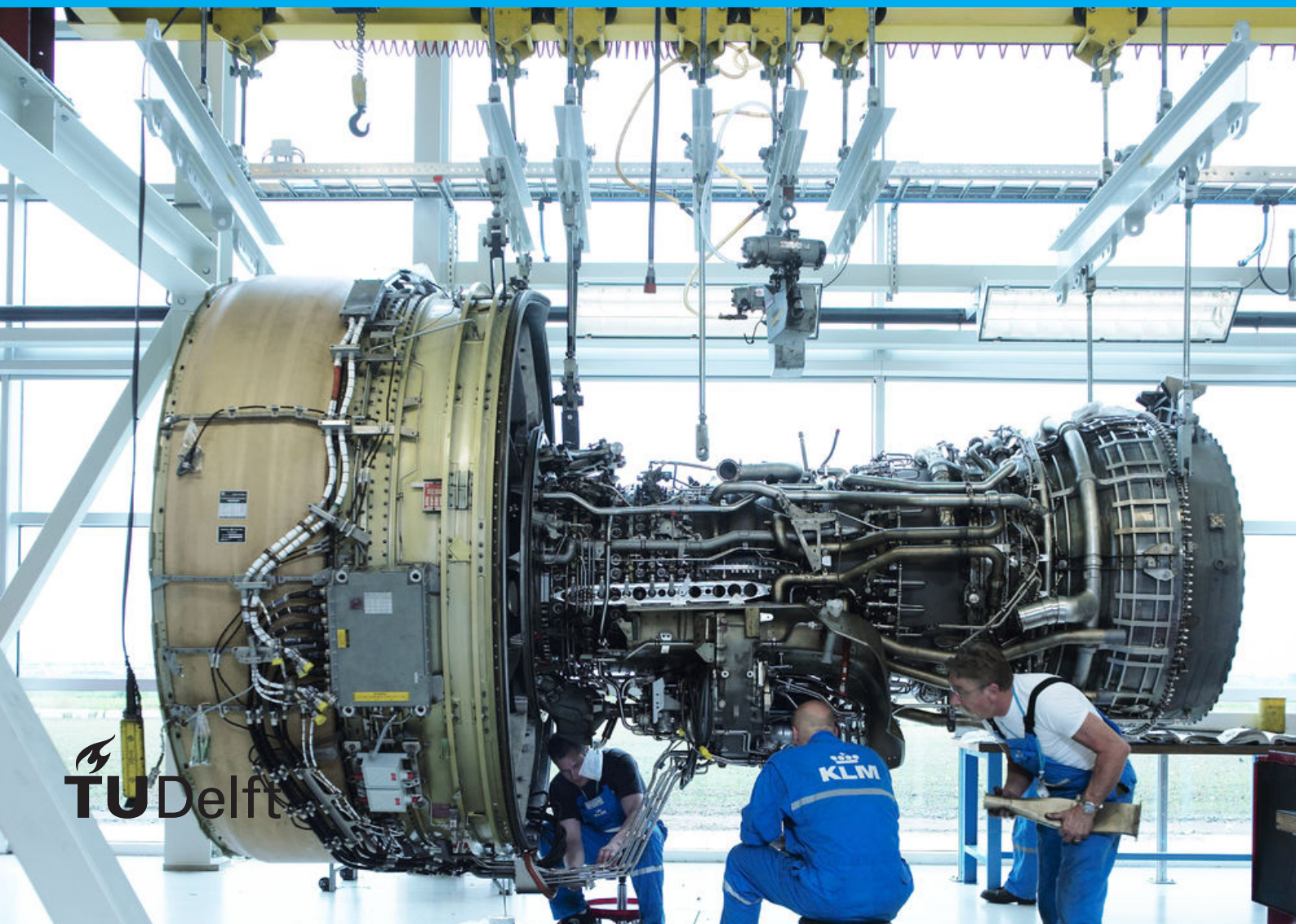
Value oriented condition based maintenance using a Grey forecasting model

A.D. Doedijns

A comparison study on the impact of different CBM optimization techniques with respect to maintenance cost savings



TU Delft

Value oriented condition based maintenance using a Grey forecasting model

by

A.D. Doedijns

to obtain the degree of Master of Science in Aerospace Engineering
at the Delft University of Technology,
to be defended publicly on Tuesday July 28, 2020 at 09:00 AM.

Student number:	1384430	
Project duration:	April 8, 2019 – July 28, 2020	
Thesis committee:	Dr. ir. W. J. C. Verhagen,	TU Delft, supervisor
	Dr. ir. B. F. Lopes dos Santos,	TU Delft
	Dr. ir. E. Mooij,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Difficult things take a long time,
Impossible things a little longer.

ANDRE A. JACKSON

What we know is a drop,
what we don't know is an ocean.

ISAAC NEWTON

Preface

This work is the culmination of my academic education at the Technical University of Delft. It is the final hurdle that had to be crossed before one can calls himself a Master of Science in Aerospace Engineering. This report constitutes of 9 months of full-time dedication to show my ability in conducting research on an academic level

By choice I selected the topic of data-driven condition based maintenance, not only because I believe that that is the future for the airline MRO industry, but because of my intrinsic interest of applying data-driven solutions to problems. During this research I increased this passion even further and I am keen on using the knowledge and experiences I learned further on in my professional life as well.

As this research uses publicly available data and because I am not affiliated to a third party I only held responsibility to my thesis mentor Wim Verhagen and myself. This proved to be a blessing and a curse. Whilst this enable me to combine my thesis work with my part-time job I also struggled in finding a balance between the two. The patience and involvement of Wim helped my find that balance and this eventually resulted in an increase in discipline and professionalism in my character.

I wish that reading this report will increase the readers knowledge and interest in the topic of data-driven CBM as it did for me.

Acknowledgements

During my time as a student of Aerospace engineering I have encountered ups and downs on both an academic as well as on a more personal level. My friends and family helped me to overcome these difficulties and without them I would not be able to be the person that I am today. They encouraged me to persevere and finish my studies. This was not just to keep up appearances, but because they had confidence in my capabilities and knew that I would find pride in reaching my goal of becoming an Aerospace engineer.

I want to thank Wim Verhagen in particular who showed great feats of patience and held confidence during the numerous trials I encountered during this final stretch of my studies. He even supported me and provided guidance whilst he was abroad continuing his career at the University of Melbourne. In my opinion that shows a great deal of altruistic involvement that is a rare quality these days. Furthermore I also want to thank V. Viswanathan and A. Bombelli who helped review my research during vital milestones even though they had little connection with me during my academic career or during this final phase. Subsequently I would like to thank the remaining members of the exam board, B. Lopes dos Santos and Erwin Mooij for evaluating my work.

Finally to my father Bram, Inge, roommate Laurens and sister Titia I would like to give my deepest gratitude. They supported me when I struggled to find motivation. They also kept me on track by continuously asking about my progress and hurdles along the way. This not only helped me gain a more structured work ethic, but also increased my interest and enthusiasm for the work I was performing. Without those two elements I would not have reached this turning point in my life. During the final stretch of my studies I may not have been a very thoughtful, caring or present to them during this final stretch and I aim to make it up to them from now on.

Executive summary

Over the past decades the aerospace industry has been investigating a shift in maintenance policy from preventive maintenance towards predictive maintenance, often called condition based maintenance (CBM). This maintenance schedule will not feature predefined maintenance intervals but adapts according to the conditions of the systems within an aircraft. To be able to adjust the maintenance schedule the current system state has to be monitored with numerous sensors. These sensors collect vast amounts of data, which combined with historical failure and degradation events, can be used to train intelligent predictive algorithms. These algorithms are able to find underlying patterns that are often not known beforehand. The data is stored either locally (on-board the aircraft) or externally and thus directly fed through to the maintenance department database. This collected *"big-data"* requires a new type of model that can overcome the three V's that characterize it, namely the data volume, variety and velocity.

The predictive algorithms used within maintenance typically fall in 4 classes. The model classes are: Physical, knowledge based, data driven and hybrid/combination models. Each type of class has its distinct advantages and disadvantages. Because aircraft are inherently complex systems there is a shift towards data driven and hybrid models. These type of models are better suitable in finding complex hidden relations. Furthermore these models can be built more flexible than purely physical models and are therefore cheaper to integrate to a wide variety of subsystems.

Although the future of CBM sounds promising there are still numerous obstacles that need to be addressed before interval based preventive maintenance becomes a legacy practice. Current research focusses most of its efforts on improving the accuracy and flexibility of prognostic models. Making them increasingly capable of forecasting different sets of data with high precision. However there is a lack of integration of these newly developed models within a practical maintenance architecture. This is mainly because of either relaxing reality by incorporating assumptions, or simply because the overall effect of a validated implementation is not analysed. Furthermore there have been limited attempts to make a financial value based assessment of such an integrated CBM system.

This research will aim to fill this gap in the state-of-the-art. It will do so by implementing a data-driven Grey model (GM) in a CBM framework and evaluate its financial benefits over a run-to-failure maintenance paradigm. Not only will this research focus on this alone, but a comparison will also be performed on different optimization strategies of such an integrated CBM system. As this is not an easy feat to conquer it is a too large and complex problem to implement on an aircraft as a whole within the scope of this research. It is therefore proposed to focus on a specific subsystem, namely a turbofan engine. The selection for this specific subsystem is driven by two considerations. First there is open-source data available from the NASA PHM-08 challenge that perfectly fits data requirements. Secondly as the power unit takes up around 47% of the total aircraft maintenance cost [17], and is increasing (as of 2018), it is also the most interesting from a financial standpoint as well. By limiting the study to this specific use case it will ensure that the fewer compromising assumptions and simplifications need to be made, keeping the model a good representation of realistic maintenance practices.

The steps that have to be addressed for a fully integrated CBM framework, as found by the reviewed literature can be summarized as follows. First data needs to be collected. This has already been provided by the C-MAPS simulation program used for the NASA PHM data set. This data then has to be filtered and converted to a suitable format to allow the prognostic algorithm to process its information. The filtered data will then be used to train the algorithm finding its emergent relations and trends. This trained algorithm will then be used in conjunction with failure records to provide accurate remaining useful life (RUL) predictions. These predictions will be validated using another set of the collected historical data. Once validated the RUL estimations will be fused with maintenance knowledge to form a realistic model that can assess the benefit of the fully integrated CBM model. Furthermore the integrated system has a modular approach wherein each subsequent segment will be evaluated on its respective performance metrics. This will allow to investigate the advantages of taking a holistic

approach of the CBM system as a whole, rather than optimizing for each specific component individually as is tradition in current literature.

This study does show that a form of gestalt is reached when each module is combined and the CBM model is optimized on a KPI that encapsulates the performance of the integrated system in its entirety, such as end-to-end financial impact. Furthermore it shows that the system configurations that focus on reaching local optima on the individual components that a CBM system comprises of does not lead to overall optimal system performance.

Finally the financial impact metric will be normalized and assessed for a range of maintenance cost relations to create an insight of its applicability on a spectrum of future potential use cases. This will allow to provide an analysis of the economic benefit for not only the subsystem in question, but also gives an additional financial incentive for other subsystems in aircraft that lie within the the region where this model will have a positive financial net benefit as well. This can encourage the aircraft maintenance industry to further its efforts in developing an integrated CBM framework for an ever growing number of subsystems. By expanding the domain of subsystems where CBM is proven to be effective the system can be made even more complex. This complexity can bring forth even more efficiency gains by, for instance, allowing grouping of maintenance tasks together to minimize resources required to keep a functional aircraft fleet.

Contents

List of Figures

List of Tables

List of abbreviations

Abbreviation	Description
AGO	accumulated generated operation
AOG	aircraft on ground
AR	auto-regressive
ARMA	auto-regressive moving average
AUROC	area under receiver operating characteristic
C-MAPSS	commercial modular aircraft propulsion system simulation
CBM	condition-based maintenance
CPL	continuous parameter logging
CSV	comma separated values
FC	flight cycles
FH	flight hours
FN	false negative
FP	false positive
GM	grey model
HPC	high pressure compressor
HPT	high pressure turbine
IAGO	inverse accumulated generated operation
JIT	just in time
KPI	key performance indicator
LPC	low pressure compressor
LPT	low pressure turbine
MAPE	mean absolute percentage error
MRO	maintenance repair and overhaul
$MTBR$	mean time between repair
NASA	national aeronautics and space agency
PHM	prognostic and health management
RMSE	root mean square error
RUL	remaining useful life
SNR	signal-to-noise-ratio
SOTA	state-of-the-art
TN	true negative
TP	true positive

1 Introduction

The aerospace industry has made large improvements regarding maintenance policies over the past decades [18]. This is mainly due to the shift from preventive maintenance towards condition-based maintenance (CBM). The term prognostics and health management (PHM) is also used instead of CBM, but to keep with consistency only CBM will be mentioned from now on (except when referencing to existing literature). CBM is a maintenance policy where maintenance is not performed on solely a time interval period, flight hours (FH) or a specific number of flight cycles (FC), but rather on the actual condition of the aircraft components/parts/subsystems themselves. Traditionally CBM can be divided in two main categories: failure diagnostics and failure prognostics. The former focusses on detecting and categorizing a failure when it occurs, whilst the latter predicts when a failure is likely to occur. This study will mainly focus on the techniques that are used for failure prognostics.

To be able to accomplish accurate fault prognostics data has to be accumulated. This is already being done on a large scale within the industry as a single jet engine can generate 20TB of data over a period of one year [3]. The collection of data is either being done by placing sensors or by performing routine inspections. By doing this the component state can be monitored in either real time or on an interval basis, this is called condition monitoring.

The collection of the data can then be subjected to algorithms with forecasting abilities to predict future deterioration. This forecast can subsequently be used to deduce an optimal maintenance policy. This can be done in two manners; the first being implementing a predefined threshold for the status of the monitored parts where after maintenance is required, this is called fault diagnostics. Or the latter by implementing the prognostics evaluations in an integrated condition based maintenance (CBM) framework. These programs use predictions of the remaining useful life (RUL) of individual components/subsystems of an aircraft allowing for a more sophisticated maintenance schedule specifically tailored to the actual state of the aircraft to be made. This integrated CBM program can subsequently lead to improvements such as cost reduction, by decreasing downtime and delays for example.

Whilst most academic research has been focussed on either fault diagnostics, fault prognostics or defining a theoretical CBM framework (with predefined underlying fault probability), limited efforts have been made to integrate data prognostics into a CBM framework. Furthermore a optimization of such an integrated CBM system from an practical financial perspective is also lacking in academic research. This research will aim to fill these gaps. Therefore the proposed research question can be formulated as:

Does a condition based maintenance model that follows a maximization strategy on maintenance cost reduction lead to lower overall maintenance cost compared to more traditional performance metrics maximization strategies of its individual components, such as: forecast error, AUROC and recall-precision performance metrics.

In order to research this an integrated CBM system will be divided in three main modules. These are the prognostic forecast model, the CBM decision framework and economic value assessment segment. The prognostic model and CBM framework will both be optimized locally, whilst a third optimization

will take a integrated system approach. Furthermore a "standard" open-loop simulation will also be performed that uses the configuration settings used in literature.

The study will be presented by first describing the experimental setting and use case in chapter 2. Herein the need for this research will be highlighted and the fundamental design setup will be described to the reader. Furthermore the data that will be used for this research, the NASA PHM-08 challenge set, will also be presented in this chapter. This run-to-failure simulation data of aircraft turbofan engines has been collecting using the C-MAPSS program that has been developed by NASA for the specific purpose. This program thus allows for an unique insight in the deterioration of a vital aircraft system until failure. Theoretical context of the prognostic model of choice (GM(1,1)) will also be given in this chapter. As a GM(1,1) does not assume a certain mathematical function it is capable of dealing with complex evolving time-series, as is the case in aircraft engine degradation. Additionally the model is not a *"black-box"* system and as such more insight in the process can be attained as well.

Next chapter 3 will explain the integrated CBM model in detail. This will be done by explaining a comprehensive flowchart that illustrates the integrated CBM system. In that chapter it will be made clear that a iterative modular approach has been used in order to assess the system on a system level as well as on sub-system level. The modular approach also aids in the verification and validation, described in a later stage of this report.

Following this chapter 4 will aim to describe how the individual components listed in the precious chapter were implemented in practice. This will be done by describing all considerations of the individual subsystems of the aforementioned flowchart in detail. First the data pre-processing will be explained and a method for obtaining the most suitable features from the data will also be presented. Furthermore a proposal how the prognostic model has to be altered to use the selected filters will also be described. Next these prognostics will be used in a maintenance decision making model wherein dynamically adjustable thresholds are used to assess an optimal strategy for repair. The performance of this model will be measured by metrics such as the receiver operating characteristic (ROC) and precision-recall score. Finally the maintenance strategy that follows from that CBM decision framework will be evaluated on its financial impact. To refrain from creating a overly specific research an economic feasibility region will be analyzed instead. This also limits the amount of assumptions to be made and will enable a more generic assessment of the research question at hand.

The results from this proposed methodology will then be collected and presented in chapter 5. In this chapter special dedication to the comparison of the various optimization strategies will be given. furthermore it will show that a holistic approach to a CBM system increases its financial impact over traditional localized optimization strategies on each segment that an integrated CBM system comprises of.

Chapter 6 will then aim to verify and validate these findings. It will do so by first explaining the steps taken to ensure that the proposed models are implemented correctly. The modular approach of the system will be one of the key methods used to warrant this correct implementation. Secondly the procedures used to validate will also be described. One of these was comparing results obtained with data selected for the training of the system with results from the test data. This will prevent that the model will over fit the training data and gives a view on the robustness of the system.

Chapter 7 will then further the analysis of the robustness of the system and will aim to find more information on its underlying dependencies on key parameters. By adjusting those parameters and analyzing the impact on performance metrics, both on a system level as well as on a local level, it will give valuable information on the applicability of using such a CBM system. It will also point out potential shortcomings that can be addressed in future research as well.

Finally the conclusion and recommendations, chapters 8 and 9 respectively, will summarize all the findings of this research and provide ideas for improvements made by future research. The conclusion will inform the reader of the benefits of using CBM in the aircraft industry, with an integrated CBM system that has been optimized on a KPI that describes performance on a system level in particular. This provides another stimulant for the aircraft maintenance industry to focus their efforts on incorporating data driven condition based maintenance within their standard maintenance practices.

2

Experimental setup

This chapter will explain the necessity and the construction of this research. It will do so by first explaining the need for this particular research in section 2.1. This will be done by illustrating the gap in the state of the art within CBM and how this study aims to fill this gap. In the following section, 2.2, the specific use case will be discussed. The experimental setup will be explained and the data set composition will also be described. From then a selection of the available data must be made. How this has been done will be showed in section 4.2. The method for obtaining the most valuable data fields will be shown and which features will be used to predict system degradation. Finally in section 2.3 the choice for the forecast model of choice, GM(1,1), will be explained.

2.1. Study motivation (gap in state of the art)

Unscheduled maintenance is a costly part of the aircraft industry. Between $10 - 20\%$ of total life cycle costs (LCC) are caused by maintenance [17]. In 2018 this amounted to a total spent of 69$ billion on MRO. This in a sector that relies on small profit margins and huge financial investments. It is therefore highly beneficial to reduce cost effectively wherever possible. The aim of incorporating a CBM policy is to minimize the cost of aircraft maintenance by reducing unexpected failures that lead to downtime of aircraft and additional delay related costs. The difficulty lies in finding the optimum replacement point. One wants to reduce unscheduled failures but a too conservative replacement/repair strategy will reduce the useful life of components which can, in effect, negate any reduction in cost gained by switching to a CBM policy.

When reviewing the literature on CBM and prognostic maintenance forecasting it can be stated that, although there has been numerous research on the topic, there still exist gaps to be filled. Current research focuses on one of two aspects. Either an algorithm that achieves a certain (often improved) prognostic accuracy is proposed, or it is proposed that reliability data can be used as a tool to develop a CBM framework without explaining how this data is obtained. It can thus be stated that these two aspects, while mutually dependent on each other, are not thoroughly combined in research.

This is especially true for research that focuses on the development of improving prognostic models. Although these models reach higher forecasting accuracy, or are able to predict further ahead with a similar degree of accuracy, the implementation of the prognostics model to assess its effects is lacking. Furthermore the data used are usually manufactured in some way or another. Either the data are simulated or faults are induced artificially to achieve a desirable controlled data set for the specific research. These aspects therefore reduce the practicality of implementing the methods and findings of the reviewed research in existing platforms. Furthermore there is a lack of run-to-failure data sets. In practice most data collected from real world scenarios end when only a fault is present and not until the actual failure. The NASA PHM data, explained in more detail in section 2.2, was developed to aid in combating this challenge by offering run-to-failure data to the public in an effort to improve research on this topic.

There is thus a gap to be filled in the state-of-the-art (SOTA). This gap can be illustrated with figure 2.1 wherein the missing links between prognostics and a CBM framework are represented by the red lines. An orange line is also present. This line represent the limited amount of research that has been

dedicated to this interaction of elements. An expanded version of this figure is presented in chapter 3, namely figure 3.1. In that chapter the individual steps will also be explained in further detail as well. Even more detail on the implementation of those individual steps will be given in chapter 4.

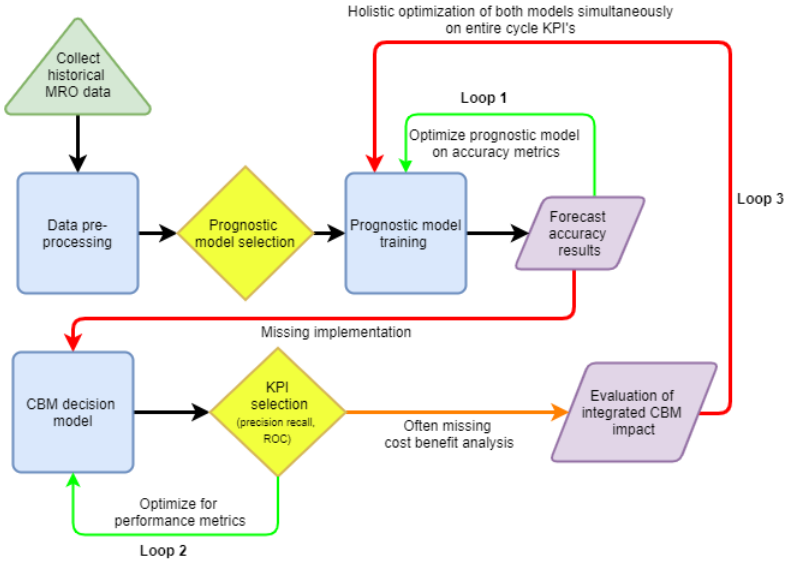


Figure 2.1: A simplified representation of an integrated CBM framework.

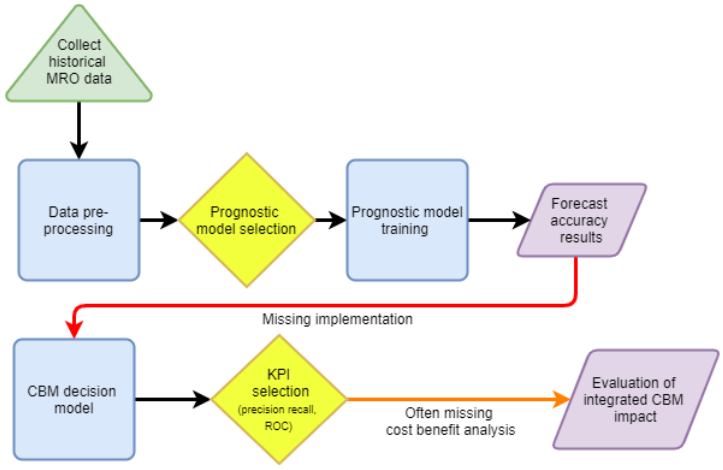


Figure 2.2: A simplified representation of an integrated CBM framework without feedback loops (open-loop).

There are several feedback loops present (loops 1, 2 and 3) in this figure that will used throughout this research. Each feedback loop represents the optimization of the model in an effort to reach optimal performance for each segment. The two green loops (listed as number 1 and 2) have been extensively studied by peer academics, whilst the red feedback loop is currently missing. Loop 1 will have the aim of reaching the highest forecast accuracy for the prognostic model. The next loop, number 2, will use a wide spectrum of prognostic outputs instead and will find a "new" optimum on CBM performance metrics. It will do so by adjusting both the GM(1,1) parameters as well as the CBM settings. The final loop, number 3, will try to find a global optimum by evaluating a wide array of settings from both the prognostics and CBM framework from a financial perspective. Finally there is also another setting, namely using only "standard" configurations found from literature. Herein no sensitivity analysis or wide band of configurations is used, but simply the stock configuration will be passed through. This can by illustrated by figure 2.2. One can see this as an "open-loop" model with no feedback at all.

2.2. NASA PHM challenge data

As explained in the previous section the NASA study offers a unique standpoint that it offers run-to-failure data to the public. This enables a wide variety of researchers the opportunity to compare the effectiveness of CBM policy's. furthermore the data is divided in training and test data over different failure modes. This gives the opportunity to validate the findings based on the training set as well. This section will provide the reader how the NASA PHM challenge data set was constructed. It will do so by briefly describing the NASA research performed.

The NASA study focuses on damage propagation in aircraft turbofan engines. As these are one of the most complex, safety critical and expensive systems of an aircraft. It can therefore be considered an area in which a successful data-driven CBM approach can have significant importance. Furthermore C-MAPSS (Commercial Modular Aircraft Propulsion System Simulation) [12] enables the user to vary input parameters that effect health and compute output sensor data that can be used for prognostic purposes. These two facts highlight the importance and possibility of implementing a CBM method for aircraft turbofan engines.

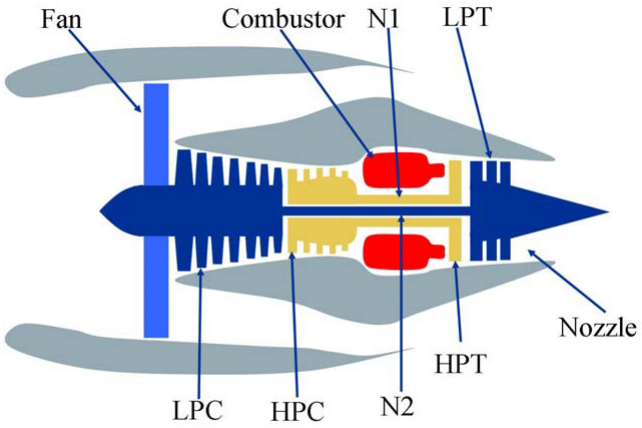


Figure 2.3: Simplified diagram of engine simulated in C-MAPSS [12].

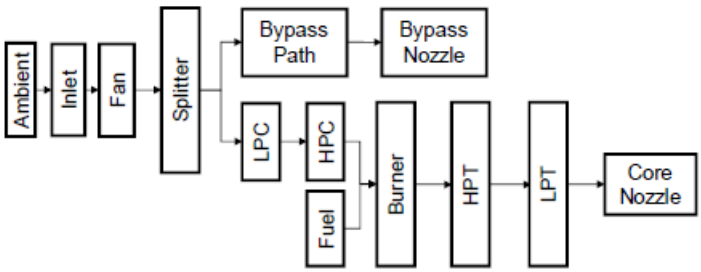


Figure 2.4: A layout showing various modules and their connections as modeled in the simulation [12].

Figure 2.3 shows a simplified schematic of such a turbofan engine. Figure 2.4 illustrates how the elements in figure 2.3 are implemented in the C-MAPSS simulation. The study uses findings found in literature from Chatterjee et al. and Goebel et al. [7] and [14] that efficiency and flow rate deviation of the compressor and turbine relates to system health, and as such the study uses these parameters to determine the system health accordingly. The model can be used in both open-loop or closed-loop, but was only used in a closed-loop configuration as this is more representative of a real-world scenario. The simulation records 21 outputs that are available for the prognostic health assessment. Table 2.1 lists these outputs.

