A Design for Availability Approach for Use with PHM

Taoufik Jazouli and Peter Sandborn

CALCE Center for Advanced Life Cycle Engineering
Department of Mechanical Engineering
University of Maryland, College Park, MD 20742
tjazouli@calce.umd.edu
sandborn@calce.umd.edu

ABSTRACT

Prognostics and Health Management (PHM) methods are incorporated into systems for the purpose of avoiding unanticipated failures that can impact system safety, result in additional life cycle cost, and/or adversely affect the availability of the system. Availability is the probability that a system will be able to function when called upon to do so. Availability depends on the system's reliability (how often it fails) and its maintainability (how efficiently and frequently it is pro-actively maintained, and how quickly it can be repaired and restored to operation when it does fail). Availability is directly impacted by the success of PHM. Increasingly, customers of critical systems are entering into "availability contracts" in which the customer either buys the availability of the system (rather than actually purchasing the system itself) or the amount that the system developer/manufacturer/supplier is paid for the system is a function of the availability achieved by the customer. Predicting availability based on known or predicted system reliability, operational parameters, logistics, etc., is relatively straightforward and can be accomplished using existing methods. However, while determining the availability that results from a set of events is straightforward, determining the events that result in a desired availability is not, and prediction of a system's attributes to meet an availability requirement can only be accomplished using "brute force" search-based methods that are not general and become quickly impractical for real systems and when uncertainties are introduced. This paper presents a "design for availability" approach that starts with an availability requirement and uses it to predict the required logistics, design and operation parameters. The method is general and can be applied when the inputs to the problem are uncertain (even the availability requirement can be a probability distribution). The method is demonstrated on several examples with and without PHM.

1 INTRODUCTION

Availability is the ability of a service or a system to be functional when it is requested for use or operation. The availability of an item is a function of its reliability and logistics management (including repairs, replacements and inventory management). Availability accounts for both the frequency of the failure (reliability) and the ability to restore the service or system to operation after a failure (maintainability). The maintenance ramifications generally translate into how quickly the system can be repaired upon failure, which is usually driven by the logistics management and is directly influenced by Prognostics and Health Management (PHM) approaches that may be used. The frequency of the failure is related to the reliability of the system, i.e., how long it will be operational (or "up") before it fails.

Many real world systems are significantly impacted by availability. A failure, i.e., a decrease of availability, of an ATM machine causes inconvenience to customers; the unavailability of a point-of-sale system to retail outlets can generate a huge financial loss; the failure of a medical device or of hospital equipment can result in loss of life; and failures of aircraft cause airlines to cancel or delay flights and military missions to be canceled. In these systems, insuring the availability of the system is important and the owners of the systems may be willing to pay a premium for higher availability.

Several different types of availability can be measured (e.g., inherent, achieved, operational, etc.) (Kececioglu, 1995). This paper is focused on the operational availability since it implicitly incorporates other forms of availability and it is the most commonly used form of availability specified in availability contracts.

Operational availability is the probability that a system or piece of equipment operates ordinarily, i.e. functional and available for operation when requested, over a specific period of time under stated conditions (Oliveto, 2001; Macheret, 2005). Operational availability (A_o) accounts for all types of maintenance and logistics downtimes. It is computed as the ratio of the accumulated uptime and the sum of the accumulated uptime and downtime:

$$A_o = \frac{\text{uptime}}{\text{uptime} + \text{downtime}} \quad (1)$$

where uptime is the total accumulated operational time during which the system is up and running and able to perform the tasks that are expected from it. Downtime is generated when the system is down and not operating when requested due to repair, replacement, waiting for spares or any other logistics delay time. The summation of the accumulated uptimes and downtimes represents the total operation time for the system.

"Availability-based" contracting originated because customers with high availability requirements are in many cases interested in buying the availability of a system, instead of actually buying the system itself. Availability based contracts are a subset of outcome based contracts (Ng *et al.*, 2009); where the customer pays for the delivered outcome, instead of paying for specific logistics activities, system reliability management or other tasks. Basically, in this type of contract, the customer pays the service or system provider to ensure a specific availability requirement is met. Examples of availability and outcome-based contracts include the Availability Transformation: Tornado Aircraft Contract–ATTAC (BAE, 2008) and PBL – Performance Based Logistics (Beanum, 2006; Hyman, 2009). Availability contracts, and most outcome-based contacting, include cost penalties that could be assessed for failing to fulfill a specified availability requirement in a defined time frame.

Two other mechanisms that include elements of availability contracting are Product Service Systems (PSS) and leasing models. PSS provide both the product and its service/support based on the customer's requirements, which may include an availability requirement. PSS involve the product's specifications, the product manufacturing and support, and the product supply chain and parts management; i.e., PSS engage all customer, manufacturer and supplier (Bankole *et al.*, 2009). Lease contracts (Yeh and Chang, 2007) fall into the use-oriented class of PSS, where the ownership of the product is usually retained by the service provider. The lease contract may indicate not only the basic product and service provided but also other use and operation constraints such as the failure rate threshold and the availability of the product or system when requested for operation.

The evaluation of an availability requirement is a challenging task for both suppliers and customers. From a suppliers' perspective, it is not trivial to estimate the cost of delivering a specific availability. From customers' perspective, the amount of money that should be spent on a specific availability contract is also a mysterious quantity.

This paper presents a new methodology that uses an availability requirement as an input to the process of determining the optimal management of a system (as opposed to an availability output that is a consequence of system management and logistics inputs). The next section describes the design for availability problem and our proposed approach and Section 3 provides simple case study examples that demonstrate the operation of the approach for systems with and without PHM.

2 DESIGN FOR AVAILABILITY APPROACH

Most availability and life cycle cost predictions used during the design and support of real systems are performed using discrete event simulators, e.g., (Juan *et al.*, 2009; Bazargan and McGrath, 2003). In general, discrete event simulators order the failure and maintenance events for a system temporally, and the times associated with the failure and maintenance events can be readily accumulated to estimate availability. Thus, it is straightforward for a discrete event simulation to compute the availability based on a particular sequence of failures, logistics and maintenance events.

