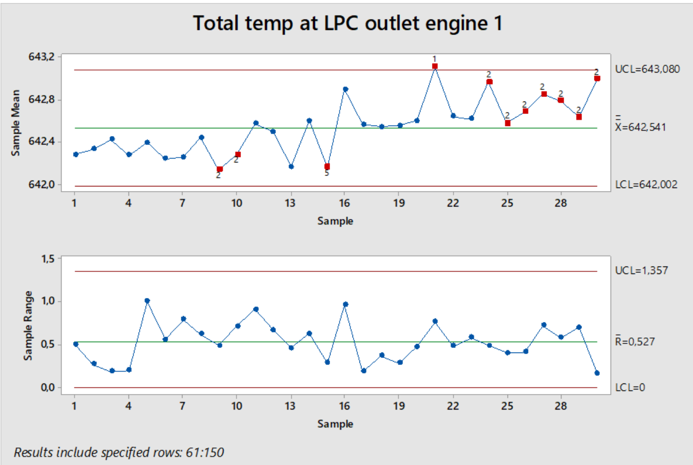
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 28
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 14
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL).
Test Failed at points: 29
Results include specified rows: 41:130
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



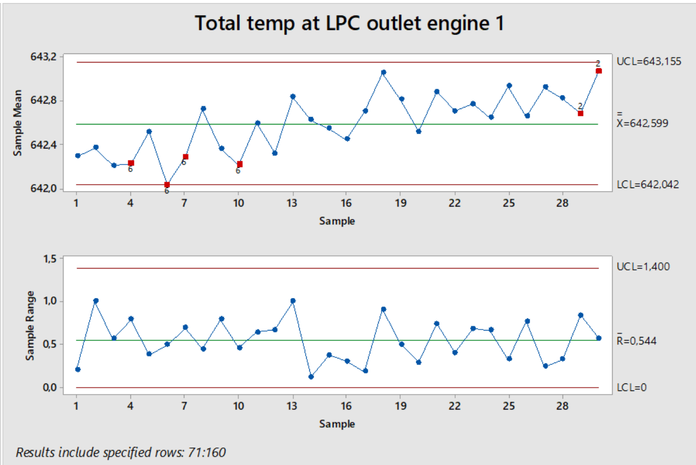
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 10
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL).
Test Failed at points: 25; 30
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 13; 28
Results include specified rows: 51:140
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



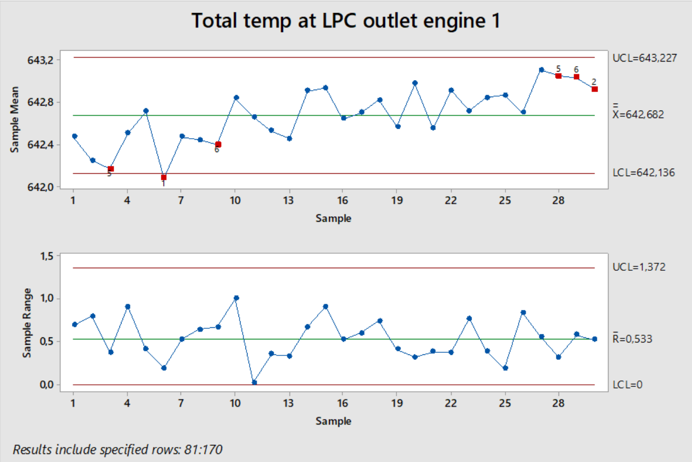
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 21
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 9; 10; 24; 25; 26; 27; 28; 29; 30
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL)
Test Failed at points: 15
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 10
Results include specified rows: 61:150
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