The model also adjusts the wear introduced under different operating conditions for each engine and flight accordingly. The operational conditions also specify what the current system health is by adjusting the system health margins. This means that the allowable system health is dependent on

Available parameters as sensor data

Symbol	Description	Units
T2	Total temperature at fan inlet	°R
T24	Total temperature at LPC outlet	°R
T30	Total temperature at HPC outlet	°R
T50	Total temperature at LPT outlet	°R
P2	Pressure at fan inlet	psia
P15	Total pressure in bypass-duct	psia
P30	Total pressure at HPC outlet	psia
Nf	Physical fan speed	rpm
Nc	Physical core speed	rpm
epr	Engine pressure ratio (P50/P2)	–
Ps30	Static pressure at HPC outlet	psia
phi	Ratio of feul flow to Ps30	pps/psi
NRf	Corrected fan speed	rpm
NRc	Corrected core speed	rpm
BPR	Bypass Ratio	–
farB	Burner fuel-air ratio	–
htBleed	Bleed Enthalpy	–
Nf_dmd	Demanded fan speed	rpm
PCNfR_dmd	Demanded corrected fan speed	rpm
W31	HPT coolant bleed	lbm/s
W32	LPT coolant bleed	lbm/s

Table 2.1: C-MAPSS outputs available for system health analysis [12]

the operational conditions and can thus vary between flights. However as the effects and operating conditions are not shared as output parameters one is subjected to use only the data from table 2.1. Furthermore only standard usage wear is simulated and failure events are omitted. Another interesting fact is that between-flight maintenance is not omitted, but is included as process noise. This allows the degradation to not be locally monotonic. This is in contrast to a lot of other simulated studies found in the literature and more accurately presents a realistic scenario. Initial wear has also been included in the simulation. as this is also prevalent in real-world situation as found in [23].

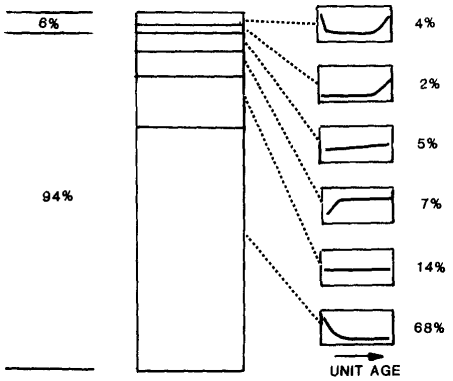


Figure 2.5: Experience age-reliability relationships.

As discussed above noise has been added to the simulation to include effects such as: manufacturing and assembly variations, process noise and measurement noise. It is therefore important that there are methods in place to deal with this. Special attention to this will be given in the decision of a suitable prognostic model. This is one of the reasons that a GM has been used. Further details on the benefits of using said prognostic model and its properties will be given in section 2.4.

2.3. Prognostic model selection

The selection of a specific prognostic model has been extensively explained in [11]. The most important elements from that report has been summarized to give the reader an understanding why a Grey model (GM) was chosen. More specifically a GM(1,1). This is a model that has an order of 1, meaning only a first degree of differential equations are used, and the model takes only a single parameter (sensor data) into account. A multi dimensional model is also possible, but this was deemed beyond the scope of this research.

As GM's do not rely on an underlying function to fit the data and thus makes it suitable to predict non-linear complex functions with high accuracy. This eliminates the assumption that wear and degradation follow a specific distribution. It therefore enables prognostics of multiple simultaneous failure modes more accurately. As this is the case for the underlying data that has been used this is a vital element why GM's are of particular interest. Furthermore they are also suitable in dealing with partial or limited data-series thus enabling quick and broad implementation.

Another positive aspect that the GM technique has is that it automatically filters noise and randomness due to the conversion of the AGO and IAGO [10, 15, 19, 22, 27, 29]. This feature is important since it reduces the fitting of random events that are not necessary linked to the health of the system. By filtering the noise the forecast will also be less erratic. This makes it possible for a simpler wear threshold to be applied without the need of complex outlier checks to be put in place.

It is also by nature less of a *"black box"* system such as a ANN system and can therefore give additional insights. The modifications to the standard model proposed by Xia et al. [29] allow the forecasting horizon to be increased further than the standard 1 time-step ahead. This makes it far more suitable in practice since maintenance lead times need to be factored in. This does come at computational price which has to be taken into consideration.

Some elements to be considered are that special care needs to be taken into account to not over fit the data by selecting a too high number of historic data points for the GM. This will primarily be done by validating the accuracy of the forecast with the test data. Work done by Chen et al. [8] proposes an improved GM with binary weights that can limit the possibility of over fitting the model. However incorporating this is beyond the scope of this research especially as there are already mitigating measures (validation testing) in place. Another point that needs to be taken into account is the difficulty GM's have when faced with fast growing time series. This is not an obstacle for this specific use case as the data are given on a per FC basis. There is thus ample time between each data point and the time series are in the realm of 12-16 FC.

2.4. Mathematical formulation of a standard Grey forecasting model

As previously stated the prognostic model of choice of this study is a GM, or GM(1,1) specifically. This section will explain the mathematical basis and the properties of the model in detail. However this is only an unaltered version and as such is not directly suitable for the use case of a CBM prognostic model. Alterations to this model that had to be made in order to increase the forecast horizon and to make it more robust as well will be explained later on in this report in subsection 4.3.

Grey models (GM's) were first developed by Deng [19] in 1982. Although they have already been used in fields such as management, economy, engineering, and finance [24] they are not widely used in the forecasting of condition based maintenance. Grey systems have the ability to predict and model data that are incomplete and/or limited in size. Grey systems are therefore able to be readily implemented at an early stage of development when large data sets are not yet available. The name *"Grey system"* comes from the nature of the model. It can be characterized as being neither a *"black box"* system, where the underlying model is not known, nor as a completely known *"white"* system, where the model has been predefined by the user. In GM's each parameter is given a specific range, typically $0-1$, where the parameter is selected by optimizing the statistical accuracy of the model. They are thus directly linked to the data set itself and are not subjected to an underlying assumption of following a specific distribution or function such as an exponential or Poisson distribution or a linear, quadratic or cubic relation respectively. GM's also have the added benefit that they are easy to implement in programming software such as MATLAB or python due to the matrix computations involved.

Basic GM's can be built by five steps. These steps will be listed below and explained in further detail

later in this report. The five steps are:

1. Collect the original data series $X(0)$
2. Perform an accumulated generating operation (AGO)
3. Estimate the parameters
4. Predict future point $X(1)$
5. Perform the inverse accumulated generating operation (IAGO) to predict values for the original data series $X(0)$

Grey models uses these steps to obtain a set of differential equations that can be used to forecast future values. The original time series data $x^0 = x^0(0), x^0(1), x^0(2)...x^0(i)...x^0(n)$ with $x^0(i)$ being the value of a selected parameter at time i.
The AGO is then used to transform this data set in a data set that contains less noise and is therefore better suitable to indicate trends. The basic AGO operation is as follows:

$$x^{(1)}(k) = \sum_{i=1}^{k} x^0(i) \qquad k = 2,3,...,n \tag{2.1}$$

With $x^1 = x^1(0), x^1(1), x^1(2)...x^1(i)...x^1(n)$. Afterwords the first differential equation can be built as:

$$\frac{dx_{(1)}}{dt} + ax_{(1)} = b \tag{2.2}$$

Wherein parameters a and b are determined using the least-squares method and give the relation of the rate of change for the state of the system to its current state. To calculate parameters a and b relations 2.3, 2.4, 2.5 and 2.6 can be used:

$$A = \begin{bmatrix} a \\ b \end{bmatrix} = \left[\beta^T \beta\right]^{-1} \beta^{-1} Y \tag{2.3}$$

$$\beta = \begin{bmatrix} Z^1(2) \\ Z^1(3) \\ \vdots \\ Z^1(n) \end{bmatrix} \tag{2.4}$$

$$Z^1 = \frac{x^1(i+1) + x^1(i)}{2} \tag{2.5}$$

$$Y = \begin{bmatrix} x^0(2) \\ x^0(3) \\ \vdots \\ x^0(n) \end{bmatrix} \tag{2.6}$$

The next step involves calculating the AGO. This can be done by integrating equation 2.2 and thus retrieving equation 2.7:

$$\underline{x}^1(i+1) = \left(x^1(1) - \frac{b}{a}\right) e^{-ai} + \frac{b}{a} \tag{2.7}$$

$$\underline{x}^0(i+1) = \underline{x}^1(i+1) - \underline{x}^1(i) \tag{2.8}$$

Combining equation 2.7 with 2.8, which is the IAGO, the transformation back to the original data set is complete and can be given by equation 2.9:

$$\underline{x}^0(i+1) = \left(x^0(1) - \frac{b}{a}\right)\left(e^{-ai} - e^{-a(i-1)}\right) \tag{2.9}$$

The steps described above shows only the basic unmodified GM(p=1,n=1) with p denoting the order of differential equation and n denoting the number of parameters used. As previously mentioned this model is ideally suitable for slow growth time series. For this use case this is not a problem as already described in 2.3. Adjustments to take this into account will also be presented in the next section as well.

2.5. Classification or regression analysis

The decision for a regression model over a classification model will be treated in this section. When analysing the literature a significant amount of research has been done on system health classification. The different health classed are then linked to a varying preset RUL estimates. This research on the other hand will use a regression model where the remaining time-steps until the failure threshold is crossed will be used directly. This is done by extrapolating the prognostic forecasts N time steps, between 12 and 18, ahead and retrieving the first entry where the forecast crosses the threshold limit. The RUL is then set to this point in time and the maintenance action will subsequently be performed.

Furthermore a regression model gives additional insights on the deterioration speed near end of life. That is if system health and sensor measurements are indeed linked as hypothesized. Finally as the true RUL of the test data is also available and can be easily compared to the estimated RUL from the model. This gives a measure of the performance of the system and will be described in more detail in section 5.2.

3

Methodology

This chapter will describe how the selected methods in chapter 2 are combined and configured to create an integrated CBM model. This will be done by first describing the model architecture step by step in section 3.1. This will be done by an illustrative flow chart, figure 3.1, which is a more detailed version of figure 2.1. The integrated system has been divided in four main segments that will be explained in further detail as well.

The clear definition of the separate modules that make up the integrated CBM model has the added benefit as that each segment can be analyzed individually. This approach is by design to aid in the verification and validation of each part of the system. Furthermore the modular approach also helps to maximize local and/or system performance metrics by adjusting the configurable parameters of the model due to the iterative nature of the framework with its feedback loops. The decision and reasoning for this approach will be highlighted in section 3.2.

3.1. Detailed flow chart of the integrated CBM model

When studying the literature a gap in the SOTA was determined. This gap was illustrated in a simple manner in figure 2.1. This section will divide this figure in more detailed steps and will also highlight each iterative loop present. Figure 3.1 gives this detailed overview by visualization of a more elaborate flow chart. From this figure the following steps can be found:

- **Data acquisition and processing**
  - Data collection
  - Initial data set analysis
  - Data filtering/cleaning
  - Feature selection
- **Prognostic model development**
  - Prognostic model selection
  - Prognostic model training
  - Validation of the prognostic forecast accuracy
- **Integration in CBM program**
  - Integration of the prognostic results in a maintenance decision framework
  - Optimizing the configuration of said model
  - Validating the CBM decision framework performance with test data
- **Cost evaluation module**
  - Combining the CBM performance metrics with cost breakdown knowledge to create a value model
  - Analysis of the results obtained from the value model
  - Optimize the total system for its over-arching value KPI
  - Validate the optimized value results
  - Document the findings in this report

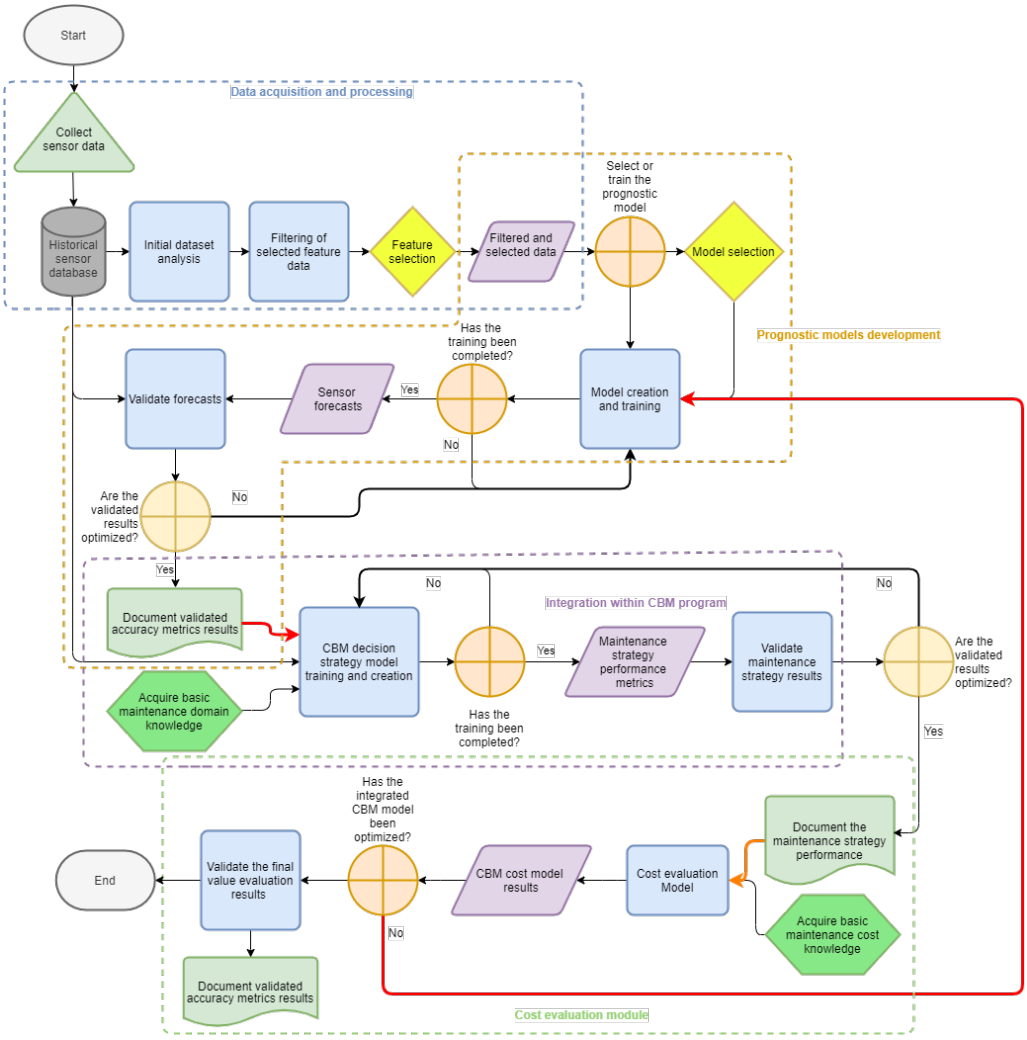


Figure 3.1: A detailed overview of the integrated CBM framework used in this study.

The first module in the figure describes the data acquisition and pre-processing steps. As these steps are already discussed in detail in sections 4.2 and 4.1 they will not be treated separately here. However the most important outcome of this process will be mentioned: this is the selection of the prognostic model of choice which is a GM(1,1).

Now that a prognostic model type has been selected a specific model has to be built. From the studied literature the mathematical structure of a GM(1,1) can be found. This also means that the model does not have to developed from scratch and the known assumptions and limitations are already identified. These two feats will ensure that the development of the prognostic model will not be too time consuming and it will also limit inherent flaws in designing the respective model.

When the model has been built it has to be trained using the simulated training data. This training will allow the model to gain the forecasting accuracy's required for the condition based RUL. It also gives information on the optimal configuration of the operational parameters of the model such as: the number of historic time steps to be incorporated in the model, the forecast horizon and *alpha* value. Multiple accuracy metrics such as the MAPE, RMSE and Theil's U statistic will be used to indicate the performance of the newly built model. This is an iterative process. If the validation results prove that the prognostic model does not give a good approximation the following actions can be performed: different features will be selected as the input data for the prognostic model, the prognostic model needs adjustment or in the worst scenario a different type of prognostic model needs to be selected. Special attention has to be given to prevent over-fitting of the trained model as well. This is to ensure that the model is not optimized for the trained data only, but still has flexibility and accuracy for future data.

Over-fitting of the prognostic model will be checked by comparing the forecast accuracy of the training data compared to the test data.

This process is an iterative loop that will be continued until the accuracy metrics show that the model is optimized whilst still producing validated results when the training data results are compared with the test data results. The results can then be documented and the model can be implemented to create a theoretical CBM system. The implementation of this theoretical CBM system will be elaborated in chapter 4.

Once the prognostic model has been trained it can be implemented in a CBM program. Herein a decision making method will be used in the form of failure thresholds. Once these thresholds are crossed by predicted values a repair will be scheduled. It is then interesting to see what the impact is of integrating the prognostics model in the maintenance schedule. To do this effectively a list of KPI's that assess the performance of the integrated model have to be established. These KPI's are for instance: true positive, false positive, false negative assessments of the maintenance strategy. The precision an recall of the model will also be evaluated as well as the impact on mean time between repairs ($MTBR's$). These will be elaborated in more detail in section 4.4.3. The acquired results will again be validated and optimized for these specific performance metrics.

The performance metrics obtained in the previous segment are then used in conjunction with domain knowledge on maintenance cost to form a realistic cost evaluation model. As it proved to be difficult to determine exact values for the cost breakdown a simplified and factor based approach is chosen. This process is elaborated in more detail in subsections 4.4.1 and 4.5. The entire integrated system will then be optimized based on these value evaluation results. Finally the obtained results are again validated and once proven to be valid the findings are documented.

It is important to note that implementing a CBM policy will not necessarily lead to benefits on all of these KPI's individually but will more likely reach a global efficiency gain. For instance it could well be true that the CBM policy will reduce unscheduled maintenance by replacing a component before failure. However this will also lead to shortening the operational life of the component. The opposite may also be true. This is also already illustrated by table 4.3. wherein the hypothetical $H0$ hypothesis is that the component should be replaced.

A trade-off has to be made in unison with the aircraft fleet owner and/or maintenance provider which of the KPI's are most valued. Some airlines value their availability or on-time arrival/departure more than their direct maintenance cost for example. By adjusting the parameters defined in the prognostic model and failure threshold values the performance can be tailored to better suit specific needs.

The adjustment of these parameters is also a key component in the sensitivity analysis of the model. This analysis will give not only the developer but also the implementer of the model valuable insight what factors contribute the most to the KPI's. For instance it may be the case that extending the forecast horizon too far will lead to a worse performance in general. Furthermore lowering the failure threshold will lead to lower unexpected failures but high maintenance costs. Metrics such as the AUROC and precision-recall curves as explained and illustrated in subsection 4.4.3, can be used to analyse the effects of adjusting these parameters.

Finally it is also very important to verify and validate the aforementioned KPI results. Verification can be done by checking the mathematical correctness of each of the models individual components. This can for instance be done by adjusting the parameters of the underlying model in order to simplify the model and check whether the results show similar results when analysed by hand. Validation on the other hand is more difficult as the model will only be simulated and will therefore produce no real world results. However similar studies have been performed on the same data using different methods and therefore a comparison analysis could be done. Furthermore the performance of the model an the training data compared to the test data will also be done. By closely comparing these results the validity of the model can thus be assessed.

3.2. Iterative and modular approach of the integrated CBM framework

As can be seen from figure 3.1 there are several individual modules that, when integrated, create a fully integrated CBM system chain. Because this study will evaluate whether the final feedback loop (based on cost evaluation) will increase performance over a more traditional localized feedback loop paradigm the complete system has been divided into 4 modules. These clear boundaries are the result of research in the current literature and are in line with the most common approaches to current CBM models. As mentioned in section 2.1 most research focusses on one of these segments in particular, often with the aim of reaching the maximizing performance metrics associated with these kind of models (i.e. forecast error, AUROC, precision-recall etc.). The modular approach enables the comparison of a system approach to the more traditional linear approach of the individual modules, which is one of the main goals of this research.

Constructing an integrated CBM system is complex and as such there are numerous errors that may arise. Locating these errors or bugs in the code that was used for the model can be a difficult and time consuming process. The modular approach played a vital role in keeping this to a minimum, as each module can be evaluated on performance separately. One can therefore simply follow the flowchart along its course until an anomaly is detected. To increase the ease of locating faults in the design even further all outputs from the individual modules are stored locally and can be assessed and analyzed in order to find discrepancies. This proved to be of the utmost importance in the verification and validation part of the design process as there were numerous bugs that were easily captured and corrected.

4

Implementation of the integrated maintenance framework

Now that a detailed framework for an integrated CBM paradigm has been established, as illustrated in figure 3.1 form chapter 3, it is important to elaborate how this was implemented in practice to create a model that adheres to this framework. This chapter will describe the individual components of figure 3.1 in more detail and will give the reader a better understanding in the considerations that have been taken to construct the model. It should be noted that due to the iterative and complex nature of the integrated CBM system that there is no clear linearity of the separate components of the flowchart. As such some considerations and decisions will already use elements that will be discussed at a later point of this report. Effort has been put into making this chapter as clear and coherent as possible, by cross-referencing as much as possible to aid in reading and understanding the taken steps.

From the flowchart it can be seen that the data pre-processing is the initial step of the system. This will thus be described first in section 4.1. Note that no special attention will be given to the acquisition of the data, since this has already been covered in section 2.2. This data does need to be filtered and a subset of the data has to be selected also to keep it within the scope of the research and to limit the computing power required for analysis. These considerations are described in subsections 4.1.2 and 4.2.1 respectively.

Second is the implementation of the prognostic model, which is a vital part of the CBM model, in subsection 4.3. In section 2.3 it was concluded that a GM will be used as the prognostic model. A description of this method has already been given in section 2.4. Herein the standard mathematical composition and explanation was presented. However further modifications are necessary to implement the prognostic model with the data format that was used and also to fulfill requirements such as the capability of forecasting more than one time-step ahead. These modification will be presented in subsection 4.3.

The next step is to use the prognostics obtained by the GM in a maintenance decision making framework. This will be defined in section 4.4. Herein required domain knowledge will first be presented in 4.4.1. A proposal how this information will be used to complete the maintenance decision making model will be given in 4.4.2. This will be done by proposing a dynamic failure threshold model capable of predicting imminent failures.

Finally there is also a segment that uses the performance of the aforementioned models to estimate the financial impact of implementing this integrated CBM system in subsection 4.5. Again there is a need for maintenance domain knowledge to make a valid assessment of the potential of the system. This knowledge is provided in subsections 4.4.1 and 4.5.

4.1. Data pre-processing

The first step in a big-data based model is understanding the data to be used. It could even be stated that the primary step is to consider how and which data are to be collected, but as this research uses open-source data that has already been simulated this step will be omitted. The methods used to analyze this data in this section.

In chapter 2.2 it is discussed in short how the parameters listed in table 2.1 were simulated by NASA, however the usability of this data still requires further analyses. First the data is extracted and evaluated on noise characteristics and monotonic behaviour. This will be described in subsection 4.1.1. Then subsection 4.1.2 will explain how a subset of the data was selected and how the data was organized to create understandable and scaleable data format.

4.1.1. Preemptive data analysis

First the data must be loaded and formatted in a useful manner. The data is originally contained in a single CSV file for each training set. Each training data has 100 run-to-failure simulation each comprising of different amount of flight cycles (FC). For analysis purposes it is imperative to organize the data on a per engine basis. The data structure used is that of the Python dictionary format. The dictionary format stores data within keys that can be specified by the user. As such calling a specific slice of the data is intuitive and the data can also be more easily understand by visual inspection.

When analysing the raw data for complexity and noise levels some interesting conclusions can be made. The entropy for each sensor is provided in figure 4.1.

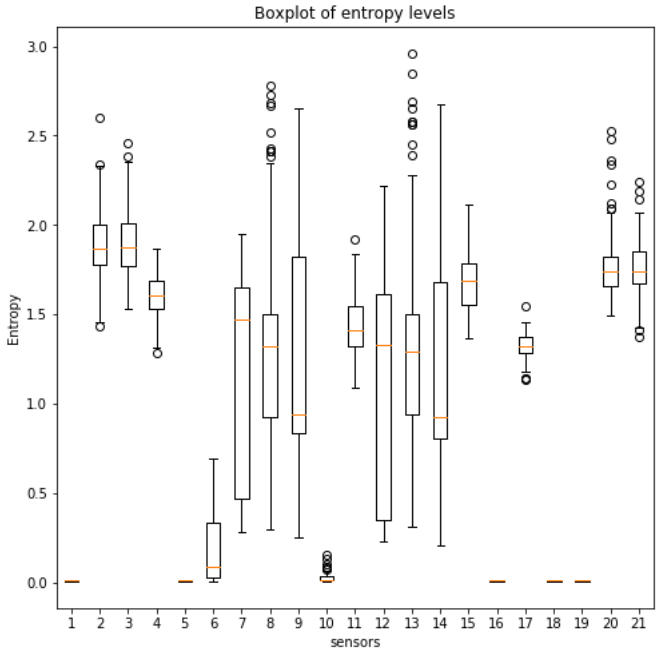


Figure 4.1: Box-plot of the raw sensor entropy of the NASA PHM data.

By inspecting figure 4.1 it can immediately be seen that there is large variation of the nature of the data for different sensors. Most notably sensors such as: fan inlet temp', 'fan inlet press', 'bypass total press', 'engine press ratio', 'burner fuel-air ratio', 'demanded fan speed' and 'demanded corrected fan speed' have a really low entropy value. When checking the data source it can be seen that the data remains (nearly) constant over the whole simulation period. It is therefore reasonable to assume that these hold no information on the system health and is removed from further analysis.

Whilst higher entropy levels mean a more complex data structure it is important to note that entropy levels of the raw data alone do not reveal the total picture. It is more important to look at the entropy of the residuals between the prognostic model forecast and raw data. By doing this any trends that are present that are captured are removed from the raw data and should, in theory, increase the entropy as only noise remains. This is illustrated in figure 4.2.

Although it can be seen from this figure that most residuals indeed have a higher entropy value (this is good since the present trend is captured by the model) it can also be immediately noticed that there are some new sensors that show totally different entropy values when compared to the raw sensor analysis. When investigating the raw sensor data and residual error of these sensors several aspects come to light. The deviations of the raw sensor values are relatively small compared to the absolute value. This causes the GM(1,1) to develop some huge spikes in the forecast as there exist the possibility

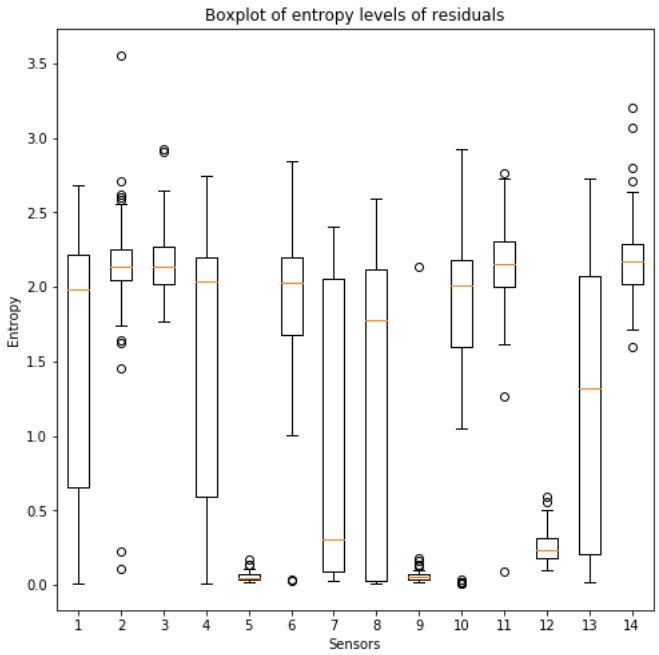


Figure 4.2: Box-plot of the raw sensor entropy of the NASA PHM data.

of near singular matrices. These spikes have to be accounted for by removing them from analysis or, as is the case of this study to limit the scope, the sensor is simply bypassed from analysis altogether.

4.1.2. filtering the data and data conversions

From the preliminary data analysis performed in subsection 4.1.1 several design decision can be made. These are: removing non information holding sensor data and data that where the residuals are high in SNR. Furthermore by visually investigating the remaining sensors for monotonic behaviour one can determine which sensors are especially suitable for regression analysis.

Furthermore the data was stored in a Python *"Dict"* format. By organizing it in this style data is stored in arrays that are associated to specific keys. These keys can then be used to identify the data giving the user valuable information of the content of the data. Nested Dictionaries are also possible and extensively used. This allowed the use of multi dimensional arrays without losing oversight of the captured information. As the data can be analyzed and identified due to this structuring it also allows for accessing on at later stages of the research without comprehension issues.