Availability requirements can be satisfied by running discrete event simulators in the forward direction (forward in time) for many permutations of the system attributes and then selecting the inputs that generate the required availability output, e.g., (Janakiraman *et al.*, 2004; Castro and Cavalca, 2006). These "brute force" search-based approaches are computationally impractical for real problems (particularly for real-time problems), are unable to deal with general uncertainties, and can't accommodate an availability requirement that is represented as a probability distribution.

In general, determining design parameters from an availability requirement is a stochastic reverse simulation problem. There have been attempts to perform reverse simulation (run discrete event simulators backwards in time) (Raffo and Setamanit, 2003; Reynolds and McKeown, 2007), but this has only been demonstrated on extremely simple non-availability analysis problems with no applicability to the real world systems. While determining the availability that results from a sequence of events is straightforward, determining the events that result in a desired availability is not, and has not in general been done. Alternatively stated, availability is straightforward to predict based on the system's reliability, operation, sparing, etc., however, the general prediction of the system attributes to meet a required availability ("design for availability") has never been done.

The goal of this work is to reverse the problem setup; this means, instead of solving for the availability that results from a specific set of system attributes, we will solve for system attributes that guarantee a specific availability. The design for availability model could be used to generate system reliability, operation, sparing, etc., for a specific availability, i.e., for a specific uptime (time that the system is up and running as requested) and downtime (time that the system is down undergoing a repair or waiting for spares). The approach presented here is not based on running backwards discrete event simulation, i.e., the proposed model runs a forward discrete event simulation, but, instead of using system attributes to compute uptimes or downtimes, the new model imposes the appropriate uptimes and downtimes based on the specified availability (input) to compute the system attributes (output).

The proposed approach consists of the following four steps:

1. Determine where (during the operation of the system) the availability requirement is imposed.
2. Impose either a downtime or uptime requirement.
3. Define a relationship between downtime (or uptime) and the system attributes.
4. Update the downtime (or uptime) requirement, and compute other quantities of interest (e.g., life cycle cost), using the updated system attributes.

2.1 Determine Where the Availability Requirement is Imposed

Availability contracts specify an availability requirement at some defined time or in some defined time period. The time period in which (or after which) a specific availability must be met is dependent on the specific contract terms. Therefore, to generalize our design for availability model, we have adopted a conservative approach by fulfilling an availability requirement at all times during the entire support life. This means, the model satisfies any availability contract requirement, regardless of the specified availability evaluation time frame. However, if needed, the model could be adjusted to evaluate the availability requirement only at the contract's defined times (which could be less conservative).

For the remainder of this work we will assume that the availability requirement implies that the operational

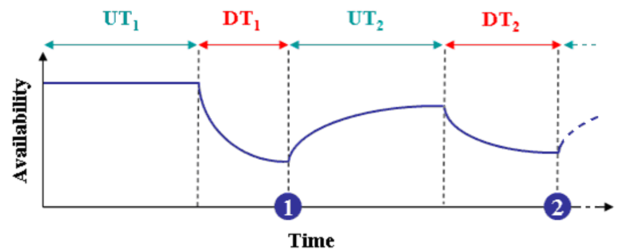

Figure 1: Availability variations as a function of time.

availability (A_o) should not drop below the availability requirement value at any time during the entire support life. Thus, whenever the A_o drops to a minimum value this value should be greater than or equal to the availability requirement. Based on (1) and Figure 1, where DT and UT represent the downtime and uptime durations respectively, the A_o reaches its local minimum values at the end of every downtime (e.g., points 1 and 2 in Figure 1). Implicitly, if the availability requirement is satisfied at the end of every downtime, it will be satisfied at all times during the support life of the system, therefore our approach is to impose the availability requirement at the end of every downtime.

Notice in this case, where the availability requirement is imposed at all times throughout the entire support life, that it doesn't make sense to impose the availability requirement at the start of the system life if the system management is starting with a downtime (instead of an uptime); since starting with a downtime will keep the availability at zero (below the contract's availability threshold) until the start of the first uptime.

2.2 Impose Either a Downtime or Uptime Requirement

The operational availability A_o is a function of accumulated uptimes and downtimes. Therefore, imposing an availability requirement means either imposing an uptime requirement while the downtime is automatically generated by the known set of system attributes (e.g., system's reliability, operational profile, logistics, PHM parameters, etc); or imposing a downtime requirement while the uptime is automatically generated by the known set of system attributes. Note, either an uptime or a downtime requirement is imposed, not both. Since availability is a function of uptime and downtime, we need to know at least one of these two quantities to be able to impose the other one. For example, if the imposed quantity is the downtime; then the imposed downtimes are computed at defined times or events as a function of the required availability and the system generated uptimes. Then, the unknown set of system attributes is computed based on the imposed downtime. The methodology described in this paper is applicable to system attributes that explicitly affect either uptime or downtime, but not both.

The criteria of imposing either an uptime or downtime requirement is based on the unknown set of system attributes that we desire to determine to fulfill a specific availability requirement. For example, if the uptime remains constant while varying an unknown system attribute, meaning that the uptime is independent of the unknown parameter, then we must impose the downtime to meet the availability requirement; and vice versa.

2.3 Define a Relationship Between Downtime (or Uptime) and the System Attributes

Assume that the set of system attributes that we are interested in computing to meet the availability requirement is explicitly related to the downtimes (based on the criteria discussed in Section 2.2). Thus, we want to impose the downtime requirement. Then, we need to establish a relationship between the unknown set of system attributes and the imposed downtimes, so that the unknown set of system attributes can be computed based on the imposed downtimes.

For example, to determine the appropriate inventory lead time (ILT) to ensure meeting a specific availability requirement we need to define a relationship between the downtime requirement and ILT. In this example, the ILT is the unknown system attribute, where ILT is the amount of time it takes to receive spare replenishment when additional spares are ordered at the inventory spares threshold (ST) value (i.e., minimum quantity of held spares); this example assumes that only when the maintenance event is a replacement (the unit cannot be repaired on-site) that there is a call for a spare from the inventory. Also, the inventory downtime (when the inventory runs out of spares, and the system is down waiting for replenishment spares) is assumed to be larger than the concurrent maintenance downtime. Thus, once the spares are received, the part can be immediately installed in the system. The ILT requires imposing a downtime requirement since varying the ILT only affects the downtime values, i.e., how long the system will be down waiting for spares to be replenished.

