A RISK ASSESSMENT METHODOLOGY AND TOOL FOR PROBABILISTIC DAMAGE TOLERANCE-BASED MAINTENANCE PLANNING

Michael C. Shiao[a], Kevin Boyd[b], and Scott Fawaz[c]

[a] William J. Hughes Technical Center
Federal Aviation Administration

[b] Aerospace Operations
S&K Technologies, Inc.

[c] Center for Aircraft Structural Life Extension
United States Air Force Academy

Abstract

This paper demonstrates a generic concept for probabilistic damage tolerance-based maintenance planning. This concept is based on the integration of damage tolerance methodology, simulation-based probabilistic methods[1] for inspection-free structures, and a recursive probability integration (RPI) method[24] for maintenance-related uncertainties. Based on this concept, a generalized probabilistic methodology for risk and damage tolerance analysis-based maintenance planning was developed to account for uncertainties associated with materials, structures, loadings/environments as well as uncertainties related to maintenance practices such as inspection scheduling and techniques, replacement and retirement decision, and repair quality (including maintenance-induced damage). Prototype probabilistic software that integrates the damage tolerance analysis software AFGROW[5], the Monte Carlo simulation method, and the RPI method was also developed for the proof of concept.

The risk assessment is accomplished by a two-stage numerical procedure. Damage accumulation histories without inspection are first generated through simulation-based probabilistic methods such as the Monte Carlo simulation or Importance Sampling methods. AFGROW is used for damage accumulation in each simulation. The RPI method is then applied to account for uncertainties related to inspection scheduling and techniques as well as repair and replacement decision. As a result of the two-stage numerical procedure, computational time is reduced significantly compared to full Monte Carlo simulations, including maintenance-related uncertainties. In addition, repeated risk assessments for various maintenance planning can be performed using the same set of damage accumulation histories generated in the first numerical stage at a fraction of the total computational time. The study also shows that this methodology is suitable for risk assessment with different damage scenarios and failure requirements, and there is no restriction on the number of random variables to be considered in the assessment.

Introduction

In the last decade, several probabilistic methods and associated software have been developed for damage tolerance applications considering uncertainties and inspection planning. While each of the methods work well for the damage scenario considered and uncertainty modeling, these methods are limited in their ability to solve general problems. Therefore, general and accurate methods are needed for damage tolerance analysis with maintenance planning under various uncertainties, including equivalent initial flaw size (EIFS) and location, material properties, loads and environmental effects, inspection schedules, probability of detection (POD), and repair quality and frequency. Realizing the importance and difficulty in maintenance risk management, the Federal Aviation Administration William J. Hughes Technical Center, Wright Patterson Air Force Base, the Aging Aircraft Support Squadron of the Aeronautical Systems Center, and S&K Technologies began a joint effort in methodology and software development.

Monte Carlo simulation (MCS) offers the most robust and reliable solution framework for general problems. The only, but major, issue is that MCS is usually time-consuming. For maintenance optimization, the computational issue is further amplified because the conventional approach requires an MCS for each different maintenance plan. As a result, many sets of MCSs are required to search for the optimal solution by exploring the design space that consists of many possible combinations of inspection scheduling, techniques, and repair/replacement/retirement strategies.

To relieve the computational burden in generating crack growth histories for many sets of MCSs, this paper describes and demonstrates an efficient method that combines the generality of MCS with the efficiency of analytical probabilistic methods. The core of the method is a recursive probabilistic integration (RPI) method that allows repeated use of baseline MCS-based crack growth histories for various maintenance plans. The fundamental concept of RPI is based on branching out the probable events after each maintenance action following an inspection where the POD is applied. The probability of occurrence of each branched event is then determined based on the probability of crack detection. In addition to allowing the reuse of crack growth histories, RPI has an additional benefit of improving the MCS sampling efficiency, especially when a maintenance plan significantly reduces the probability of failure.

The accuracy and efficiency of MCS with RPI has been demonstrated[4] using an infinite plate with a center crack subjected to uniform tension stress. A simple Paris crack growth equation with an analytical stress-intensity factor solution was implemented. A failure occurs when the crack size reaches a critical value. In this paper, risk assessment with various failure mechanisms and requirements as well as different definitions of probability of failure were performed using fixed inspection intervals as well as a risk-controlled inspection strategy. The generality, accuracy, and efficiency of this methodology are demonstrated in this paper using three representative examples reflecting general damage scenarios and various definitions of probability of failure.

General Formulation of Probabilistic Damage Tolerance Analysis With Maintenance Planning

To formulate probabilistic damage tolerance analysis with maintenance planning properly, several key technical issues are discussed in this section. Those issues are (1) definitions of probability of failure, (2) maintenance event tree development, (3) repair quality modeling, (4) probabilistic analysis frame work, and (5) software development requirements.

Definitions of Probability of Failure

There are several different measures of probability of failure due to various definitions. Those used in this study are defined below.

- The unfailed probability of failure (UPOF) is the probability of failure determined, assuming there is no prior failure up to the time of assessment. This definition of probability of failure is used in the software PROF[6]. As a result, those that have failed before time t can be repaired in the UPOF calculation if they are detected during inspections.

- The cumulative probability of failure (CPOF), *F(t)*, is the cumulative probability of time to failure due to failure occurrences before time t. This definition is the same as the total probability of failure defined in reference 7. Unlike the definition for UPOF, those that have failed will never be fixed under the CPOF calculation.

- The instantaneous failure rate[8], or hazard function, *h(t)*, is defined as the probability that a failure occurs in a unit time interval, given that a failure has not occurred prior to t, the beginning of the interval. This can be considered as the probability of failure per unit time. The unit time can be a cycle or a flight hour. The instantaneous failure rate is derived from CPOF, as shown in equation 1.

  $$h(t) = f(t)/(1 - F(t)) \qquad (1)$$

  where *f(t)* is the probability density of time to failure.