This proved vital for the modular approach of the constructed model. Any anomalies encountered at late stages of the design could be backtracked to its root location by a process of retracing data flows. Knowing the origin of the discrepancy considerably reduced the time needed to find the bugs that were encountered.

It also allowed the model to be scalable for future adaptations or configurations. This also proved to be hugely beneficial for the sensitivity analysis as the range of the keys that construct the dictionary could simply be expanded. The only downside that arose was the computational effort required to create increasingly larger Dictionaries. As the number of variable parameters rose it proved to form limit for the resolution and range of those parameters to keep computation time within bounds. One can see that if a list of N parameters with 10 different configuration that are independently studied a dictionary of N^10 key pairs has to be created. Combine this with a two dimensional threshold setting of 100 different heights and calculations for 4 different cost breakdowns a total of $4 * 100^2 * N^10$ simulation arcs have to be performed. With, for instance, 3 different variables the number of simulations comes down to 40 million. And as there are multiple calculations required for each simulation run time and memory issues can arise.

This was one of the reasons to keep the number of adjustable variable low by design and to also limit the range and resolution of those variables.

4.2. Sensor and feature selection from the data

The steps taken in section 4.1 gave insight in where potential value lies within the data. This section will delve into this a bit further by also looking at the underlying study where the data originates from. Furthermore statistic features from this data will be taken as entries for the prognostic model to reduce noise and compromise the data even further. This will be done by selecting the most suitable sensor in subsection 4.2.1 and why this sensor is selected in particular. Finally two statistic features will be selected that will be used as input data for the prognostic model. These will be presented in subsection 4.2.2.

It should be noted that a more sophisticated feature analysis is definitively possible, but as this is not the main focus of this research only 2 basic features are used. By searching for more complex features and using a multi-dimensional analysis of several of these features a, potentially, more accurate model can be obtained. However this also increases the workload linearly and the comparisons between the different analyses exponentially. This will result in a study that is beyond the scope of this research.

4.2.1. Sensor selection

When looking at the information gathered in section 4.1 a subset of the data can already be made. When this is combined with information given by the NASA research paper several remaining sensors are of particular interest. The study namely gives information about the failure mode namely high pressure compressor (HPC) degradation, and in training and test sets 3 and 4 HPC and fan degradation. It is also known that to simulate the HPC failure mode the parameters HPC efficiency and flow modifier are used. Furthermore the parameters that are used to assess the system health are the following: Total temperature at HPT outlet, Fan stall margin, LPC stall margin and the HPC stall margin.

With this information and looking at figure 4.2 It can thus be concluded that the most relevant parameters are: HPC and LPT outlet temperature, bypass ratio and LPT coolant bleed. To get an overview which sensor data is best captured by the GM(1,1) the residual forecast accuracy can be investigated. To do this properly multiple error metrics need to be used. This is to limit potential biases that a single error metric can have. Also using an absolute error metric as well as a relative accuracy metric will give a more informed idea of the accuracy. The used metrics are as follows: the mean average percentage error (MAPE) equation 4.1 from [21], Theil's U statistic U equation 4.2 from [28] and the root mean square error (RMSE) in equation 4.3 from [6].

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y-\widehat{y}}{y}\right| * 100 \tag{4.1}$$

$$U = \frac{\left[\frac{1}{n}\sum_{i=1}^{n}\left(y-\widehat{y}\right)^2\right]^{\frac{1}{2}}}{\left[\frac{1}{n}\sum_{i=1}^{n}\left(y\right)^2+\right]^{\frac{1}{2}} + \left[\frac{1}{n}\sum_{i=1}^{n}\left(\widehat{y}\right)^2+\right]^{\frac{1}{2}}} \tag{4.2}$$

$$RMSE = \sqrt{\sum_{i=1}^{n}\frac{\left(y-\widehat{y}\right)^2}{n}} \tag{4.3}$$

When computing the forecast error metrics of these sensors table 4.1 can be formed.

Residual accuracy for different metrics

Sensor	RMSE	MAPE	Theils U statistic
HPC outlet temp.	6.42 (5.30–7.55)	0.298 (0.282–0.314)	0.00199 (0.00163–0.00234)
LPT outlet temp.	6.16 (6.07–6.26)	0.356 (0.350–0.363)	0.00212 (0.00209–0.00216)
Bypass Ratio	0.0323 (0.0276–0.0372)	0.292 (0.278–0.305)	0.00188 (0.00160–0.00217)
LPT coolant bleed	0.0711 (0.0696–0.0725)	0.243 (0.240–0.247)	0.00149 (0.00147–0.00151)

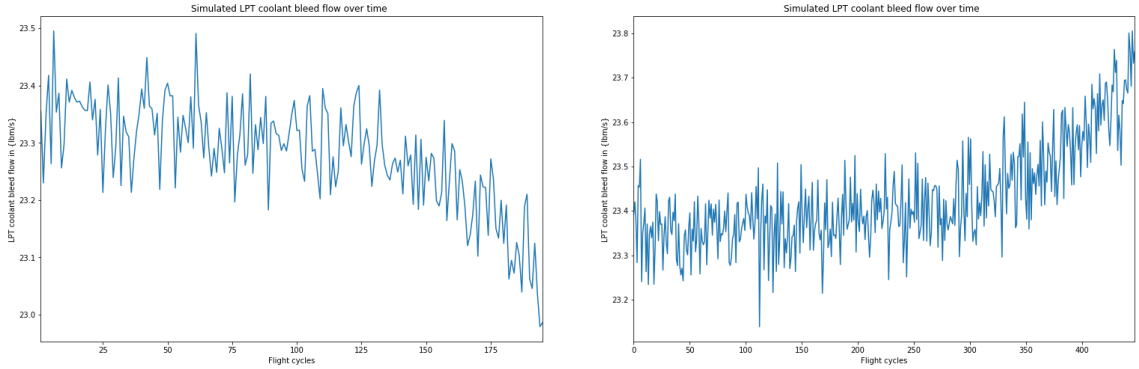
Table 4.1: Table listing the residual error of each sensor using several accuracy metrics (less is better).

It is clear form this table that the LPT coolant bleed sensor shows both a low absolute forecast error as well as a normalized forecast error. This sensor was therefore selected to be the primary focus of this study.

4.2.2. List the most suitable features

With a specific sensor being selected it is still necessary to select which features will be used from that sensor data. There are limits to what features can be used as there are only one time step measurements made available for the public. As such intra flight features are not an option. An extensive approach to feature selection can be done via packages such as *tsfresh*, but whilst this ensures that no information is missed from the data it does increase the number of variables dramatically. To limit the scope, run time and memory usage for this research it is therefore decided to only keep the most basic features.

To make an informed decision as to what these features may be one can simply look at the data at hand and see if there are some obvious patterns emerging. Figure 4.3 gives the LPT coolant bleed flow over the lifespan of a particular engine.



(a) HPC degradation only.

(b) HPC and fan degradation.

Figure 4.3: Simulated data of the LPT coolant bleed sensor until failure.

From these figures it can clearly be seen that the signal has noise but develops an upward or downward trend from a specific point in its lifespan. It could be that this is a direct result of HPC and/or fan degradation as are modeled in the simulation. The goal of this research is answering the question if this is indeed the case and if the amount of degradation of said element can be used as a reference to indicate system health.

Furthermore in Chatterjee et al. [7] it is found that the engine degradation system health can be linked to flow capacity deviations. These findings were also used to simulate the system health by the C-MAPSS program. Although not the LPT bleed flow was adjusted but the HPC flow it is assumed that will have a detectable and useful impact on the LPT coolant bleed flow as well.
Concluding there are three features that are going to be used as a health indicator. These are:

1. An absolute upper limit
2. An absolute lower limit
3. A deviation from the mean (taken from the initial X flight cycles of the training data)
4. An independent absolute upper and lower limit

These will hold a key role in the condition based maintenance decision model described in more detail in subsection 4.4.2, which is a part of the whole CBM module described in section 4.4. But before they can be used for failure prediction the underlying parameter, LPT coolant bleed, still needs to be entered in the prognostic model of choice, an adjusted for of a GM(1,1), to get the accurate forecast data mentioned above. How this was performed will be discussed in the next section.

4.3. Necessary modifications of the standard GM(1,1)

Whilst the mathematical formulation of a standard GM(1,1) were already given in section 2.4 it was already mentioned that the default model will not be suitable for the purposes of failure prognostics. Therefore some alterations and additions to the model need to be made. These will be described in this section.

The most notable changes to an original $GM(1,1)$ model is that the improved model is capable in forecasting multiple time steps ahead. It does so by using the one step ahead forecast as the final input for a next iteration, and increasing its first data point by one. This will be done until the required forecast horizon is met. This is also called a rolling $GM(1,1)$ model. This is also described mathematically by equations 4.4 through 4.6 and has been described and used by Akay et al. [2].

$$\underline{x}^0(i+1) = \left(x^0(1) - \frac{b}{a}\right)\left(e^{-ai} - e^{-a(i-1)}\right) for(x^0 = (x^0(1), x^0(2)...x^0(i))) \tag{4.4}$$

$$\underline{x}^0(i+2) = \left(x^0(1) - \frac{b}{a}\right)\left(e^{-ai} - e^{-a(i-1)}\right) for(x^0 = (x^0(2), x^0(3)...x^0(i+1))) \tag{4.5}$$

$$\underline{x}^0(i+3) = \left(x^0(1) - \frac{b}{a}\right)\left(e^{-ai} - e^{-a(i-1)}\right) for(x^0 = (x^0(3), x^0(4)...x^0(i+2))) \tag{4.6}$$

By doing this multiple times the forecast horizon can be extended for $i\ n$. As such the number of historic data points used for the first forecast also imposes a limit to the forecast horizon length. Increasing this forecast horizon does come at a cost of accuracy as each further forecast relies on more estimated values. This can effect forecast accuracy and stability. A fine balance has therefore be found in being able to forecast far enough ahead in order to take maintenance lead times into account and forecast accuracy. Forecasting further ahead in time does allow for a less sensitive failure threshold, which will be explained in 4.5.

Another aspect of the GM that can be adjusted is the factor *alpha*, present as the variable Z in equation 2.5. This factor can be described as the "sensitivity" of the model. A higher value for *alpha* will dampen the model and make it less erratic and more robust. It will also filter noise in greater extent as well. This does come at a cost. As the model is dampened it will also be slower to adept to introduced trends from incipient deterioration.

A delicate balance of these configurable parameters: forecast horizon, *alpha* value and number of historic data points analyzed has to be found. Finding the optimal value for those parameters and their influence on system performance will be one of the key aspects treated in chapter 7.

4.4. Condition based maintenance model

After the prognostic model has reached its accuracy level and has been found to be valid when comparing training with test data it can be used as an input for the maintenance decision making framework. How this will be done will be described in the upcoming section. First some domain knowledge has to be obtained that contains basic information of the maintenance task of the corresponding failure. An estimate has to be found for elements such as necessary lead-time, replacement cost and costs associated with unexpected maintenance to name a few. These will be treated in subsection 4.4.1. After that a method has to be developed that assesses system health. This is done by implementing a failure threshold that can be adjusted dynamically to find the optimal value for said threshold. How this is done will be explained in subsection 4.4.2. If, on the basis of this threshold, a failure is predicted it needs to be determined if this assessment is indeed correct. This will be explained in subsection 4.4.3. Several classification metrics will be described that can be used to give a quantifiable performance indicator of the model. Finally subsection 4.5 will translate this performance into a final value metric of maintenance cost reduction of the integrated CBM model compared to a run-to-failure maintenance paradigm.

4.4.1. Maintenance domain knowledge

To correctly configure the integrated CBM model knowledge of the maintenance practice is needed in order to make the model representative of the reality. This section will highlight some of the elements that were taken into consideration in order to make the model able to produce results that can be used to assess the value of said model. The main items that will be discussed are as follows:

- RUL tolerance cut-off point
- Minimum forecast horizon length
- Maintenance cost definition
- Maintenance cost approximation

As previously mentioned and visualized in figure 5.1 ideally one would be able to predict failures as far ahead as possible. However the difficulty in forecasting the RUL becomes greater with increasing forecast horizon lengths. Furthermore to assess whether a maintenance action is performed just-in-time (JIT) or too early gives an overview of the performance of the CBM model. A JIT forecast will result in a true positive prediction, whilst a too early prediction will lead to a false positive instead. The mathematical representation of these metrics is given in subsection 4.4.3. A cut-off point of 10 FC was selected that will make the distinction between those two assessments. This is a standard that is widely used in the industry. This parameter will henceforth be called the RUL tolerance window, or RUL tolerance. Increasing this value will result in higher scores for the ROC and precision-recall curves, which has been reported in greater detail in subsection 7.2.2. It should be noted that this parameter will not effect the eventual value metric since earlier repairs will still result in shorter $MTBR's$ and it has no effect on the number of false negatives. It can thus be stated that the cut-off point is somewhat arbitrary from a financial perspective, but still necessary to assess the performance and validity of the CBM model.

A second item that needs to be addressed is the capability of actually performing the maintenance action without resulting in expensive aircraft on ground (AOG) scenario's. This is the reason that a forecast horizon window range of 12 till 16 FC was used for the prognostic model. This will ensure that there is ample time to schedule a check/repair action without creating too much disruptions in the flight planning schedule.

In order to make a value based assessment on the impact of the model an understanding in the financial that make up the aircraft maintenance industry has be acquired. According to Galar et al. aircraft maintenance can be divided in several components as illustrated by figure 4.4 [13].

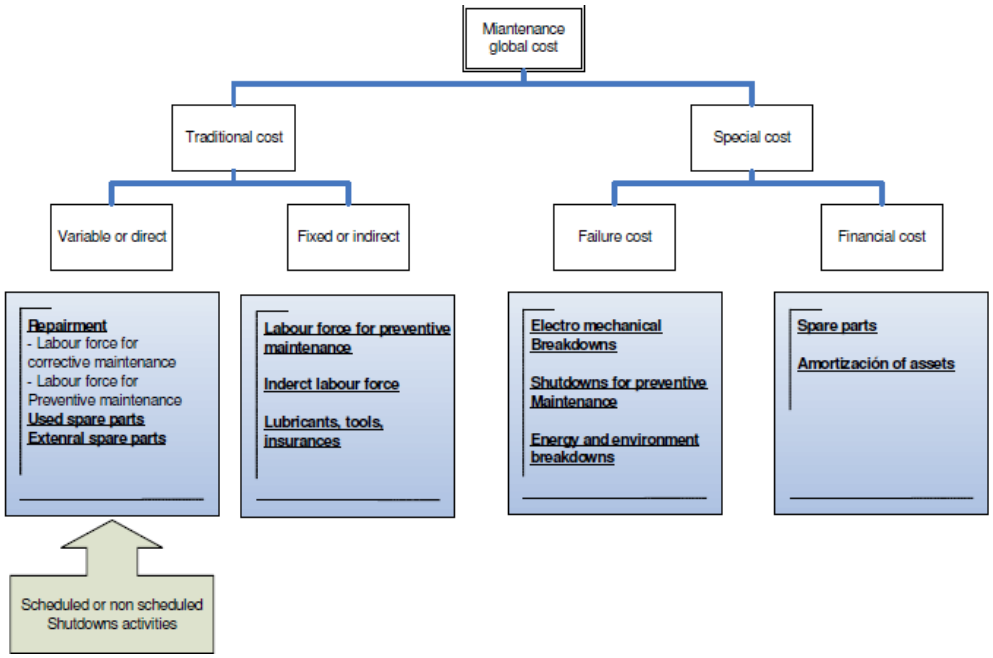


Figure 4.4: Breakdown of maintenance global costs in its different concepts [13]

Using this figure the global costs of aircraft maintenance can be formulated as equation 4.7 [13]:

$$C_g = C_i + C_f + C_s + C_{oi} \tag{4.7}$$

Herein C_g is the global cost of maintenance, C_i the cost of intervention, C_f the cost associated to a failure, C_s the cost for storage and C_{oi} the cost of over investment. This equation is simplified by removing the variables C_s and C_{oi} in order to make the model not too complex. The simplified costs are then represented by formula 4.8.

$$C_g = C_i + C_f \tag{4.8}$$

Furthermore the variable C_i can be sub-divided in the components C_{labour}, $C_{material}$ and C_{repair}. According to Ackert [1] these take up $20-30\%$, $60-70\%$ and $10-20\%$ respectively. Furthermore Batalha [4] listed averages for C_f which are represented in table 4.2.

Costs	Estimates [per thousand $]
$SVC_{g/f}$ Shop visit after on ground / in-flight failure	12-700
CDC_f Potential contingency damage costs	500
Potential (line) inspection costs IC due to a false alarm	0.2
SVC_{FP} potential shop visit costs for a false alarm	2
LCg/f Logistic costs to replace an engine off-base	100-250
$CL_{g/f}$ AOG cost	266-372

Table 4.2: Maintenance cost values based on a study from Batalha [4].

As these values vary greatly, as well as in other studied literature, it is difficult to take a single estimate for the variable C_f in equation 4.8. Secondly although C_i can be divided in its 3 main components the values will still vary for different aircraft, personnel, failures etc. So in an effort to limit the amount of assumptions made, or to be too limited to a specific failure event it was decided to opt for a factor based cost breakdown. This dimensionless factor is simply the relation between C_f and C_i as represented in equation 4.9.

$$C_{factor} = \frac{C_f}{C_i} \tag{4.9}$$

This factor will allow to make the necessary calculations for financial impact of the integrated CBM model by simply evaluating if the outcome is positive or negative. It will also define a range where such a model will yield a positive net benefit. This gives airline operators an indication whether or not a CBM approach will be beneficial for them in a specific use case, provided that the CBM model has similar ROC, precision-recall performance. Finally if all parameters are known to the operator they can simply be plugged in and an estimate for the overall value can be calculated. Section 5.4 will provide additional information regarding the use of these parameters and how they will be used to compute a singular value KPI.

4.4.2. Maintenance failure threshold explanation

Once a mathematical model that is able to forecast future sensor data there is still a need to transform those forecasts in a maintenance decision making tool. As previously explained in subsection 4.2.2 lists the features that will be used to assess system health. This will be done in the following manner. A range of thresholds will be created wherein the height of the threshold will be altered dynamically. This can most easily be described by using figure 4.5. The orange bands dictate the range of tested thresholds for the model. The thresholds heights are varied along these ranges independent from one another creating a 2D map of possible threshold combinations. This is only the case for the model that uses both the upper and lower bound together as the single threshold (upper, lower and deviation) only has a single design parameter. This is visualized in figure 4.6.

Note that in figure 4.5 and 5.2 only the final forecast values are represented by the red dots. In reality the model predicts a path of 12-18 time steps ahead from each individual FC after the necessary time steps have passed to compute the model. This is illustrated in figure 4.7.

As can be seen from figure 4.7 the forecast trajectory crosses the threshold multiple times, whilst the final estimated value still lies above it. The model has been built in such a way that this is taken

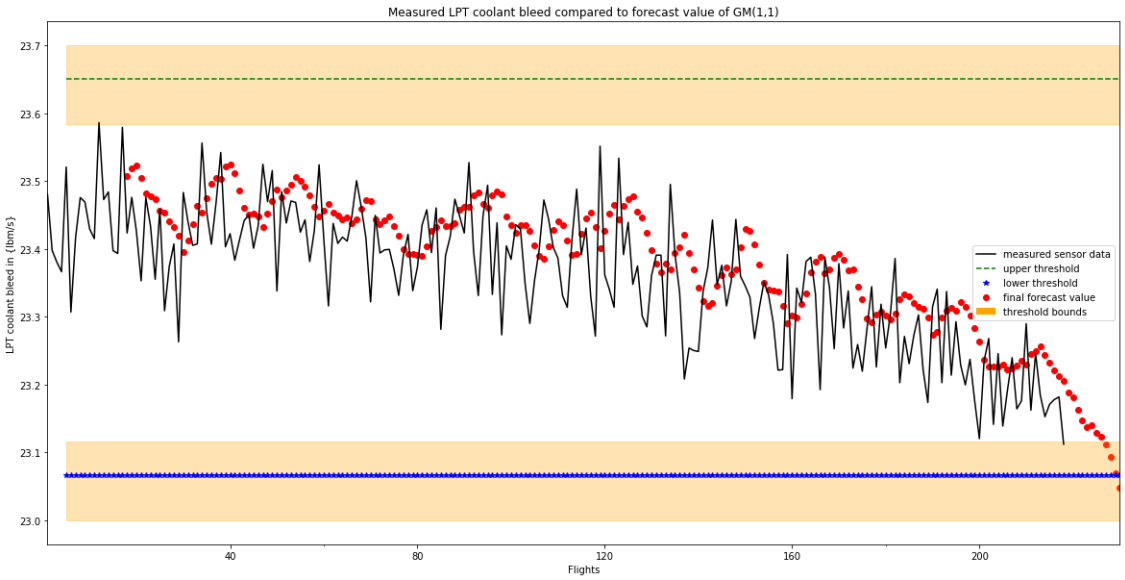


Figure 4.5: A graphical overview threshold configuration. The orange bands are the ranges for the thresholds and the red dots represent the final forecast values computed by the prognostic model.

into account. Once a trajectory crosses the threshold boundary it will declare that a maintenance task needs to be performed.

4.4.3. Model performance metrics

In order to describe the effectiveness of the forecasting model several metrics have to be defined first. In classification models the following metrics are commonly used. These are listed below.

- Model precision $P_{precision}$
- Model recall rate P_{recall}
- True positive (TP) rate TP_{rate}
- True negative (TN) rate TN_{rate}
- False positive (FP) rate FP_{rate}
- False negative (FN) rate FN_{rate}
- Area under receiver operating characteristic $AUROC$
- Prognosis time horizon $T_{prognostic_{horizon}}$
- Maintenance lead time $T_{lead_{maintenance}}$
- Prognosis failure time $T_{failure_{prognostic}}$
- Minimum time before maintenance action can be performed $T_{prognostic_{action}}$
- Actual failure time $T_{failure_{actual}}$
- Mean time between repair $MTBR$

Table 4.3 and equations 4.10 through 4.21 can be used to explain the parameters.

		predicted	
		positive	negative
actual	positive	true positive (TP) (correct maintenance action)	false negative (FN) (unexpected failure)
	negative	false positive (FP) (unnecessary maintenance)	true negative (TN) (correct no maintenance)

Table 4.3: Forecasting errors possible within a CBM policy.

From these classes several computations can be made that can be used to measure the performance of the integrated CBM model [5]. These metrics are presented mathematically in equations 4.10 through 4.21.

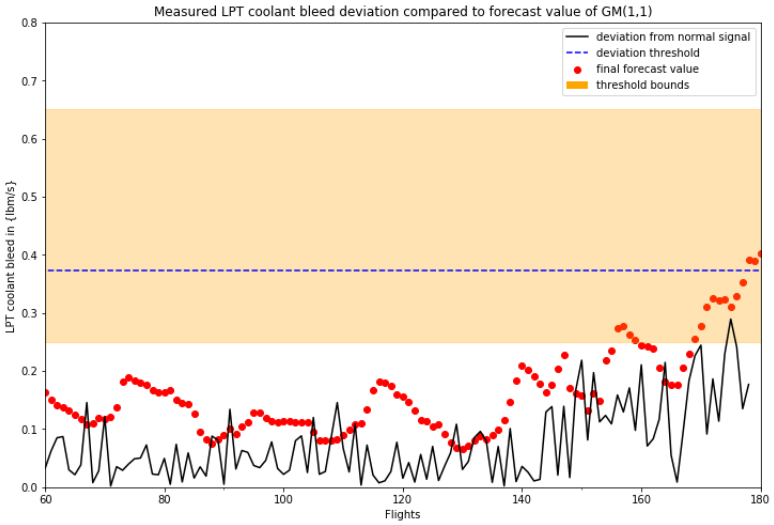


Figure 4.6: System health threshold schematic for LPT coolant bleed signal deviation feature. The orange band depict the range of the failure threshold and the blue line represents a specific selected threshold value.

$$TP \quad if \quad T_{prognostic_{action}} \; T_{failure} \tag{4.10}$$

$$FN \quad if \quad T_{failure} \; T_{prognostic_{action}} \tag{4.11}$$

$$FP \quad if \quad T_{prognostic_{action}} \; T_{failure} + RUL_{tolerance} \tag{4.12}$$

$$TN \quad if \quad no\,action\,and\,no\,failure \tag{4.13}$$

$$TPT_{failure_{prognostic}} \; T_{prognostic_{horizon}} \tag{4.14}$$

$$T_{prognostic_{action}} = max(T_{failure_{prognostic}}, T_{lead_{maintenance}}) \tag{4.15}$$

$$P_{precision} = \frac{TP}{TP + FP} \tag{4.16}$$

$$P_{recall} = \frac{TP}{TP + FN} \tag{4.17}$$

$$TP_{rate} = \frac{TP}{TP + FN} \tag{4.18}$$

$$TN_{rate} = \frac{TN}{TN + FP} \tag{4.19}$$

$$FP_{rate} = \frac{FP}{TN + FP} \tag{4.20}$$

$$FN_{rate} = \frac{FN}{TP + FN} \tag{4.21}$$

Equations 4.10 through 4.13 list the mathematical conditions that need to be true for each class of table 4.3. Furthermore the prognosis time horizon $T_{prognostic_{horizon}}$ is the amount of time wherein a prognostic model can *"predict"* the prognosis failure time $T_{failure_{prognostic}}$, depicted by equation 4.14. The $T_{prognostic_{horizon}}$ needs to be sufficiently large otherwise the maintenance lead time $T_{lead_{maintenance}}$

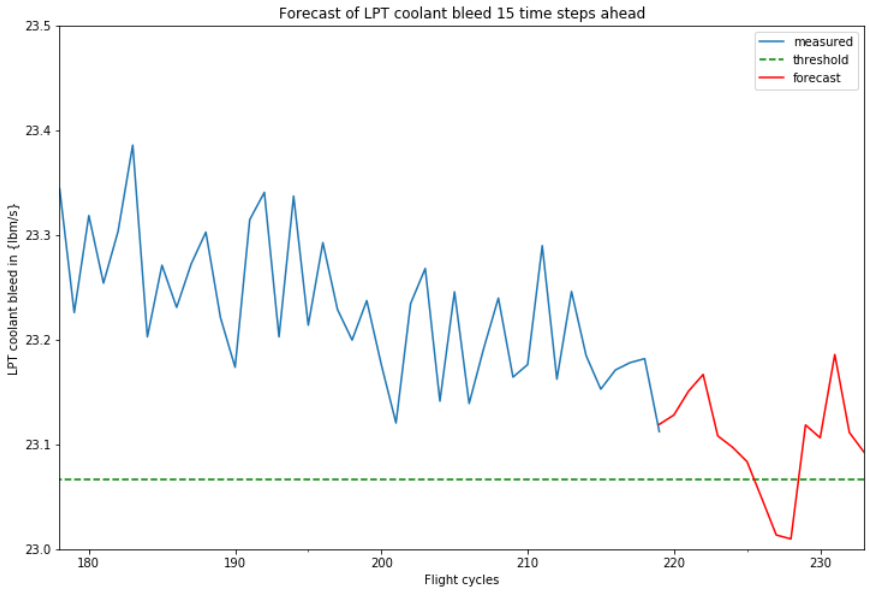


Figure 4.7: A single forecast trajectory. Note that the trajectory does cross the threshold whilst finishing above it.

will exceed the $T_{failure_{prognostic}}$. Resulting in an inability to make the adjustments necessary within the maintenance schedule to prevent the impending failure and nullifying the effect of the prognostic model. This is because the minimum time before a maintenance action can be performed is defined by equation 4.15. Wherein the maintenance lead time can be explained as the time required to acquire the necessary resources for the maintenance action. Finally the actual failure time is necessary to determine if the maintenance action is indeed successful or not. From these parameters the probabilities of $P_{success}$ and $P_{failure}$ can be determined.

The model success rate $P_{success}$ is the probability that the correct maintenance action is performed. Meaning that a part is maintained/replaced before failure (TP) or the part is not maintained and does not fail (TN). The model error rate P_{error} is 1 minus that probability (ie. the chance of a failure before maintenance took place or a unnecessary maintenance action). Finally the occurrence rates from all classes from table 4.3 are defined by equations 4.18 through 4.21.