In this case (Figure 2), the decision to impose the inventory downtimes (IDT) to meet the availability requirement, instead of imposing uptimes, is based on the fact that the unknown system attribute, i.e., ILT, is only dependent on the downtimes; and it is independent of the uptimes. Varying the ILT generates different IDT values; however the uptime values remain constant since they are only a function of the inventory spares threshold (ST) and maintenance downtimes (MDT). Where the inventory held spares is a function of the quantity of initial spares (IS) and quantity of replenishment spares (RS). The ILT only defines the start of the next uptime, but it does not define the uptime duration.

Because, the IDT is purely a function of ILT and spares threshold time (STT), where STT is the corresponding period of time to use all remaining spares; the IDT only depends on how low the inventory level is allowed to drop before ordering additional spares and how long it will take to receive those spares,

$$ILT = STT + IDT_1 \qquad (2)$$

For this example we assume that the MDT are given and cannot be modified, i.e., the maintenance lead time, replacement time and repair time are already specified as inputs. To fulfill the availability requirement at the end of the first IDT; IDT_1 should satisfy the A_o requirement (as defined in (1); where IS-MDT_1 corresponds to the accumulated uptime and IS+IDT_1 corresponds to the sum of the accumulated uptime and downtime), thus satisfy the following equation:

$$IDT_1 = \frac{IS - MDT_1}{A_o} - IS \qquad (3)$$

Once IDT_1 is determined, the ILT can be computed by satisfying (2).

Finally, the ILT is updated as the downtime requirement gets updated throughout the entire support life; which is presented in the next step (Section 2.4) in the proposed approach. In this simple example, we have defined a relationship between the ILT (the unknown system attribute) and the imposed downtimes,

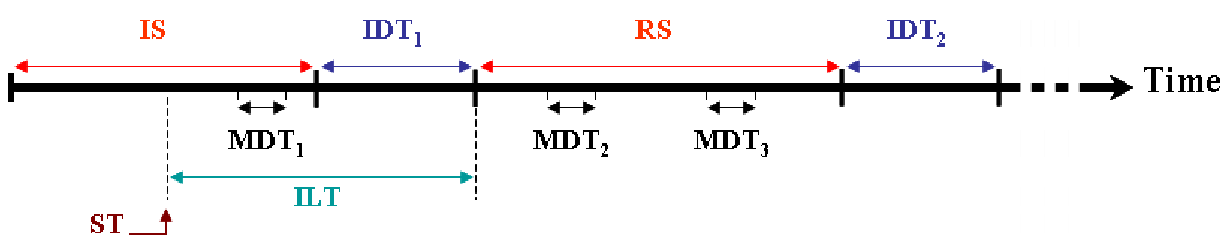

Figure 2: Implication of the inventory model parameters on the timeline.

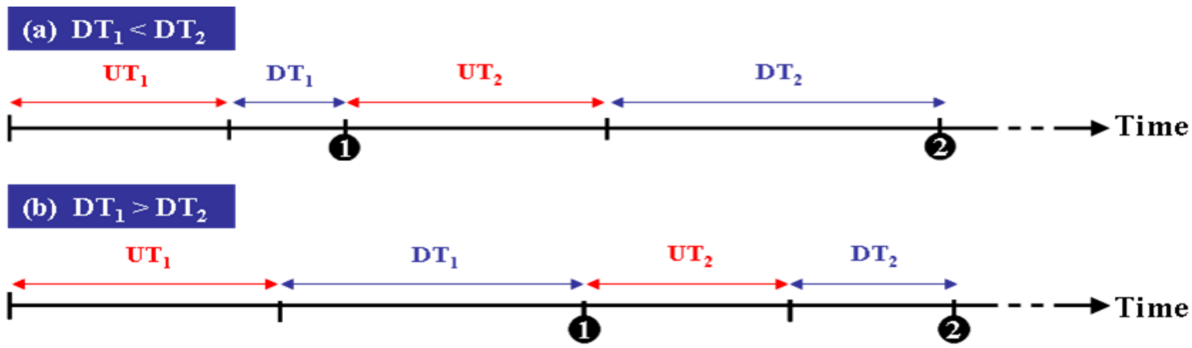

Figure 3: Special cases of downtimes requirement.

to satisfy an availability requirement. The approach can be generally applied to any set of system attributes that are explicitly related to either downtimes or uptimes, to fulfill an availability requirement.

2.4 Update Downtime (or Uptime) Requirement

In this section, for demonstration purposes, we will assume that the unknown set of system attributes is explicitly dependent on the required downtimes to satisfy the availability requirement, while the rest of the system attributes are given and responsible for generating the uptimes.

The challenge in this step is that the availability is not determined by a single downtime value, but rather a sequence of downtime values that are not necessarily identical; each resulting in different computed values for the system attribute. As a result, by the end of the simulation we could generate multiple values for the same system attribute with no way to determine which value to use to fulfill the availability requirement.

In the simplest case, if all downtimes were identical, the same value for the system attribute would have been generated at the conclusion of every downtime. To achieve this, we wish to select a single downtime value that is the maximum allowable downtime to meet a specific availability requirement, and then use this quantity as a constant downtime value that will fulfill the availability requirement at every point throughout the entire support life. To derive the maximum allowable downtime value, we have explored two scenarios.

The first scenario is illustrated in Figure 3a, where the first imposed downtime (DT_1) duration is shorter than the second imposed downtime (DT_2), where both downtimes have been imposed based on the same availability requirement. In this case, averaging the two downtimes would generate an average downtime ($DT_{Average}$) that is larger than DT_1, thus if DT_1 is substituted for $DT_{Average}$ then the availability requirement will not be fulfilled at the end of DT_1, as expressed in (4),

$$\frac{UT_1}{UT_1 + DT_{Average}} < \frac{UT_1}{UT_1 + DT_1} \tag{4}$$

In this situation the maximum allowable downtime duration that the system can accommodate without failing to satisfy the availability requirement is constrained by the value of DT_1. Therefore, DT_2 value should be substituted for DT_1.

