Requirements Flowdown for Prognostics and Health Management

Abhinav Saxena[1], Indranil Roychoudhury[2], and Jose R. Celaya[3]
SGT Inc., NASA Ames Research Center, Moffett Field, CA, 94035

Bhaskar Saha[4], and Sankalita Saha[5]
MCT Inc., NASA Ames Research Center, Moffett Field, CA, 94035

and

Kai Goebel[6]
NASA Ames Research Center, Moffett Field, CA, 94035

Prognostics and Health Management (PHM) principles have considerable promise to change the game of lifecycle cost of engineering systems at high safety levels by providing a reliable estimate of future system states. This estimate is a key for planning and decision making in an operational setting. While technology solutions have made considerable advances, the tie-in into the systems engineering process is lagging behind, which delays fielding of PHM-enabled systems. The derivation of specifications from high level requirements for algorithm performance to ensure quality predictions is not well developed. From an engineering perspective some key parameters driving the requirements for prognostics performance include: (1) maximum allowable Probability of Failure (PoF) of the prognostic system to bound the risk of losing an asset, (2) tolerable limits on proactive maintenance to minimize missed opportunity of asset usage, (3) lead time to specify the amount of advanced warning needed for actionable decisions, and (4) required confidence to specify when prognosis is sufficiently good to be used. This paper takes a systems engineering view towards the requirements specification process and presents a method for the flowdown process. A case study based on an electric Unmanned Aerial Vehicle (e-UAV) scenario demonstrates how top level requirements for performance, cost, and safety flow down to the health management level and specify quantitative requirements for prognostic algorithm performance.

Nomenclature

α	=	accuracy parameter to specify allowable prediction error bounds
α^+	=	maximum late prediction error allowable
α^-	=	maximum early prediction error allowable
β	=	total probability sum of the predicted RUL pdf between α-bounds
β^+	=	total probability of late prediction beyond a specified α^+
β^-	=	total probability of early prediction below a specified α^-
λ	=	time window parameter specifies the time when performance should meet desired specifications

[1] Research Scientist, Intelligent Systems Division, MS 269/4, AIAA Member.
[2] Computer Scientist, Intelligent Systems Division, MS 269/3.
[3] Research Scientist, Intelligent Systems Division, MS 269/4.
[4] Research Scientist, Intelligent Systems Division, MS 269/4.
[5] Research Scientist, Intelligent Systems Division, MS 269/4.
[6] Senior Scientist, Intelligent Systems Division, MS 269/4, AIAA Member.

I. Introduction

Prognostics and health management (PHM) is an active field of research in the systems health management community. Besides its benefits to the safety of a system, PHM is a critical element for the lifecycle management of a system. As such, it needs to be considered during the design of a system because the resulting product may be designed differently to enable the functionality of PHM. Since there have been very few implementations of prognostics for critical systems, to improve the Technology Readiness Level (TRL), more progress needs to be made in overcoming certain shortcomings, such as a lack of available run-to-failure data, accelerated ageing environments, real-time prognostics algorithms, Uncertainty Representation and Management (URM) techniques, prognostics performance evaluation, and methods for verification and validation, to name a few. One other important need is a systematic method to derive specifications for prognostics. Even when PHM is (as is often the case today) an add-on functionality, its requirements need to be expressed and handled in a similar fashion. Today, there is a lot of Science and Technology (S&T) development under way that produces PHM solutions at the technology level. However, these products (e.g. algorithms, methods, software architectures, etc.) cannot presently be consumed because there is a lack of understanding of how to express the needs at the different levels of the requirement ladder. It is also necessary to express the requirements for PHM properly and to flow down the requirements in accordance to system engineering principles.

To this end, this paper (i) provides a perspective of how important requirements are for Validation and Verification (V&V) of Prognostics and Health Management (PHM), and (ii) establishes a method to flow down these requirements to a level where they translate to performance numbers at the algorithmic level. The paper presents one (out of many possible) connection between high-level requirements and performance at the algorithmic level. Specifically, a step-by-step process is provided for requirement flowdown and it is shown via a case study how the requirements can be derived in a methodical manner. Generally, such a process would set performance goals for algorithm developers and at the same time allow them to escalate concerns or negotiate if some of the performance goals are practically not feasible. This negotiation would ultimately result in reasonable expectations on between developers and users. It would also provide a better understanding of what may or may not be achievable at the high level given the constraints at low levels.

From an engineering perspective some key parameters driving the requirements for prognostics and health management include: (i) maximum allowable Probability of Failure (PoF) of the prognostic system to bound the risk of loss of asset, (ii) maximum tolerable probability of proactive maintenance to bound unnecessary maintenance, (iii) lead time to specify the amount of advanced warning needed for appropriate actions, and (iv) required confidence to specify when prognosis is sufficiently good to be used. However, it is not completely clear how to derive these requirement specifications. Initial steps show promise, such as a generalized PHM-Value model[1] that defines performance metrics from Original Equipment Manufacturer (OEM)/service provider and customers' points of view, and then connects them to high level goals to extract requirements.

In a similar spirit, this paper extends our previous work on various aspects of requirements specifications[2] and takes a systems engineering view towards the requirements specification process and delineates what drives performance requirements for a prognostics system. This paper also identifies various components that must come together to specify requirements; and then investigates what has been done in the industry in those areas, and whether some or any of it can be reused, or enhanced, to incorporate prognostics requirement specification process. The paper concludes with a case study for a requirements flow down for prognostics for the power storage on an electric Unmanned Aerial Vehicle (e-UAV) Edge 540. It demonstrates how top level requirements relating to performance, cost, and safety flow down to the health management level and specify more quantifiable requirements on prognostic algorithm performance. Several prognostic performance metrics were developed in a previous effort[3] to allow quantifying algorithm performance that consider several health management aspects related to time criticality and prediction accuracy. These metrics include several performance parameters that are governed by top level requirements. This case study presents an example of how these parameters can be derived from high level customer requirements in a systematic manner. In addition, some lessons learned during the requirements flow down exercise are presented, which in our opinion give valuable insights into the process.

This paper is organized as follows. Section II presents findings from an extensive literature review in the PHM domain. A brief overview of various prognostic performance metrics and significance of corresponding design parameters is provided in Section III, which provides the infrastructure for specifying requirements for prognostic algorithm performance. Section IV describes the e-UAV application scenario including top level mission objectives that are then used to derive metrics design parameters though a systematic flow down. Section V concludes the paper.

II.   Requirements in PHM Literature

Looking back over a decade, the understanding of prognostic principles and its potentials has gained substantial ground. It has become apparent that prognostics and health management needs to integrate seamlessly into the systems engineering (SE) process in a system's life cycle. Out of many other key steps in SE, a systematic procedure for defining requirements and specifications for PHM systems have been identified as one of the key gaps that must be bridged before end-to-end integrated successful PHM systems may be realized. While looking for methods and approaches for specifying requirements on PHM systems, and prognostics algorithms in particular, it is evident that this shortcoming is poorly addressed, and often ignored, because of its complexities. Lack of established SE processes for PHM is hindering PHM from being part of the design cycle of any production system. A review of existing literature shows that requirements flowdown and specification for PHM, and diagnostics and prognostics in particular, have not been covered, and there exists a lack of methodology that connects the effects of prognostic performance to high level system goals. The papers can be roughly categorized into following classes, shown in Table 1. The column 3 in the table provides the distribution of papers into these seven categories and corresponding references to the papers are listed in column 4. The last category (#7) is the main interest of this paper.