Although the outcome of these parameters do not give a direct underlying value for a CBM system they can be very useful in interpreting the current status of the implemented framework. For instance if P_{error} is high then either the maintenance action lead time $T_{lead_{maintenance}}$ has to be decreased, the prognostic failure time, $T_{failure_{prognostic}}$, overestimates the actual failure time and has to be decreased as well or the prognostic time horizon, $T_{prognostic_{horizon}}$, needs to be extended.

Finally the area under receiver operating characteristic (AUROC) metric can also be computed by adjusting the parameters for $T_{prognostic_{horizon}}$ and the reference component condition threshold after which a component is replaced. The AUROC is defined by the area under the curve of the TP_{rate} set against the FP_{rate}. It thus gives a measure on how well the model is able to classify between the classes depicted in table 4.3. Figure 4.8 illustrates the definition of the AUROC [20].

It is also interesting to note that if a particular operator sets a limit to maximum allowable false positive rates the AUROC curve can also be interpreted differently. That is the area until that limit on the X-axis has to be maximized instead.

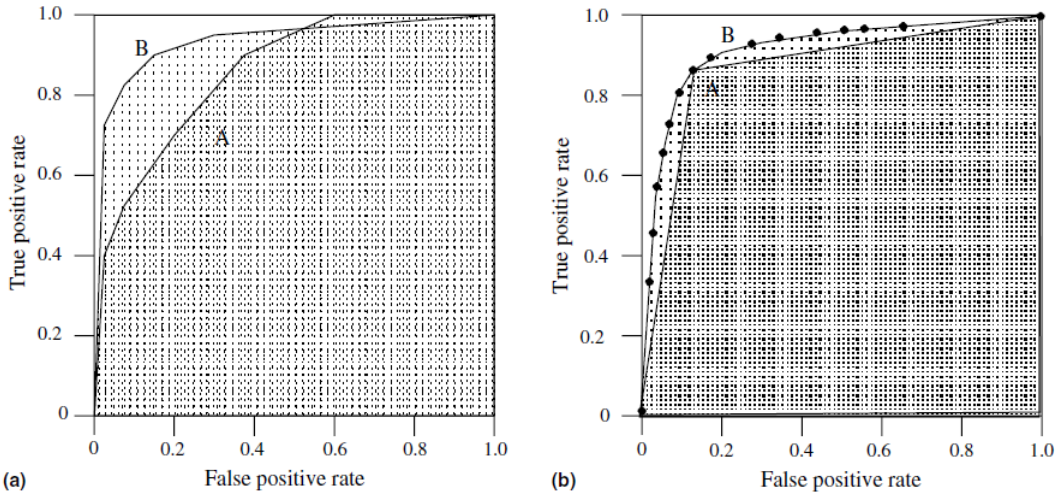


Figure 4.8: Two ROC graphs. The graph on the left shows the area under two ROC curves. The graph on the right shows the area under the curves of a discrete classifier (a) and a probabilistic classifier (b). [20].

4.5. Cost evaluation module of the integrated CBM framework

The final step necessary to complete the integrated CBM framework is the maintenance cost evaluation, i.e. the net financial impact of incorporating the steps mentioned in the previous section. From the domain knowledge acquired and listed in subsection 4.4.1 and table 4.2 it was concluded that there exist a large deviation in maintenance cost associated with unexpected engine failures. As such the maintenance cost evaluation formulation has been kept simple on purpose.

In order to make a simple formulation of the cost of the CBM model the following parameters are necessary: the FN rate, recall and $MTBR$. The FN rate and recall are given in equations 4.11 and 4.17 respectively. This leaves the $MTBR's$ to still be formulated as this parameter will account for the frequency of maintenance actions and replacements/depreciation of the components in question, which will have a direct effect on the associated costs.

The $MTBR$ depicts the amount of FC between each executed maintenance action. As such it can be formulated as equation 4.22. It should be noted that as the data comprises of a set amount of run-to-failure simulations the total amount of maintenance task actions is also set. The number of FC's that occur in total is thus the only variable parameter in this equation. This value will be largely dependent on the height, and thus the level of conservatism, set for the failure threshold.

$$MTBR = \frac{FC_{total}}{\#\ maintenance\ tasks} \tag{4.22}$$

As mentioned before it should be noted that the formulation is kept simple by design to limit the amount of uncertainty added. A more complex formula is only better if the underlying parameters that are used to create such a model can be determined with certainty. In order to keep the emphasis of this study on the difference in performance of the various optimization strategies modeled it is not required to develop a too complex model. This will keep the scope within reach for his project and by keeping a simpler model it will also enable a clearer analysis of the differences between strategies.

The cost per engine can be divided in three components: $C_{failure}$, C_{repair} and $C_{component}$. Herein $C_{failure}$ is the cost associated by an unscheduled failure occurrence, C_{repair} is the cost of the maintenance task itself and $C_{component}$ is the cost of the component associated with the acquisition of a new component. These parameters are slightly altered forms from the ones described in subsection 4.4.1. The mathematical description is represented in equations 4.23, 4.24 and 4.25.

$$C_{action} = C_{labor} + C_{repair} \tag{4.23}$$

$$C_{unexp.\ maintenance} = C_{failure} * FN_{rate} \tag{4.24}$$

$$C_{depreciation} = \frac{C_{component}}{MTBR} \tag{4.25}$$

Note that for equations 4.23 and 4.24 this is the average cost associated for each maintenance task and equation 4.25 is the cost per FC. Equation 4.25 has to be multiplied by $MTBR$ to get the the cost per repair as well. Combining these elements gives equations 4.26 and 4.27. The FN_{rate} factor is not present in the run-to-failure case as each engine has failed before it will be replaced.

$$\frac{C_{total}}{repair}_{CBM} = C_{action} + FN_{rate} * C_{failure} + C_{depreciation} = C_{repair} + C_{failure} * FN_{rate} + C_{component} \tag{4.26}$$

$$\frac{C_{total}}{repair}_{run-to-fail.} = C_{action} + C_{failure} + C_{depreciation} = C_{repair} + C_{failure} * FN_{rate} + C_{component} \tag{4.27}$$

These formula's have to be transformed to the overall total cost by multiplying by the total amount of FC of each simulated aircraft divided by the $MTBR$ to get the amount of repairs that will be performed under the different maintenance paradigms. The final equations can thus be stated as follows:

$$C_{total_{CBM}} = \frac{\#FC}{MTBR_{CBM}} * (C_{action} + FN_{rate} * C_{failure} + C_{component}) \tag{4.28}$$

$$C_{total_{run-to-failure}} = \frac{\#FC}{MTBR_{run-to-failure}} * (C_{action} + C_{failure} + C_{component}) \tag{4.29}$$

The main difference between the two formula's lie in the absence of the FN_{rate} in the run-to-failure model. As the FN_{rate} lies between 0 and 1 by definition the inclusion of this factor in the CBM cost formula results in a cost reduction. The $MTBR$ on the other hand will be shorter for the CBM model compared to a run-to-failure maintenance strategy. The relation of the magnitudes of the individual cost components, the FN_rate and the $MTBR_{CBM}$ compared to $MTBR_{run-to-failure}$ will determine the value of using an integrated CBM model and can be described by equation 4.30.

$$\begin{aligned} Value_{model} =& C_{total_{run-to-failure}} - C_{total_{CBM}} = \\ & \frac{\#FC}{MTBR_{run-to-failure}} * (C_{action} + C_{failure} + C_{component}) \\ & - \frac{\#FC}{MTBR_{CBM}} * (C_{action} + FN_{rate} * C_{failure} + C_{component}) \end{aligned} \tag{4.30}$$

As mentioned in subsection 4.4.1 it is difficult to define exact values for these parameters. A factor based approach is therefore performed. Herein $C_{component}$ and C_{repair} are combined to a single parameter C_i. Furthermore the number of FC are also excluded from the equation to get a normalized value. The equation then simplifies to 4.31.

$$\begin{aligned} Value_{model_{normalized}} =& \frac{1}{MTBR_{run-to-failure}} * (C_i + C_{failure}) \\ & - \frac{1}{MTBR_{CBM}} * (C_i + FN_{rate} * C_{failure}) \end{aligned} \tag{4.31}$$

$$\begin{aligned} Value_{relative} =& \frac{Value_{model}}{C_{total_{run-to-failure}}} = \frac{C_{total_{run-to-failure}} - C_{total_{CBM}}}{C_{total_{run-to-failure}}} = \\ & \frac{\frac{(C_i + C_f)}{MTBR_{run-to-failure}} - \frac{(C_i + FN_{rate} * C_f)}{MTBR_{CBM}}}{\frac{(C_i + C_f)}{MTBR_{run-to-failure}}} \end{aligned} \tag{4.32}$$

All the steps from figure 3.1 have now been described and as such the next chapter, chapter 5, will be dedicated to the results obtained from the integrated system.

5
Results

This chapter will be dedicated to the evaluation of the performance of the integrated CBM model. This will be done per segment. These segments are: the prognostic model forecast accuracy, CBM policy classification score (AUROC & precision-recall) and the overall value of the integrated system on a monetary basis.

It is important to note that for the forecast accuracy, ROC and precision-recall and financial value analysis different optimal model configurations are used. This is to illustrate the fact that an local optimum in model configuration for the prognostic accuracy and CBM metrics will not necessarily translate to a global optimum from a financial perspective. Moreover it can be seen that by integrating the two a new configuration set will emerge as the optimal model configuration setting. This will be highlighted in more detail in chapter 7 as well.

To assess the accuracy of the prognostic model the same multiple accuracy metrics used in subsection 4.2.1 will be used. These results will be presented and discussed in section 5.1. Next the performance of the CBM model will be treated in section 5.2. This will primarily be done by evaluating the AUROC curve and the precision-recall curve. Thereafter a look on the financial impact of incorporating a CBM policy over a run-to-failure paradigm will be performed in section 5.4. Finally some intermediate conclusions that combines all findings from these results in section 5.5.

5.1. Prognostic model results

It is important to know how well the prognostic model, GM(1,1) in this case, is able to estimate future values. Not only because any occurring trend will be captured but also to give additional insight on the importance of fluctuations of the measured sensor data. If for instance an increase in in value relates to, and even causes, system health degradation, engineers can then improve the design to limit it from happening in the first place. This is also one of the benefits over a pure "*Black-box*" model.

Furthermore accuracy metrics by itself might hold some information, but it is even more insightful when those metrics are compared between different prognostic models on the same data. This gives not only an overall performance idea, but also a relative indication as well. In the case that the data is "noisy" by nature it might be extremely hard to get a certain forecast accuracy, but if the GM(1,1) outscores different other models it can still be concluded to be superior for this use case.

Therefore subsection 5.1.1 will treat the absolute and relative performance. The robustness of the GM(1,1) will also be tested. This will be done by comparing the forecast accuracy of the GM(1,1) on the NASA PHM data with data obtained from KLM. The data from KLM will also be tested on noise so the effect of this noise can also be investigated. This will be done in subsection 5.1.2. It also gives an additional validity check of the prognostic model which will be treated later on in this report.

5.1.1. Prognostic forecast model accuracy

It has been hypothesized that deviations in the LPT coolant bleed can be used to predict future failures. However to account for maintenance lead-times one has a minimum required time ahead necessary for those predictions. Figure 5.1 by Hess et al. [16] illustrates the importance of this. Ideally one would have an expected RUL as far ahead as possible but this does decrease the model accuracy. Some models

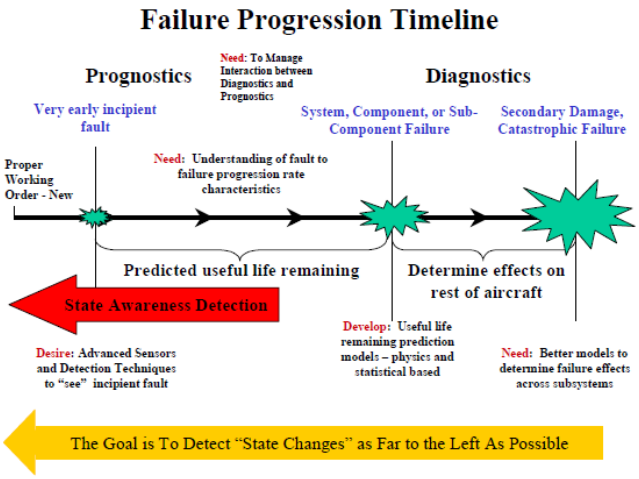


Figure 5.1: Failure progression timeline [16].

GM(1,1) forecast error on raw signal compared to other forecast models

Accuracy metric	GM(1,1)	Naive forecast	ARMA model
Theil's U statistic	0.00149 (0.00147, 0.00152)	0.00222 (0.00217, 0.00226)	0.00152
RMSE	0.0713 (0.0693, 0.0733)	0.103 (0.101, 0.105)	0.0740
MAPE	0.243 % (0.239, 0.248%)	0.355 % (0.348, 0.363%)	0.246 %

Table 5.1: Multiple prognostic forecast error metrics comparison of GM(1,1), ARMA model and naive forecast (including 95 % confidence interval).

are more suitable to predict further ahead and it is therefore important to have context of the accuracy numbers (ie. different model accuracy comparison). Table 5.1 gives this context. Herein the same underlying data has been subjected to different prognostic models (the GM(1,1), an auto-regressive moving average model and the naive model) to compare the forecasting accuracy performance across those different forecasting algorithms.

When computing the same forecast accuracy metrics for the signal deviation feature there are surprising results. These can be found in table 5.2. The data from table 5.2 also shows that the model does not accurately predict future values for the signal. When the forecasts are plotted it can be seen where the problem lies. Figure 5.2 shows this clearly. There are instances that the prognostic algorithm is unstable and and has large prediction errors. It seems from first analysis that this decreases once an upward trend in the signal becomes prevalent. It can thus be hypothesized that for oscillating small absolute values the model portrays erratic behaviour. The detailed reasoning for this will not be dealt with in more detail as this is not the focus of this study. Furthermore a mitigation to this problem that allows for a similar, but not identical feature to be used instead has already been implemented. That is by using both a lower and upper threshold simultaneously one still can detect both positive and negative deviations from the reference signal.

It should be noted that no 95% confidence window for the accuracy metrics was listed in table 5.2 as there was a huge difference in accuracy between different simulations due to the erratic nature of the forecasts. The most accuracy metric of the most "stable" simulation was presented instead to still give a measure of the incapability of prognoses of this feature. Furthermore the larger difference of the relative error metrics (MAPE and Theil's U) versus the absolute error metric (RMSE) can be attributed to the fact that the values of the underlying data are smaller in magnitude in general. This highlights the importance of using both relative and absolute error metrics in conjunction as stated earlier.

Note that there is also no confidence window provided for the ARMA model. This is due to the fact that the ARMA model behaves highly unstable and there are therefore very frequent forecast outliers. The number presented is not the mean of the error but the lowest accuracy for any simulation. The following can therefore be concluded about using a (rolling) ARMA model with a forecast horizon of

GM(1,1) forecast error on deviation feature

Accuracy metric	GM(1,1)
Theil's U statistic	0.223
RMSE	0.081
MAPE	49 %

Table 5.2: Prognostic forecast error metrics of GM(1,1) on the signal deviation data (including 95 % confidence interval).

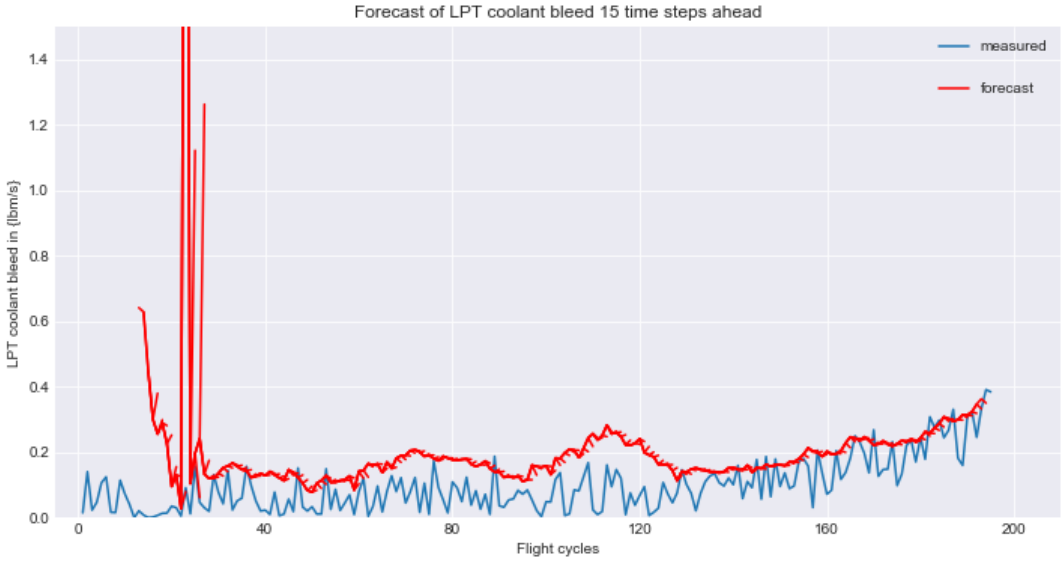


Figure 5.2: GM(1,1) forecast plotted against the deviation of the reference signal. Note the instability when the reference signal has small values.

15 time steps. The unstable character of the ARMA model will trigger early replacements when they are not needed. And even in the best case scenario the overall forecast accuracy is lower than the GM. Suffice it to say that it is not advisable of using an ARMA model for this specific use case.

When the final forecast values are plotted, such as in figure 5.3, it can be seen that the model dampens the input signal. This is due to the intrinsic property of a GM(1,1) as described in section 2.4. This dampening effect becomes greater when *alpha* is increased. Although this dampening deals with the noise of the signal it also caused a delay in the forecast. It can be seen that the trends in the data are captured but the future estimates trail behind. This delay caused by a high *alpha* can be "cured" by extending the forecast horizon, as with a further forecast any trends will be amplified. However a further forecast horizon will also relate to a lower forecast accuracy negating the damping effects. A balance has therefore be found that finds the optimum for those parameters. These adjustable parameters of the GM(1,1) are listed in table 5.3.

Optimal configuration w.r.t. prognostic model forecast accuracy

prognostic model parameters	values
GM(1,1) alpha	1.1
# historic data points	18
forecast horizon	13

Table 5.3: Prognostic model configuration setting to reach maximum forecast accuracy based on multiple error metrics.

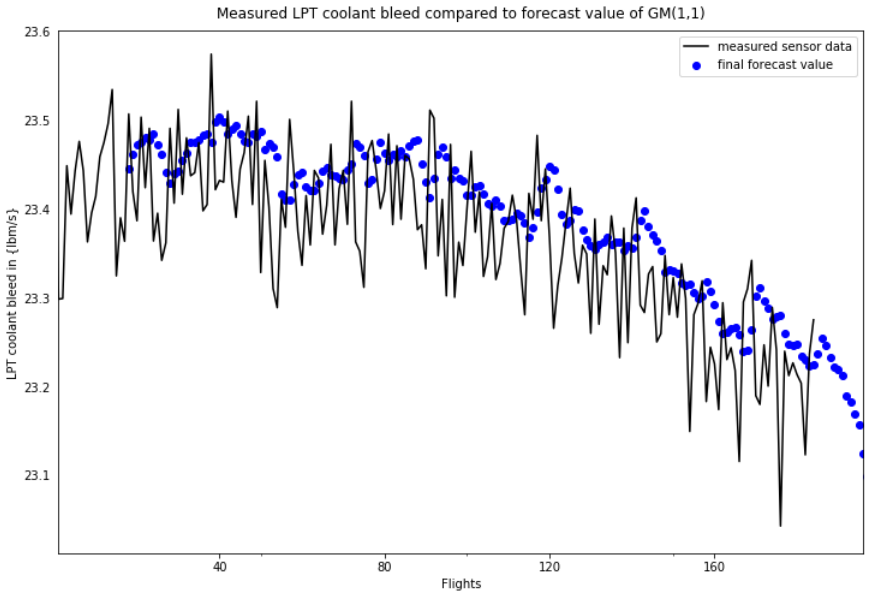


Figure 5.3: Final forecast values plotted against the reference signal. Note that the forecast model has a dampening effect but also suffers from lag.

GM(1,1) forecast error comparisons between different data sets

Accuracy metric	NASA PHM data (entropy @ 1.84)	KLM data (entropy @ 2.02)	KLM data (entropy @ 2.15)	KLM data (entropy @2.58)
Theil's U statistic	0.00149	0.00443	0.0155	0.0314
RMSE	0.0713	0.878	2.77	4.55
MAPE	0.243 %	0.755%	2.56%	4.86%

Table 5.4: Forecast error of GM(1,1) of the NASA PHM data set compared to the KLM data with varying noise levels.

5.1.2. Relation between signal noise and forecast accuracy

In order to test the robustness of the GM(1,1) it has also been used on a different set of data and the same forecast accuracy metrics are analyzed. The data in question is continuous parameter logging (CPL) data from the 747 bleed air system provided by KLM. Since this data is logged continuous instead of only having 1 data entry per FC, as is the case for the NASA PHM data, it was compressed to the same format. The data from a variety of sensors was then analysed on its levels of noise by calculating their entropy in order to make an informed comparison. Table 5.4 lists the performance of the GM(1,1) on the NASA data set compared to the KLM data of various sensors with their corresponding entropy levels.

It can be concluded that the GM(1,1) has more difficulty to track the KLM data than anticipated. Although the MAPE is still under 1% for the KLM data that has the least entropy it is still proving to be less accurate. In an effort to find the reason for this discrepancy a study performed by Costa et al. [9] showed that for shorter time series the entropy level decreases. It can therefore be assumed that noise is relatively more prevalent in the KLM data, visual inspection does conform this also.

The results of this analysis does confirm the previous statement that the GM(1,1) has a better forecast accuracy for lower entropy values. The decision for the LPT coolant bleed as the reference signal for the model is therefore strengthened.

5.2. Condition based maintenance decision model results

Following the steps from figure 3.1 the next phase of the integrated CBM model is to use the prognostic forecasts in a system health model. This model, that has been described in section 4.4, will be analysed on several metrics. Most notable the ROC curve, precision/recall curve and the monetary value will be investigated. The results from the different features (lower threshold, upper threshold, lower and upper threshold and deviation) will all be treated. These metrics can then be used to get a clear overview of the classification capabilities of the a GM(1,1) regression model to assess system health. First the ROC curves will be presented in section 5.2.1. Then subsection 5.2.2 will present the precision/recall curves. This performance metric gives vital context to the ROC curves from the aforementioned section. In practice it has little significance if a model can have "good ROC" properties if the recall is poor. In that case it only "cherry picks" a few instances correctly but has no impact in the grand scheme of things.

5.2.1. ROC performance analysis

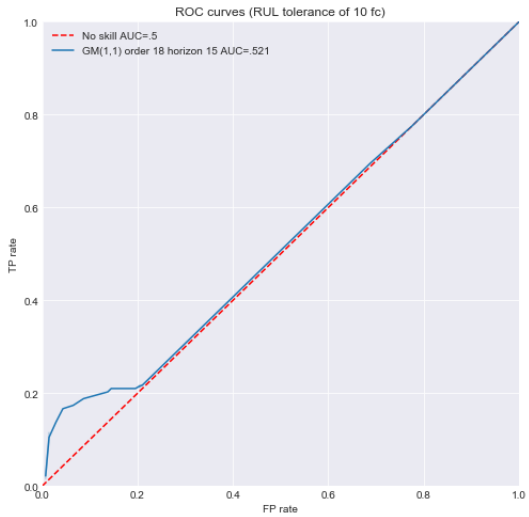
To evaluate the performance of the CBM framework introduced in subsection 4.4.2 ROC curves will be used. These give an insight on the balance between TP_{rate} vs. FP_{rate}. It can be seen that to increase the TP_{rate} beyond a certain point the number of false positives rises considerably. Note that the curve represents a whole set of model configurations and that a particular design point can be selected that has an optimal balance between TP_{rate} and FP_{rate}.

The area under the curve can also be calculated to give the $AUROC$ score. This is a metric that encapsulates the whole figure in one single numeric value. This can be used to easily compare different model configuration to one another, however it could be suggested that the model configuration with the highest $AUROC$ value will not always be the most desirable. This can be attributed to the fact that recall might still be low with a high $AUROC$ model configuration, which will be expanded upon in sections 5.2.2 and 7. Furthermore one can impose limits on maximum false positives rate which skews the results also. Figure 5.4 gives the performance of the CBM framework for the different failure features introduced in subsection 4.4.2.

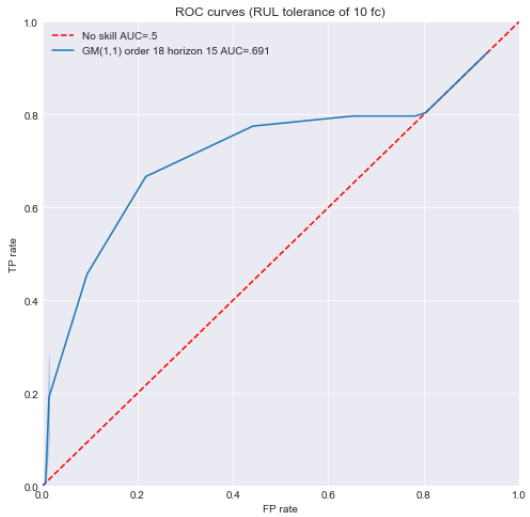
It can be concluded that using a single threshold, either upper or lower, as a health assessment fails to recognize all imminent failures. This is due to the fact that deterioration will not result in only upward or downward trends of the LPT coolant bleed signal but in one of either. This is especially the case for the data that simulates both HPC and fan degradation. As such a combined upper and lower threshold ensures that both deteriorating effects are captured. However as seen by the analysis from 5.1 the GM(1,1) behaves highly erratic for this feature input data. It is therefore unsuitable. Even if there was a method in making the model more stable there is still a downside of using this feature. This is because the deviation feature does not discriminate between an upward or downward deviation, but only an absolute one. In reality it could be that a downward deterioration relates to a lower system health and will fail sooner than when degradation caused an upward trend of the LPT signal. By splitting the two and dynamically adjusting both simultaneously one gets a two dimensional threshold setting that can be optimized. This leads to the best performance overall.

It must be stated that there are potentially other features that can also be used beside the aforementioned 4, but to limit the scope of this aspect of the research design constraints had to be imposed.

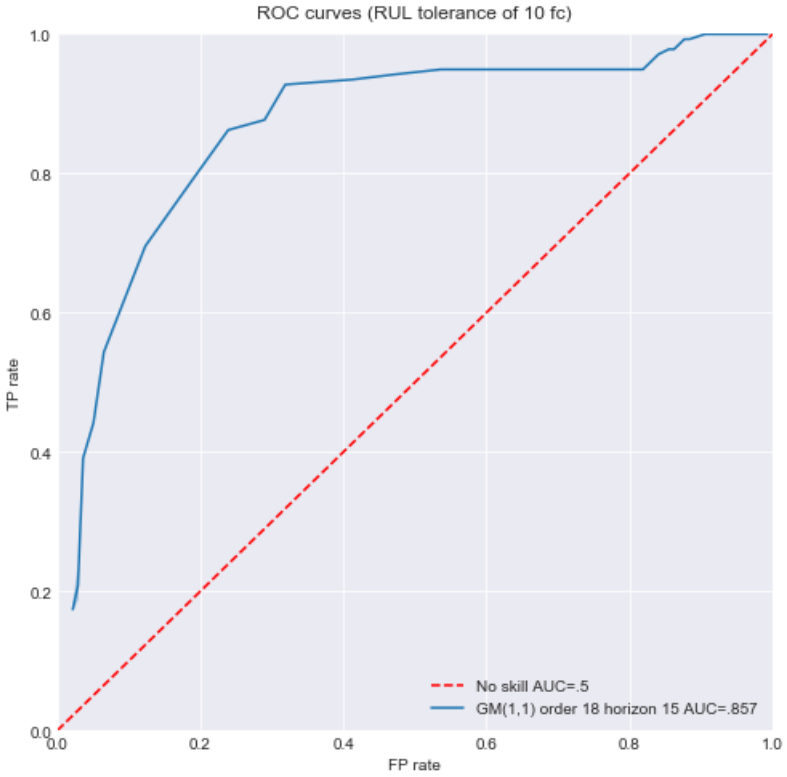
Furthermore when the RUL tolerance is increased the $AUROC$ score increases as well. This can be attributed to the fact that a classification of false positive is less likely to occur. It was also found that the sensitivity of the threshold adjustments has a large effect on the overall performance of the model. If the steps between threshold values are too large the model fails to find the sweet spot where failures are just about to happen. By increasing the RUL tolerance this sensitivity is reduced. This will be treated in further detail in chapter 7. Finally by having model that is too finely tuned, that is to say with very small increments in sensor thresholds and with a low RUL tolerance window, it is prone to over fitting. This can also be attributed to the fact that the model is capable of finding the small sweet spot of just-in-time failure recognition, but only on the training data. This will be expanded upon in section 6.2.