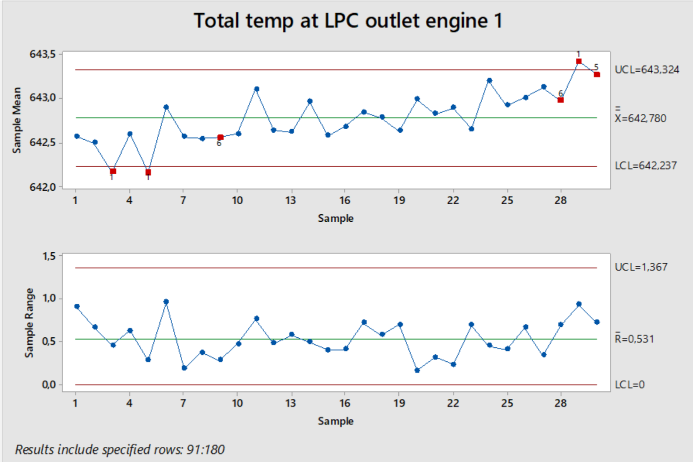
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 29; 30
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 4; 6; 7; 10
Results include specified rows: 71:160
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



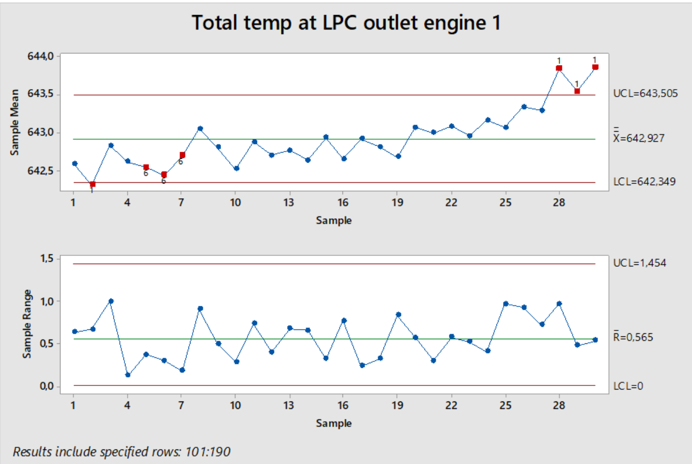
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 6
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 30
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL)
Test Failed at points: 3; 28
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 9; 29; 30
Results include specified rows: 81:170
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



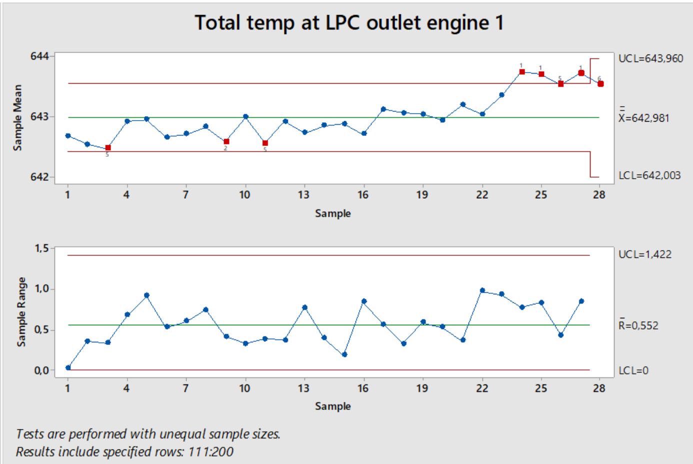
Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 3; 5; 29
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL)
Test Failed at points: 5; 30
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 5; 9; 28; 29; 30
Results include specified rows: 91:180
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 2; 28; 29; 30
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 28; 29; 30
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL)
Test Failed at points: 28; 29; 30
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 5; 6; 7; 28; 29; 30
Results include specified rows: 101:190
102 rows are excluded.
* WARNING * If graph is updated with new data, the results above may no longer be correct.



Test Results for Xbar Chart of Total temp at LPC outlet

TEST 1. One point more than 3,00 standard deviations from center line.
Test Failed at points: 24; 25; 27
TEST 2. 9 points in a row on same side of center line.
Test Failed at points: 9
TEST 5. 2 out of 3 points more than 2 standard deviations from center line (on one side of CL)
Test Failed at points: 3; 11; 25; 26; 27
TEST 6. 4 out of 5 points more than 1 standard deviation from center line (on one side of CL).
Test Failed at points: 25; 26; 27; 28
Results include specified rows: 111:200
110 rows are excluded.
* NOTE * 8 empty rows ignored.
* WARNING * If graph is updated with new data, the results above may no longer be correct.