The second scenario is illustrated in Figure 3b, where DT_1 is larger than DT_2. In this case, averaging the two downtimes would generate a $DT_{Average}$ smaller than DT_1, thus if DT_1 is substituted for $DT_{Average}$ then the availability requirement will be fulfilled at the end of DT_1, as expressed in (5).

$$\frac{UT_1}{UT_1 + DT_{Average}} > \frac{UT_1}{UT_1 + DT_1} \tag{5}$$

Notice in this case that the availability requirement at any specific time includes all accumulated previous downtimes. Thus, when using the average value at the end of DT_2, the availability requirement will still be satisfied,

$$\frac{UT_1 + UT_2}{UT_1 + UT_2 + DT_{Average} + DT_{Average}} = \frac{UT_1 + UT_2}{UT_1 + UT_2 + DT_1 + DT_2} \tag{6}$$

Since the accumulated averages are just the accumulated downtimes, i.e., summation of DT_1 and DT_2. Therefore, in this situation the maximum allowable downtime duration that the system can accommodate without negating the availability requirement is the average downtime.

Figure 4 summarizes both cases in one general case. The model imposes the required downtime to meet the availability requirement, and then it evaluates the current downtime with respect to the previous one. If the currently imposed downtime is larger than the previous one, then the model substitutes the current downtime value for the previous one. But if the currently imposed downtime is shorter than the previous one, then the model averages the current downtime value with all previous ones. The goal of this procedure is to generate one unique value of the maximum allowable downtime that meets the availability requirement. Note, during this procedure, the unknown system attribute that is determined based on the downtime requirement, gets updated as soon as the downtime values are updated.

If the unknown system attributes are explicitly generating the uptime, instead of the downtime, then, by analogy, we can use a similar procedure to impose and update the uptimes to derive one unique value of the system attributes. In this case, we would derive the minimum allowable uptime that meets the availability requirement; then use the derived quantity to compute the corresponding system attributes.

Finally, the proposed model derives and updates the required system attributes to meet the A_o requirement. Once these are determined, the model uses the final updated values of the system attributes to compute other quantities of interest (e.g., life cycle cost, investment cost, avoided failures, etc.).

3 APPLICATION OF THE METHODOLOGY

The design for availability methodology described in Section 2 has been implemented within a PHM Return on Investment (PHM ROI) tool, for demonstration and testing (i.e., verification).

The PHM ROI tool (Sandborn and Wilkinson, 2007; Feldman *et al.*, 2009) is a discrete event simulation that follows a population of sockets (a socket is an instance of an installation location for an LRU) through their lifetime from the first line replaceable unit (LRU) installation in the socket to the retirement of the socket. In this tool a probabilistic model is implemented as a Monte Carlo simulation. It is a stochastic simulation of a timeline where specific events are added to the timeline and the resulting event order and timing can be used to analyze failure avoided, life cycle cost, availability (as an output), etc.

The prediction of the remaining useful life (RUL) is determined by the sampling of both the time-to-failure (TTF) values and the distributions that are used to model the effectiveness of a particular PHM approach. The sampling of the TTF values is defined differently for each PHM sustainment approach (e.g., data-driven, model-based, fixed interval scheduled maintenance and unscheduled maintenance), see (Sandborn and Wilkinson, 2007). The PHM ROI tool includes the modeling of other quantities as well (e.g., operational profile, false positives, cost of money, inventory management, etc).

3.1 A Simple Demonstration of the Proposed Methodology for Unscheduled Maintenance

In this subsection the proposed design for availability methodology will be demonstrated for two different availability distribution inputs assuming an unscheduled maintenance policy. The objective in these example cases is to determine the maximum allowable spares replenishment lead time, i.e., inventory lead time (ILT), to fulfill a specified availability requirement (focusing on ILT prediction is only for example purposes, the approach can be applied to other parameters as well). In order to use this demonstration as a qualitative verification of the methodology, we will perform the following steps:

- Using the availability distribution requirement as an input, determine the distribution of maximum allowable ILTs.
- Use the generated ILT distribution as an input to the existing PHM ROI simulation (described in the introduction to this section) to predict an availability distribution as an output.
- Compare the availability distribution input requirement to the availability distribution determined as an output – they should be equivalent.

The first step is sufficient to achieve the design for availability task, since the ILT will be determined for a

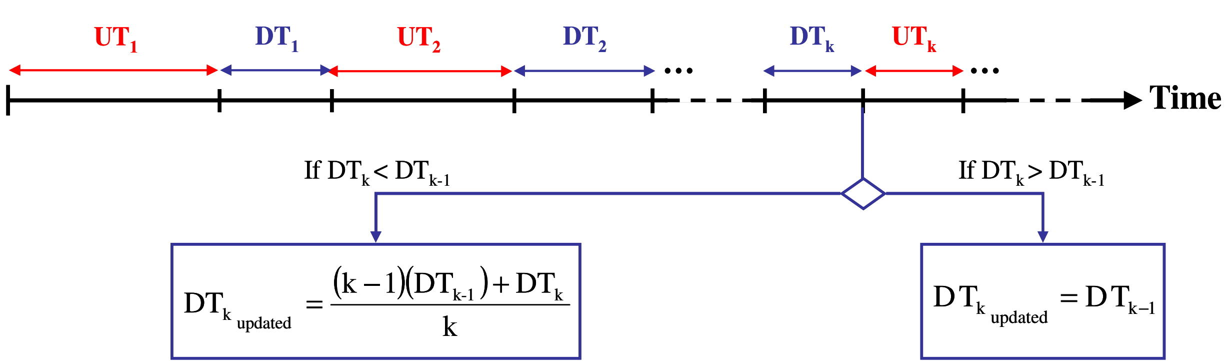

Figure 4: General case of downtimes requirement.

specific availability requirement. The second and third steps are performed as a verification of the methodology.

A detailed description of all of the case study inputs is provided in the Appendix including reliability information, LRU description, implementation and maintenance costs, operational profile and inventory management parameters.