Maintenance Event Tree Development

In the case of damage accumulation processes, it may be uneconomical to design a structure in such a way that the reliability is sufficient for an entire design life. A more economical solution can be obtained by establishing a maintenance scheme. In those cases, failures will not occur if the inspection detects damages and repair action is taken immediately. The sequence of inspection and repair events can be represented in an event tree, as shown in figure 1.

Let the first inspection I_1 be planned at time t_1. At this node, there exists a branch with three possible events: (1) event of failure F—a failure occurs before t_2, (2) event of repair R—the inspection detects a serious defect and repair is conducted, and (3) event of no-

finding I_2—no serious defect is detected and the next inspection at $t = t_2$ is planned. When a detected defect has been repaired, a new sequence is started and a new branch is generated at the next inspection.

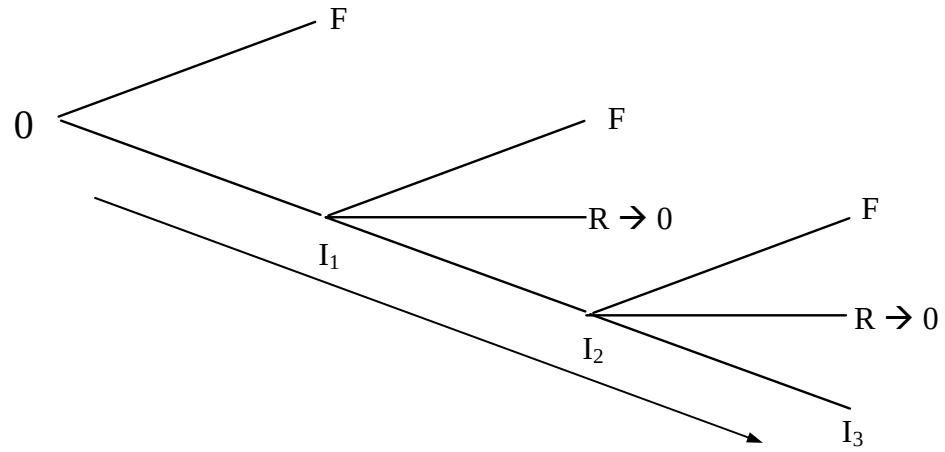

Figure 1. Maintenance Event Tree

Repair Quality Modeling

Perfect repair is unlikely. Often, field repair may produce inferior quality due to insufficient equipment, inability to acquire high-quality materials in time, poor repair skills, etc. In addition, repair most often removes the crack tip of the propagating crack, yet it often leaves behind potential initiation sites for cracks to grow during future operation. For this reason, assuming a perfectly repaired structure would create unconservatism. In general, a postrepair crack size distribution that is worse than the original parts has to be estimated. Hovey, et. al [6] used the equivalent repair quality to quantify the possible repair flaws inherited in the structures after repairs. Although the effect of repair quality on the failure probability is not immediate, it can have a major effect on aging structures. In addition, the effect of the potential maintenance-induced damage should be modeled by further adjusting the postrepair crack size distribution.

Probabilistic Analysis Framework

Aircraft structures are subjected to in-service inspections and subsequent maintenance actions, if warranted, to maintain reliability and minimize risk. Since the effectiveness of an inspection depends on the POD and the location and size of a crack, the optimal design and maintenance of reliable structures requires methods to evaluate time-dependent reliability analysis subject to inspections. For practical purposes, such methods should provide a quick turnaround of reliability analysis for design and inspection planning. The framework for probabilistic and damage tolerance-based

maintenance planning is summarized in figure 2. A comprehensive framework should include a wide range of uncertainties, including:

- Random or uncertain parameters in material (e.g., threshold of the stress-intensity factor, modulus of elasticity)
- Defect or flaw (including size, shape, location, and the frequency of occurrence)
- Loading, type of usage (with frequency of occurrence)
- Finite element model (including modeling error)
- Crack growth model (including modeling error)
- Maintenance (including inspection schedules, frequency of inspections, POD curves, repair/replacement methods and effects)
- Human factors