(a) ROC curve for the CBM model with upper threshold only (alpha = 0.9, horizon = 15)



(b) ROC curve for the CBM model with lower threshold only (alpha = 0.9, horizon = 15)



(c) ROC curve for the CBM model with upper and lower threshold in place (alpha = 0.9, horizon = 15)

Figure 5.4: ROC curves for the different analyzed features. Note the improvement of the area under the curve for the cases where both an upper and lower limit are set.

Optimal configuration w.r.t. ROC curve	
prognostic model parameters	
GM(1,1) alpha	0.9
# historic data points	18
forecast horizon	15

Table 5.5: Prognostic model configuration optimized for maximum AUROC value.

5.2.2. Precision-recall curve analysis

A similar trend can be noticed when analyzing the precision-recall curves as in the previous section. Namely that the model performs best when both an upper and lower threshold are combined. Figure 5.5b also gives the context that explains why the one sided thresholds are not suitable for detection system health degradation for the investigated failure modes. It can be seen that whilst the accuracy of the model tends to be quite high for the single threshold scenario, a maximum accuracy of 0.85 at a recall of , there is a distinct drop-off for higher recall values.

When comparing figures 5.5a with 5.5b it is clear that the precision recall curve for the lower threshold only has a much sharper shape. This relates to a more narrow defined threshold where failures are prone to happen. Although it can be seen from the figure that using only this threshold will not yield a well performing model it does give the valuable insight that a lowering of the LPT coolant bleed flow relates to a poorer system health state than an increase. This is one of the benefits of not using a completely "black-box" system.

When the two threshold boundaries are combined it can be seen that the model is able to catch 75% of the impending failures with a 75% precision. Although there are still failures that are not detected a large portion of unscheduled maintenance can still be avoided. Note that this graph is computed using a RUL tolerance window of only 10 FC. The $MTBR$ will therefore not be dramatically influenced for high precision values. Once the precision drops it will automatically mean that the model will infer a failure at a too soon point in the life-cycle of the component and will result in a higher overall cost for the operator. This will be treated in section 5.3 in more detail.

Optimal configuration w.r.t. precision-recall curve	
prognostic model parameters	
GM(1,1) alpha	0.7
# historic data points	18
forecast horizon	13

Table 5.6: Prognostic model configuration optimized for maximum area under the precision-recall curve.

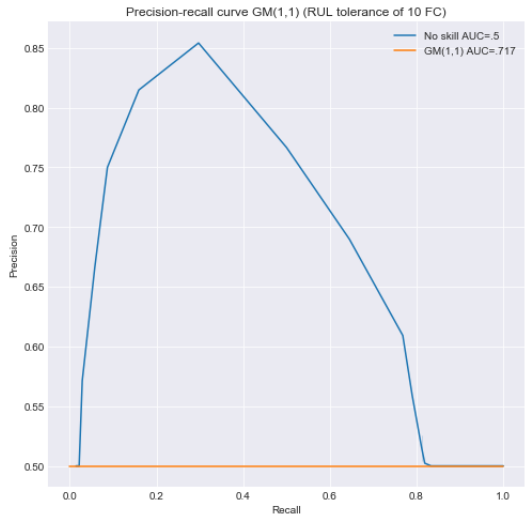
5.2.3. $MTBR$ versus recall

From an operational perspective it is interesting to investigate the impact on the $MTBR$ of incorporating a CBM over a run-to-failure paradigm. The useful life will be lower when replacements are performed before failure can occur, but to what extent and at what recall. This section will describe this in detail. Figure 5.6 shows this relation.

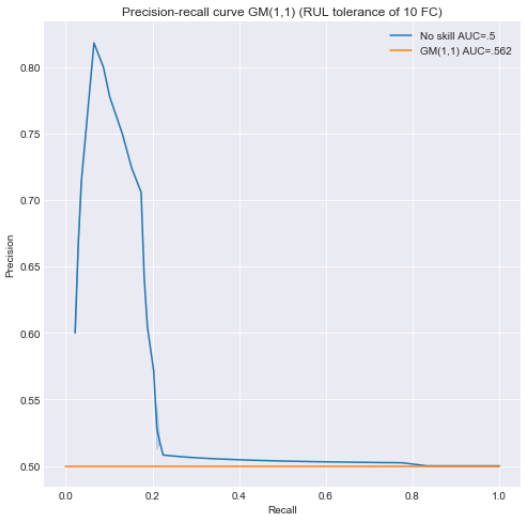
When analyzing figure 5.6 it can clearly be seen that for increasing recall values the $MTBR$ goes down. This is due to the fact that a more conservative replacement strategy is being incorporated. This will result in more failures that are being prevented, but the failures that were already predicted will now be issued at an earlier timestamp. It therefore is important to note that a balance has to be found between the two. Where the optimum lies relies on the value of preventing failures over a too soon replacement policy.

Furthermore it must be stated that there are also secondary effects that can come in effect when the $MTBR$ becomes increasingly short. It will become harder to plan all the necessary maintenance actions in a tight airline operating schedule. It will also relate to a higher throughput of components which have to be in inventory.

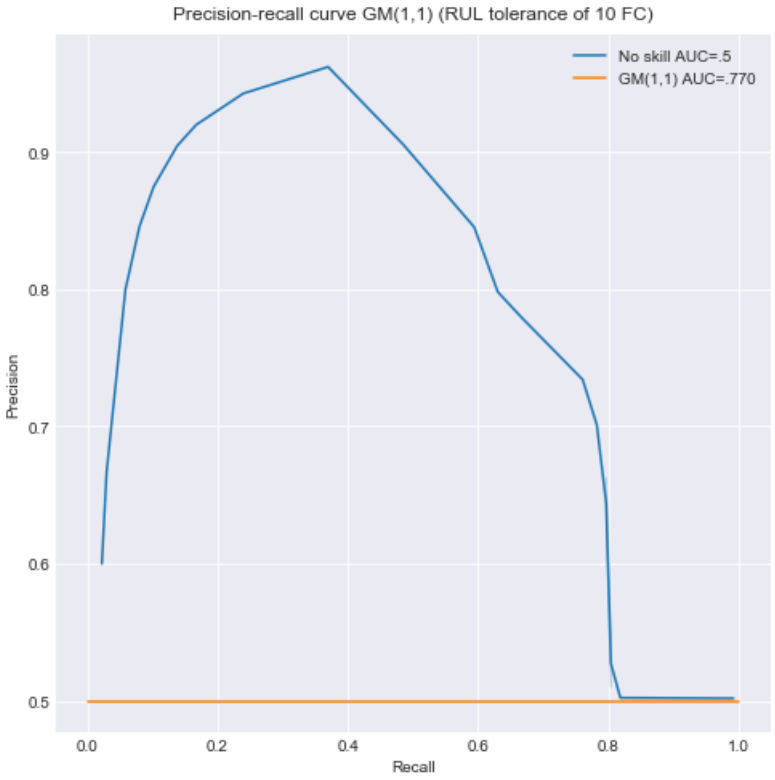
The next section, where the financial optimal strategy will be computed, will evaluate where the balance of $MTBR$ vs. recall lies. As can be seen from that section the change in $MTBR$ will remain under 10% for recall values up to 80% (which is around the optimum recall for a cost factor of 1 in 5).



(a) Precision-recall curve for the CBM model with lower threshold only (alpha = 0.7, horizon = 13)



(b) Precision-recall curve for the CBM model with upper threshold only (alpha = 0.7, horizon = 13)



(c) Precision-recall curve for the CBM model with upper and lower threshold in place (alpha = 0.7, horizon = 13)

Figure 5.5: Precision-recall curves for the different analyzed features. Note the improvement of the area under the curve for the cases where both an upper and lower limit are set.

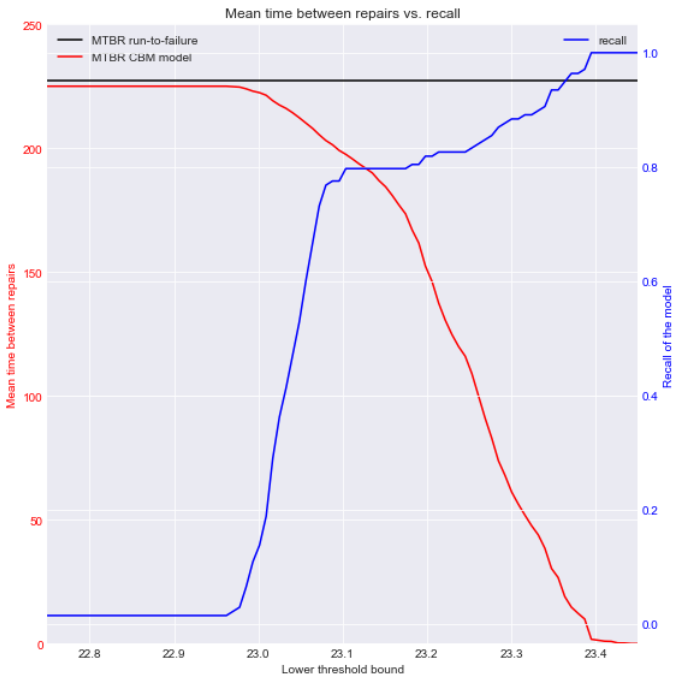


Figure 5.6: Relation between mean time between repair for increasing recall.

It will be assumed that this change will not need radical reorganization of maintenance policy.

5.3. Maintenance cost reduction when comparing CBM with run-to-failure policy

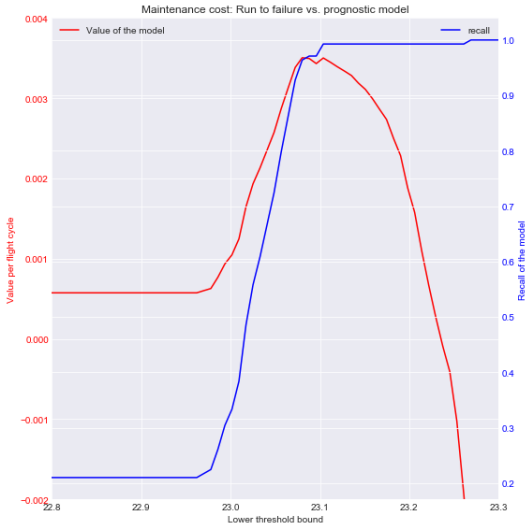
The capability of the model to make accurate identifications of imminent failures has been described by the previous two sections. However the economics impact of the model have to be analyzed as well. Without a financial gain to be had the implementation of this model will not be as desirable for an airline operator. The main contributor for this financial value comes down to the relation of the FN_{rate} and the $MTBR$ as illustrated in equations 4.28, 4.29 and 4.30 from section 5.4.

As previously described it is difficult to ascertain the exact values that can be inserted in those equations. Costs may vary from airline to airline and will surely differ from aircraft to aircraft as component will vary as well. It is therefore more practical to compute the economic value in a more abstract manner. This has been done by using the several cost breakdown factors mentioned in 4.4.1.

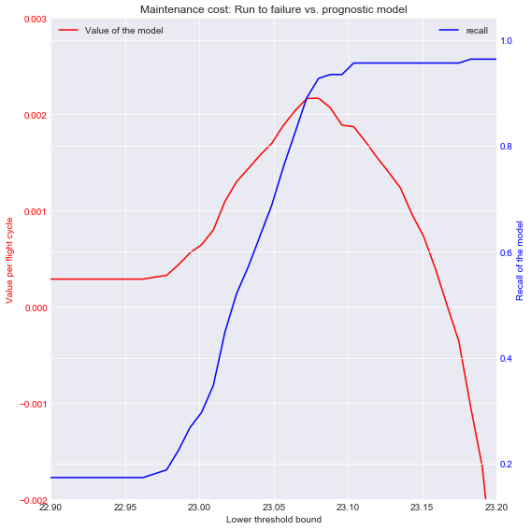
It can be seen that for a maintenance cost factor of 1 the model will have the best financial value. This is to be expected as in this case preventing unscheduled failures will result in the most cost saving. One thing to note from all the figures is that there is a very sharp drop-off where the model will become financially unfeasible.

It should also be noted that changing the RUL tolerance value (standard of 10 FC) has no effect on the performance of the model with respect to the financial evaluation. As the optimum scenario and configuration relies on the FN_{rate} in combination with the $MTBR$ and not on the area of the ROC, precision-recall curve or prognostic accuracy metrics. As such a higher RUL tolerance window, which will improve the ROC and precision-recall performance of the model, it will result in lower $MTBR$ values leading to more expenses. The impact of the RUL tolerance value on the difference between optimal configuration from a ROC and precision-recall perspective compared to the financial based assessment will be treated in more detail in sections 5.4 and 7.

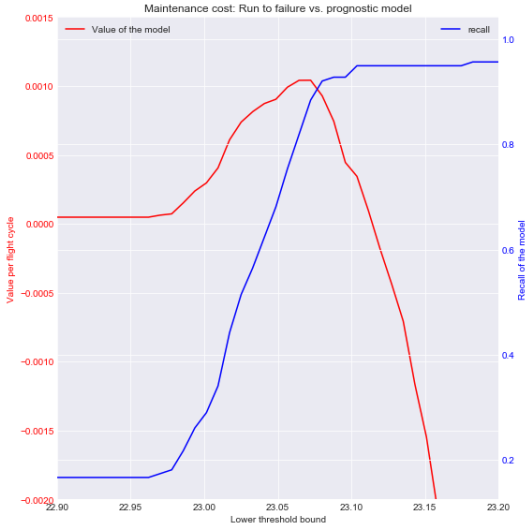
Although tables 5.7, 5.8 and figure 5.7 gives insight on the financial value at specific cost factors C_f they do not provide information on the performance of the model at intermediate cost factors. Figures 5.8a and 5.8b illustrate the effect the cost factor C_f has on the impact of the CBM system by directly plotting the maximum cost saving for all these intermediate values of C_f. Notice that there are two lines present in each figure. The red line shows the model performance for static threshold



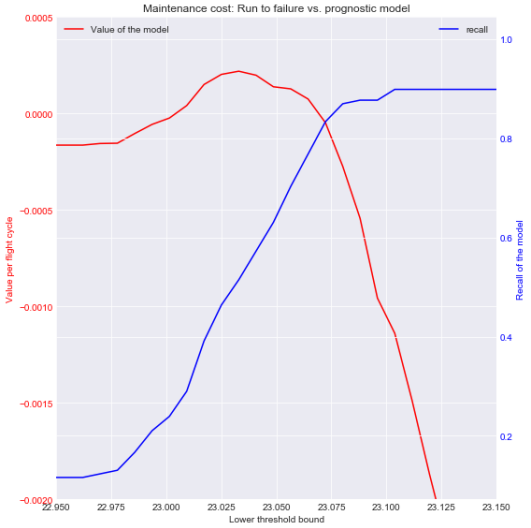
(a) Maintenance cost factor is 1.



(b) Maintenance cost factor is $\frac{1}{3}$.



(c) Maintenance cost factor is $\frac{1}{5}$.



(d) Maintenance cost factor is $\frac{1}{7}$.

Figure 5.7: Net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors.

Optimal configuration w.r.t. financial value

prognostic model parameters			Threshold settings			
			C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
GM(1,1) alpha	0.6	Lower	23.088	23.088	23.064	23.040
# hist. data points	18	upper	23.803	23.831	23.831	23.888
forecast horizon	15	**Norm. value {€ per FC}**	**0.00352**	**0.00220**	**0.00127**	**0.000566**

Table 5.7: Prognostic model configuration optimized for maximum financial impact at various maintenance cost factors.

Relative normalized financial value benefit

	C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
run-to-failure cost {€ per FC}	0.00866	0.0173	0.0260	0.0346
CBM model cost {€ per FC}	0.00514	0.0151	0.0247	0.0340
Relative cost reduction	**40%**	**12.7%**	**4.9%**	**1.7%**

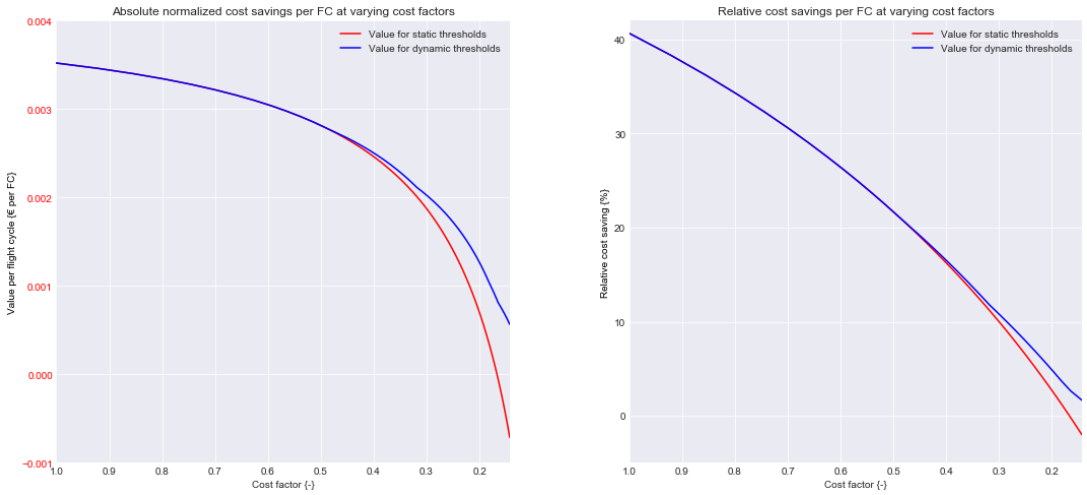
Table 5.8: Relative normalized financial impact at various maintenance cost factors.

settings, whilst the red line shows performance at dynamic threshold settings. The static threshold setting assumes a single threshold value that remains constant throughout changes in C_f. The height of the threshold setting was selected on the basis that it maximized performance at a C_f of 1. The dynamic threshold on the other hand varies the threshold height with cost factor in order to maximize performance throughout the whole cost factor range. As such it is unsurprising that this approach will lead to better performance as the cost factor shift further away from 1.

This can be explained by the following fact. As the cost factor decreases (repair costs increase relative to unscheduled failure costs) it becomes more expensive to adhere to a too conservative maintenance strategy with subsequent conservative failure threshold bounds. These conservative threshold height will reduce the amount of false negatives, but as the $MTBR$ is lower (compared to a less conservative approach) it will still result in higher costs and thus less savings.

Another logical assessment that can be made on the basis of these figures is that the relative cost savings has a different shape when compared to the absolute cost saving. Most notably the relative benefit decreases faster for C_F values close to 1 and slower for small values of C_f. This is to be expected as a similar absolute cost saving will have a lesser relative effect when actual maintenance task costs are increasing (for C_f decreasing). On the other hand when C_f becomes increasingly small a potentially large increase in absolute maintenance cost will have a smaller relative effect.

Note that this is the case with the assumptions in mind that an unexpected failure has a normalized cost equal to 1 and costs associated with a maintenance intervention C_i range between 1 and 7 (giving the range of C_f between 1 and $\frac{1}{7}$ in the figures).



(a) Normalized absolute cost saving.

(b) Relative cost saving.

Figure 5.8: Normalized absolute and relative cost saving for varying cost factors. Note the higher cost savings for the dynamic versus the static threshold heights.

5.4. Cost reduction comparison on the different configuration strategies

When using the optimal settings from each respective module (prognostic model, and CBM model) based on their respective performance metrics and the open-loop scenario (no optimization and standard configuration) there are detectable differences when global cost reduction is concerned. The different parameter configurations are listed in table 5.9. Wherein the most indicative feature of overall performance (normalized value) is highlighted in bold font. This performance metric depicts the normalized cost saving per FC as described in subsection 4.5. The relative cost saving is also present which is also explained in the same subsection. The graphs that correspond to the values listed in this table can also be found in appendix A. Furthermore appendix B contains the graphs that show the cost savings/value for a continuous range of cost factors as well.

Note that after each optimization loop has been performed the optimal threshold values for that specific optimization can vary as well. The threshold values that perform best were all chosen. It should also be noted that although these threshold values show similarities between the respective configuration, there are still some minor differences present. These small variations can be explained by the high level of sensitivity the threshold height has on model performance. This will be further discussed in chapter 7. From figures 5.8a and 5.8b it can also be seen that these small variations do lead to a notable increase in performance, most notably when the cost factor becomes small (relative high repair vs. failure cost see equation 4.9). Furthermore as all simulations use the same 50 step-sizes for threshold heights it is to be expected that there are similar values due to the discrete nature of the threshold boundary values. Finally the RUL tolerance window is 10 FC for all simulations.

It can be seen that there are noticeable differences in the prognostic model configuration that yields the highest performance for each respective CBM segment metric. The different configurations will also lead to a different set of threshold boundaries that are to be used to maximize financial benefit. Furthermore there is a significant difference in the actual value for each respective optimization strategy visualized in appendix C in figure C.1. In that figure all different configurations are plotted next to each other and one can see clearly that the value oriented approach leads to higher cost savings throughout the cost factor spectrum that was investigated. The difference becomes more noticeable as C_f decreases.

Although the difference is in the single percentage points, this is still a significant gain as unexpected maintenance cost can contribute to 47% of total maintenance cost. This coupled with an industry that has low profit margins and major difficulties in the current post COVID-19 era is still a stone that cannot be left unturned to keep public aviation a financially sustainable business.

All these factors emphasize the hypothesis that to maximize the effectiveness of a CBM framework a holistic system approach has to be done.

Optimal configuration w.r.t. forecast accuracy

prognostic model parameters			Threshold settings			
			C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
GM(1,1) alpha	1.1	Lower	23.136	23.112	23.112	23.112
# hist. data points	18	upper	23.774	23.803	23.803	23.860
forecast horizon	15	**Norm. value**	**0.00341**	**0.00199**	**8.96e-04**	**-1.08e-04**
		perc. saving	**39.4%**	**11.5%**	**3.45%**	**-0.312%**

Optimal configuration w.r.t. ROC performance

prognostic model parameters			Threshold settings			
			C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
GM(1,1) alpha	0.9	Lower	23.159	23.112	23.112	23.088
# hist. data points	18	upper	23.774	23.774	23.803	23.831
forecast horizon	15	**Norm. value**	**0.00337**	**0.00201**	**8.41e-04**	**-1.88e-04**
		perc. saving	**38.9%**	**11.6%**	**3.23%**	**-0.543%**

Optimal configuration w.r.t. precision-recall performance

prognostic model parameters			Threshold settings			
			C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
GM(1,1) alpha	0.7	Lower	23.136	23.112	23.088	23.088
# hist. data points	18	upper	23.774	23.774	23.803	23.831
forecast horizon	13	**Norm. value**	**0.00344**	**0.00203**	**9.45e-04**	**2.090e-05**
		perc. saving	**39.7%**	**11.7%**	**3.63%**	**0.060%**

Open-loop configuration

prognostic model parameters			Threshold settings			
			C_{factor} 1	C_{factor} 1/3	C_{factor} 1/5	C_{factor} 1/7
GM(1,1) alpha	0.5	Lower	23.088	23.088	23.064	23.040
# hist. data points	18	upper	23.803	23.831	23.860	23.888
forecast horizon	12	**Norm. value**	**0.00336**	**0.00191**	**7.60e-04**	**2.19e-05**
		perc. saving	**38.8%**	**11.0%**	**2.92%**	**0.063%**

Table 5.9: Prognostic model optimized for each different segment of the integrated CBM model. Note the normalized value and percentage of maintenance cost saved (both highlighted in bold font) over a traditional run-to-failure maintenance paradigm as the indicative performance indicator for each configuration.

5.5. Intermediate conclusions of the results

When comparing the values from the tables listed in the previous sections, table 5.9 in particular, it can clearly be seen that configurations that aim to maximize performance metrics of a specific subsection of the integrated CBM model does not result in maximizing a maintenance cost reduction. This highlights the fact when determining the value of a CBM model a holistic approach has to be done where each of the elements are communicating to each other. The system than has to be optimized on KPI's that supersede individual modules, but reflect a metric that encompasses the whole chain of elements.

Furthermore this proves that the approach found in most of the literature (illustrated in section 2.1 and figure 2.1), wherein only the performance of a single aspect of the whole CBM chain is considered, is not optimal. The hypothesis that a system approach will lead to an increase in performance is therefore valid.

Another point that has to be considered is that threshold height has to be set differently for various cost factor values, as shown most clearly by figure 5.8. Moreover that figure gives a clear insight on the financial feasibility region at various cost factors. As such the relation between cost factor and relative financial impact of the integrated CBM model can be used to quickly assess whether or not a CBM approach is financially viable once the actual cost factor of the respective component of the aircraft is known.

It should be noted that each segment of the integrated CBM module used in this study does not reflect the most advanced method currently available it does show the importance of combining the individual elements. This is also the most prevalent academic novelty of this research. Furthermore

it shows that it is not necessary that each component of the integrated system has been configured to reach maximum performance on its local performance metrics, but that it is more important that an integral approach is being performed.

6

Verification and validation

It is important to ensure that the model proposed in incorporated correctly and that it is also realistic in terms of performance. This can be done by a verification and validation process. During the verification special attention is payed to the mathematical correctness of all the models that were used to create the integrated CBM model. These are: the prognostic model forecasts, CBM performance metrics and cost calculations. This will be treated in section 6.1. Then when it is shown that all metrics and values are correctly acquired it is time to validate the results that they represent. This is done in the validation in section 6.2. In that section special attention will be paid to check whether the trained model does not over-fit the data to a too large degree and thus exaggerates the performance.

6.1. Verification

To verify means: “ensuring that the computer program of the computerized model and its implementation are correct" according to Sargent [25]. As most of the calculations of this study is performed using *Python*, as the number of calculations and complexity is too big to be done by hand, it becomes even more important that results and computations are frequently checked on their correctness. There exist several methods to do such checks and below are some techniques used for verification purposes during this research:

1. Modular model approach
2. Internal consistency checks (inspection)
3. Analytical trace steps
4. Visualisation checks

The most important method used during this research is that the model has been built using several modules on purpose. This is due to the fact that an integrated CBM model consists of multiple facets, as illustrated by figure 2.1. This allows for each segment to be evaluated separately. All variables are also stored on local memory in so called *"Dictionary"* formats. Storing them as a Dict allows the use of informative labels to be attributed to each row/column which makes routine inspections of the data that much easier.

Each module (prognostic model, CBM model and cost evaluation) uses the outcomes from the previous module as its entries. Special checks are therefore put in place after each module where the outputs are visualized, analysed and critically dissected before passed through as inputs for the next module. This method of dividing has lead to discovery of numerous bugs and anomalies which could be easily located and attributed to a single section of the integrated CBM model.

Some more detail of each verification measure at each component of the model will will now be provided. Subsection 6.1.1 will highlight some key methods used during the prognostics of the model. Then in subsection 6.1.2 the verification of the CBM framework will be treated in detail. This includes checks on CBM performance metrics and the value calculations in particular.

6.1.1. Prognostic model

To check whether the GM(1,1) model behaves as intended several verification methods have been applied. These will be described in this section. The first method to verify if the model works as intended was by using a simple set of dummy data that was created with the sole purpose of configuring the prognostic model for future use. The values used in the data were simple by design such that the outcome could be computed analytically.

Another method that has been used is the visualisation of the model forecasts. Figures such as 5.3 are analysed for anomalies. This has been combined with the previous method as well. By visualizing the performance of the model on the basic input data used for testing the model it could be clearly seen that the model showed the required behavioral properties. This method was also used to explain the differences found in the prognostic accuracy results for the ARMA model and the GM(1,1) model on the sensor deviation data compared to the GM(1,1) accuracy results. From the plots, such as figure 5.2, it could clearly be seen what caused the difference. This indicated that the calculations were not wrong, but the model itself just showed unstable behaviour.