Table 1. Summary of literature review from papers discussing PHM requirements.

No	Category Description	#Papers (%)	References
1	Papers by PHM practitioners discussing SE principles and concepts on requirements from a theoretical standpoint	4 (10)	4-7
2	PHM related papers with the word 'Requirement' in title but requirements not discussed explicitly or in a relevant context in the main discussion	2 (5)	8, 9
3	Papers discussing only high level functional requirements but no flowdown to algorithm level, or that discuss the importance of requirements, specifications, flowdown, etc. for PHM	9 (22.5)	2, 4, 6, 9-14
4	Requirements are expressed quantitatively, but specification numbers chosen arbitrarily and not based on actual flowdown	9 (22.5)	8, 12, 15-21
5	Papers discussing Cost-Benefit analyses without taking PHM performance into account	14 (35)	15, 19, 21-32
6	Papers discussing Cost-Benefit analyses taking PHM performance into account. Performance metrics are an input and the CBA models used can be used for a flowdown exercise	12 (30)	14, 16-18, 20, 33-39
7	An attempt to carry out actual flowdown or develop a flowdown method for PHM performance requirements	2 (5)	3, 21, 37

It is worthwhile to mention that almost all papers discuss the importance of requirements in implementing a PHM system, however, only a very small fraction actually presents a methodical process for it. Furthermore, a large fraction of initiatives on PHM efforts resort to selecting performance numbers on an ad-hoc basis without any careful flow down from the top level requirements. Lastly, there exist several modeling and simulation studies that are primarily aimed at cost-benefit analysis (category #6 in Table 1), which we feel, could also be used in deriving requirements from top levels goals. Since various performance specifications are inputs to these models, it may be possible to establish a right set of specifications to achieve targets defined at the high level. However, we do not see much evidence of such efforts. In this paper we illustrate how one could translate top level requirements to concrete performance specifications at the algorithmic level. Another key observation worth noticing is that whenever there are references to any prognostics metrics in literature, a common intent is reflected to make timely and correct decisions based on estimated system health. Notions of accuracy and precision of predictions, timeliness, and establishing confidence under uncertainties are common to most references. This supports the choice of metrics that were developed in Saxena *et.al.*[3] and that the four requirements stated in Section I above are central to prognostics based decision making. In the remainder of the paper, we first describe prognostics performance metrics and the associated parameters that were developed in earlier works[3, 40-42]. Then, using a case study on an electric-UAV we show how one can derive these performance parameters such that the top level requirements are satisfied.

III.   Prognostics Performance Metrics

The paradox of prognostics is that performance can only be evaluated in retrospect, i.e. not until the predicted event has actually taken place. Metrics were designed for an offline evaluation assuming the availability of the ground truth information about the EOL to evaluate the prediction performance. The premise is that an algorithm will conform to specified requirements and prove so by meeting the specifications on several relevant validation cases. There are other performance indicators[43] that have been proposed for an online application but they only qualify the prediction process quality such as stability, precision, but do not address accuracy aspects, since it

requires the knowledge of ground truth that would be realized in the future. The discussion in this paper is focused on two key metrics that we developed in earlier works[40-42], namely the Prediction Horizon (PH) metric and the α-λ metric that represent accuracy, precision, timeliness, and prediction confidence. Several parameters in these metrics definitions encode requirements, and are described below.

A. Metrics parameters driving requirements

1. Lead time to specify the amount of advanced warning needed for appropriate actions - λ

Extending the definitions as originally described in Saxena *et.al.*[3]. In this section, we generalize the interpretation of some of the metrics parameters to a broader class of application scenarios. A time window parameter, λ, specifies the time at which the performance is evaluated during the system's lifetime. This parameter is expressed as a percentage of the lifetime left from the onset of fault (often approximated as the time when the fault is detected) to end of life of the system. It is assigned a value such that using that information; a reasonable actionable decision will be possible in a timely manner. For comparing multiple algorithms on a single dataset, the value of λ remains the same and does not change with the algorithm, and hence, expression of this parameter as a percentage of remaining life time is acceptable, as originally suggested in Saxena *et.al.*[3]. However, to compare the performance of a particular algorithm on multiple run-to-failure datasets that are likely to have different lifetimes, expression of λ as a percentage would yield different absolute values since different runs are likely to exhibit different EOLs, whereas the time required for an actionable decision may not change for those different cases. For instance, time required to perform a particular maintenance action is likely to remain the same irrespective of how varied the failure times of different systems in a fleet are. Therefore, we suggest expressing λ in absolute time units that is indicative of the time required to take an action and does not change based on the actual value of the EOL. It is also connected to PH such that it must be achieved before the time instant t_λ. It must be noted that PH specifies the time when an algorithm performance converges to within desired specifications and, hence, specifies the maximum advance warning an algorithm can provide before EoL occurs[3].

2. Tolerable limits on early prediction to bound unnecessary maintenance – α^-, β

The notion of desired accuracy (or acceptable error) is represented by α parameters (see Figure 1), where α^+ denotes the maximum bound on late prediction and α^- represents the lower bound on an early prediction[3].
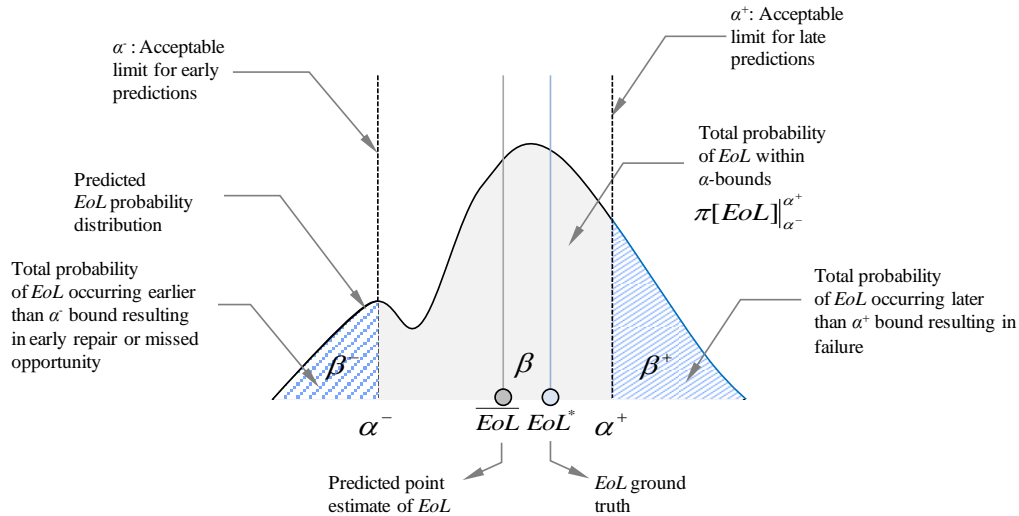

Figure 1. Schematic illustrating various prognostic performance metrics parameters.