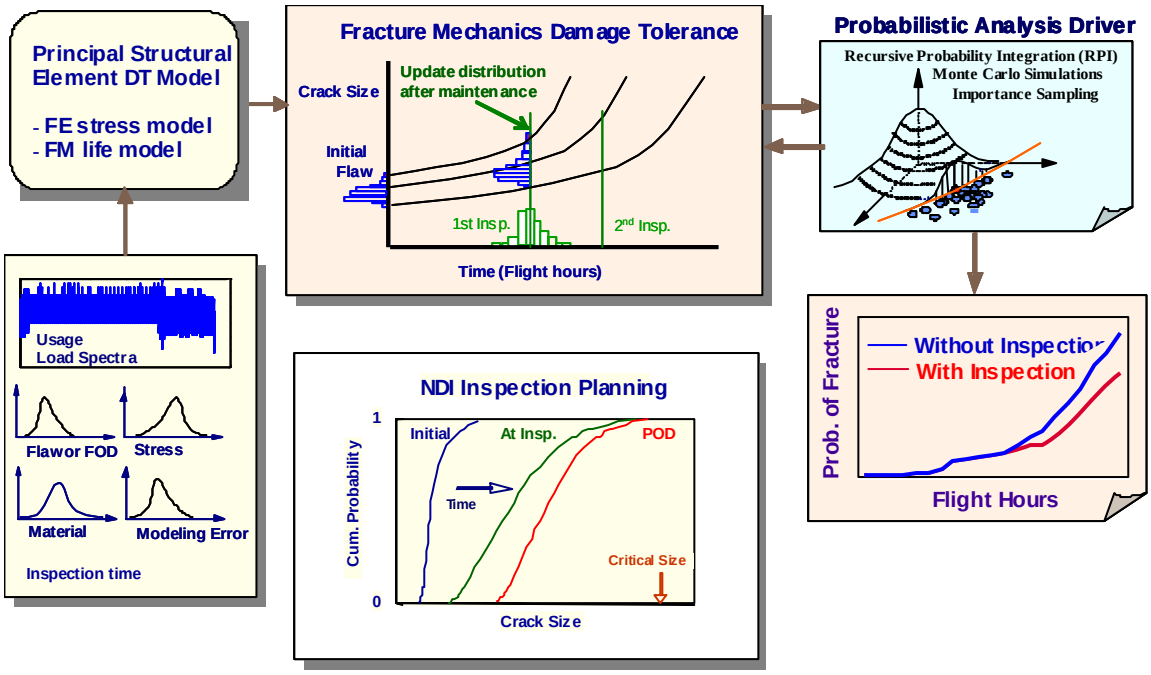

Figure 2. Probabilistic Damage Tolerance Analysis-Based Maintenance Planning

Software Development Requirements

The methodology adopted in this software, shown in figure 2, represents a generic risk assessment for maintenance planning and optimization. The associated analysis software will be capable of treating various damage scenarios and mechanics and various forms of uncertainties, including those that are maintenance-related. Currently, only MCS and RPI methods specifically addressing the maintenance-related problems are integrated in the prototype probabilistic software. In the future, importance sampling methods should be implemented to further reduce the computational time. A parallelization feature, which uses supercomputers with multiple CPUs for improved efficiency, is also desirable. The software will also be developed in such a way that the burden of

integration with any deterministic analysis software, such as NASGRO, AFGROW, and other in-house damage tolerance software, can be eased and minimized. In addition, this software will be used to verify new probabilistic methods and provide error-checking to ensure the reliability of failure prediction.

Recursive Probabilistic Integration Method

Two-Stage Numerical Procedure

The RPI method is a two-stage numerical procedure. The first stage computes the crack growth history without inspections for original and/or repair parts. The second stage accounts for maintenance-related uncertainties by the RPI method together with the crack growth histories generated from stage 1. The new method reduces the computational burden of using MCS alone. In addition, the new probabilistic method allows the reuse of stored crack growth histories without inspections to compute risk as a function of inspection time and/or inspection techniques without additional stress and life analyses. This method is particularly useful for inspection optimization or maintenance planning.

Recursive Probabilistic Integration Formulation

The concept of RPI is discussed in this section. Figure 3 shows the possible fatigue crack growth paths from an MCS without inspections. Failure of each fatigue path is indicated by the cross symbol for any failure requirement, such as net section yield, critical crack size, or fracture criteria. In the figure, PDF stands for probability density function.

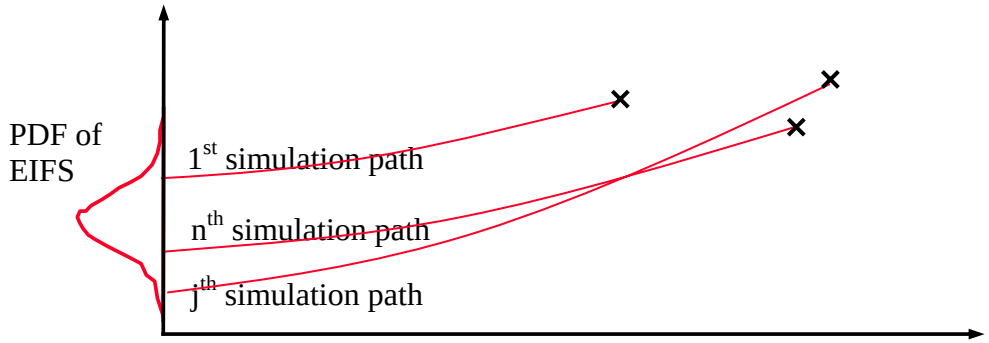

Figure 3. Fatigue Life and Fatigue Crack Growth Paths Without Inspection

To illustrate the RPI concept using graphical presentation for clarity, fatigue life simulations without inspections are shown in figure 4. In the figure, the horizontal line indicates the beginning and end of fatigue life of a simulation without inspection. This line is called “fatigue path,” since it implicitly represents the fatigue crack growth path from the beginning to the end. Keep in mind that at any instance t along j^{th} fatigue path, there is a crack size *a(j,t)*.

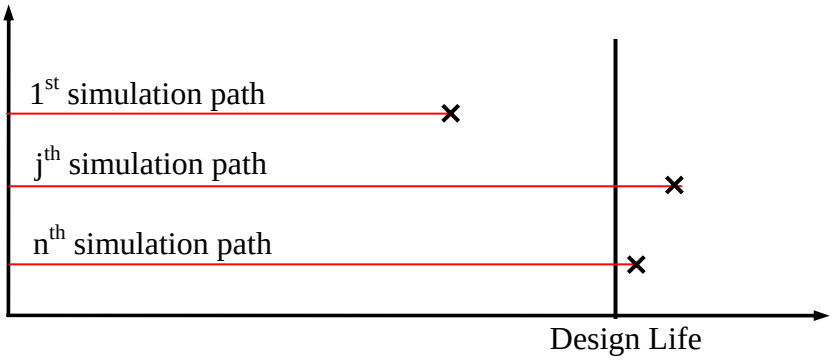

Figure 4. Fatigue Paths of an MCS Without Inspection

Now let's investigate the probability calculation. When n simulations are conducted in an MCS, the probability of occurrence of each path (simulation) is 1/n. The conditional probability of failure of jth fatigue (simulation) path, given the occurrence of this path, $P_f^{\,c}(j)$, is defined by equation 2, where c stands for conditional.