Another method used prolifically is inspection. As mentioned earlier the output variables of the forecast model are saved in the local memory. When the forecast values are then used further on in the system, such as in threshold crossing scenario's, the values are checked whether the satisfy the conditions that were imposed on them.

6.1.2. CBM model

For the CBM model metrics verification methods are mostly similar compared to the prognostic model, however there are some additional checks that can and have been performed. This is mainly due to the fact that the metrics that were used to assess the performance of the CBM decision framework have boundary characteristics that are always true. That is there are bounds for the $TP_{rate}, FP_{rate}, FN_{rate}, TN_{rate}, precision and recall$ as they are all defined between 0 and 1. Furthermore in the setup used for this research the following relations also apply: $FN_{rate} = 1 - TP_{rate}$ and max count of TP and FP is limited to number of simulated engine time series. There are also boundary conditions for the TBR and subsequent $MTBR$, namely the time between repair cannot exceed each engine unit time series length and the mean time between repair for the CBM model lies between the $MTBR$ of the run-to-failure scenario and 1 FC.

Knowing these relations allows the user to check them consistently when making adjustments and improvements on the model. It also allows easy verification of the code by invoking conditions where the outcome is known and then checking whether these compare by the simulation as well. For instance when computing the $MTBR$ of the CBM model for threshold conditions that are always/never satisfied (i.e. by having very conservative or progressive threshold values) they should coincide with the extremes the $MTBR$ can have. However when performing this check it was concluded that there was a bug in the code that resulted in a wrongly calculated TBR for each true positive case. This resulted in an inaccurate result for the $MTBR$ which subsequently influenced the outcome of the maintenance value assessment as well. As previously stated the modular approach allowed for quick identification and localization of the issue and it could be fixed without much trouble.

Another check that will be highlighted in the sensitivity analysis, chapter 7, is that all adjustable parameters were varied to study their impact on the computed results. One of those parameters is the RUL tolerance window which dictate what the cut-off point is between a TP and a FP maintenance call by the system. Increasing this window size should always result in an at least equal or greater number of TP and less FP. However it should have no impact on the actual time in between subsequent repairs and the total number of failures prevented by the system. This indeed proved to be the case.

6.2. Validation

The previous section discussed checks that were performed to make sure all calculations were made correctly. The answer now is: is the model and are the calculations representative of real-world conditions. Or to rephrase validation can be stated as the: "substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model" from Schlesinger et al. [26].

This section will discuss how this is done for this study. The following validation measure were used in particular:

GM(1,1) forecast error validation check

Accuracy metric	Training data	Test data
Theil's U statistic	0 00149 (0.00147, 0.00152)	0 00161 (0.00157, 0.00166)
RMSE	0 0713 (0.0693, 0.0733)	0 0763 (0.0744, 0.0782)
MAPE	0.243 % (0.239, 0.248%)	0.261 % (0.255, 0.268%)

Table 6.1: Prognostic forecast error metrics of GM(1,1) (including 95 % confidence interval) on the training and test data respectively.

1. Historical data validation (splitting of training and test set)
2. Sensitivity analysis
3. Extreme condition test
4. Comparison to other models

Firstly the data was divided from the beginning into two components. A training set, that comprised 50% of the total data, and a test set containing the remaining 50%. All calculations and optimizations were performed on the training set only and at a later stage compared to the results obtained from the test set. This check can inform the user if there is a case of (significant) over-fitting of the data. As there is an optimization strategy in place small discrepancies are to be expected, but if large differences arise one must look closer to the cause of this difference.

Furthermore by performing a sensitivity analysis a clear view of the robustness of the model and the dependencies on the underlying variables can be observed. Many relations are directly/inversely proportional to the underlying variables and when these underlying variables are changed the subsequent effect can be checked for expected behaviour.

Another easy check is to use extreme conditions. These conditions simplify the calculations and allow them to be performed by hand as well. These can then be compared to the results obtained from the model to see if these are comparable.

Finally the results from the segments that make up the integrated CBM model are compared to other models as well. In the case of the prognostic model an ARMA model was used on the same data and a naive forecast was also used to compare accuracy results. The same prognostic model was also subjected to another data series to check whether similar results could be obtained as well. For the CBM model this proved to be more difficult as the RUL tolerance window highly influences the CBM performance metrics (discussed in more detail in section 7.2.2). As for the economic value of the model there are large variations in the costs documented in literature, which also made it difficult to compare the findings with other performed research.

6.2.1. Prognostic model

To validate the findings from the prognostic model accuracy the data is split up in two. The data is split 50-50 meaning that the first 100 engines split between both deterioration simulations are used for the training phase and the remaining 100 were used to validate the results. When analysing the forecast accuracy error of the validation set compared to the training set it can be seen that the prognostic model show similar performance. Table 6.1 shows these results.
There are some discrepancies which show a minor case of over fitting. These could be attributed to the fact that the optimal configuration that has the highest forecast accuracy was selected for the training data. When looking at the range of accuracy metrics for slightly different configurations of the GM(1,1), displayed in section 7.1, it shows that the validation accuracy lies within the range from the training set. The difference will therefore not be considered to be significant.

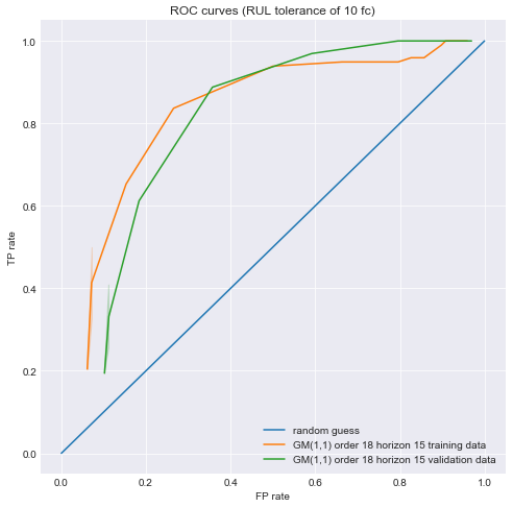
Furthermore the model has also been tested on the KLM data (in section 5.4). Although the forecast accuracy on that particular data shows a significant worse fit it can be accredited to the difference on the type of data itself. This test does shed light on the robustness of the model. It can be concluded that the prognostic model fares better for lower noise data, as most forecast models, but still reach accuracy levels that can be used for prognosis properties.

Finally it will be interesting to see what the impact on the slightly lower forecast accuracy has on the performance of the other segments of the integrated CBM model. It could be that the slight difference

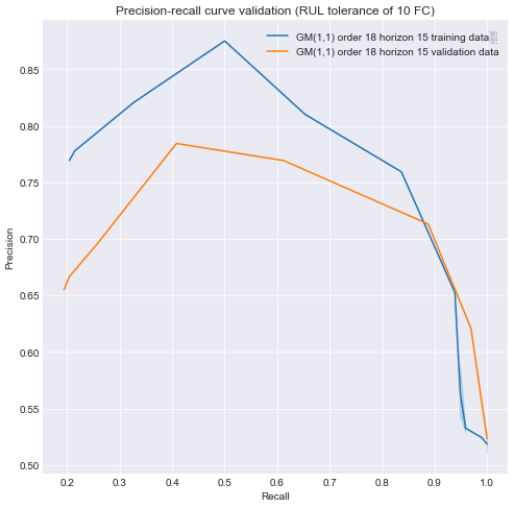
will not propagate further downstream and will contribute to a large detriment in performance. This will be analysed in the next section in more detail.

6.2.2. CBM model

When analysing the CBM results for both the training data and the test data an interesting conclusion can be made. Namely there exist a substantial difference between the two. This can be seen when analysing figure 6.1a and 6.1. This indicates that there is a distinct degree of over-fitting, or that the optimal threshold boundaries are not totally comparable between the training and test data. At least when ROC and precision-recall metrics are concerned. This can be attributed to the high sensitivity on threshold height combined with the sensitivity on RUL tolerance window on these metrics, both of which will be treated in more detail in chapter 7.



(a) ROC curve comparing the training and test data.



(b) precision-recall curve for training and test data.

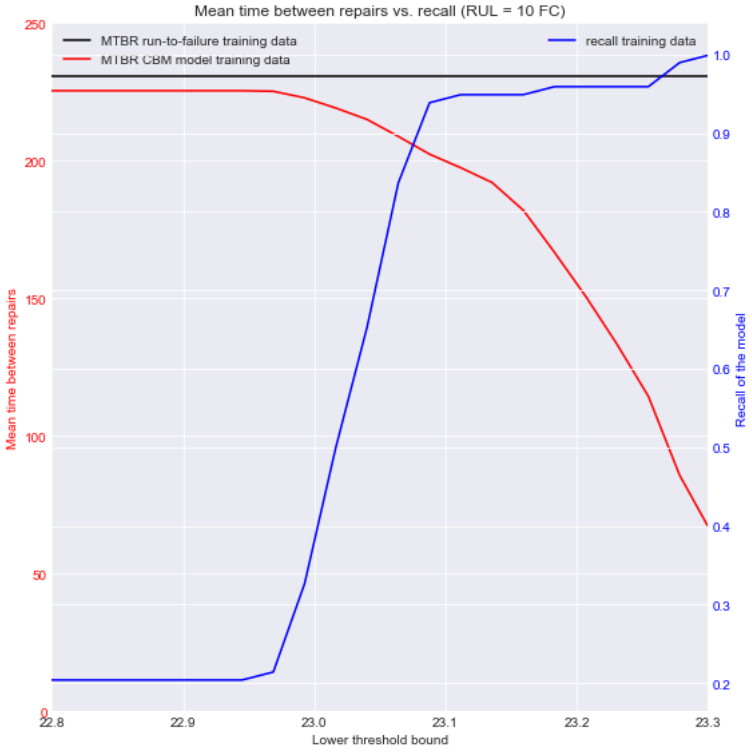
Figure 6.1: ROC and precision-recall curves for both the training and test data. These are computed using identical prognostic forecasts GM(1,1) configuration: alpha value = 0.9; forecast horizon = 15.

When the recall vs. $MTBR$ plots are compared the difference tends to be much smaller. As figure 6.2 clearly shows. However when closely examined there are still some differences detectable. The training data has a slightly lower $MTBR$ as standard and the model $MTBR$ lies closer to the theoretical ceiling as well. This can be attributed to the higher forecast precision illustrated in figure 6.1b for the training data. However the recall rate is slightly steeper for the test set which can counter the loss in $MTBR$.

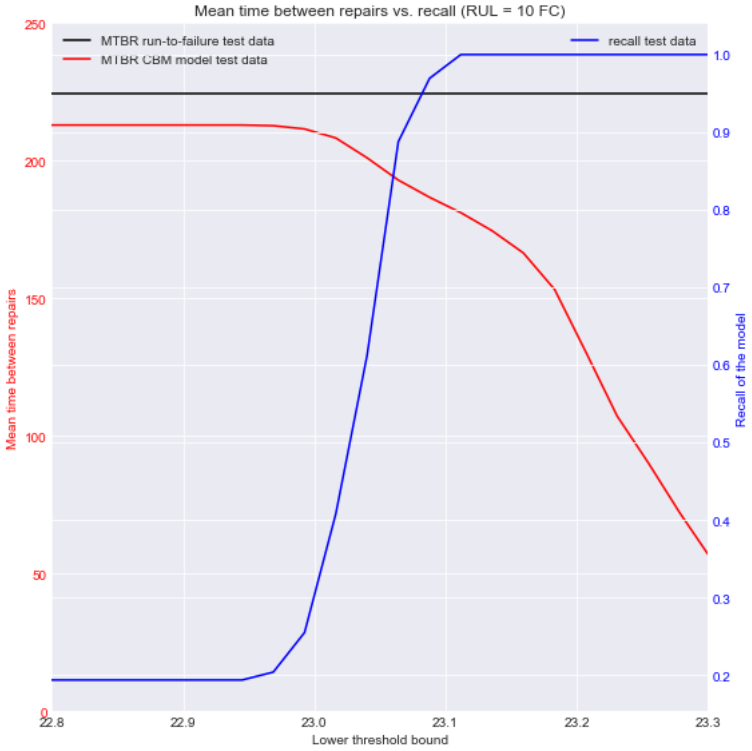
Figure 6.3 shows the financial benefit charts for both the validation set and the training set. It can be seen that as the cost factor C_{factor} increases the difference between the training and test set increases as well. This can be explained by the fact that recall becomes less important for lower cost factors ($\frac{1}{3}$, $\frac{1}{5}$ and below) and precision and $MTBR$ is more important.

However for cost factors above 1 in 5 there is still a normalized net benefit detectable. Even more important the peaks of the two value plots coincide for that range as well. Meaning that for cost factors within the range 1 in 1 up to 1 in 5 static threshold boundaries can be used.

To address the issue of potential over-fitting one could alter the approach for the failure threshold determination. In the current design the threshold(s) height(s) are adjusted dynamically for the training set in an effort to maximize the performance metric of interest. Due to the high sensitivity that the threshold height has on performance a small difference in the underlying data will result in significant changes on the performance metrics as well. By not fixing the threshold height, but by continuously updating threshold heights based on newly gathered information this could be limited to a large extent. Unfortunately this does come at a cost of added complexity and computation power. These are also the main reasons that this was omitted in this study in an effort to keep it within the scope of a thesis project.

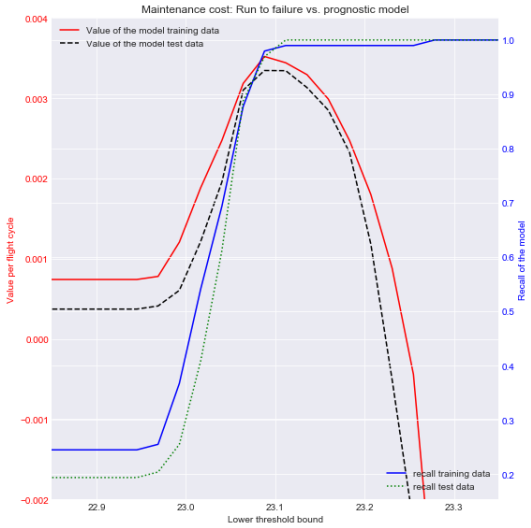


(a) $MTBR$ vs. recall training data

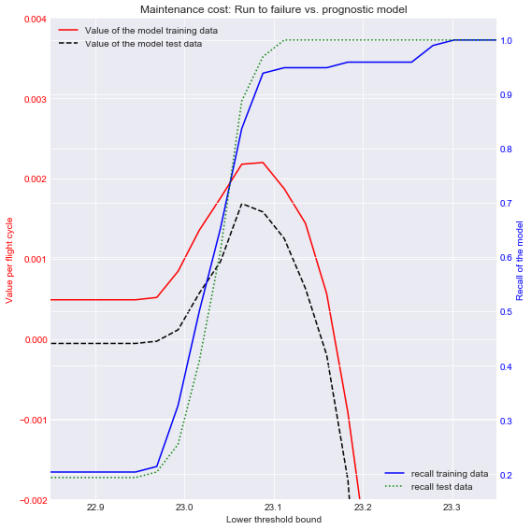


(b) $MTBR$ vs. recall test data

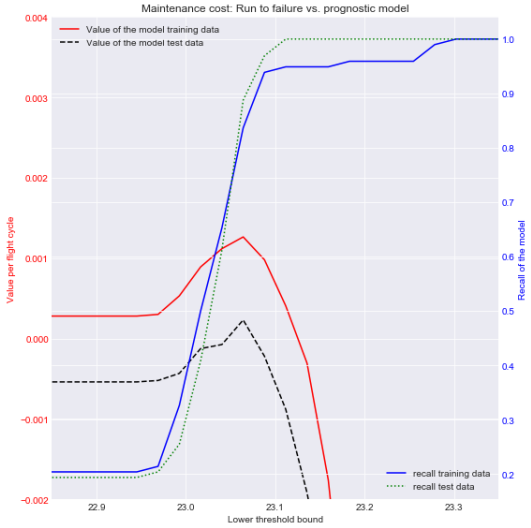
Figure 6.2: $MTBR$ vs. recall for both the training and test data. These are computed using identical prognostic forecasts GM(1,1) configuration: alpha value = 0.7; forecast horizon = 15



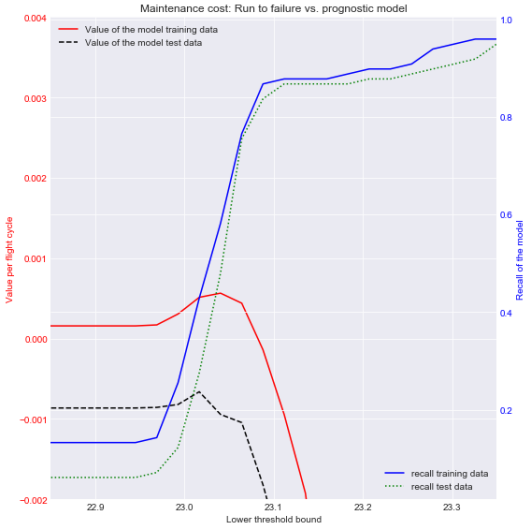
(a) Maintenance cost factor is 1.



(b) Maintenance cost factor is $\frac{1}{3}$.



(c) Maintenance cost factor is $\frac{1}{5}$.



(d) Maintenance cost factor is $\frac{1}{7}$.

Figure 6.3: Normalized net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors comparison of training data and test data.

7

Sensitivity analysis

When presenting the results and validation of the model in chapter 5 and 6.2 it has been mentioned that the configurable parameters that are used to construe the model have significant effects on the outcome and performance. This chapter will perform an in depth analysis on that very subject. It will do so in a staggered fashion where each element that makes up the integrated CBM framework visualized in figure 2.1 will be treated separately. That is the sensitivity of the prognostic model will be evaluated in section 7.1. Then the fluctuations of the CBM metric on changes on the parameters will be evaluated in section 7.2. And lastly the value of the integrated CBM framework as a whole in section 7.3.

Each additional step further along the total framework will introduce more configurable parameters and as such if a fully coupled dependency sensitivity analysis would be performed will introduce an additional degree of freedom as well. To limit the amount of figures, analysis and computation time needed this will not be done in this fashion, but rather the most influential parameters will be treated instead.

The list of all adjustable parameters discussed during the sensitivity analysis are as follows:

- Number of historic data points used in the prognostic model
- Prognostic model value alpha
- Prognostic model forecast horizon length
- CBM threshold value and sensitivity
- RUL tolerance cut-off point
- Maintenance cost factor

Note that all these parameters are discrete, but some are also limited to integer values. For the parameters that are not integer based that step size of the fluctuations also plays an important role in the performance of the results and the validation thereof as previously mentioned in section 6.2. This will be briefly reiterated in section 7.4 as well. Furthermore whilst the $MTBR$ and FN_{rate} are one of the key parameters that define the value of the integrated CBM model they can not be adjusted by the user, but are based on outputs from earlier segments of the integrated framework. As such they will not be treated in much detail.

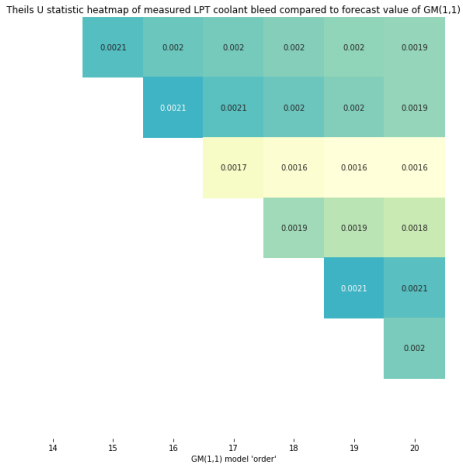
7.1. Sensitivity of the prognostic model

The parameters that can be influenced by the designer of the GM(1,1) are as follows: # of historic data points, alpha value and forecast horizon. The influence of these parameters will be treated on their effect on the accuracy metrics described in section 5.1. The number of historic data points used by the model will be treated separately as it imposes limitations on the maximum forecast horizon. This will be done in subsection 7.1.1. From that analysis a single value has been selected to be used for the remainder of this study to limit the amount of adjustable parameters. Then in subsection 7.1.2 both the alpha and horizon length effect on prognostic model performance will be treated together. This will be done by using heat-maps, which enable to see any coupling effects of the two.

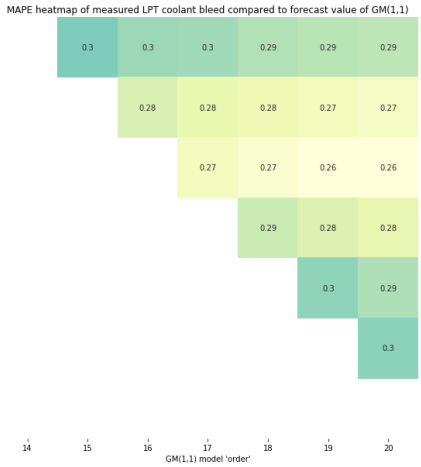
7.1.1. Prognostic model "history order" and horizon length effects on accuracy

First of all it should, again, be noted that there are 2 "orders" of that comprise a GM(m,n). There is the actual model order m that states which degree of differential equation is used in obtaining AGO and IAGO respectively. And there is the model "history order" mentioned throughout this report which is the number of historic data points used from the original data set as inputs for the GM. It is therefore possible to have a Grey model of order 1 that uses 18 historical values for training. Furthermore the number of training values used has a direct implication on the length of the maximum forecast horizon it can produce.

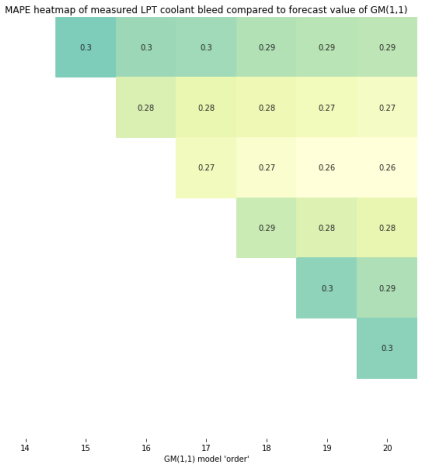
In order to meet the preset minimum forecast horizon of 12 FC to allow for maintenance action lead times the spread of time series points taken into account will be 14 till 20, with forecast horizon lengths varying from 12 till 18. The results from the accuracy analysis is depicted in figure 7.1. It should be noted that the portion of the heat-map that is not depicted forecast horizon smaller than 12 FC and the results are therefore excluded from the figure.



(a) Heat-map of the Theil's U statistic.



(b) Heat-map of the MAPE values.



(c) Heat-map of the RMSE values.

Figure 7.1: Heat-map showing the prognostic model forecast performance on the accuracy metrics at varying model "historic orders" and forecast horizon values.

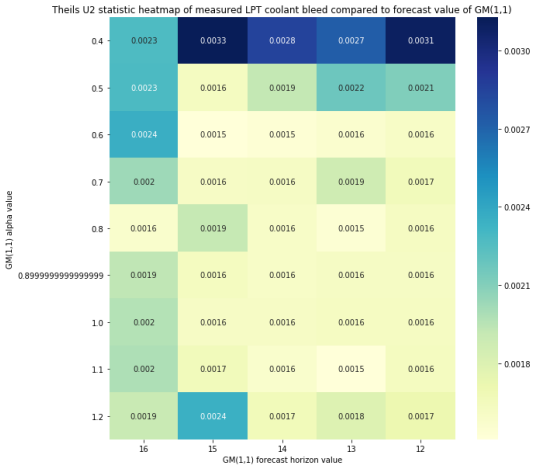
The heat-maps can be interpreted as follows. The model "historic order" depicts the amount of data points used as inputs of the GM(1,1) and each additional vertical block increases the forecast horizon by 1, with a minimum of 12. When analysing the results it can be seen that when a higher number of points are used, for a similar forecast horizon, to create the GM(1,1) the accuracy increases. However there is also a higher number of historical points required for obvious reasons. As there is not a large increase in forecast accuracy from 18 points onward, whilst still being able to provide a useful range of

forecast horizon lengths, this was the selected number of points for the rest of the study.

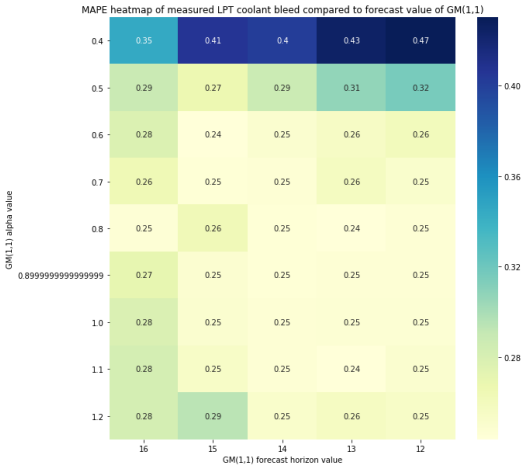
Furthermore it seems that there is an optimum forecast horizon length of 15 where, irrespective of the number of data points used by the model, the accuracy is highest. The reason for this is unsure when analysing these graphs alone and further analysis on this with regards to the GM alpha value might give additional information why this is the case.

7.1.2. Prognostic model forecast horizon and alpha effects on accuracy

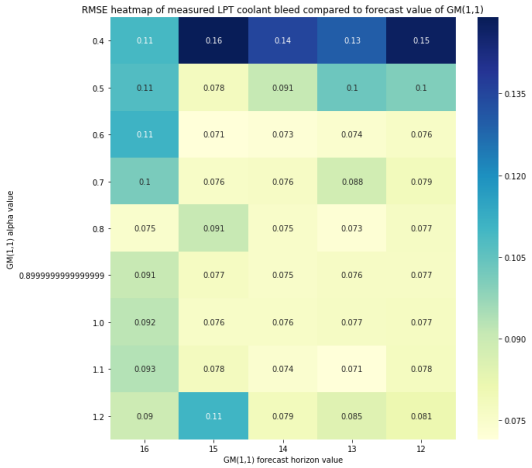
This section will investigate the (coupled) effect of forecast horizon and model alpha value. It has been hypothesized that an increase in horizon length will lead to a less accurate prediction since the nature of the rolling GM will use more values obtained from previous predictions than actual historical values. This will lead to an exaggeration of trends and will induce a more erratic forecast in return. The value alpha on the other hand acts as a damping agent and has an opposite effect when increased. It is therefore especially interesting how these two parameters perform when used in conjunction of each other. The provided heat-maps in figure 7.2 will give a clear image of this coupling effect.



(a) Heat-map of the Theil's U statistic.



(b) Heat-map of the MAPE values.



(c) Heat-map of the RMSE values.

Figure 7.2: Heat-map showing the prognostic model forecast performance on the accuracy metrics at varying alpha and forecast horizon values.

When analyzing the difference in for the different configurations it can be seen that the value alpha has a far greater influence on the performance than the forecast horizon. When taking a look at a plot, figures 7.3a and 7.3b, with a GM(1,1) with alpha values of 0.4 compared to 1.0 you can clearly see the difference in behaviour. The value alpha determines the rate of change that the GM infers on differences in the reference signal. As the input data behaves quite erratic a low alpha value will amplify this to extreme degree. This is undesirable as it increases the chance that the maintenance threshold is crossed

prematurely, especially since the sensitivity of threshold height are quite high (this will be elaborated upon in subsection 7.2.1).

It should be noted that a higher alpha value is not better by definition as the dampening effect will also reduce its capacity of detecting newly emerging trends and will therefore experience lag. There is thus an optimum between those two extremes. The heat maps are an excellent way to find that balance.

When comparing figures 7.3b to 7.3c it can again be seen what was already touched upon in section 5.1. That a forecast horizon that lies further in the future will be more erratic. The difference in the two is more subtle compared to the difference detected for the alpha values, but it is present nonetheless. A more surprising fact is that when the forecast horizon becomes less than 16 leads to a significant drop in residual error. Furthermore decreasing the forecast horizon even further will not lead to the same better forecast accuracy. It can therefore be argued that selecting a higher forecast horizon with similar accuracy has the priority as it will be able to detect failures from an earlier point in time.

7.2. Sensitivity of the CBM model

It is not only interesting to see what the sensitivity is of the CBM model from an academic point-of-view but this also gives insight in the operation range that the turbofan engine is designed for. Furthermore the dependency of the used performance metrics often used in literature, ROC and precision-recall, on parameters such as the RUL tolerance window will also shed some light on their practicality in general. This section will answer those two points by first addressing the CBM sensitivity on variations in threshold values in subsection 7.2.1 and secondly on the RUL tolerance window in subsection 7.2.2. Finally the impact that the threshold height has on the $MTBR's$ will also briefly be discussed.