It has been argued in the literature that both early and late predictions are attributed with penalties. However, the distribution of these penalties is often non-symmetric. A late prediction, i.e. the failure occurs before the predicted time, may have catastrophic consequences. In contrast an early prediction cuts down on the utilization of the available resources or represents a missed opportunity. In such cases we expect heavier penalties for late predictions, which may sometimes suggest favoring conservative predictions to avoid heavy penalties when optimizing an overall cost function for decision making. Therefore, in order to bound the predictions from being overly conservative, the α^- parameter is defined, which signifies the tolerable limits on early predictions. Furthermore, it may be useful to limit the rate of early predictions that violate the bound. This requirement may be specified limiting

the probability of early predictions, which is computed by integrating the portion of the predicted EoL pdf prior to the α^- bound (see Figure 1). Limiting the maximum value of this probability, by specifying a parameter β^-, limits the acceptable bound on early predictions i.e. the scenarios that the predicted event occurs earlier than the time specified by α^-.

3. Tolerable limits on late predictions to bound the risk of failure – α^+, β^+

Late predictions (failure occurring before predicted EoL) often have more significant impact especially when a late action may result in a failure or complete loss of assets. Safety critical systems therefore have very little or no tolerance for late predictions and often work with conservative estimates of when a failure would occur. However, there are cases where late predictions may result in downtime but are not safety critical or where the end-of-life (EoL) criteria are intentionally defined before the actual failure threshold. In such cases it is still not desirable to make late predictions. However, there is some tolerance for failures that occur due to an incorrect (late) prediction. The goal in such cases is to minimize the probability of late predictions. In Saxena *et.al.*[3], a bound for acceptable late prediction is expressed by the parameter α^+ expressed as a percentage of EoL time. For a more stringent performance requirement, α^+ is suggested to be expressed as a percentage of RUL value at any given time so that the allowable error actually shrinks as RUL decreases and the system nears its EoL. This requires prediction accuracy to improve as time goes by. In order to define the risk posed by the portion of estimated EoL probability density function (pdf) that violates the α^+ bound, another parameter β^+ is introduced, which is computed as sum total of the probability crossing the α^+ bound (see Figure 1). This allows quantitatively connecting the tolerable risk of loss to performance specifications for an algorithm whose output is an estimate of EoL pdf.

4. Required confidence to specify when prognosis is sufficiently good to be used – β

Prognostic algorithms inherently contain uncertainties that arise from a variety of factors and, therefore, it is expected from a prediction algorithm to account for such uncertainties and provide an estimate for them. Consequently, prognostic algorithms often estimate the uncertainties in the predicted quantity and express them as probability distributions, sampled outputs, histograms, etc. These estimates can be used to infer the variability (spread) in predictions. In order to make decisions based on prognostics point estimates are first computed from these pdfs to identify the most probable time for expected failure. Choosing mean of the pdf has been the most favored choice in most applications, however, Saxena *et.al.*[41] argues that choosing a median is better in practice. Furthermore, it was suggested that a comparable result is obtained by guaranteeing that the total probability between the two α bounds is at least 50% represented by the parameter β. The higher the value of β, the more is the confidence in a prediction that it will remain within acceptable α bounds. If the sum total of probability between the α-bounds is less than 50% for any pdf, it should be interpreted as a low confidence prediction and should not be used for decision making.

IV. A Case Study

In order to derive concrete specifications for prognostics requirements, first, one must define what prognostics can do for the system of interest in qualitative terms. For instance in a battery operated electric UAV scenario, prognostics on remaining battery charge would help in planning of waypoints based on current state of charge (as measured/estimated) and expected future operational loading. Specifically, it allows determining which maneuvers to carry out and which ones not to in order to keep the system safe while accomplishing the mission to the best possible extent. For safety critical systems the foremost advantage of prognostics is in avoiding a catastrophic failure and then to accomplish the mission to the best possible extent by providing actionable information that allows operating for longer durations than without that information. Additionally, the financial benefit that commercial operators would enjoy is also significant and has been the main motivation for fielding PHM. Since prognostics is always accompanied by an associated confidence level (often expressed as the probability of predicted outcome being true), the user must assess the risks for decisions made based on prognostics where the predictions may not hold true in future. This risk assessment should be associated with loss of vehicle, life, damage, loss of mission, and so on, and preferably be expressed in monetary terms. At the same time the user should express a maximum bound of such losses that can be tolerated if an incorrect decision were in fact made. These concepts are further explained and exemplified through a case study presented in this section.

A. e-UAV Scenario

We take the case of an electric UAV with battery operated propulsion as an illustrative example. It should be noted that the power and propulsion systems have been identified as a major source of UAS accidents[44]. Loss of battery power would result in loss of aircraft control leading to accidents typically categorized under 'early flight

termination'[45] potentially resulting in a crash. Loss of aircraft to such accidents must be minimized to avoid not only the loss of asset itself but also to the secondary damage on the ground due to uncontrolled impact. Traditionally, in the aviation community, there exists a procedure to be followed to contain the risk due to aircraft accidents. A brief summary of that procedure as applicable to UAVs is provided here for a quick reference.

1. Safety Assessment and Derivation of Functional Requirements for UAVs

Accidents related to UAVs are divided into three broad categories (primary accidents) that are used in safety assessment, namely: (i) unintended or abnormal system mobility operation, (ii) mid-air collision, and (iii) early flight termination[45]. Secondary damage effects of these accidents are first analyzed to identify their possible outcomes (e.g. fatality rates) and accordingly a Target Level of Safety (TLS)[46] is determined based on mandated safety standards for manned aviation (specified in terms of acceptable frequency of ground impact or expressed as mean time between ground impacts). A TLS for ground impact accident rate cannot be directly used as a design specification but is used to determine reliability requirements on various components of the system through Operational Safety Assessment (OSA)[46]. First Operational Environment Definition (OED) is carried out to create a detailed list of all functions that UAV will perform and to subsequently identify the factors that can influence safety (Function Hazard Assessment (FHA)). This is done by decomposing various system level functions (e.g. aviate, navigate, communicate, mitigate hazard, etc.) into lower level basic functions. Possible failures are identified that are associated with these basic functions and categorized into severity levels (no safety effect, minor, major, hazardous, and catastrophic) as defined in the risk reference system for manned aviation. The risk reference system also defines a TLS for each of these severity levels. The next step in OSA is to evaluate the effects of component failures using the Fault Modes, Criticality, and Effects Analysis (FMECA) and a Fault Tree Analysis (FTA), with a final goal to determine appropriate component reliability levels, and if there are other means to control hazards.