The first availability distribution considered is shown on Figure 5a. This distribution could represent the requirement specified in an availability contract.* Note, availability contracts may specify the availability requirement as a single value (which can easily be accommodated by the methodology described in this paper), but, to accommodate more general problems, in this example we will use availability requirements that are represented as probability distributions. To generate a distribution of ILTs, the model samples the required availability distribution and other quantities that may be described as probability distributions (e.g., reliability), and then uses the quantities and the assumption of an unscheduled maintenance approach to solve for a value of the ILT using the methodology in Section 2. This process is repeated for each socket of the population, resulting in a histogram of ILTs. Figure 5b shows the distribution of allowable ILTs that result from the availability requirement shown in Figure 5a.

In order to qualitatively verify the methodology, the ILT distribution (Figure 5b) was used as an input to the PHM ROI tool discussed in the introduction of this section. The PHM ROI tool used the ILT distribution along with the other inputs in the Appendix and generated a resulting availability distribution. Figure 5c shows the availability prediction that resulted from the PHM ROI tool. The two availability distributions (Figures 5a and 5c) are not expected to be absolutely identical (since this is a stochastic solution), but the mean and standard deviation are very similar.

Now consider a more challenging case. Figure 6a shows a different availability input distribution (we are not addressing the actual reason for requiring an availability distributed as shown in Figure 6a in this paper, it is only used as a verification exercise). The model generates the maximum allowable ILT to meet the availability requirement. The ILT distribution is shown in Figure 6b.

As in the first example, to verify the methodology, the ILT distribution (Figure 6b) was used as an input to the PHM ROI tool, along with the other inputs specified in the Appendix (and assuming unscheduled maintenance). The PHM ROI tool used these inputs to generate a resulting availability distribution, Figure 6c. The shape of the distribution and its mean and standard deviation are well matched (comparison of Figures 6a and 6c).

* The interpretation of an availability requirement as a probability distribution means that not every socket in the population has to have the same availability, but that as a population they must satisfy the required availability distribution. This is consistent with the fact that the reliability of the LRU is a probability distribution (each LRU fails at different times); thus using a logistics management plan that is common across the population, each socket will have a different availability value depending on the failure dates of the LRUs that occupy it and socket-to-socket operating profile variations. Note, an availability requirement that is a single value, could refer implicitly to the average value of the availability distribution of the population.

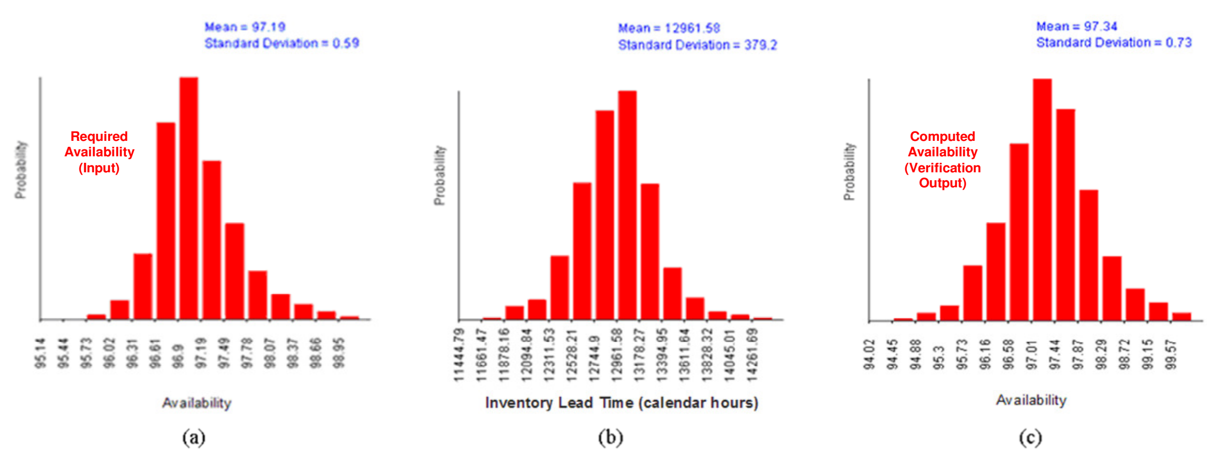

Figure 5: (a) Required availability distribution (input to the model). (b) Computed Inventory Lead Time (ILT) in calendar hours. (c) Availability probability distribution generated using the computed ILT (Figure 5b).

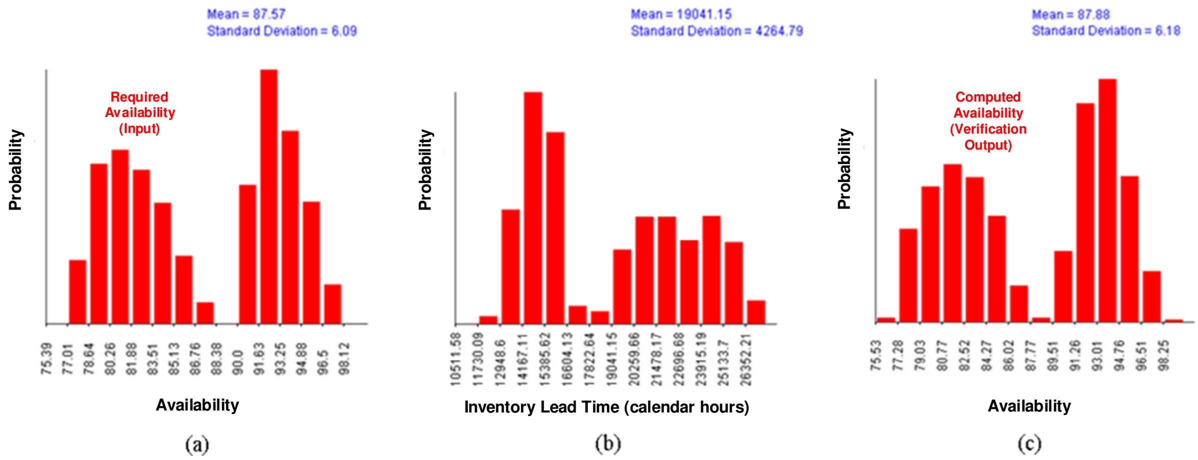

Figure 6: (a) Required availability distribution (input to the model). (b) Computed Inventory Lead Time (ILT) in calendar hours. (c) Availability probability distribution generated using the computed ILT.

The simple examples presented in this subsection demonstrate qualitatively that the design for availability approach is satisfying the input availability requirement.