7.2.1. Threshold sensitivity on CBM performance

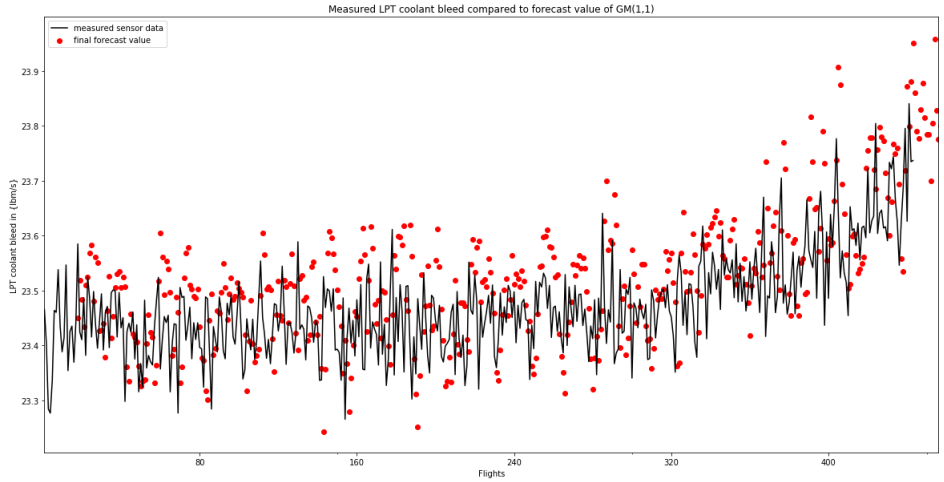
The sensitivity of the CBM performance metrics on threshold height is a different story compared to the previously described parameter, as the ROC and recall curves rely on a range of values for the threshold to be calculated in the first place. There is therefore always a certain sensitivity in place that will be illustrated when calculating those metrics. Nonetheless as mentioned above it is still interesting from a designer standpoint to see what the ranges of threshold values are for a recall of 100% compared to 0%. Furthermore the resolution of the threshold values, how many increments comprise the total range of threshold values, also has a significant effect on the CBM performance metrics.

It should be noted that all sensitivity analysis evaluations performed in this section use the two independent threshold model. As the figures used are only able to visualize one floating threshold parameter the upper bound was used as the static threshold. It is therefore not possible to visualize the total range of recall as the upper bound already captures some impending failures. The decision to use a static value for the upper threshold over the lower threshold comes from analysis from 5.5a where a distinct sharp peak in precision could be detected. It can therefore be interpreted that the upper threshold has an even smaller range and thus more feasible to use as a single value for these analyses. It should also be noted that for actual use of the CBM model a single value for these thresholds has to be selected eventually as well.

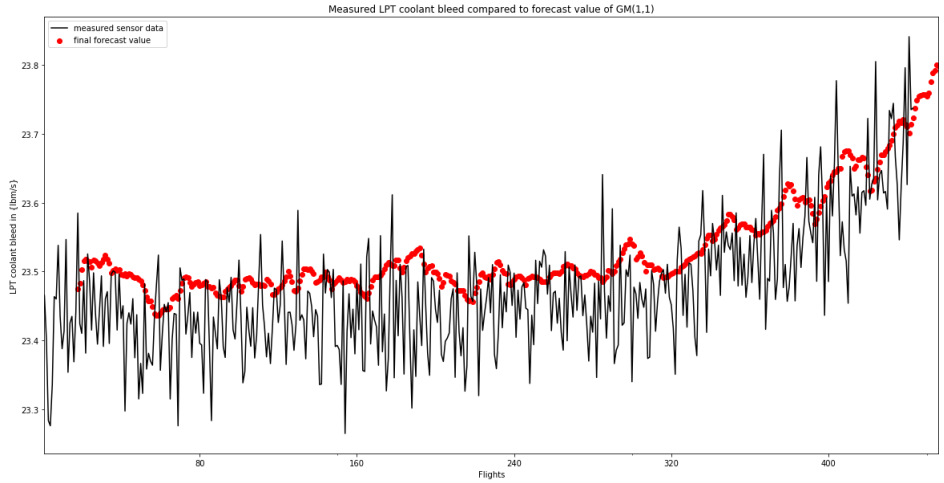
When analysing the figures in 5.7 one can see that when the lower threshold bound is set at 22.95 the recall is around 20%, whilst it is around 95% at 23.10. This does change slightly for the various cost factors used in these graphs as these have different upper threshold bounds, but it can clearly be seen that a only a slight difference in threshold results to a dramatic difference in recall. the difference is only 0.65% in absolute terms. It can thus be concluded that the LPT coolant bleed flow has a very narrow operating range. As the system is run in a closed control loop it is likely that this sensor is one of the control variables.

The sensitivity of the threshold values compared to $MTBR's$ can also be seen from figure 5.6. It can be seen that a difference in lower bound of 23.4 to 23.0 results in an almost 100% decrease in modeled $MTBR$. It should be obvious that a threshold value of 23.4 makes no sense as this is higher than the average value of the sensor signal of the engine when newly installed (minimal degradation), but it still should be taken into consideration. For instance if a particular operating condition will reduce the LPT coolant bleed flow, a significant drop in atmospheric pressure or sensor that is not properly calibrated, it can instantly trigger a false positive engine failure. This is highly undesirable and mitigating measures to this have to be taken into consideration if such a system was used in real-world cases.

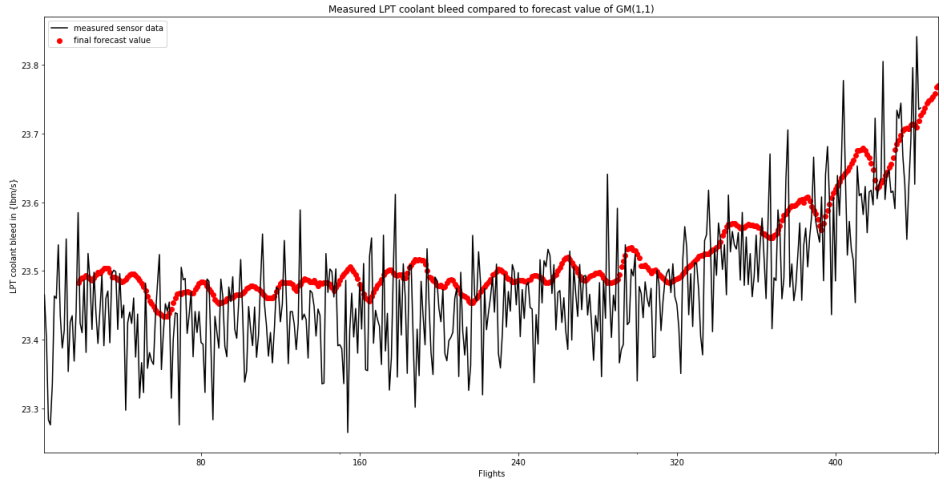
As mentioned before in subsection 6.2.2 the resolution of the threshold parameter has a great effect



(a) GM(1,1) configuration: alpha value = 0.4; forecast horizon = 12.

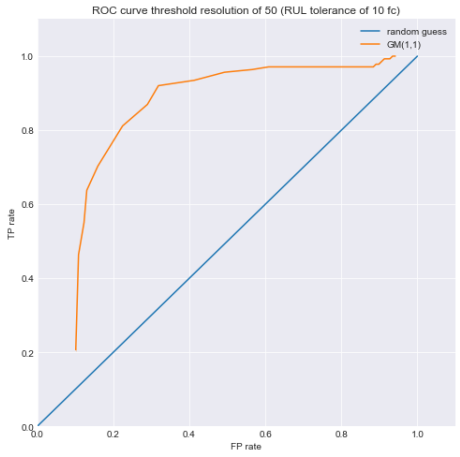


(b) GM(1,1) configuration: alpha value = 1.0; forecast horizon = 16.

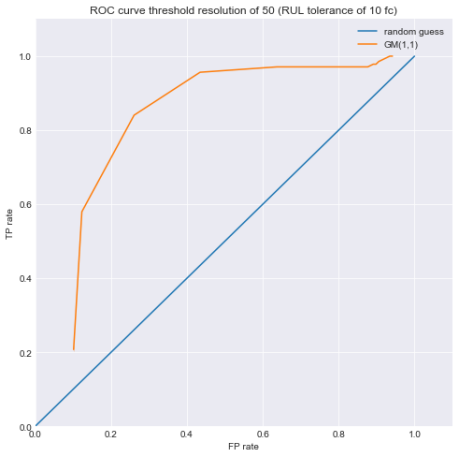


(c) GM(1,1) configuration: alpha value = 1.0; forecast horizon = 12.

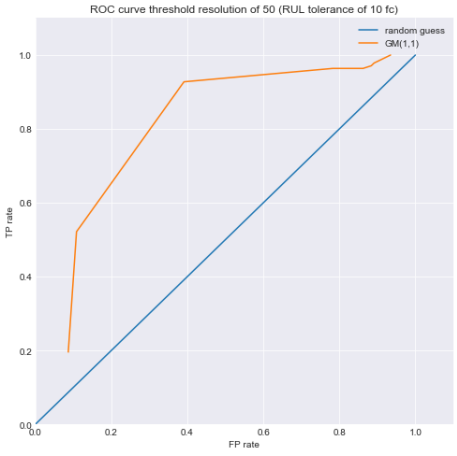
Figure 7.3: Final forecast values of a GM(1,1) for various values of alpha and forecast horizon.



(a) Threshold resolution of 150.



(b) Threshold resolution of 50.



(c) Threshold resolution of 25.

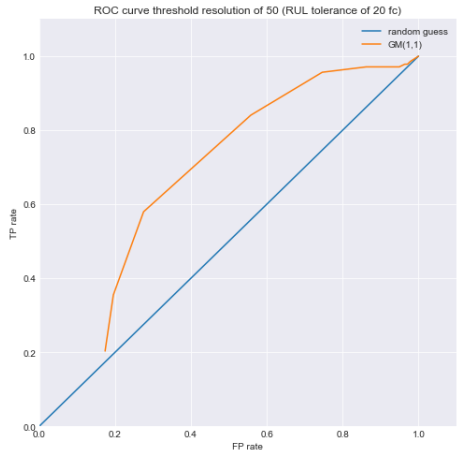
Figure 7.4: Differences in ROC curves based on number of threshold step-sizes (resolution). All calculations were performed with the same prognostic data (GM(1,1) configuration: alpha value = 0.7; forecast horizon = 15).

on how the ROC an precision-recall metrics are perceived. The terminology of perceived is done on purpose as two a identical threshold values will lead to identical CBM metrics such as TP_{rate} FP_{rate} precision and recall but their respective figures will look different and as such the area that is calculated and is often used as an indicator for global performance of a model will also differ. It is quite simple as there is such a fine margin of between the "healthy" and "near faulty" sensor signal a higher resolution of threshold values will be able to capture the JIT point, whilst a low resolution will easily overshoot and will go from FN to FP in one threshold boundary step.

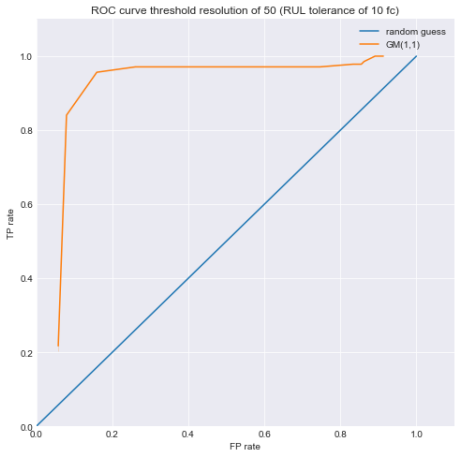
This increase in step-size of threshold values will thus lead to a less fluid ROC and precision recall-curve. This can be seen in figures 7.4. Here the same identical prognostic model forecasts are used, but only the step-size (resolution) of the threshold range is varied. Note that distinct points should coincide, but because there are less points the interpolation in between those points becomes coarser. Also note that it is possible that when the threshold step size is changed it is not possible to have the exact same threshold points and this will also result in slight variations in the plots.

7.2.2. Remaining useful life tolerance window sensitivity on CBM performance

Changing the so called "RUL tolerance window" is like changing your measuring stick from imperial to metric. It does nothing to the underlying model, but greatly alters the rated performance in terms of CBM metrics such as TP_{rate} FP_{rate} and precision. This is no surprise as the fundamental basics of those metrics is determined whether a predicted failure lies within said tolerance window. If a certain failure is predicted to occur 15 FC before actual failure it will only counts as a FP in the case of a RUL

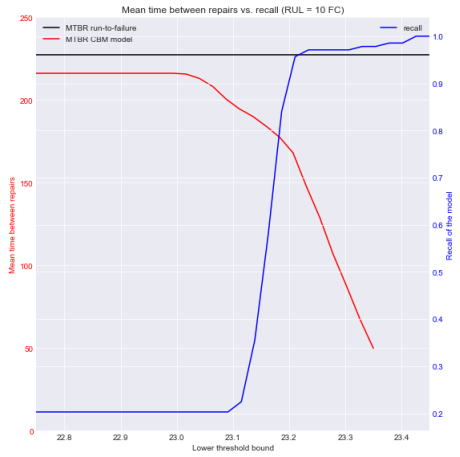


(a) RUL tolerance window of 5 FC.

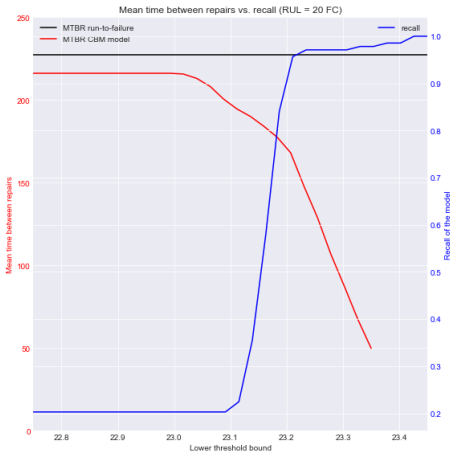


(b) RUL tolerance window of 20 FC.

Figure 7.5: ROC curves for different RUL tolerance windows. These are computed using identical prognostic forecasts (GM(1,1) configuration: alpha value = 0.7; forecast horizon = 15).



(a) RUL tolerance window of 5 FC.



(b) RUL tolerance window of 20 FC.

Figure 7.6: $MTBR$ plotted against different RUL tolerance windows. Note that the RUL window has no effect on these metrics and both plots are identical.

tolerance window of 10 FC and not if it has been set to 15 FC or even higher. This can often lead to misleading results as the model still predicts exactly the same time that a repair is needed, but in one case will be perceived as a bad forecast, whilst in the second will be evaluated as the opposite. This is one of the main downsides of relying solely on ROC and precision-recall performance as they can be interpreted incorrectly. Furthermore the industry standard of 10 FC will have a different impact for components that have a shorter lifespan than ones that have a large lifespan.

Plotting the ROC and precision-recall curves with identical input data shows this in clear detail, as figure 7.5 illustrates. It looks like the curves of the higher RUL tolerance window indicate a better performing model, whilst this is entirely untrue. As previously discussed the RUL window has no effect on the recall and $MTBR$ as these metrics are not defined by it. A quick glance at figure 7.6 confirms this.

Finally it should be noted that increasing the RUL window will reduce the assessment on over-fitting (based on the previously mentioned metrics) as there is a less strict zone that define these. Furthermore high dependency of RUL window size on perceived performance does not mean that those metrics are not suitable to gauge performance. It only illustrates that context is vital to do a correct assessment. A model with a high AUROC (with relative to lifespan small RUL window) will indicate that it is able to classify well.

7.3. Sensitivity with respect to variations in maintenance cost on value

From figure 5.7 it can be seen that there is a distinct upper limit concerning the cost factor C_{factor} where this CBM model has a positive financial benefit. Furthermore changes in ratio of component cost versus repair cost changes from $\frac{1}{5}$ to $\frac{1}{3}$ and 1 signifies a normalized value increase of 69% and 59% respectively. It should be noted that this is only a direct cost saving and will not take any secondary benefits into account such as for instance the benefit of better on time performance of the airline.

As the value calculations are normalized the overall benefit will increase linearly with the actual height of these costs. Also these calculations are on a per FC basis so if the total number of FC's increases the value will also increase accordingly.

Finally the main driver that determines cost savings/additional costs is the relationship between $MTBR$ and recall (1-FN_{rate}). As costs are inversely proportional to $MTBR$ they tend to skyrocket if $MTBR$ becomes increasingly small. A key point to take into consideration is that the current model acts strictly binary. If a failure is predicted a maintenance action will be performed always resulting in the same cost profile. This is of course not representative of reality as a no fault found assessment will not necessarily lead to the same costs as a repair when system degradation is indeed noticeable. It could therefore be stated that this evaluation gives a minimal baseline of financial benefit and will perform even better if the maintenance actions are better tailored to the actual deterioration assessment when a failure has been predicted.

7.4. Sensitivity analysis intermediate conclusions: risk of over fitting

There are several key points that can be taken from this section. One of the most important is that the CBM model is highly sensitive to threshold value fluctuations. There is a very small change in the signal value between the system near failure and at start of life. Due to this it is difficult to find the JIT point without increasing the threshold resolution that enables very fine step sizes.

By doing this one is able to define the JIT point, but over fitting becomes an issue. As signal noise has an increasingly large effect as threshold step size decreases the performance of the model will differ more from test to test.

Another issue is that the small window of operation that signifies the difference between a "healthy" engine and a "failed" engine it becomes increasingly important to limit signal noise and outside influences on signal value. A small deviation due to different operating conditions, environmental properties or calibration of the sensor can result in a faulty classification easily. It is therefore important to add additional safety guards in place, such as incorporating a minimal of faulty classifications before an actual failure event will be called. Finding the right balance between the parameters alpha and forecast horizon can be used to some extent to dampen signal noise also.

It can also be concluded that ROC and precision-recall performance alone do not provide adequate information to assess the value of a model. Necessary context needs to be provided. The ROC and precision-recall curves rely to a great extent to the definition of the RUL tolerance window which could be perceived as arbitrary.

Finally fluctuations in cost have a direct effect on economic feasibility of the CBM model as expected. An upper limit can be defined from where there is a measurable cost saving compared to a run-to-failure maintenance paradigm. This is a worst case scenario as there are many additional improvements possible, most notably in differentiating maintenance actions adequately if a failure is predicted by the CBM system.

8

Conclusions

The research has shown that there it is indeed vital that a system approach is performed when it comes to condition based maintenance. The local optimization approach commonly found in literature is not sufficient to reach optimal results. The sophisticated models used in those researches are thus only part of the puzzle. An equally important step that has to be made is the integration of those sub-systems. Furthermore simply integrating all elements without taking any considerations on the impact they impose on each other will lead to a sub-optimal design. It is therefore vital that a feedback loop and performance metric that encompasses the maintenance system in its entirety is used. Only by using such an approach is it possible to find the required configurations that reach a global optimum in system performance.

It was also discovered the GM(1,1) used for the prognostic model is effective in dealing with signals that are heavily subjected to noise. However if the underlying signal becomes small and stationary the model tends to become unstable. Further development and research on mitigating measures for this undesirable behaviour are therefore suggested.

With regards to the decision making CBM framework it is found that the system has a highly sensitive nature. Small deviations in the prognostic value will result in a healthy or faulty assessment. It is therefore crucial that system noise is minimized as much as possible. Additionally the highly sensitive property of the model indicate that careful consideration has to be made in configuration of the adjustable parameters. Outside influences such as operational conditions and environmental effects that distort the reference signal are also to be avoided, or to be mitigated by adequately re-calibration of the failure thresholds. This can pose a threat to the robustness of this approach and the applicability in real-world scenario's. As such it can be recommended that further research shall be devoted in combating these issues.

It was also shown that ROC and precision-recall metrics are prone to be misread when provided without context. These metrics are highly dependent on the arbitrary distinction between a TP and FP by the user. Another noticeable findings include regarding these metrics is that when the threshold resolution is increased the performance of these metrics will increase as well as the are able to over-fit the data to a higher degree. These two facts can lead to inflated results that can thus indicate totally different performances. A CBM system primarily based on these metrics is therefore inadvisable.

To mitigate this issue it can be suggested that $MTBR$ vs. recall graphs are to be used in conjunction with the aforementioned metrics. These give the required context necessary to make an adequate assessment on performance. Moreover these parameters are unaffected by user specification/relaxation of the RUL tolerance window and as such are more fundamental performance metrics.

From this study also proved that using a cost factor approach can have priority over an ultra specific cost assessment. When studying the literature there is a vast range of possible methods to calculate maintenance cost. Not only do these differ in their approach; the values that are presented are also highly subjective to a specific condition. It is therefore not useful to use such a system to assess the financial feasibility of an integrated CBM system as it will either be too case sensitive or so many assumptions will have to be made that the added complexity is lost regardless. By performing a simplified cost factor approach a future interested party is able to quickly evaluate if there is a viable use case for a CBM system.

Finally it should be noted that the aim for this research was not to use the most sophisticated techniques known to date to reach maximum performance, but rather prove that there is an equally significant gain to be accomplished by simply taking an integrated approach to the problem at hand.

9

Recommendations for future research

One of the most notable observations found during this research is the sensitivity of the threshold height on model performance. The impact of fluctuations in sensor input values is very high and can lead to restriction on the practicality of implementing the model as is. The high sensitivity also led to over-fitting issues when if underlying data differs from the training data. It is therefore suggested that further research on this topic is necessary. There are some suggestions that can help in addressing this issue.

The most simple is to limit the scope of the applicability to systems that have a high SNR. Another is to limit signal noise and external influences imposed by the environmental and operational conditions. This can be done by either filtering the sensor readings directly or by using sensors that are less prone to these external factors. Either by using different sensors, different sensor locations or different measured parameters in its entirety. Furthermore the filtering characteristics and/or robustness of the prognostic model and failure threshold system respectively can also be improved upon with future research. Finally a more dynamic concept of failure thresholds can be constructed wherein the height of the threshold is updated continuously on newly gathered information. This will mitigate over-fitting issues and will probably also improve performance in general by decreasing the false positive rate for a given recall rate.

an additional aspect that can be improved upon is the complexity of the prognostic model. Although this study confirms the hypothesis that performance is not strictly linked to prognostic accuracy an improvement on this part can lead to better imminent fault detection, thus potentially limiting the amount of missed failures without compromising life-cycle usage of components in general. To accomplish this the prognostic model can be altered to take multiple sensors into account simultaneously, transferring it to a multi-dimensional model. This will also reduce its dependency on single parameter fluctuations and thus increasing its robustness overall.

Special attention to the feature analysis is also a point that can be touched upon in future work. Especially when combined with the former point of attention (a multi-dimensional Grey model approach). Detailed feature analysis can provide an insight to the features which hold the most information on system health and as such a combination of the most important features can increase the models capabilities on fault detection.

Whilst fault prevention by forecasting imminent failures is in itself a great feat, being able to accurately predict a gradient of system health states will allow for even more informed maintenance decisions. Coupling this with broadening the application of the this type of maintenance strategy to other aircraft sub-systems could allow for an even more complex maintenance structure wherein maintenance tasks are grouped together. This grouping strategy could for instance be based on their respective location in the aircraft or their functionality. This strategic planning of maintenance would require this more complex system health system combined with sophisticated scheduling of maintenance assets, but it could lead to a reduction in maintenance time and cost in the long run.

Another item that can be addressed is the robustness of the CBM decision making model. Currently the model is susceptible to early false positive failure assessments when the forecasting model performs erratically. This could be mitigated by adding a requirement that multiple consecutive failures have to be predicted to occur before action is undertaken. A different method to limit these false positives is

to diversify the model by adding multiple features as mentioned earlier. Both of these measures will introduce additional robustness of the system, but special care has to be taken into consideration that it does not impact the recall rate of the system negatively.

Finally concerning the cost evaluation there is also a large window for improvement to be gained as well. The current model uses a rudimentary approach by design as there exist a large variation in maintenance cost associated with unexpected failures, or engine failures in general. But when the underlying models, both the prognoses and system health monitoring, are improved in their complexity and capabilities a cost evaluation method that is more tailored to specific failure modes will certainly be possible. This would result in a complete rework on the approach of all underlying aspects of the integrated CBM system as this research has proved that to maximise performance a system approach has to be taken.

Bibliography

[1] Shannon Ackert. Basics of Aircraft Maintenance Reserve Development and Management. Technical report, Aircraft Monitor, 2012.

[2] Diyar Akay and Mehmet Atak. Grey prediction with rolling mechanism for electricity demand forecasting of Turkey. *Energy*, 32(9):1670–1675, 2007. ISSN 03605442. doi: 10.1016/j.energy.2006.11.014.

[3] Rajendra Akerkar. Analytics on big aviation data: Turning data into insights. *International Journal of Computer Science and Applications*, 2014. ISSN 09729038.

[4] E. da C. P Batalha. *Aircraft Engines Maintenance Costs and Reliability*. PhD thesis, Universidade Nova de Lisboa, 2012. URL `https://run.unl.pt/bitstream/10362/8792/3/TEGI0308.pdf`.

[5] Gustavo E. A. P. A. Batista, Ronaldo C. Prati, and Maria Carolina Monard. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, 2004. ISSN 19310145. doi: 10.1145/1007730.1007735.

[6] T. Chai and R. R. Draxler. Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3): 1247–1250, 6 2014. ISSN 19919603. doi: 10.5194/gmd-7-1247-2014.

[7] Santanu Chatterjee and Jonathan S. Litt. Online model parameter estimation of jet engine degradation for autonomous propulsion control. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2003. ISBN 9781563479786. doi: 10.2514/6.2003-5425.

[8] Ho Wen Chen, Ruey Fang Yu, Shu Kuang Ning, and Hsin Chih Huang. Forecasting effluent quality of an industry wastewater treatment plant by evolutionary grey dynamic model. *Resources, Conservation and Recycling*, 2010. ISSN 09213449. doi: 10.1016/j.resconrec.2009.08.005.

[9] Madalena Costa, Ary L. Goldberger, and C. K. Peng. Multiscale Entropy Analysis of Complex Physiologic Time Series. *Physical Review Letters*, 89(6), 2002. ISSN 10797114. doi: 10.1103/PhysRevLett.89.068102.

[10] J L Deng. Introduction to Grey System Theory. *J. Grey Syst.*, 1(1):1–24, 11 1989. ISSN 0957-3720. URL `http://dl.acm.org/citation.cfm?id=90757.90758`.

[11] Abram Durk Doedijns. *Big-data prognostics in aircraft condition based maintenance: a literature review*. PhD thesis, Delft University of Technology, 2019.

[12] Dean K Frederick, Jonathan A Decastro, Jonathan S Litt, and William W Chan. User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS). Technical report, NASA, 2012. URL `http://www.sti.nasa.gov`.

[13] D. Galar, A. Parida, U. Kumar, D. Baglee, and A. Morant. The measurement of maintenance function efficiency through financial KPIs. In *Journal of Physics: Conference Series*. Institute of Physics Publishing, 2012. doi: 10.1088/1742-6596/364/1/012112.

[14] Kai Goebel, Hai Qiu, Neil Eklund, and Weizhong Yan. Modeling Propagation of Gas Path Damage. In *2007 IEEE Aerospace Conference*, pages 1–8, Big Sky, MT, USA, 2007. IEEE.

[15] Jie Gu, Nikhil Vichare, Bilal Ayyub, and Michael Pecht. Application of grey prediction model for failure prognostics of electronics. *International Journal of Performability Engineering*, 2010. ISSN 09731318.

[16] Mr Andrew Hess. Challenges, Issues, and Lessons Learned Chasing the "Big P": Real Predictive Prognostics Part 1. In *2005 IEEE Aerospace Conference*, pages 3610–3619, Big Sky, MT, USA, 2005. IEEE. doi: 10.1109/AERO.2005.1559666. URL `https://ieeexplore-ieee-org.tudelft.idm.oclc.org/document/1559666`.

[17] IATA. Airline Maintenance Cost Executive Commentary, 2019. URL `https://www.iata.org/contentassets/bf8ca67c8bcd4358b3d004b0d6d0916f/mctg-fy2018-report-public.pdf`.