It must be noted that the desired component level reliabilities derived in this fashion are only used to schedule a maintenance and/or repair activities based on a typical schedule based or inspection based maintenance. Prognostics and health management, on the other hand, makes recommendations based on continuous condition monitoring and predictive outlook provided by the prognostic algorithms. Therefore, it is required to extend the requirements flowdown further and connect the reliability requirements derived from top level to the specifications at prognostics performance level. In developing our scenario we follow a similar approach as described above while not going into the detailed procedures and keeping the example simple and focused. We assume availability of key pieces of information that would be generated from a FMECA and FTA for the e-UAV and also assume desired TLS information is given to us. Furthermore, among various functions that the e-UAV is required to perform, we only focus on those that are the target of the health management system in this example (i.e. the battery system) and that are required to complete a specific mission as described in the scenario.

2. Risks due to Battery Failures

Batteries are the power sources for the electric motors driving the electric aircraft. Ideally, a fully charged battery should be able to last for the entire duration of the mission with some power left as a backup for extreme situations. However, due to several uncertainties inherent to an aircraft scenario, the charge consumption rates are quite different that makes it difficult to accurately estimate how long the aircraft might operate on a given set of batteries. The most common sources of such uncertainties in battery life include (but are not limited to) the following:

1. Battery state-of-charge (SOC) and Battery state-of-health (SOH) (current battery capacity, or presence of a fault) that are typically not known or measured accurately and are estimated based on best available model and/or sensing capabilities, which themselves have inaccuracies.
2. Sensor or measurement noise that accounts for uncertainties in estimated battery health.
3. Ambient conditions that comprise of varying temperatures, wind speeds and direction leading to uncertainties in estimating future load profiles.
4. Inherent variations in control inputs (pilot responses) for given maneuver sequences for a flight plan.

Characterizing these sources of uncertainty beforehand and using this information in an algorithm produces confidence bounds on the predictions that must be accurate and narrow enough to make safety related decisions. In the absence of any forewarning, operators act very conservatively and often cut down on mission duration even if the battery still had enough charge left. Employing a battery health management system, the operator can not only track the latest battery SOC but also get a predictive estimate of how much time might be left before the battery runs out of power. This can extend the duration of the mission and also reduce the risk of unwanted early flight termination scenarios. A more detailed description of this scenario and relevant prognostic algorithms providing evidence of reduction in risk of loss of flight control can be found in [47, 48].

3. High Level Requirements

Following the project development principles it can be generalized that all top level requirements fall under one of the *performance, cost, and schedule* goals, described below. Depending on the end application, relative weights for these goals change and influence lower level requirements accordingly. Furthermore, interpretations of the high level goals often changes from one application to another. For instance, in one case, performance may mean just adhering to a planned trajectory; but be broadened to ensure safety aspects in others. Likewise, costs can be tangible (operational costs, maintenance savings, etc. expressed in dollars) or intangible (perceived cost of an incomplete mission, diminished reputation, etc.) components.

These high level goals are first translated into functional requirements to specify what a system must do. Highest level requirements often originate from project management or strategic planning levels, which flow down into specifics for execution. This e-UAV case study is based on a research effort towards a 3-D demonstration of prognostics and decision making under NASA's Aviation Safety Program. Therefore, there are constraints for time schedule, budgets allocated, and research objectives comprising successful demonstration and aircraft safety.

At the highest level the goal statement read "*Carry out prognostics and decision making demonstration by flying an e-UAV to complete a specified research mission by a given milestone deadline in a cost effective and safe manner*". This statement can be further dissected and augmented with additional information to extract various requirements and constraints such as following:

- *Performance Goal*: a successful demonstration, fly for desired mission duration, fly safely
- *Cost Goal*: stay within budgetary limits, avoid loss of aircraft (safety)
- *Schedule Goal*: meet milestone deadline

Performance goals often translate into functional requirements, which can be further decomposed into requirements on various subsystems and components of the aircraft. As these requirements branch out, more granularity is achieved by gathering more details for concrete specifications. In this example top level performance, cost, and schedule goals are further expanded as shown in Table 2.

Table 2. Listing top level goals and corresponding functional requirements.

Goal	Metric	Functional Requirement
1. Performance	1.1 Successful demonstration	Carry out research objective – collect relevant measurement data
	1.2 Flight duration	Fly 20 minutes to complete the mission
	1.3 Safety	Land safely each time and contain the risk of loss to less than 4%
2. Cost	2.1 Cost of PHM Implementation	Minimize implementation cost and stay within stipulated budgets
	2.2 Cost of carrying out mission flights	Minimize cost per mission (resources, time, procurement) to allow adequate number of flights
	2.3 Tolerable risk of losing the asset	Minimize chances of loss of aircraft or inflicting damage due to unsafe landings
	2.4 Maximized mission duration	Minimize the loss of opportunity by not flying to avoid unfinished experiments
3. Schedule	3.1 Project timeline	Avoid delays and finish by project deadline

These high level functional requirements must be achieved individually or collectively by different interacting subsystems. For instance a safe flight during the mission requires that all subsystems of the aircraft function properly, availability of adequate resources to fly the aircraft, conformation of flight profile to safe flight envelop, and so on. All these requirements flow down as different branches, which may or may not intersect at lower levels. As an initial exercise a Quality Function Deployment (QFD)[49] analysis was carried out to identify various branches these top level requirements flow into. The tree quickly grew and it was decided to focus on the branches that were connected to health management aspects at lower levels. For reference a brief description and lessons learned from the QFD analysis is provided in the Appendix. In this example we illustrate the concept of flowdown by following the branch that is concerned with ensuring adequate battery power till the aircraft lands safely. Quantifiable limits specifying such requirements are arrived at by considering various factual data as well as desired performance.

Figure 2 below shows a safety assessment tree for the e-UAV used in this case study, a discussion that takes place on the customer side. This analysis derives relative contributions of different factors that can lead to failures. The nodes in this tree represent various categories the different fault modes fall into and the connections represent