3.2 Use of Design for Availability with PHM

In this subsection we compare the maximum allowable ILT for a specific availability requirement for unscheduled maintenance and a data-driven PHM approach.

Determining the maximum allowable ILT for a specific availability requirement could be used to improve logistics management and potentially reduce life cycle cost. If the availability drops below a specified threshold value, a cost penalty could be assessed; determining upfront the appropriate ILT would avoid these potential cost penalties. Also, knowing the maximum allowable ILT information, customers could require their suppliers to deliver within a specific lead time.

For the assumed set of system attributes and assumptions (see the Appendix) we want to determine the appropriate spares replenishment lead time, i.e., inventory lead time (ILT), to fulfill the availability requirement specified in Figure 5a for an unscheduled maintenance approach and a data-driven PHM approach applied to the same system (a detailed explanation of how the data-driven PHM approach is modeled is provided in Sandborn and Wilkinson, 2007).

Increasing the availability to meet the requirement implies reducing the delivery time (considered as the only variable input in this example). However, faster delivery requires larger investment cost. Basically, we want to generate an optimal solution that produces the maximum allowable ILT (i.e., to minimize cost) that keeps the availability value at or above the availability requirement.

By running the simulation with the imposed availability (input) requirement shown in Figure 5a, the ILT (output) satisfying this requirement was determined for the unscheduled maintenance policy, and the generated ILT probability distribution is shown in Figure 7 (light gray histogram bars).

We now need to apply the method to the data-driven PHM approach. Figure 8 shows the results of the analysis to determine the optimal (lowest life cycle cost) prognostic distance for the data-driven PHM approach used; where the prognostic distance is defined as the measure (e.g., operational hours) of how long the prognostic structures or prognostic cell is expected to indicate failure, before the system actually fails (Sandborn and Wilkinson, 2007).

Small prognostic distances may miss failures while

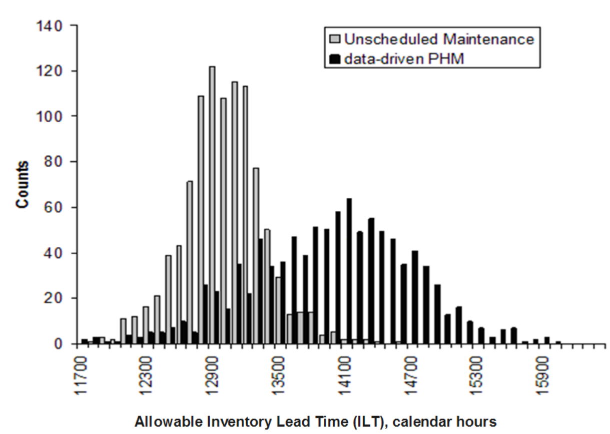

Figure 7: Computed inventory lead time (ILT) for unscheduled maintenance policy, and for a data-driven PHM approach.

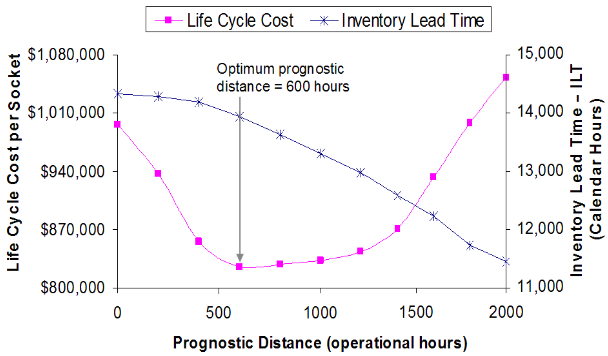

Figure 8: Variation of life cycle cost with data-driven PHM prognostic distance.

large prognostic distances may throw away significant remaining useful life. Each prognostic distance generates a corresponding ILT and life cycle cost. Note, the ILT values shown in Figure 8 are the mean values of the generated ILT distributions. Large prognostic distances may provide additional time to order spares ahead of failures; however, they could also produce solutions that require more spares, which will increase the accumulated IDT (inventory downtime, i.e., time that the system is down waiting for spares) since every spares replenishment event generates an additional IDT. Therefore, to accommodate these downtimes, while fulfilling the availability requirement, the ILT has to be reduced (i.e., reduce the IDT, thus faster delivery). This is illustrated in Figure 8; where large prognostic distances generate shorter ILT.

Using the Appendix data with the data-driven PHM approach, an optimal prognostic distance of 600 hours results in the minimum life cycle cost over the entire support life. Also, a symmetric triangular distribution with a width of 500 hours was assumed to represent the effectiveness of the data-driven PHM approach (see Sandborn and Wilkinson, 2007).

After running the simulation with the imposed availability requirement shown in Figure 5a, the ILT satisfying the contract requirement was determined, and the generated ILT probability distribution is shown in Figure 7 (black histogram bars).

In this example, the data-driven PHM approach allows for a larger ILT (mean = 13,936 calendar hours), compared to the unscheduled maintenance case (mean = 12,961 calendar hours). In other words, using a data-driven PHM approach allows a given availability requirement to be met if ILTs are longer, or alternatively stated, the use of PHM would allow a supply chain with longer ILTs to be used. The use of a data-driven PHM approach has shifted the ILT distribution by approximately 1000 hours to the right. This result is due the fact that data-driven PHM has provided early warning of failures; therefore presenting the opportunity to switch maintenance actions from unscheduled to scheduled events reducing the accumulated operational downtime. For a fixed ILT, this would result in an improved operational availability of the system. However, since in this problem the availability was set at a fixed required value, thus the accumulated operational downtime was used as a fixed quantity (imposed by the availability requirement); then the avoided unscheduled maintenance downtime was added to the IDT, resulting in a larger allowed ILT.

To summarize, our model was applied to the Appendix case study example, for two different maintenance approaches, to satisfy a specific contract availability requirement. For both approaches, unscheduled maintenance and data-driven PHM, we were able to determine the unknown system attribute (the allowed ILT in this case) satisfying the availability requirement.

4 DISCUSSION AND CONCLUSION

In this paper a new methodology for determining an unknown system attribute to fulfill a specific availability constraint has been presented. The proposed design for availability methodology is <u>not</u> a "search-based" approach and is capable of calculating unknown system parameters directly from the availability requirement even when the inputs are uncertain and the availability requirement is represented as a probability distribution.