$$\begin{aligned} P_f^{\,c}(j) &= 0 \qquad & j^{th}\ fatigue\ life > design\ life \\ P_f^{\,c}(j) &= 1 \qquad & j^{th}\ fatigue\ life < design\ life \end{aligned} \qquad (2)$$

The probability of failure for an MCS without inspection can be obtained by equation 3.

$$P_f = \frac{1}{n}\sum_{j=1}^{n} P_f^{\,c}(j) \qquad (3)$$

Now let's add one inspection to the analysis. Figure 5 shows an inspection at time *t* is conducted on the jth fatigue path.

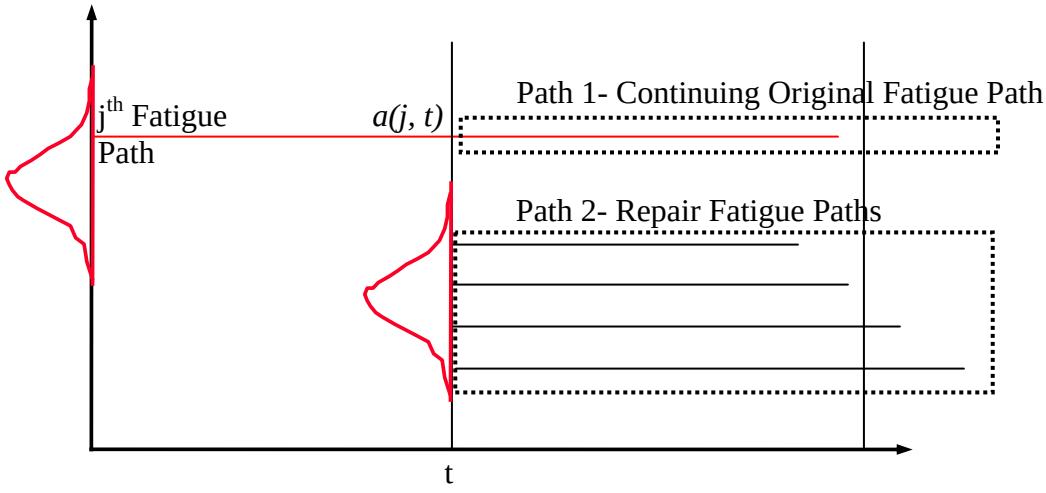

Figure 5. Probabilistic Branching Due to Inspection and Repair

The crack size at the time of inspection is denoted as *a(j, t)*. Due to the inspection, j^{th} fatigue path will branch out to two possible paths. If the crack is not detected, the j^{th} fatigue path will continue its original fatigue path as indicated by Path 1 in figure 5. When a crack is detected and repaired immediately, the continuing path can be either one of the multiple repair paths, indicated as Path 2 in figure 5. As a result of probability branching, $P_f^c(j)$ in equation 3 needs to be updated accordingly by equation 4.

$$P_f^c(j) = POD(a(j,t)) * P_f(path\,1) + (1 - POD(a(j,t))) * P_f(path\,2) \qquad (4)$$

POD(a(j,t)) is the POD of crack size *a(j,t)*. $P_f(path\ 1)$ is the conditional probability of failure of the original fatigue path, as defined in equation 2. $P_f(path\ 2)$ is the probability of failure of repair paths, which can be obtained by an MCS without inspection starting at the time of inspection using equations 2 and 3. Once $P_f^c(j)$ is updated, the probability of failure with one inspection can be computed by equation 3. Using the same analogy, probability of failure for multiple inspections can be derived. The following paragraph will give more details of the RPI method. Several key parameters used for RPI are defined below and illustrated in figure 6.

t_s	Service life (or design life)
m:	Number of inspections
MCS(k)	A Monte Carlo simulation without inspections starting at k^{th} inspection, where k = 0 to m, k = 0, representing simulations with original parts; and k>0, representing simulations with repair parts
n_k:	Number of simulations for MCS(k)
t_f(k,j):	Fatigue life of the j^{th} simulation of MCS(k)
a(k,j,i)	The crack size at the i^{th} inspection of the j^{th} fatigue path of MCS(k); k = 0 to m; j = 1 to n_k; for each k, i = k+1 to m
Full Path i	A probability event consists of complete fatigue paths of an MCS considering subsequent inspections starting at i^{th} inspection, i = 0 to m
Br(k,j,i)	A branched probability event consists of all possible fatigue paths of an MCS considering subsequent inspections starting at i^{th} inspection with the following conditions. The initial condition of the fatigue branch Br(k,j,i) is inherent from the condition at i^{th} inspection of the j^{th} fatigue path of MCS(k) where k = 0 to m; for each k, i=k to m-1; for a pair of k and i, Br(k,j,i) consists of fatigue branch Br(k,j,i+1) and Full Path (i+1)

Figure 6 show a full repair path starting at $(m-2)^{th}$ inspection. The dashed line represents the j^{th} fatigue path of MCS(m-2), and the circle indicates that an inspection has been conducted along this path. A diamond indicates that a repair action has been taken after detecting a crack. The solid line with a diamond at the beginning and an arrow at the end represents an MCS considering subsequent inspections. It also represents a Full Path starting at the time indicated by the diamond.