corresponding contributions expressed as percentages. It describes how different subsystems can exhibit failures leading to loss of aircraft. It is determined that about 80% of problems to the aircraft occur due to battery related malfunctions. Notably such information may be derived from FMECA studies. There can be several root causes of battery malfunction such as using an aged battery (with diminished capacity), presence of faults in the battery system, usage beyond recommended specifications (extreme operational loads or environmental conditions), etc. It is expected, however, that preflight inspections and regular maintenance operations detect many of these causes before the flight, and a timely maintenance action (replacement or repair as applicable) can alleviate most of the battery failures. However, unexpected loads or environmental conditions can occur and a condition based assessment using continuous monitoring must be utilized. Therefore, in this example it is estimated that of all battery problems that can occur, only about 50% can be mitigated using PHM and the rest must be covered via maintenance operations, both of which have their own success and failure rates. Once again such assessments or case analyses are carried out by, or with the help of, the customer. These analyses yield numerical constraints on achievability of targets for a PHM system. In this example it is given that a Battery Health Management (BHM) system is able to manage 95% of all failures it is monitoring. This means that risk due to BHM's failure to respond accounts for 2% of the total number of failures whereas the total risk of failure due to all factors combined is 4.7%.
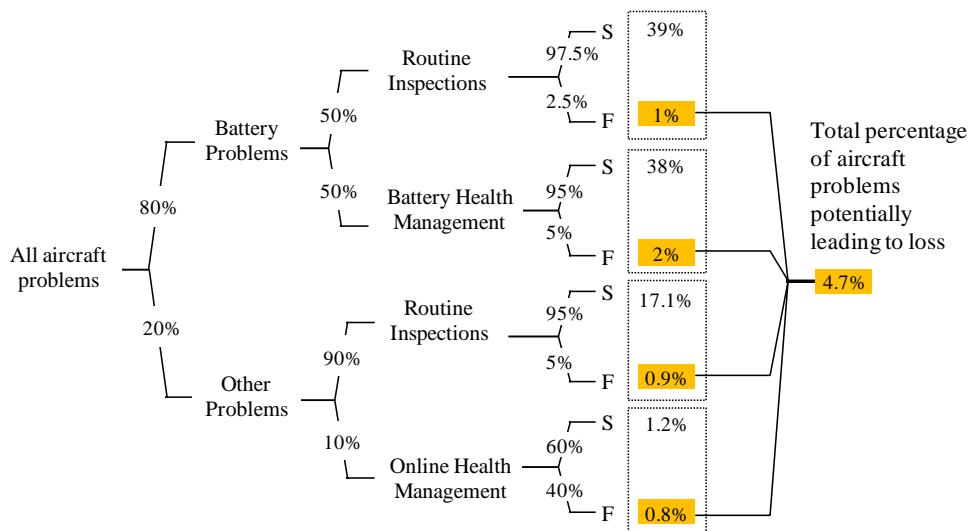
Figure 2. Safety assessment tree for e-UAV and requirements flowdown.

It must be noted that there are always other requirements that feed into a particular branch to influence or modify a requirement when flown down to a lower level. For instance there may be additional long term goals that guide the development of the PHM system; such as a goal of implementing and utilizing PHM information in an intelligent battery usage management to extend the overall life of the battery, and not just ensuring completion of a specific flight. An example of such usage management could be controlling the depth-of-discharge to avoid excessive capacity degradation, or autonomous reconfiguration of power routing to minimize further damage to a weak battery among several battery modules (from redundant or other subsystems). Clearly, this particular goal is derived from minimizing overall project cost by reductions in additional battery procurements required while carrying out multiple mission flights over the course of entire project duration. However, to maximize mission duration it might be important to use a battery to a deeper depth of discharge, which is clearly a conflicting goal and a balance must be struck.

B. Deriving Requirements for Prognostic Algorithms
In this section we will derive various prognostics performance parameters as described in Section III from the top level requirements.

1. Lead time (λ):
The top level requirements did not spell out the lead time requirement specifically. However, while flowing down the safety requirement it is important to identify the time for advance warning so an appropriate decision may be taken and a corresponding action carried out without putting the aircraft to risk. In this case, we consulted the customers (aircraft operators) to find out how much warning was needed. It is specified that conditional on the wind

conditions the aircraft operator can take between 1.8 to 2 minutes to land the aircraft. Therefore, the corresponding requirements for a prognostic algorithm are derived as shown below:

- *The prognostic algorithm shall predict no later than two minutes before the battery voltage cutoff threshold is reached*
- *The prognostic algorithm shall raise an alarm when the battery remaining useful life reduces to two minutes*
- *The prognostic algorithm performance shall meet desired prediction confidence levels at time $t_\lambda = 2$ minutes*

Following the definitions as developed in Saxena *et.al.*[3] the value of λ can be normalized by total mission duration, i.e. 20 minutes, which results in $\lambda = 0.9$ or 90% as shown below.

$$\lambda = \frac{20-2}{20} \times 100 = 90\%$$

Specifying λ determines the critical time by when the algorithm performance should converge to within the specified accuracy limits and attain specified confidence.

2. Limits on early prediction– α^-, β^-

The high level functional requirement #2.4 in Table 2 that calls for maximizing resource utilization towards completing a mission must be further qualified with specifications. In this scenario first the loss from early predictions must be quantified. Generally a monetary figure is associated with an assumed penalty to be incorporated in a cost benefit analysis on the customer's side. We assume that such analysis was duly conducted and as a result of the cost benefit analysis and the following requirement is derived:

"*The aircraft shall not land more than 0.4 minutes too early before the battery end-of-discharge more than 10% of the times*"

This requirement specifies the tolerance for an early prediction and the minimum rate (at least 90%) at which this requirement should be met. Using this information we can determine the α^-, β^- parameters at time $t_\lambda = 2$ minutes. The stated requirement will be met if an early prediction for a two minutes warning is made no earlier than 2.4 minutes to actual EoL. This can be seen from Figure 3(a) by interpreting the decision at coordinates $A(2.4, 2)$. In this figure we plot the predicted RUL versus the true RUL at any given time. Hence, the negative 45 degree line represents perfect prediction. As RUL decreases with time, note that the *x*-axis is drawn backwards. At coordinate $A(2,4,2)$, the actual RUL is 2.4 minutes but the algorithm predicts only 2 minutes RUL. If a decision to land is made at this point the aircraft would lose the opportunity to utilize remaining 0.4 minutes worth of battery power, which is the maximum limit on early prediction. Therefore any point to the left of 2.4 minutes line on the *x*-axis would violate the requirement. In order to draw the α^--bound, in accordance with the α-λ metric defined in Saxena *et. al.*[3], we connect the point $A(2.4, 2)$ to the true EoL point at coordinates (0,0) as shown by the black dotted line. Correspondingly, the point of intersection of the α^--bound and the vertical line at 2 minutes on true RUL (*x*) axis defines the limit for the predicted RUL distribution beyond which the prediction would be categorized into an early prediction. As shown in Figure 3(b) the total probability before the α^--bound, labeled as β^-, must be bounded by the specifications expressed in the requirement statement.

3. Limits on late predictions to bound the risk of failure – α^+, β^+

Flowing down from high level functional requirements #1.3 for safety and #2.3 for cost in Table 2, late predictions must be minimized. In our scenario the specified 2 minutes warning for a safe landing is assumed to be a hard limit and therefore would result in a certain system failure if full two minutes are not available for landing, which would be the case if the algorithms predicted late or did not provide specified prognostic horizon specified by the lead time requirement in our example. Therefore, the α^+-bound is determined to be the true EoL itself, i.e. α^+ is zero indicating a zero tolerance for late predictions. Any prediction beyond α^+ would not be useful and expected to result in system failure, which can be characterized by estimating the total probability from the RUL pdf extending beyond the α^+-bound and is represented as β^+. Given specifications on safety and risk at the high level we need to derive the bound on β^+ parameter. Given the specification in requirement #1.3 in Table 2, one must reduce the total risk of failure to 4% or less. Since in this case none of the other factors, i.e. health management for other problems, or improvements in routine inspection effectiveness are controllable, the entire onus of covering that additional 0.7% falls upon BHM performance. In order to achieve this goal the BHM performance must improve and should only allow 1.3% of the overall failures, which translates into 3.25% of the BHM failures. Therefore, while the current BHM allows β^+ to be 5%, it must be upper bounded at 3.25% to meet the requirements.