The proposed design for availability methodology provides the possibility for significant new capability to: a) perform (in conjunction with prognostics and health management) real-time pro-active availability analysis; b) determine requirements flow down for the development of prognostics and system health management and flow down to the supply chain; and c) perform pro-active reliability versus logistics tradeoffs, and assess the cost and resources required to deliver and support systems subject to availability contracts (e.g., Performance Based Logistics contracts).

This method will enable the use of advances in the detection of performance anomalies and degradation of systems (including prognostics), to assess (and mitigate) logistics risks that result in system downtime. Providing health assessment and advanced warning of impending failure coupled with real-time design for availability control enables decision support actions that when communicated to maintenance and logistics operations will insure timely forecasting of maintenance and logistics actions that meet required availability levels.

Simple unscheduled maintenance case study examples were presented to demonstrate the methodology, as well as providing a means for verification and qualitative validation purposes. Then, specific case study results for an example system managed using both unscheduled maintenance and a data-driven PHM approach were also included. The model predicted a larger allowable ILT for the data-driven PHM case for the same availability requirement. Note, the conclusions in this paper about the ILT associated with PHM and unscheduled maintenance approaches are specific to the example data assumed and should not be interpreted as a general conclusion. However, the example demonstrates that the use of PHM in cases where availability requirements are imposed, can provide value beyond the commonly articulated failure avoidance and minimization of lost remaining useful life.

The methodology in this paper is part of a disciplined supportability analysis strategy that could be applied early in the system development process, thus exerting influence on the system (and system supportability) design by suggesting where appropriate PHM monitors and data collection mechanisms should be included in the design.

The determination of ILT (used as an example in this paper) for a specific availability requirement was provided as a demonstration of the design for availability model operation. The model can be applied to determine any system attributes that can be explicitly related to the timeline downtimes or uptimes, for a contract availability requirement. Current research work is focused on extending the methodology to system attributes that explicitly affect both uptime and downtime.

This methodology could be extended to address the concurrent determination of multiple design variables that are dependent, i.e., if a relationship between the unknown design variables is known. However, when the unknown design variables are independent, the inclusion of an optimization approach may be required at the conclusion of the 3rd step (Section 2.3) of the methodology; since the relationship between the imposed downtime (or uptime) and the unknown design variables could accept more than one unique solution. But, even in this case (i.e., multiple independent unknown design variables), the methodology is still efficient in terms of reducing the large and complex optimization problem, i.e., determining the unknown design parameters for multiple non-identical downtime (or uptime) values that generate different availability quantities, which may or may not satisfy the availability requirement; to determining the unknown design parameters for a single downtime (or uptime) value that has been imposed to satisfy the availability requirement.

APPENDIX – DATA SUMMARY FOR CASE STUDIES

This Appendix provides model inputs and assumptions that are used for the example analyses presented in Section 3. The example in this Appendix represents a simplified version of the case study that appeared in (Feldman *et al.*, 2009). Only the most relevant inputs for this specific application of our design for availability model are provided here; for a more detailed inputs data refer to (Feldman *et al.*, 2009). The LRU used in this example is an avionics multifunction display (MFD). The implementation costs are summarized in Table A.1. The discount rate on money used is 0.07.

The cost per hour out of service is \$500 for scheduled maintenance and \$5092 for unscheduled maintenance assuming during mission failures.

Table A.1: Implementation Costs

Frequency	Type	Value
Recurring Costs	Base cost of an LRU (without PHM)	\$25,000 per LRU
Recurring Costs	Recurring PHM cost	\$155 per LRU \$90 per socket (C_{REC})
Recurring Costs	Annual Infrastructure	\$450 per socket (C_{INF})
Non-Recurring Engineering	PHM cost	\$700 per LRU (C_{NRE})

The operational profile is summarized in Table A.2 (Feldman *et al.*, 2009; Henkle *et al.*, 2002), and a 20 years support life was chosen based on (Federal Aviation Administration, 2001).

Table A.2: Operational Profile

Factor	Multiplier	Total
Support life: 20 years	2,429 flights per year	48,580 flights over support life
7 flights per day	125 minutes per flight	875 minutes in flight per day
45 minutes turnaround between flights (Henkle *et al.*, 2002)	6 preparation periods per day (between flights)	270 minutes between flights/day

Table A.3 summarizes the spares inventory (per socket) assumptions. Also, note that a spares carrying costs are incorporated into the LRU recurring costs.

Table A.3: Spares Inventory

Factor	Quantity
Initial spares purchased for each socket	5
Threshold for spare replenishment	≤ 1 spares in the inventory per socket
Number of spares to purchase per socket at replenishment	4
Spare replenishment lead time	*Solved for in Section 3 case studies*
Spares carrying cost	10% of the beginning of year inventory value per year
Billing due date when ordering additional spares	2 years from purchase date

Figure A.1 shows the assumed reliability, i.e., time-to-failure (TTF), of the LRU based on (Scanff *et al.*, 2007; Kumar *et al.*, 2000), which provides Weibull and exponential reliability distributions commonly used to model avionics.

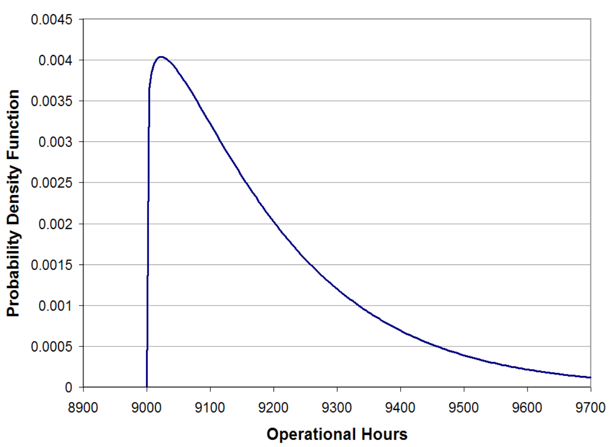

Figure A.1: Weibull distribution of TTFs (β=1.1, η=200 and γ=9000).

ACKNOWLEDGMENT

The work was supported in part by the CALCE Prognostics and Health Management Consortium.