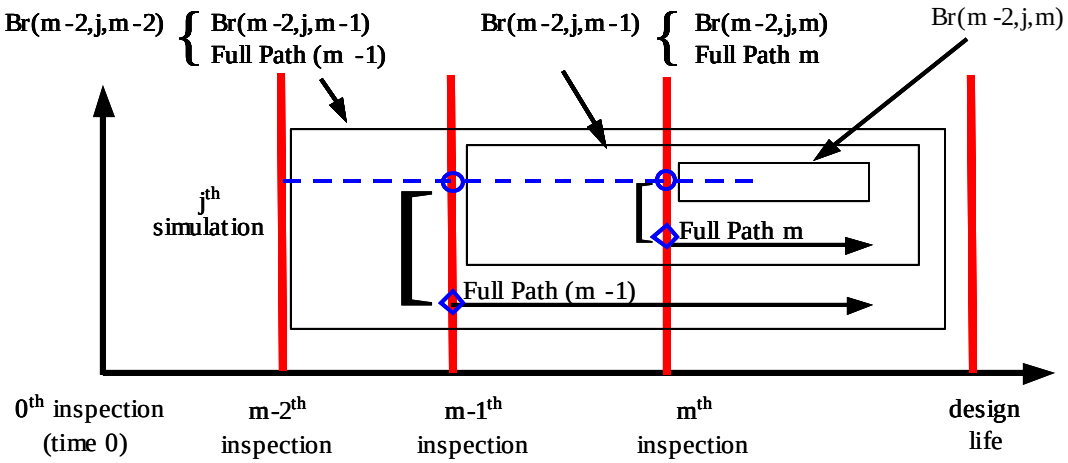

Figure 6. Full Path(m-2) With Two Subsequent Inspections

Figure 6 also shows that Br(m-2,j,m-2) consists of a single (j^{th}) realization of MCS(m-2) and two possible branched probability events, Full Path (m-1) and Full Path (m), after subsequent inspections along this realization. From the definition of Full Path, Full Path (m-2) is also represented by the probability events Br(m-2,j,m-2), where j = 1, n_{m-2}. The following are also shown in figure 6: Br(m-2,j,m-2) consists of Br(m-2,j,m-1) and the Full Path (m-1); Br(m-2,j,m-1) consists of Br(m-2,j,m) and the Full Path m; and Br(m-2,j,m) represents the last segment of j^{th} fatigue path of MCS(m-2).

Now let's follow the j^{th} simulation path to investigate the probability of occurrence of each probability event. At the $(m-1)^{th}$ inspection, if a crack is detected and a repair action is taken, the probability of occurrence of a repair action (or Full Path (m-1)) is equal to the POD for the crack size *a*(m-2,j,m-1). If no crack is detected, probability of occurrence of a fatigue branch Br(m-2,j,m-1) is equal to 1 minus POD. As defined before, Full Path (i), where i > 0, represents the probability event after a repair action is taken at i^{th} inspection. It is noted that Full Path (i) depends upon index i but is independent of indices k and j (same indices for *a*(k,j,i) and Br(k,j,i)), which means this event can be used repeatedly for any k and j. In another words, the probability of failure of this probability event just needs to be calculated once. However, the probability of occurrence of this event will be the probability of detection of crack size *a*(k,j,i), which is a function of indices k and j.

The basic concept described above leads to the development of the RPI algorithm, as shown in equations 5 and 6.

$$P_f^{\,Full\ Path\ k} = \frac{1}{n_k}\sum_{j=1}^{n_k} P_f^{\,Br(k,j,k)} \qquad where\ \ k = m\ to\ 0 \qquad (5)$$

In equation 5, $P_f^{\,Br(k,j,k)}$ for k < m is calculated recursively by equation 6.

$$P_f^{Br(k,j,i)} = (1 - pod\ (a(k,j,i+1)))* P_f^{Br(k,j,i+1)} + pod\ (a(k,j,i+1))\ * P_f^{Full\ Path\ (i+1)}$$
$$where \quad i = m-1\ to\ k \qquad (6)$$

Using RPI, only two sets of MCSs without inspections are needed, one uses the original crack size distribution and the other uses the postrepair crack size distribution (assuming there is only one such distribution in maintenance planning). The computational procedures are summarized below.

First, conduct an MCS or, when applicable, more efficient importance sampling methods for the original parts without inspection and similarly for repaired components. Probability of failure of Br(k,j,m) are then determined by equation 7 using the simulation results starting at the k^{th} inspection.

$$P_f^{Br(k,j,m)} = 0 \quad if \quad t_f(k,j) > t_s$$
$$= 1 \quad if \quad t_f(k,j) < t_s \quad where \quad k = 0\ to\ m \quad and \quad j = 1\ to\ n_k \qquad (7)$$

With the initial conditions, the probability of failure considering inspection effect can be determined by the recursive equations 5 and 6. At the end of computation, the P_f of Full Path 0 is found. It represents the combined P_f considering all the branches and events. Equation 7 is for the calculation of UPOF. For the calculation of CPOF, *pod(a(k,j,i+1))* in equation 6 needs to be set to 0 when a component is failed before inspection to avoid repair action.

Numerical Examples

The efficiency, accuracy, and generality of the risk assessment concept are demonstrated in the following, using three cases based on their respective failure requirements, structural details, and uncertainty consideration. Case 1 is based on PROF risk assessment methodology, considering three random variables with a known crack growth history as well as an analytical solution for stress-intensity factor calculation. Cases 2 and 3 demonstrate the generic probabilistic approach with an unlimited number of random variables and a generic crack growth analysis. Each case was analyzed using different inspection techniques and various inspection strategies for different definitions of probability of failure. Four POD curves were used throughout the study and are described by the following equation[6].

$$pod\ (a) = 1 / [1 + exp(- \frac{\pi}{\sqrt{3}} \frac{ln(a - a_{min}) - ln(a_{50} - a_{min})}{q})] \qquad (8)$$

In the equation, a_{50} represents the median crack size that can be detected 50% of the time and q is the scale parameter; $a_{\min}$ is the minimum crack size that can be detected. a_{50}, $a_{\min}$ and q for each POD are shown in table 1, and the POD curves are plotted in figure 7.