4. Required confidence in prognosis– β

As discussed in Section III earlier, the lower bound on β is suggested to be 50%. Generally, the larger the β more is the confidence in predictions. The bound on β works in conjunction with the α-bounds to specify performance constraints. As α-bounds get narrower (i.e. less error tolerated) the probability density functions is required to contain the spread in order to satisfy same β criterion. Furthermore, the value for β can be set to a higher value than 50% if it is desired to use higher confidence.
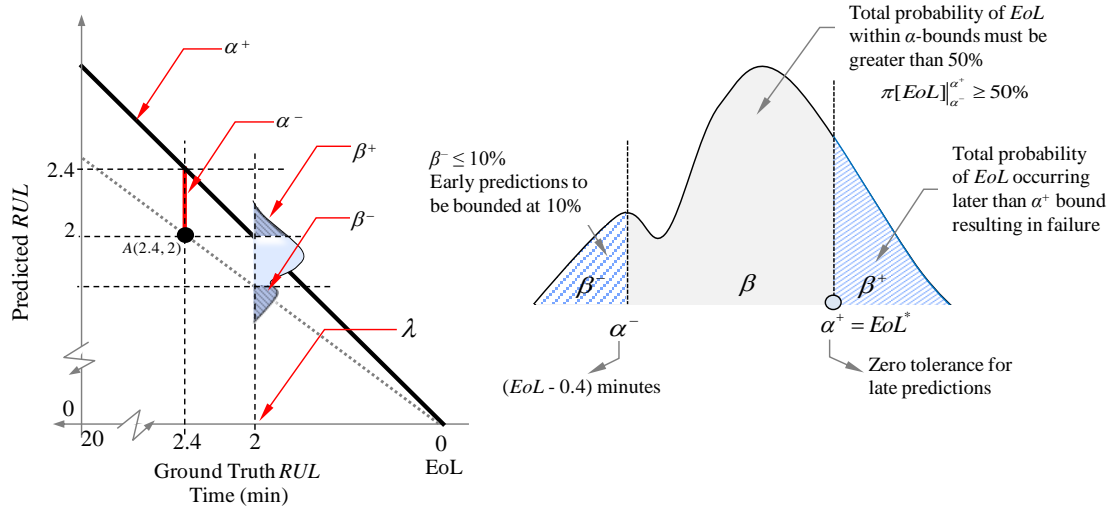
Figure 3.(a) Predicted RUL vs. True RUL (time) plot for the α-λ metric (b) Illustration of β parameters on an RUL probability density function.
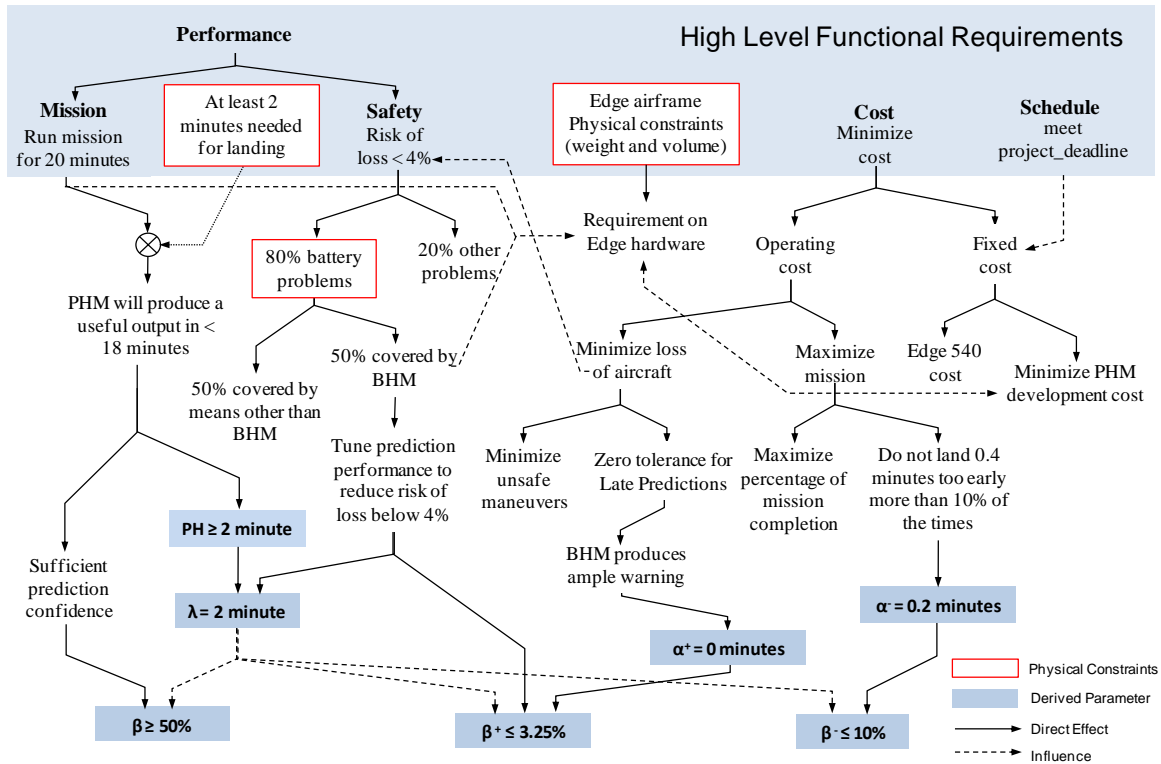

Figure 4. Requirements flowdown tree for the e-UAV scenario to derive performance parameters for the battery health management system.

American Institute of Aeronautics and Astronautics

V. Conclusions

In this paper a process to derive prognostics performance metrics parameters have been presented by means of an example based on an e-UAV scenario. The role of high levels goals and corresponding decomposition into high level functional requirements was carried out (see Figure 4). These high level functional requirements further flow down to lower levels and eventually to the lowest levels specifying requirements for prognostic algorithm performance. The prognostic algorithms are evaluated based on several prognostic metrics that take into account performance factors such as accuracy, precision, timeliness, and prediction confidence. In addition to flowing down through a primary branch, requirements often influence adjacent branches originating from other high level goals. In addition there are several physical constraints that the system must abide by, which therefore, influence the flowown as well.

While the process developed in this paper may not be the only way to carry out a flowdown, this is our first attempt to define such a process with a goal to facilitate inclusion of PHM design under the mainstream system engineering during a system's lifecycle. Several variables were intentionally left out to keep the example simple and focused, which need to be explored further. The process developed in this work needs to be expanded and applied to other examples to gauge its applicability to a general class of problems.