REFERENCES

(BAE, 2008) BAE 61972 BAE Annual Report (2008).

(Bankole *et al.*, 2009) O. O. Bankole, R. Roy, E. Shehab, P. Wardle, Affordability Assessment of Industrial Product-Service System in the Aerospace Defense Industry, *proceedings of the 1st CIRP Industrial Product-Service Systems (IPS2) Conference*, Cranfield University, 1-2 April 2009, pp. 230.

(Bazargan and McGrath, 2003) M. Bazargan and R. N. McGrath, Discrete event simulation to improve aircraft availability and maintainability, in *Proceedings of the Annual Reliability and Maintainability Symposium*, 2003, pp. 63-67.

(Beanum, 2006) R. L. Beanum, Performance Based logistics and contractor support methods, in *Proceedings of the IEEE Systems Readiness Technology Conference (AUTOTESTCON)*, September 2006.

(Castro and Cavalca, 2006) H. F. Castro and K. L. Cavalca, Availability optimization with genetic algorithm, *Reliability Engineering & System Safety*, vol. 91, no. 4, pp. 413-420, April 2006.

(Federal Aviation Administration, 2001) Federal Aviation Administration, *Investment Analysis Benefit Guidelines: Quantifying Flight Efficiency Benefits*, Version 3.0, Investment Analysis and Operations Research Group, June 2001.

(Feldman *et al.*, 2009) K. Feldman, T. Jazouli, and P. Sandborn, A methodology for determining the return on investment associated with prognostics and health management, *IEEE Trans. on Reliability*, vol. 58, no. 2, pp. 305-316, June 2009.

(Henkle *et al.*, 2002) A. Henkle, C. Lindsey, and M. Bernson, Southwest Airlines: A review of the operational and cultural aspects of Southwest Airlines, *Operations Management Course Presentation, Sloan School of Management*, Summer 2002.

(Hyman, 2009) W. A. Hyman, *Performance-based contracting for maintenance*, NCHRP Synthesis 389, ISSN:0547-5570, 2009.

(Janakiraman *et al.*, 2004) G. Janakiraman, J. R. Santos, and Y. Turner, Automated system design for availability, in *Proceedings of the International Conference on Dependable Systems and Networks* (DSN'04), 2004.

(Juan *et al.*, 2009) A. A. Juan, A. Monteforte, A. Ferrer, C. Serrat, and J. Faulin, Applications of discrete-event simulation to reliability and availability assessment in civil engineering structures, in *Proceedings of the Winter Simulation Conference*, pp. 2759-2767, 2009.

(Kececioglu, 1995) D. Kececioglu, *Maintainability, Availability, Operational Readiness, Engineering Handbook*, Volume 1, Prentice Hall PTR, 1995.

(Kumar *et al.*, 2000) D. Kumar, J. Crocker, J. Knezevic, and M. El-Haram, *Reliability Maintenance and Logistic Support: A Life Cycle Approach*, Springer, 2000.

(Macheret, 2005) Y. Macheret, P. Koehn, and D. Sparrow, Improving reliability and operational availability of military systems, in *Proceedings of the IEEE Aerospace Conference*, 2005.

(Ng *et al*., 2009) I. C. L. Ng, R. Maull, and N. Yip, Outcome-Based Contracts as a driver for Systems thinking and Service-Dominant Logic in Service Science: Evidence from the Defence industry, *European Management Journal*, vol. 27, no. 6, pp. 377-387, December 2009.

(Oliveto, 2001) F. E. Oliveto, An optimal sparing model for the operational availability to approach the inherent availability, in *Proceedings of the Annual Reliability and Maintainability Symposium*, 2001, pp. 252-257.

(Raffo and Setamanit, 2003) D. M. Raffo and S. Setamanit, Supporting software process decisions using bi-directional simulation, *International Journal of Software Engineering and Knowledge Engineering (IJSEKE)*, vol. 13, no. 5, pp. 513-530, 2003.

(Reynolds and McKeown, 2007) A. P. Reynolds and G. P. McKeown, Construction of factory schedules using reverse simulation, *European Journal of Operational Research*, vol. 179, no. 3, pp. 656-676, June 2007.

(Sandborn and Wilkinson, 2007) P. A. Sandborn and C. Wilkinson, A maintenance planning and business case development model for the application of prognostics and health management (PHM) to electronic systems, *Microelectronics Reliability*, vol. 47, no. 12, pp. 1889-1901, December 2007.

(Scanff *et al*., 2007) E. Scanff, K. Feldman, S. Ghelam, P. Sandborn, M. Glade, and B. Foucher, Life cycle cost estimation of using prognostic health management for helicopter avionics, *Microelectronic Reliability*, vol. 47, no. 12, pp. 1857-1864, December 2007.

(Yeh and Chang, 2007) R. H. Yeh, W. L. Chang, Optimal threshold value of failure-rate for leased products with preventive maintenance actions, *Mathematical and Computer Modeling*, vol. 46, pp.730-737, 2007.

Taoufik Jazouli received the B.S. degree in mechanical engineering, and the M.S. degree in mechanical engineering (Design and Mechanical Manufacturing), both from the National School of Electricity and Mechanics, University Hassan II Ain Chock, Morocco, in 2005. After two years of professional experience in engineering consulting, he is currently attending the University of Maryland College Park pursuing a Ph.D. in Mechanical Engineering (Electronic Packaging Systems). He is a member of the IEEE, PHM society and ASME.

Peter Sandborn is a Professor in the CALCE Electronic Products and Systems Center at the University of Maryland. His group develops life cycle cost models and business case support for long field life systems. This work includes: obsolescence forecasting algorithms, strategic design refresh planning, lifetime buy quantity optimization, and return on investment models for maintenance planning (including the application of PHM to systems). He is an Associate Editor of the IEEE Transactions on Electronics Packaging Manufacturing and a member of the editorial board of the International Journal of Performability Engineering. He is the author of over 150 technical publications and several books on electronic packaging and electronic systems cost analysis. He has a B.S. degree in engineering physics from the University of Colorado, Boulder, in 1982, and the M.S. degree in electrical science and Ph.D. degree in electrical engineering, both from the University of Michigan, Ann Arbor, in 1983 and 1987, respectively.