Table 1. Parameters in Equation 8 for POD Curves

	POD Curve 1	POD Curve 2	POD Curve 3	POD Curve 4
$\mathbf{a_{50}}$	0.05	0.05	0.10	0.10
$\boldsymbol{q}$	0.50	0.75	0.50	0.75
$\mathbf{a_{min}}$	0.00	0.00	0.00	0.00

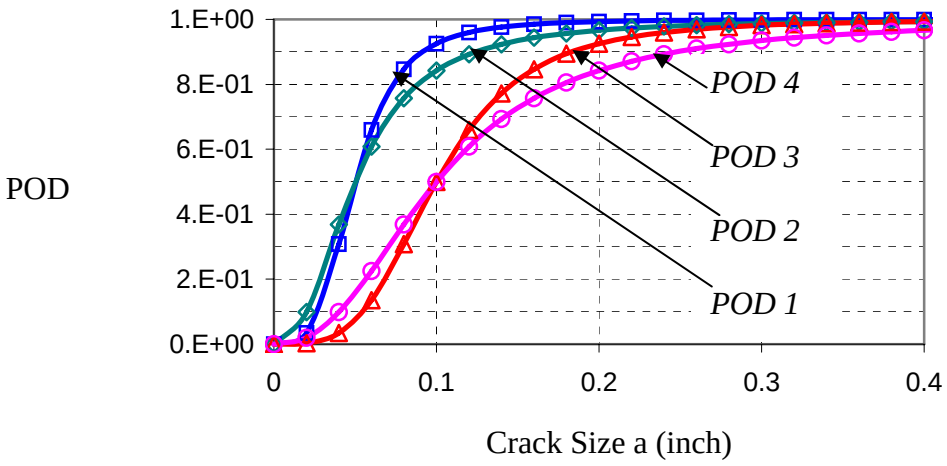

Figure 7. Probability of Detection Curves

Case 1:

Case 1 is based on PROF probabilistic analysis methodology, as described in reference 6, for crack growth initiated at a fastener hole. Three random variables, maximum stress, fracture toughness, and EIFS were considered. Failure occurs when the stress-intensity factor is greater than the fracture toughness.

The maximum stress distribution is described by a Gumbel distribution, as shown in equation 9, where A = 1.31 and B = 14.6. The EIFS is described by equation 10 for a Weibull distribution with η = 0.0061 and β = 0.996. Fracture toughness is described by a Normal distribution with mean = 35 and standard deviation = 3.1. PDFs of maximum stress and EIFS are plotted in figures 8 and 9.

$$P(\sigma) = exp(-exp((\sigma - B)/A)) \qquad (9)$$

$$G(a) = 1 - exp(-(a/\eta)^{\beta}) \qquad (10)$$

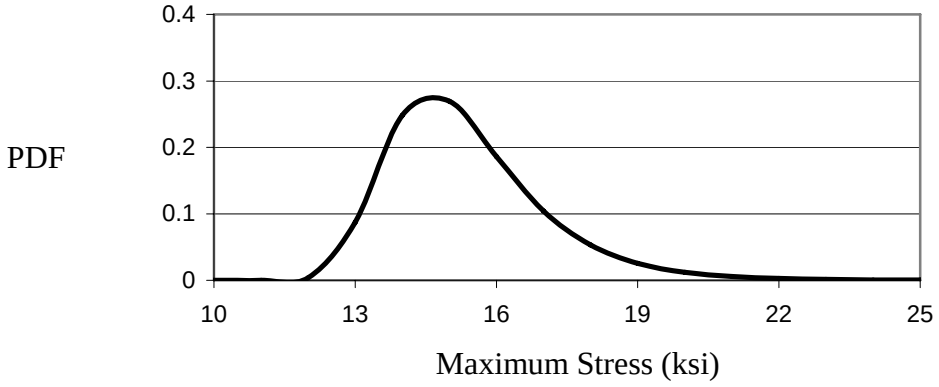

Figure 8. Probability Density Function of Maximum Stress

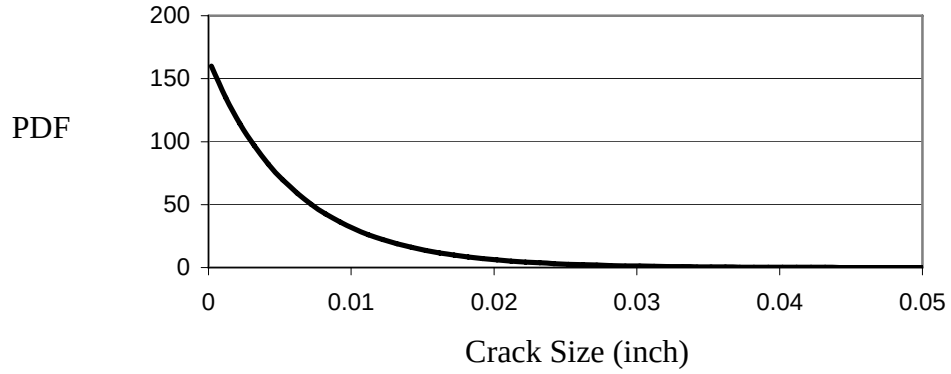

Figure 9. Probability Density Function of EIFS

The deterministic crack size *a* versus time *t* for damage tolerance analysis is shown in figure 10. The relationship between normalized stress-intensity factor (K/σ) and crack size is modeled as a deterministic input and is defined in figure 11.