Appendix II – Complete score chart and condition index score for all engines

Engine 1	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	4	1	1	0	0	1	1	1	0	95,5
11-100	0	0	10	11	2	0	2	3	0	0	4	84
21-110	0	0	10	9	2	1	11	3	0	0	2	81
31-120	2	0	8	2	10	4	6	8	1	0	3	78
41-130	3	0	5	3	8	2	9	8	0	0	4	79
51-140	5	0	7	13	8	0	13	8	2	2	2	70
61-150	12	1	4	12	16	1	18	18	11	4	7	48
71-160	6	7	14	18	18	1	18	18	14	6	7	36,5
81-170	7	6	18	18	18	3	18	18	18	18	15	21,5
91-180	10	9	18	18	18	3	18	18	18	18	18	17
101-190	16	15	18	18	18	5	18	18	18	18	18	10
111-200	13	14	18	18	18	7	18	18	18	18	18	11

Engine 2	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	1	0	1	0	1	1	0	0	0	0	0	98
11-100	0	6	2	0	1	0	0	0	0	0	0	95,5
21-110	1	0	1	0	1	1	0	0	0	1	0	97,5
31-120	1	0	1	0	0	0	0	0	4	0	0	97
41-130	0	4	2	0	1	2	3	0	0	1	1	93
51-140	0	0	0	1	0	7	4	0	0	0	1	93,5
61-150	1	0	1	1	0	2	5	0	0	1	0	94,5
71-160	1	2	1	1	0	3	4	1	1	2	1	91,5
81-170	0	1	0	2	0	9	6	3	0	0	0	89,5
91-180	1	1	0	5	1	4	6	2	6	0	0	87
101-190	3	1	2	3	2	9	0	7	1	2	2	84
111-200	4	1	3	0	6	13	5	9	1	1	5	76
121-210	5	2	2	12	5	15	12	8	5	0	6	64
131-220	10	2	8	12	5	18	9	8	10	0	18	50
141-230	18	7	10	9	12	18	18	13	16	0	18	30,5
151-240	18	9	17	15	18	18	16	18	16	7	14	17
161-250	18	18	18	18	18	18	18	17	18	14	12	6,5
171-260	16	18	18	18	18	18	18	18	18	18	18	2
181-270	16	18	18	18	18	18	18	18	18	18	18	2
191-280	18	18	18	18	18	18	18	18	18	18	18	1
200-290	18	18	18	18	18	18	18	18	18	18	18	1

Engine 3	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	4	0	0	18	2	0	11	0	0	82,5
11-100	2	2	4	0	2	18	1	0	18	1	0	76
21-110	0	0	3	0	0	18	2	0	18	0	0	79,5
31-120	2	0	0	0	1	18	2	1	18	2	0	78
41-130	0	1	4	2	5	18	1	0	18	7	2	71
51-140	0	0	4	3	4	18	4	0	18	5	10	67
61-150	5	2	9	8	2	18	11	2	18	7	17	50,5
71-160	13	14	18	18	8	18	18	0	18	15	18	21
81-170	18	18	18	18	15	18	18	3	18	18	18	10
91-180	18	18	18	18	18	18	18	18	18	18	18	1

Engine 4	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	1	0	1	1	14	0	0	15	2	3	81,5
11-100	12	1	1	0	0	18	2	0	18	0	0	74
21-110	1	0	1	6	0	18	0	0	18	1	1	77
31-120	0	1	0	2	1	18	0	0	18	0	0	80
41-130	3	1	0	0	0	18	2	0	18	2	0	78
51-140	3	2	2	2	0	18	2	0	18	2	0	75,5
61-150	1	4	0	13	1	18	5	0	18	2	5	66,5
71-160	8	2	12	13	2	18	18	0	18	8	8	46,5
81-170	10	15	18	18	5	18	18	1	18	18	13	24
91-180	18	14	18	18	8	18	18	6	18	18	18	14
101-190	18	18	18	18	18	18	18	16	18	18	18	2