[18] Andrew K.S. Jardine, Daming Lin, and Dragan Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance, 2006. ISSN 08883270.

[19] Deng Ju-Long. Control problems of grey systems. *Systems and Control Letters*, 1982. ISSN 01676911. doi: 10.1016/S0167-6911(82)80025-X.

[20] Steven Kesten, Richard Casaburi, David Kukafka, and Christopher B. Cooper. Improvement in self-reported exercise participation with the combination of tiotropium and rehabilitative exercise training in COPD patients. *International Journal of COPD*, 2008. ISSN 11769106. doi: 10.1016/j.patrec.2005.10.010.

[21] Sungil Kim and Heeyoung Kim. A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting*, 32(3):669–679, 7 2016. ISSN 01692070. doi: 10.1016/j.ijforecast.2015.12.003.

[22] Mingzhi Mao and E. C. Chirwa. Application of grey model GM(1, 1) to vehicle fatality risk estimation. *Technological Forecasting and Social Change*, 2006. ISSN 00401625. doi: 10.1016/j.techfore.2004.08.004.

[23] Thomas D Matteson. Airline experience with reliability-centered maintenance. In *Nuclear Engineering and Design*, volume 89, pages 385–390, 1985.

[24] Ying Peng, Ming Dong, and Ming Jian Zuo. Current status of machine prognostics in condition-based maintenance: A review. *International Journal of Advanced Manufacturing Technology*, 2010. ISSN 02683768. doi: 10.1007/s00170-009-2482-0.

[25] G Sargent, Robert. Verification and Validation of simulation models. In J. Himmelspach K.P. White S. Jain, R.R. Creasey and eds. M. Fu, editors, *Winter Simulation Conference*. Winter Simulation Conference, 2009. ISBN 9781424457717.

[26] Stewart Schlesinger, E El Segundo Boulevard El Segundo, Roy E Crosbie, Roland E Gagn, George S Innis, CS Lalwani, Joseph Loch, Wright Patterson Air Force Base, Richard J Sylvester ASD, En Wright Patterson Air Force Base, Richard D Wright, Ft McNair, Naim Kheir, and Dale Bartos. Terminology for model credibility. In *Simulation councils proceedings*, 1979.

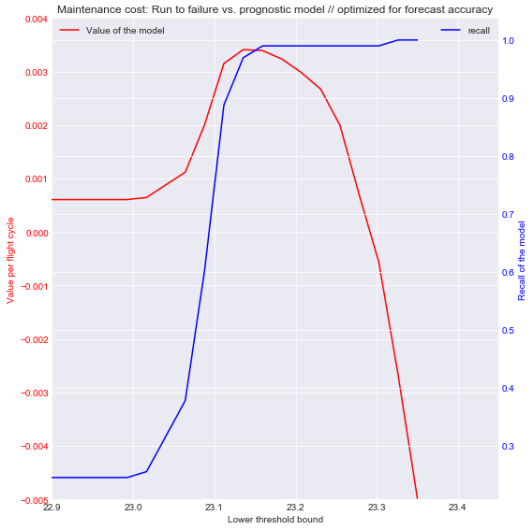
[27] D. Daniel Sheu and Jun Yuan Kuo. A model for preventive maintenance operations and forecasting. *Journal of Intelligent Manufacturing*, 2006. ISSN 09565515. doi: 10.1007/s10845-005-0017-6.

[28] H Theil. Applied economic forecasting, 1966.

[29] Tangbin Xia, Xiaoning Jin, Lifeng Xi, Yuejun Zhang, and Jun Ni. Operating load based real-time rolling grey forecasting for machine health prognosis in dynamic maintenance schedule. *Journal of Intelligent Manufacturing*, 2015. ISSN 15728145. doi: 10.1007/s10845-013-0780-8.

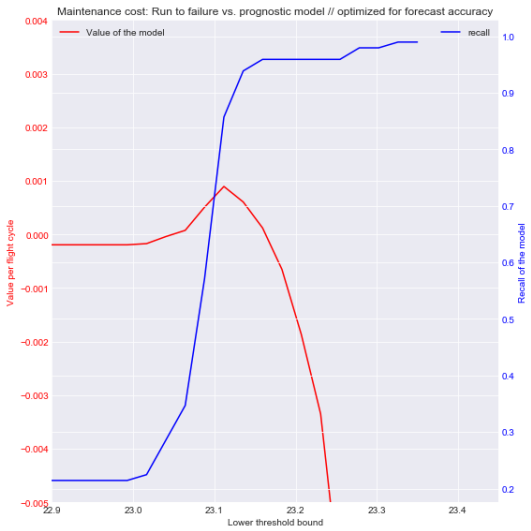
A

Maintenance cost saving graphs at discrete cost factor points

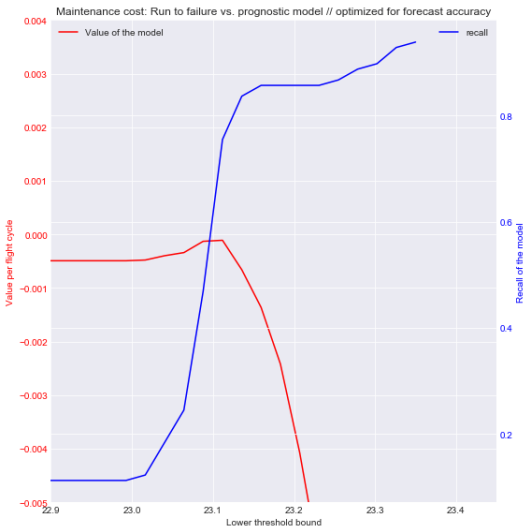


(a) Maintenance cost factor is 1.

(b) Maintenance cost factor is $\frac{1}{3}$.



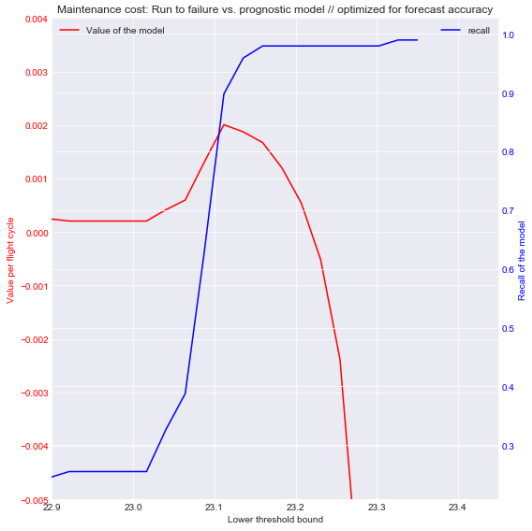
(c) Maintenance cost factor is $\frac{1}{5}$.



(d) Maintenance cost factor is $\frac{1}{7}$.

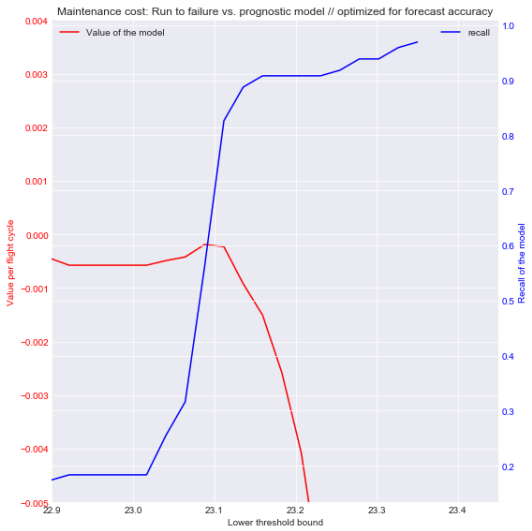
Figure A.1: Net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors. Optimized for prognostics accuracy metrics.

(a) Maintenance cost factor is 1.



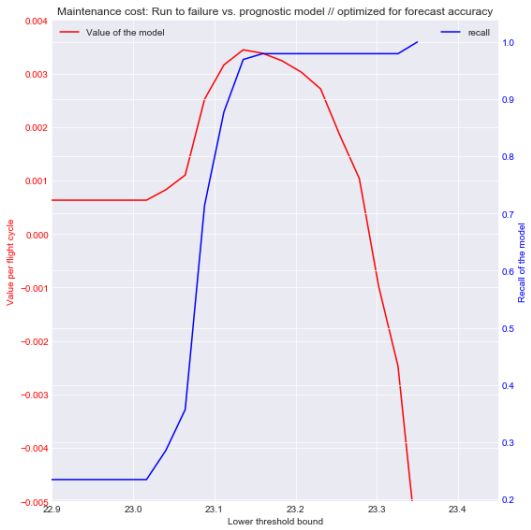
(b) Maintenance cost factor is $\frac{1}{3}$.

(c) Maintenance cost factor is $\frac{1}{5}$.

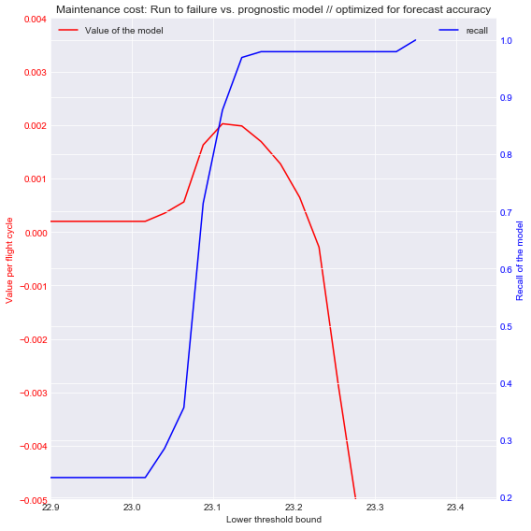


(d) Maintenance cost factor is $\frac{1}{7}$

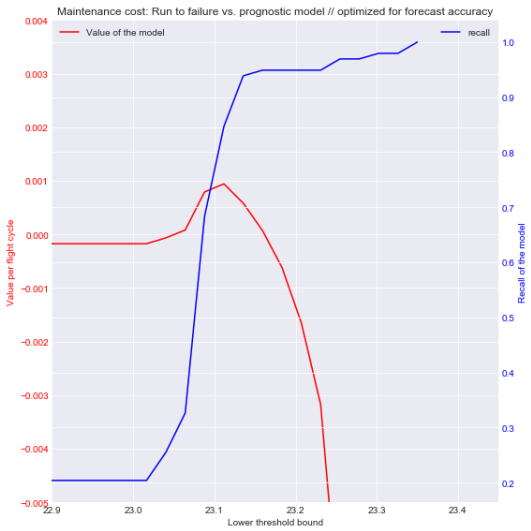
Figure A.2: Net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors. Optimized to maximize AUROC.



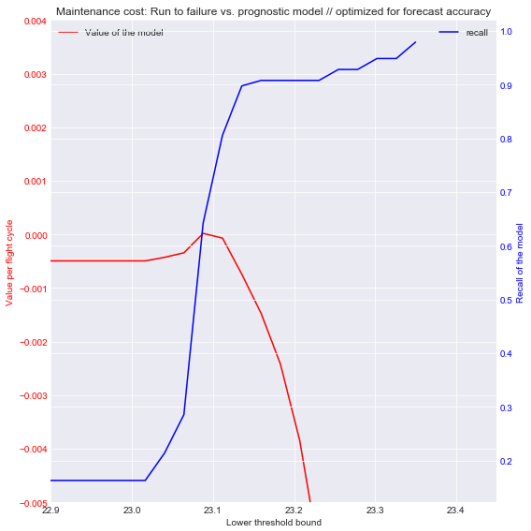
(a) Maintenance cost factor is 1.



(b) Maintenance cost factor is $\frac{1}{3}$.

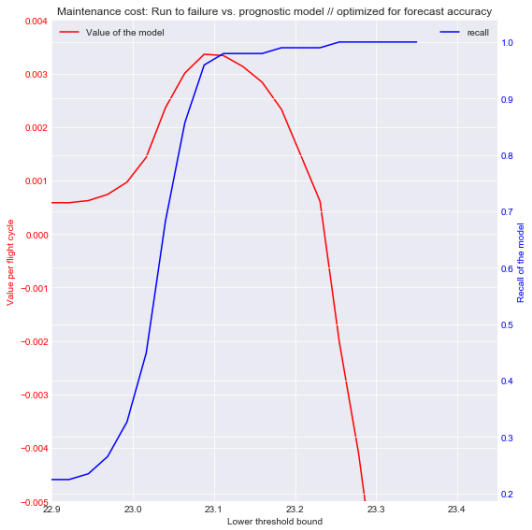


(c) Maintenance cost factor is $\frac{1}{5}$.

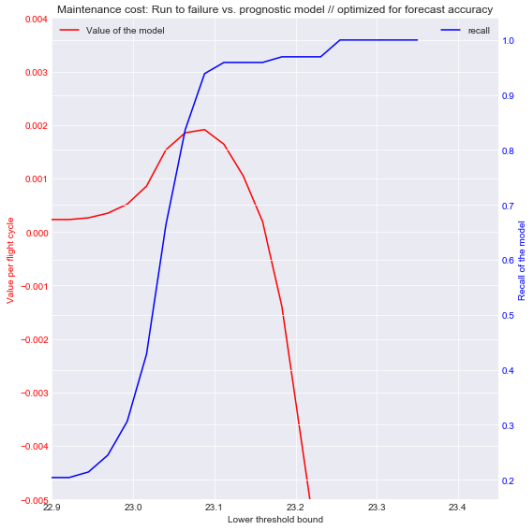


(d) Maintenance cost factor is $\frac{1}{7}$.

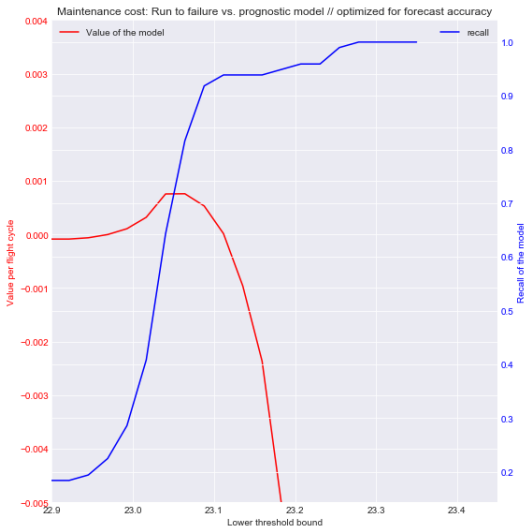
Figure A.3: Net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors. Optimized to maximize area under precision-recall curve.



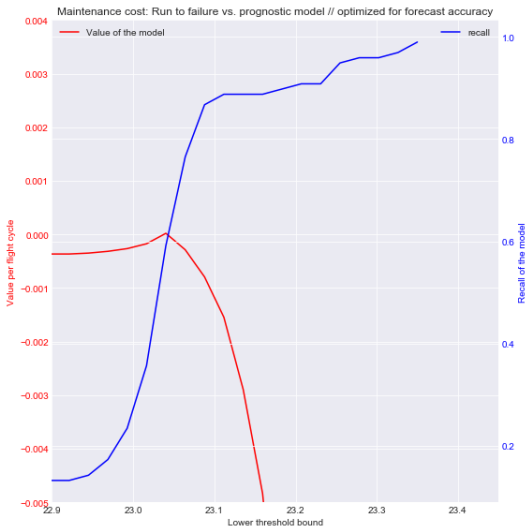
(a) Maintenance cost factor is 1.



(b) Maintenance cost factor is $\frac{1}{3}$.



(c) Maintenance cost factor is $\frac{1}{5}$.

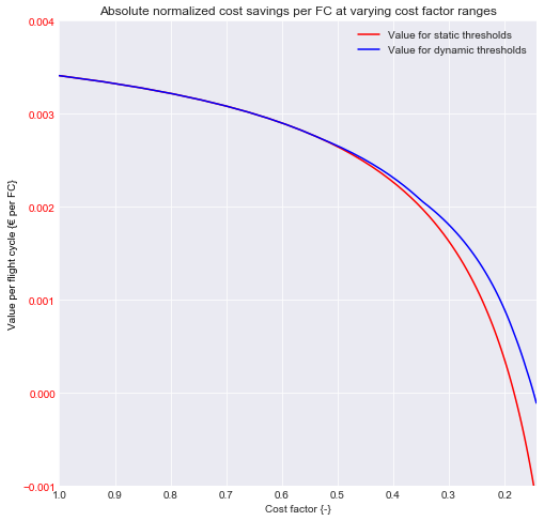


(d) Maintenance cost factor is $\frac{1}{7}$.

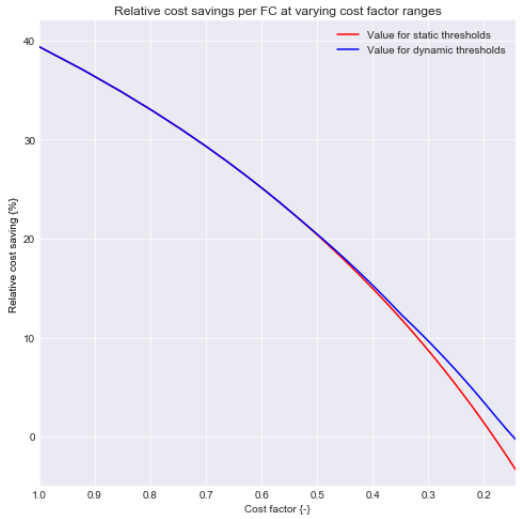
Figure A.4: Net value per unit of time of the integrated CBM model over a run-to-failure paradigm for different cost factors. No optimization strategy.

B

Absolute and relative financial impact graphs on a continuous range of cost factors for different model optimization configurations

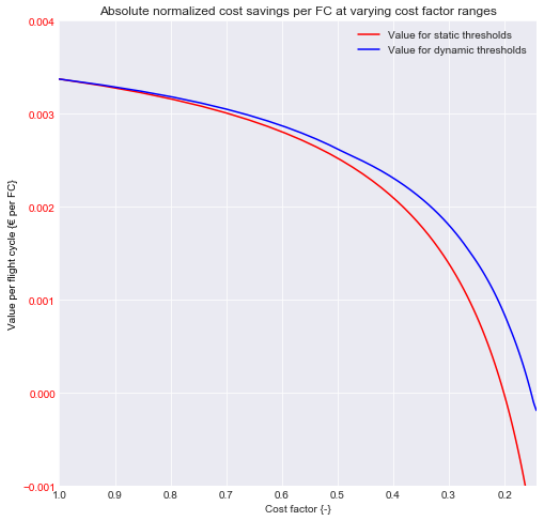


(a) Normalized absolute cost saving.

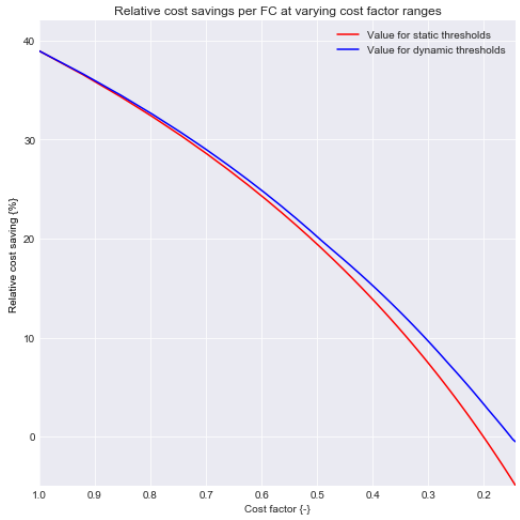


(b) Relative cost saving.

Figure B.1: Normalized absolute and relative cost saving for varying cost factors with model configuration that minimizes forecast error (GM(1,1) alpha = 1.1 and forecast horizon 15 FC). Note the higher cost savings for the dynamic versus the static threshold heights.

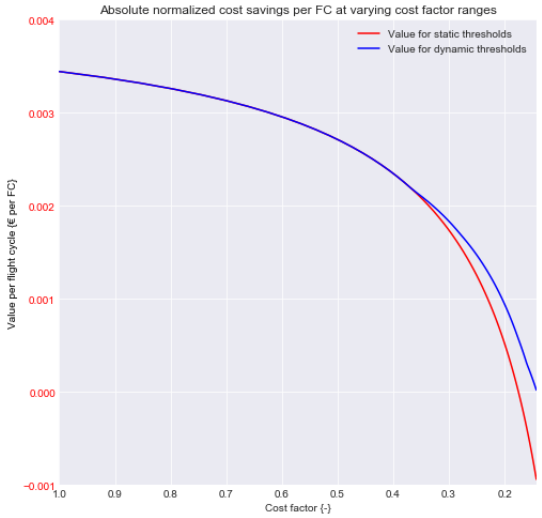


(a) Normalized absolute cost saving.

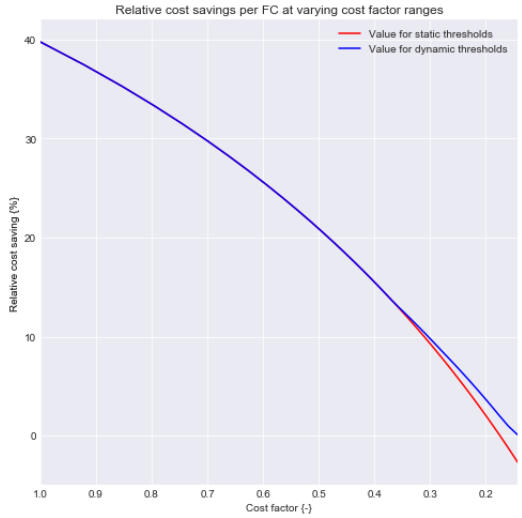


(b) Relative cost saving.

Figure B.2: Normalized absolute and relative cost saving for varying cost factors with model configuration that maximizes the AUROC performance metric (GM(1,1) alpha = 0.9 and forecast horizon 15 FC). Note the higher cost savings for the dynamic versus the static threshold heights.

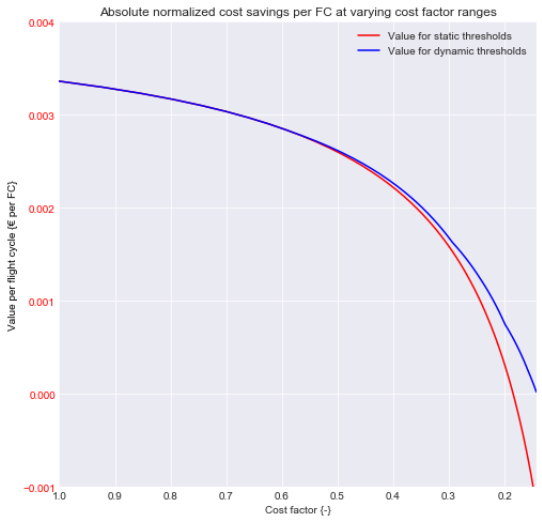


(a) Normalized absolute cost saving.

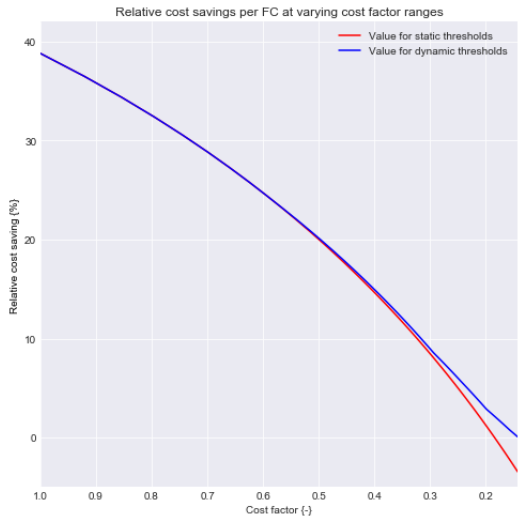


(b) Relative cost saving.

Figure B.3: Normalized absolute and relative cost saving for varying cost factors with model configuration that maximizes the precision-recall curve (GM(1,1) alpha = 0.7 and forecast horizon 13 FC). Note the higher cost savings for the dynamic versus the static threshold heights.



(a) Normalized absolute cost saving.

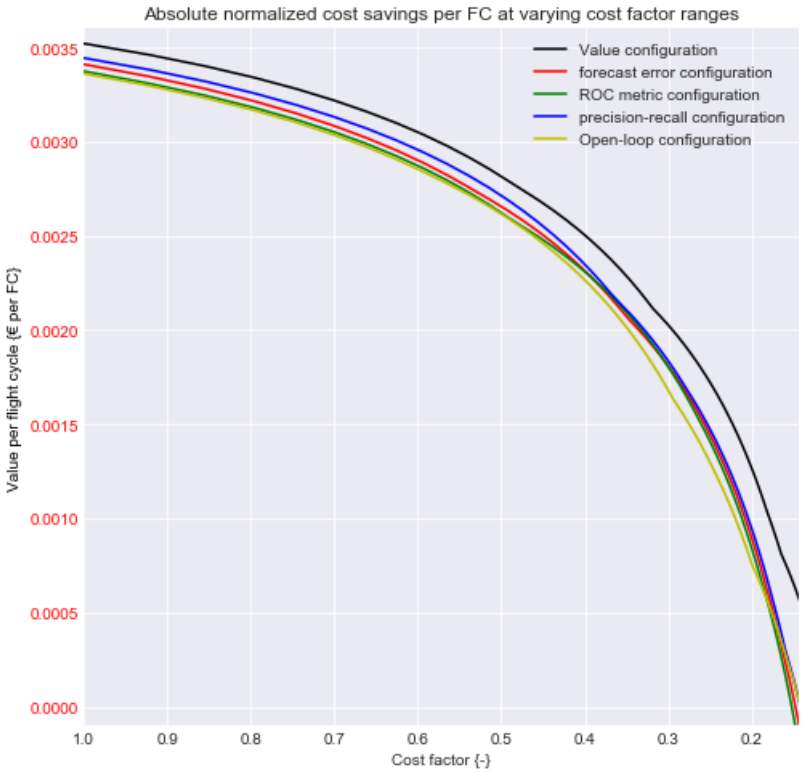


(b) Relative cost saving.

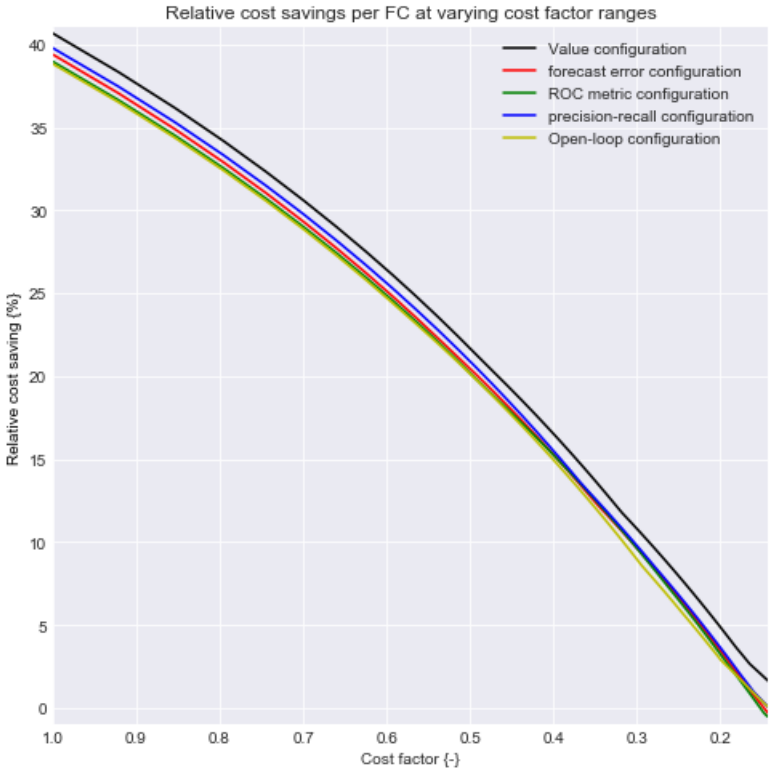
Figure B.4: Normalized absolute and relative cost saving for varying cost factors with "default" model configuration ("open-loop") (GM(1,1) alpha = 0.5 and forecast horizon 12 FC). Note the higher cost savings for the dynamic versus the static threshold heights.

C

Absolute and relative financial impact graphs comparison of the different model optimization configurations



(a) Normalized absolute cost saving.



(b) Relative cost saving.

Figure C.1: Normalized absolute and relative cost saving comparison of the different model configurations.Note the higher cost savings for the value oriented approach configuration