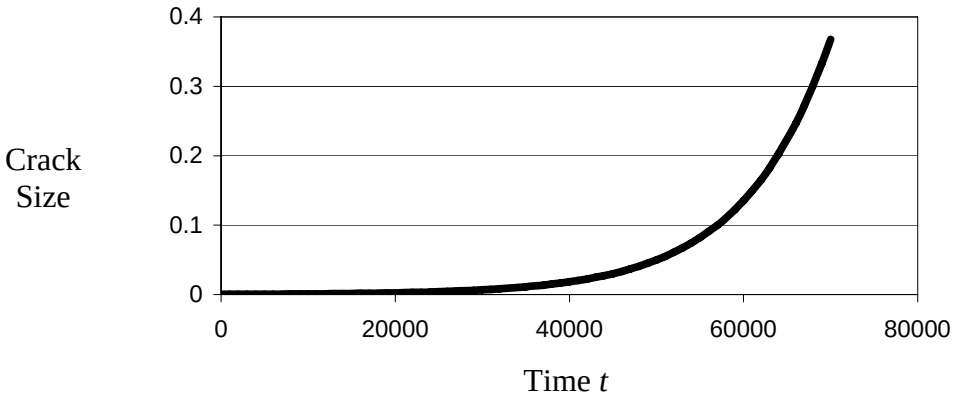

Figure 10. Crack Size *a* Versus Time *t*

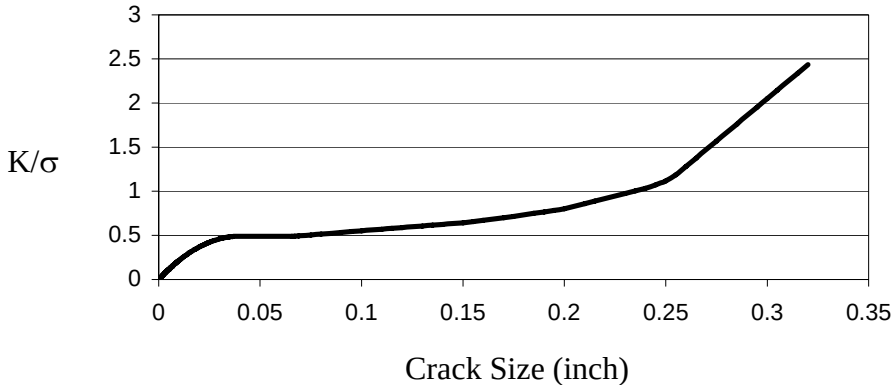

Figure 11. Normalized-Stress Intensity Factor versus Crack Size

For simplicity, the quality of repair parts is assumed to be as-manufactured. In addition, the failure mechanism remains the same after repair. Therefore, only one MCS set, without inspections, is needed for RPI method. The same assumption for repair quality is used for all three cases. The UPOF was calculated based on three probabilistic methods: PROF, MCS, and RPI at 20,000, 25,000, and 30,000 cycles. The accuracy and efficiency of probability predictions among various probabilistic methods were compared. The results are shown in tables 2 to 5.

Table 2. Results Using POD Curve 1

		POD Curve 1			No Inspection
	LIFE	PROF	MCS	RPI	MCS
Unfailed Probability of Failure	20,000	3.4E-08	3.9E-08	3.7E-08	9.1E-04
	25,000	2.5E-07	2.2E-07	2.4E-07	1.4E-02
	30,000	7.9E-07	7.3E-07	7.5E-07	7.4E-02
Number of simulations required for 20% error and 95% confidence	20,000		2,500,000,000	110,000	
	25,000		450,000,000	7,000	
	30,000		140,000,000	1,500	

Table 3. Results Using POD Curve 2

		POD Curve 2			No Inspection
	LIFE	PROF	MCS	RPI	MCS
Unfailed Probability of Failure	20,000	9.3E-07	7.7E-07	7.7E-07	9.1E-04
	25,000	5.8E-06	5.2E-06	4.7E-06	1.4E-02
	30,000	1.7E-05	1.3E-05	1.3E-05	7.4E-02
Number of simulations required for 20% error and 95% confidence	20,000		130,000,000	110,000	
	25,000		20,000,000	7,000	
	30,000		8,000,000	1,500	

Table 4. Results Using POD Curve 3

		POD Curve 3			No Inspection
	LIFE	PROF	MCS	RPI	MCS
Unfailed Probability of Failure	20,000	1.6E-05	1.4E-05	1.4E-05	9.1E-04
	25,000	1.6E-04	1.4E-04	1.4E-04	1.4E-02
	30,000	5.6E-04	4.7E-04	5.1E-04	7.4E-02
Number of simulations required for 20% error and 95% confidence	20,000		70,000,000	110,000	
	25,000		800,000	7,000	
	30,000		220,000	1,500	

Table 5. Results Using POD Curve 4

		POD Curve 4			No Inspection
	LIFE	PROF	MCS	RPI	MCS
Unfailed Probability of Failure	20,000	3.9E-05	3.4E-05	3.2E-05	9.1E-04
	25,000	3.8E-04	3.1E-04	3.2E-04	1.4E-02
	30,000	1.4E-03	1.2E-03	1.2E-04	7.4E-02
Number of simulations required for 20% error and 95% confidence	20,000		30,000,000	110,000	
	25,000		330,000	7,000	
	30,000		80,000	1,500	

As shown in tables 2 to 5, the results predicted by the three methods compare well. However, the RPI method shows significant computational time reduction compared to the MCS method when better inspection techniques were used. In addition, the efficiency of the RPI method directly depends on the failure probability level of inspection-free structures, as explained below.