Engine 5	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	14	1	1	1	0	0	0	0	4	2	0	88,5
11-100	3	2	1	0	3	0	0	0	0	0	0	95,5
21-110	3	0	0	1	1	2	0	0	0	0	0	96,5
31-120	7	0	1	5	0	2	2	1	1	1	0	90
41-130	2	0	1	3	0	6	3	0	2	0	0	91,5
51-140	0	0	2	0	1	3	1	1	6	0	2	92
61-150	0	1	1	0	0	4	5	2	1	0	0	93
71-160	0	0	1	0	0	4	10	1	5	1	1	88,5
81-170	1	4	1	2	6	10	5	0	9	1	2	79,5
91-180	0	4	0	4	0	12	7	1	16	0	0	78
101-190	1	1	4	4	3	18	6	2	18	4	0	69,5
111-200	4	0	5	18	4	18	7	2	18	4	4	58
121-210	7	9	9	18	6	18	18	6	18	1	2	44
131-220	8	4	14	18	5	18	18	7	18	11	6	36,5
141-230	12	10	18	18	8	18	18	16	18	13	12	19,5
151-240	18	18	18	18	9	18	18	18	18	18	18	5,5
161-250	18	18	18	18	18	18	18	18	18	18	18	1
171-260	18	18	18	18	18	18	18	18	18	18	18	1
181-270	18	18	18	18	18	18	18	18	18	18	18	1

Engine 6	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	1	2	2	2	4	0	0	0	0	1	94
11-100	3	1	4	1	1	1	3	0	1	3	1	90,5
21-110	0	3	0	0	0	2	0	0	0	0	0	97,5
31-120	1	3	2	2	7	3	0	0	0	2	0	90
41-130	2	0	4	2	9	2	4	3	2	1	1	85
51-140	2	0	0	5	12	1	6	8	1	1	3	80,5
61-150	2	1	6	13	12	1	12	6	18	3	2	62
71-160	9	2	14	12	18	2	18	18	14	4	7	41
81-170	9	2	18	18	18	7	18	18	16	7	18	25,5
91-180	18	6	18	18	18	10	18	18	18	14	18	13
101-190	18	18	18	18	18	16	18	18	18	18	18	2

Engine 7	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	3	0	2	0	0	1	0	1	0	96,5
11-100	1	0	1	0	0	1	0	2	0	0	0	97,5
21-110	1	0	1	0	0	1	0	0	4	3	5	92,5
31-120	0	0	0	0	3	0	4	5	0	5	0	91,5
41-130	0	0	0	0	0	1	2	2	1	2	0	96
51-140	0	0	2	1	0	1	4	1	5	5	0	90,5
61-150	0	0	0	3	3	4	13	1	3	2	2	84,5
71-160	2	0	2	4	6	2	12	2	2	0	0	84
81-170	0	1	5	2	7	1	8	3	4	1	0	84
91-180	2	1	5	11	13	3	10	4	0	1	1	74,5
101-190	5	1	1	16	17	1	13	3	0	1	3	69,5
111-200	4	7	17	17	18	1	18	7	2	4	11	47
121-210	7	18	11	18	14	0	18	17	0	2	10	42,5
131-220	3	16	18	18	11	0	18	18	1	9	13	37,5
141-230	15	18	18	18	18	1	18	18	1	18	18	19,5
151-240	18	17	18	18	18	0	18	18	0	18	18	19,5
161-250	18	15	18	18	18	0	18	18	1	18	18	20
171-260	18	14	18	18	18	0	18	18	1	18	18	20,5