Appendix

As a first step of flowing down the top-level requirements of performance, cost, and schedule to various subtasks that will enable the fulfillment of these requirements, we started with the well-established technique of Qualitative Function Deployment (QFD). QFD is a well established methodology for enumerating and prioritizing different subtasks and requirements. However, the QFD analysis brought several of its shortcomings (with respect to our needs) to light. For example, the QFD task decomposition tree grew exponentially with the number of decomposition levels, and we had to leave out several branches that did not show clear connections to algorithmic performance at the lower levels.
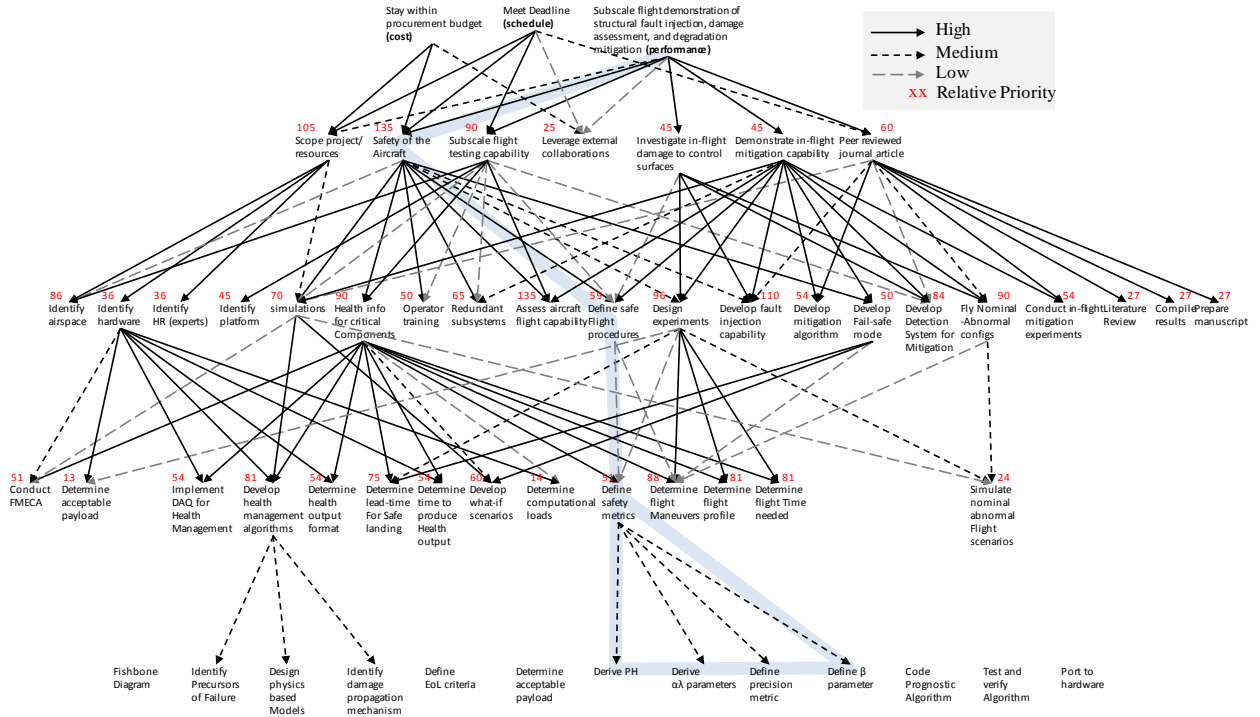
Figure A1. A portion of QFD tree for the e-UAV scenario presented in this paper. The branch concerning PHM performance requirements flowdown has been highlighted.

In our particular example, highlighted in Figure A1, we isolated a single branch to illustrate the flowdown process with an example, however, it may not easily be possible to isolate an individual branch in practice as various subsystems interact and often impose conflicting goals on each other. Also, QFD, as the name suggests, only deals with qualitative requirements, and does not provide provisions to flow down quantitative specifications. These

shortcomings motivated our research for improved methodologies for flowing down top-level requirements to low-level quantitative specifications dealing with algorithm performance. Our requirement flowdown approach, however, is inspired by QFD. Like in QFD, we too indentify several stakeholders, broadly classified as customers and vendors that dictate requirements and impose constraints in a system's development. While the customers are concerned with needing a system built to specifications and using it, the vendors are involved in building the system but not using it themselves. Just like in QFD, the requirements flow down first within the customer's context where high level requirements are broken and prioritized into desired functions and constraints. These specific functions and constraints then flow down on the vendor side, where first an assessment of feasibility is carried out keeping in mind the constraints of resources and if needed an iterative refinement and negotiation process takes place between the two sides. For reference the QFD analysis tree is illustrated here with four levels of qualitative flowdown (see Figure A1). The type of connecting lines indicate the relative weights with which a top level node influences the connected node at the lower level. An aggregate priority is also indicated with numbers, which ranks all nodes at a particular level to prioritize various subtasks. It must be noted that there is not one unique way of carrying out QFD or a requirements flowdown. It is greatly influenced by the skills of the analyst and is fairly subjective in nature.

Acknowledgments

References

[1] O'Flarity, S. *PHM Experience at UTC and Pratt & Whitney: Challenges and Opportunities*. Annual Conference of the Prognostics and Health Management Society (PHM09) 2009 [cited 2009 Dec 2009]; Available from: https://www.phmsociety.org/sites/phmsociety.org/files/FieldedSystems_OFlarity.pdf.

[2] Saxena, A., et al., *Requirements Specification for Prognostics Performance – An Overview*, in *AIAA Infotech @ Aerospace*2010: Atlanta.

[3] Saxena, A., et al., *Metrics for Offline Evaluation of Prognostic Performance.* to appear in International Journal of Prognostics and Health Management, 2010. **1**(1): p. 21.

[4] Cocheteux, P., et al., *Prognostics Design: Requirements and Tools*, in *11th International Conference on The Modern Information Technology in the Innovation Processes of the Industrial Enterprises (MITIP09)*2009: Bergame, Itlay p. 8.

[5] CPDA, *Requirements Management: Solutions Review (Product Lifetime Management [PLM] Integration/Product definition).*

[6] Lee, G., J. Howard, and P. Anderson, *Safety-Critical Requirements Specification and Analysis Using SpecTRM.*

[7] Uckun, S., K. Goebel, and P.J.F. Lucas, *Standardizing Research Methods for Prognostics*, in *International Conference on Prognostics and Health Management (PHM08)*2008: Denver, CO. p. 10.

[8] Goebel, K., M. Krok, and H. Sutherland. *Dianostic Information Fusion Requirements Flowdown and Interface Issues*. in *IEEE Aerospace Conference Proceedings*. 2000. IEEE.

[9] Kurtoglu, T., I.Y. Tumer, and D.C. Jensen, *A functional failure reasoning methodology for evaluation of conceptual system architectures.* Research in Engineering Design, 2010. **online**.

[10] Byer, B., A. Hess, and L. Fila, *Writing a convincing cost benefit analysis to substantiate autonomic logistics*, in *IEEE Aerospace Conference*2001. p. 3095-3103.

[11] Millar, R.C., *Defining Requirements for Advanced PHM Technologies for Optimal Reliability Centered Maintenance.*

[12] Wheeler, K.R., T. Kurtoglu, and S. Poll, *A Survey of Health Management User Objectives Related to Diagnostic and Prognostic Metrics*, in *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*2009: San Diego CA. p. 15.