Engine 8	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	1	0	2	0	8	1	2	1	0	1	2	91
11-100	1	3	1	0	10	0	3	14	0	4	0	82
21-110	2	0	2	2	18	2	3	8	3	1	0	79,5
31-120	2	0	8	1	18	0	11	3	1	6	3	73,5
41-130	7	4	15	13	18	3	18	17	0	18	8	39,5
51-140	18	4	18	18	18	3	18	18	4	18	18	22,5
61-150	18	9	18	18	18	0	18	18	3	18	18	22

Engine 9	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	4	1	3	5	1	2	10	0	1	86,5
11-100	1	1	3	0	0	13	2	1	18	0	0	80,5
21-110	0	1	1	0	1	18	4	0	18	0	0	78,5
31-120	1	0	2	1	1	18	3	2	18	0	0	77
41-130	6	2	2	1	0	18	6	2	18	1	0	72
51-140	7	8	0	0	0	18	7	1	18	0	1	70
61-150	10	6	2	0	0	18	3	7	18	1	2	66,5
71-160	8	3	3	1	1	18	4	1	18	5	6	66
81-170	9	1	11	1	0	18	9	3	18	10	17	51,5
91-180	10	5	18	6	1	18	18	8	18	14	18	33
101-190	18	18	18	18	10	18	18	18	18	15	18	6,5
111-200	18	18	18	18	18	18	18	18	18	18	18	1
121-210	18	18	18	18	18	18	18	18	18	18	18	1

Engine 10	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	1	1	1	1	0	2	1	4	0	3	0	93
11-100	0	1	1	0	0	5	0	5	2	2	1	91,5
21-110	2	1	1	4	0	2	1	2	0	1	2	92
31-120	1	1	0	2	2	3	6	3	4	0	2	88
41-130	2	1	2	2	1	11	2	4	5	1	0	84,5
51-140	9	0	1	5	3	11	1	1	3	2	5	79,5
61-150	10	1	1	10	3	18	6	1	9	2	7	66
71-160	9	0	10	11	3	18	12	2	17	6	14	49
81-170	14	1	9	15	10	18	18	5	18	8	3	40,5
91-180	18	3	18	14	16	18	18	6	18	17	10	22
101-190	18	4	18	18	13	18	18	14	18	18	17	13
111-200	17	6	18	18	12	18	18	18	18	18	18	10,5
121-210	18	15	18	18	18	18	18	18	18	18	18	2,5
131-220	18	18	18	18	18	18	18	18	18	18	18	1
141-230	18	18	18	18	18	18	18	18	18	18	18	1

Engine 11	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	2	0	0	0	1	0	1	3	1	0	96
11-100	0	6	0	0	0	2	0	0	0	1	1	95
21-110	0	1	0	1	1	1	0	1	7	2	0	93
31-120	1	2	0	0	1	4	0	0	3	0	0	94,5
41-130	4	1	1	1	0	8	0	0	2	0	7	88
51-140	1	0	4	2	2	10	1	0	3	0	13	82
61-150	0	0	3	0	2	13	4	1	8	0	2	83,5
71-160	1	0	4	8	3	18	1	0	18	0	5	71
81-170	3	0	6	9	5	18	4	0	18	1	8	64
91-180	0	0	7	14	8	18	12	1	18	2	6	57
101-190	2	2	11	18	7	18	13	3	18	8	8	46
111-200	12	8	12	18	2	18	15	4	18	6	6	40,5
121-210	12	18	18	18	2	18	18	3	18	4	11	30
131-220	18	18	18	18	12	18	18	9	18	13	18	11
141-230	18	18	18	18	12	18	18	15	18	18	18	5,5
151-240	18	18	18	18	11	18	18	18	18	18	18	4,5

Engine 12	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	0	3	1	0	2	0	0	0	1	96,5
11-100	1	0	0	7	1	0	5	2	0	1	2	90,5
21-110	0	1	0	1	4	0	10	1	0	0	3	90
31-120	3	2	6	1	9	1	4	2	0	2	4	83
41-130	2	4	5	4	13	0	12	6	0	3	6	72,5
51-140	5	4	18	5	10	0	18	18	0	5	11	53
61-150	7	6	18	14	18	2	17	18	0	7	3	45
71-160	13	8	18	16	18	0	18	18	4	18	11	29
81-170	18	7	18	18	18	4	18	18	0	18	18	22,5

Engine 13	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	2	0	0	2	1	0	4	0	0	95,5
11-100	0	2	7	1	0	16	4	1	17	1	5	73
21-110	1	0	4	4	1	18	2	0	18	0	1	75,5
31-120	5	0	7	4	1	18	5	0	18	3	1	69
41-130	5	0	7	18	0	18	18	3	18	7	4	51
51-140	3	7	17	18	0	18	16	5	18	3	4	45,5
61-150	3	8	18	18	4	18	16	2	18	11	11	36,5
71-160	9	18	18	18	9	18	18	6	18	18	18	16
81-170	17	18	18	18	9	18	18	12	18	17	18	9,5

Engine 14	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	5	1	2	3	0	3	7	0	4	0	0	87,5
11-100	0	0	7	0	0	7	1	0	10	1	5	84,5
21-110	0	0	2	3	0	18	2	0	11	0	1	81,5
31-120	1	0	18	7	5	18	2	1	17	0	3	64
41-130	0	1	18	13	4	18	1	0	18	1	8	59
51-140	1	2	18	10	7	18	6	0	18	7	9	52
61-150	0	10	18	12	11	18	9	2	18	5	6	45,5
71-160	0	5	14	17	11	18	15	4	18	8	10	40
81-170	3	14	18	18	3	18	18	18	18	16	13	21,5
91-180	9	18	18	18	15	18	18	18	18	18	18	7

Engine 15	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	6	0	0	0	2	2	3	0	0	2	0	92,5
11-100	1	0	1	7	5	1	1	2	1	8	2	85,5
21-110	0	1	3	11	1	0	3	1	0	0	0	90
31-120	2	0	0	18	1	1	4	4	0	2	0	84
41-130	0	0	2	13	11	2	4	8	3	6	5	73
51-140	0	0	9	18	18	0	5	10	2	3	2	66,5
61-150	1	0	8	15	18	2	18	18	3	12	3	51
71-160	9	0	10	11	18	3	18	18	9	11	10	41,5
81-170	9	2	18	11	18	0	18	18	16	18	16	28
91-180	18	4	18	18	18	0	18	18	18	18	18	17
101-190	18	11	18	18	18	3	18	18	18	18	18	12
111-200	18	18	18	18	18	12	18	18	18	18	18	4
121-210	18	18	18	18	18	16	18	18	18	18	18	2

Engine 16	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	3	1	0	0	0	1	0	1	1	4	0	94,5
11-100	0	0	0	0	0	0	0	1	1	1	0	98,5
21-110	0	1	3	0	0	0	0	3	1	1	2	94,5
31-120	3	0	0	0	0	0	0	4	0	0	0	96,5
41-130	0	0	0	6	1	0	5	9	0	0	0	89,5
51-140	0	0	1	6	1	0	8	3	3	2	0	88
61-150	2	2	3	3	5	1	6	14	0	2	0	81
71-160	1	2	0	6	9	1	5	12	0	1	0	81,5
81-170	0	4	0	11	8	1	11	11	1	2	5	73
91-180	1	5	4	9	9	1	11	14	0	3	6	68,5
101-190	2	8	9	13	7	1	15	18	0	9	4	57
111-200	6	4	18	18	18	1	18	18	1	16	16	33
121-210	11	2	18	18	18	1	18	18	0	13	17	33