[13] Xu, P., Z. Wang, and Y. Li, *Prognostics and Health Management (PHM) System Requirements and Validation.*

[14] Yang, C. and S. Letourneau, *Model evaluation for prognostics: estimating cost saving for the end users*, in *Sixth International Conference on Machine Learning and Applications*2007. p. 304-309.

[15] Drummond, C. and C. Yang, *Reverse engineering costs: How much will a prognostic algorithm save?*, in *International Conference on Prognostics and Health Management*2008: Denver, CO.

[16] Feldman, K., T. Jazouli, and P.A. Sandborn, *A Methodology for Determining the Return on Investment Associated With Prognostics and Health Management.* Reliability, IEEE Transactions on, 2009. **58**(2): p. 305-316.

[17] Feldman, K., P. Sandborn, and T. Jazouli. *The analysis of Return on Investment for PHM applied to electronic systems*. in *Prognostics and Health Management, 2008. PHM 2008. International Conference on*. 2008.

[18] Goodman, D.L., S. Wood, and A. Turner, *Return-on-investment (ROI) for electronic prognostics in mil/aero systems*, in *IEEE Autotestcon*2005: Orlando, FL.

[19] Hoyle, C., et al., *On quantifying cost-benefit of ISHM in Aerospace Systems*, in *IEEE Aerospace Conference*2007: Big Sky, MT.

[20]Keller, K., et al., *A process and tool for determining the cost/benefit of prognostic applications*, in *IEEE Autotestcon*2001: Valley Forge, PA. p. 532-544.

[21]Leao, B.P., et al., *Prognostics performance metrics and their relation to requirements, design, verification and cost-benefit*, in *Prognostics and Health Management (PHM08), International Conference on*2008: Denver, CO. p. 1-8.

[22]Ashby, M.J. and R.J. Bryer, *An approach for conducting a cost benefit analysis for aircraft engine prognostics and health management functions*, in *Reliability and Maintainability Symposium (RAMS)*2002. p. 2847-2856.

[23]Banks, J. and J. Merenich, *Cost benefit analysis for asset health management technology*, in *Reliability and Maintainability Symposium (RAMS)*2007: Orlando, FL. p. 95-100.

[24]Banks, J., et al., *How engineers can conduct cost-benefit analysis for PHM systems*, in *IEEE Aerospace Conference*2005: Big Sky, MT. p. 1-10.

[25]Byington, C.S., et al., *Prognostic Enhancements to Gas Turbing Diagnostic Systems*, in *IEEE Aerospace conference*2001: Big Sky, MT.

[26]Greitzer, F. and R.A. Pawlowski, *Embedded Prognostics Health Monitoring*, in *International Instrumentation Symposium: Embedded Health Monitoring Workshop*2002.

[27]Hines, J., et al., *Cost-Benefit Analysis Trade-Space Tool as a Design-aid for the U.S. Army Vehicle Health Management System (VHMS) Program*, in *1st Annual Conference of the Prognostics and Health Management Society*2009: San Diego, CA. p. 18.

[28]Hoyle, C., et al., *Cost-benefit quantification of ISHM in Aerospace Systems*, in *ASME International Design Engineering Technical Conference*2007.

[29]Kacprzynski, G.J., M.J. Roemer, and A.J. Hess. *Health management system design: Development, simulation, and cost - benefit optimization*. in *IEEE Aerospace Conference*. 2002. Big Sky, MT.

[30]Kurien, J. and M.ı.D. R-Moreno, *Costs and Benefits of Model-based Diagnosis*, in *IEEE Aerospace Conference*2008: Big Sky, MT.

[31]Leao, B.P., et al., *Cost-Benefit Analysis Methodology for PHM Applied to Legacy Commercial Aircraft*, in *IEEE Aerospace Conference*2008: Big Sky, MT. p. 1-13.

[32]Reimann, J., et al., *Using condition based maintenance to improve the profitability of performance based logistic contracts*, in *Annual Conference of the Prognostics and Health Management Society*2009: San Diego, CA.

[33]Hecht, H., *Prognostics for electronic equipment: An economic perspective*, in *Reliability and maintainability symposium (RAMS)*2006: Newport Beach.

[34]Kacprzynski, G.J., et al., *A prognostic modeling approach for predicting recurring maintenance for shipboard propulsion systems*, in *ASME Turbo Expo*2001.

[35]Kacprzynski, G.J., et al., *Extending FMECA – Health management design optimization for aerospace applications*, in *IEEE Aerospace Conference*2001: Big Sky, MT.

[36]Khalak, A. and J. Tierno, *Influence of Prognostic Health Management on Logistic Supply Chain*, in *American Control Conference*2006.

[37]Luna, J.J., *Metrics, models and scenarios for evaluating PHM effects on logistics support*, in *Annual Conference of the Prognostics and Health Management Society*2009: San Diego, CA.

[38]Scanff, E., et al., *Life cycle cost impact of using prognostic health management (PHM) for helicopter avionics*. Microelectronics Reliability, 2007. **47**(12): p. 1857-1864.

[39]Williams, Z., Concepts of Operations: Program Analysis and Modeling Environment.

[40]Saxena, A., et al., *Metrics for Evaluating Performance of Prognostic Techniques*, in *International Conference on Prognostics and Health Management (PHM08)*2008: Denver, CO. p. 17.

[41]Saxena, A., et al. *Evaluating Algorithmic Performance Metrics Tailored for Prognostics*. in *IEEE Aerospace Conference*. 2009. Big Sky, MT.

[42]Saxena, A., et al., *On Applying the Prognostics Performance Metrics*, in *Annual Conference of the Prognostics and Health Management Society (PHM09)* 2009: San Diego, CA. p. 16.

[43]Orchard, M., et al. *A Novel RSPF Approach to Prediction of High-Risk, Low-Probability Failure Events*. in *Annual Conference of the PHM Society*. 2009.

[44]DoD, *Umanned Aircraft Systems Roadmap 2005-2030*, 2005, Office of Secretary of Defence, US.

[45]Clothier, R., et al. *A Causlaty Risk Analysis for unmanned Aerial System (UAS) Operations Over Inhabited Areas*. in *12th Australian International aerospace congress, 2nd Australian Unmanned air Vehicles Conference*. 2007.

[46]Dalamagkidis, K., K.P. Valavaanis, and L.A. Piegl, *UAS SAfety Assessment and Functional Requirements*, in *On Integrating Unmanned Aircraft Systems into the National Airspace*, S.G. Tzafestas, Editor 2009, Springer.

[47]Saha, B. and K. Goebel, *Model Adaptation for Prognostics in a Particle Filtering Framework*. International Journal of Prognostics and Health Management, 2011. **2**(1): p. 10.

[48]Saha, B., et al. *Battery Health Management System for Electric UAVs*. in *IEEE Aerospace Conference*. 2012. Big Sky, MT.

[49]Bouchereau, V. and H. Rowlands, *Quality function deployment: the unused tool*. Engineering Management Journal, 2000. **10**(1): p. 45–52.