Engine 17	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	0	3	0	0	4	0	1	2	2	1	93,5
11-100	0	0	0	1	0	1	2	1	2	0	0	96,5
21-110	0	0	5	0	0	1	6	0	3	1	0	92
31-120	0	0	8	1	3	3	0	1	4	1	0	89,5
41-130	0	0	0	0	0	8	1	0	1	0	3	93,5
51-140	0	0	6	0	0	8	7	0	5	1	2	85,5
61-150	0	0	2	3	6	17	1	0	10	0	0	80,5
71-160	0	0	0	0	1	18	3	0	18	0	0	80
81-170	0	0	1	3	2	18	10	0	18	0	4	72
91-180	0	0	0	3	0	18	6	0	18	1	0	77
101-190	0	0	0	0	0	18	3	1	18	0	1	79,5
111-200	2	1	0	2	0	18	4	2	18	0	0	76,5
121-210	0	0	4	2	0	18	7	1	18	4	5	70,5
131-220	0	0	9	2	1	18	9	0	18	6	6	65,5
141-230	5	0	15	8	3	18	18	2	18	5	9	49,5
151-240	6	2	18	18	3	18	18	2	18	8	18	35,5
161-250	16	5	18	17	18	18	18	5	18	14	18	17,5
171-260	18	10	18	18	18	18	18	13	18	18	18	7,5
181-270	18	18	18	18	18	18	18	18	18	18	18	1
191-280	18	18	18	18	18	18	18	18	18	18	18	1

Engine 18	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	3	0	4	1	8	1	1	3	3	2	87
11-100	1	1	0	2	0	18	2	2	12	0	0	81
21-110	1	0	0	4	0	18	0	3	18	0	0	78
31-120	1	11	0	2	0	18	1	2	18	1	3	71,5
41-130	0	1	2	2	0	18	1	3	18	0	0	77,5
51-140	3	2	1	3	0	18	0	0	18	1	1	76,5
61-150	12	7	3	1	0	18	4	2	18	1	0	67
71-160	2	2	7	2	0	18	8	1	18	6	0	68
81-170	2	5	11	6	0	18	7	0	18	8	8	58,5
91-180	12	16	18	6	6	18	18	5	18	15	18	25
101-190	16	18	18	18	15	18	18	18	18	18	18	3,5
111-200	18	18	18	18	18	18	18	18	18	18	18	1

Engine 19	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	6	1	4	2	0	0	0	2	0	1	4	90
11-100	4	3	2	4	3	0	1	4	0	6	3	85
21-110	12	2	2	1	8	0	8	3	1	4	6	76,5
31-120	10	3	6	3	10	0	4	8	0	7	7	71
41-130	8	5	18	14	18	0	11	12	1	16	10	43,5
51-140	5	2	18	18	18	0	18	18	17	16	16	27
61-150	8	5	18	18	18	0	18	18	18	18	16	22,5
71-160	8	10	18	18	18	3	18	18	18	18	18	17,5

Engine 20	T at LPC o/l	T at HPC o/l	T at LPT o/l	PSI at HPC o/l	Fan RPM	Core RPM	Rat of FF to Ps 30	Corr fan RPM	Corr core RPM	HPT coolant	LPT coolant	CI value
1-90	0	2	0	0	0	4	1	0	0	1	0	96
11-100	0	7	0	3	0	2	0	0	1	2	0	92,5
21-110	2	0	0	5	0	1	0	0	0	2	0	95
31-120	0	3	0	2	0	3	1	0	0	1	0	95
41-130	0	3	7	2	0	4	0	0	0	3	0	90,5
51-140	3	3	5	0	0	8	0	0	0	5	0	88
61-150	0	3	0	1	1	7	0	1	0	0	1	93
71-160	0	0	9	4	2	1	0	2	0	6	0	88
81-170	1	1	5	1	3	1	5	1	0	1	3	89
91-180	0	4	3	4	10	0	7	11	0	2	0	79,5
101-190	3	12	5	5	8	3	7	12	4	5	0	68
111-200	9	5	8	18	10	1	16	9	10	4	11	49,5
121-210	4	2	12	18	18	4	18	14	3	4	9	47
131-220	10	6	18	18	18	4	18	18	3	8	11	34
141-230	18	6	18	18	18	8	18	18	7	11	16	22
151-240	18	6	18	18	18	10	18	18	3	9	12	26

NTNU
Norwegian University of
Science and Technology

SINTEF