The computational time needed for an MCS depends on the probability level P. In this study, the number of simulations required for a traditional MCS with inspection, N_{mcs}, is based on equation 11 for 95% confidence and 20% error with P equal to $P_{f\text{-}i}$, where $P_{f\text{-}i}$ represents P_f with inspections.

$$N = 100 / P \qquad (11)$$

Since the RPI method uses the results of inspection-free components, its computational time depends directly on the probability of failure at service life without inspection, $P_{f\text{-}o}$. Therefore, using the same equation but letting P equal $P_{f\text{-}o}$, the number of simulations required for RPI, N_{rpi}, can be calculated. Since the P_f without inspections is always no less than the P_f with inspections, RPI requires fewer samples in the MCS. One can expect that a more effective detection method will produce greater time savings.

Tables 2 to 5 also show the number of simulations required to achieve a probability prediction with 20% error and 95% confidence. As shown in the tables, the number of simulations required for both RPI and MCS methods without inspection are the same. This finding is further demonstrated in figure 12. In the figure, the effectiveness of the crack detection method is represented by the probability ratio, $P_{f\text{-}o}/P_{f\text{-}i}$. The larger the ratio, the more effective the detection method. The efficiency of the RPI method is measured by the ratio of N_{mcs}/N_{rpi}, where N_{mcs} and N_{rpi} are the number of simulations required to achieve 20% error and 95% confidence of the probability prediction for MCS and RPI, respectively. It shows that the RPI method is more efficient when the probability ratio is high. On the other hand, the time savings using RPI is minimal when the POD is ineffective.

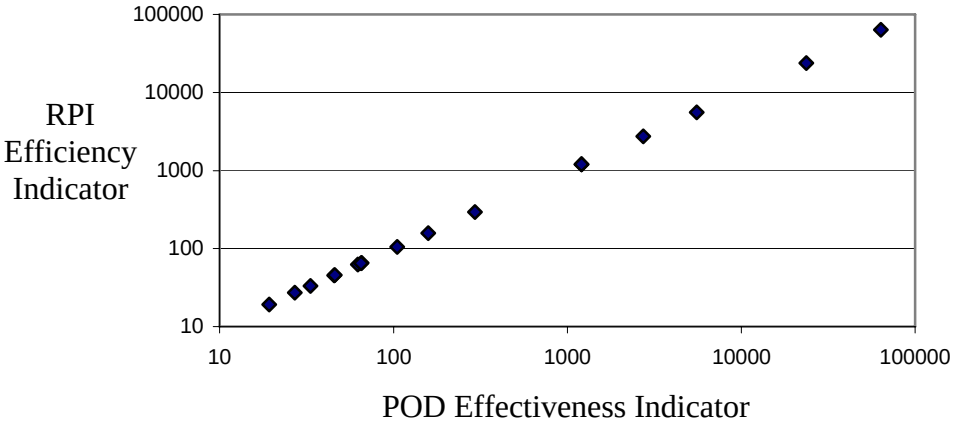

Figure 12. Efficiency of the RPI Method

Case 2:

The structural detail of case 2 is shown in figure 13. A failure occurs when the crack size reaches a critical value of 0.20 inch. The UPOF was calculated using two different inspection intervals: a fixed inspection interval of 4000 cycles and a reliability-based inspection interval that ensures the UPOF to be less than 10^{-6}. The random variables in the study include the EIFS, stress amplitude, thickness, yield strength, hole diameter, and C in the Walker equation. Crack growth histories were generated using the general-purpose fracture mechanics software AFGROW.

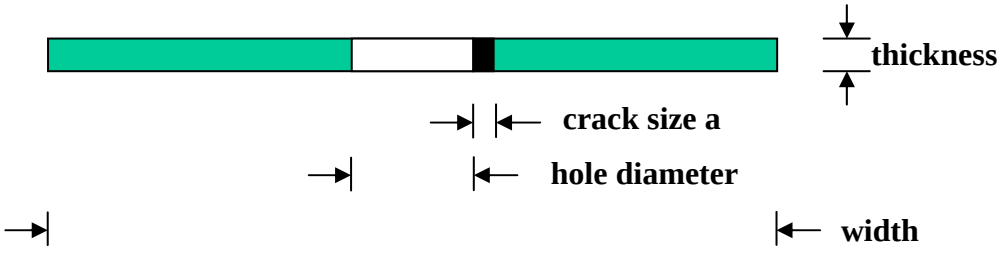

Figure 13. Structural Detail for Case 2

The probability distribution of the EIFS is the same as that used in Case 1. The stress amplitude has a Gumbel distribution with A = 0.75 and B = 15, as described in equation 9. The statistics of the remaining random variables are shown in table 6.

Table 6. Statistics of Uncertainties for Case 2

	Distribution Type	Mean	Coefficient of Variation
Thickness	Normal	0.063 in.	0.03
Crack Growth	Lognormal	2E-09 in./cycle	0.07
Hole Diameter	Normal	0.1875 in.	0.03
Yield Strength	Lognormal	40 ksi	0.03

POD curve 1 was used for the demonstration, and the results are shown in figure 14. In the figure, the line with triangular markers represents the probability of failure without inspection. The line with square markers represents the unfailed P_f with inspections at every 4000 cycles. The line with diamond markers represents the unfailed P_f with a reliability-based inspection strategy that unfailed $P_f < 10^{-6}$.

In this case, two strategies of setting inspection interval were demonstrated for a given inspection technique using the same crack growth histories (database) without inspection. Similarly, repeated risk assessments with different inspection techniques can be accomplished with high efficiency using the same database. This information can be used later to determine the optimal maintenance plan with cost and accuracy considerations.

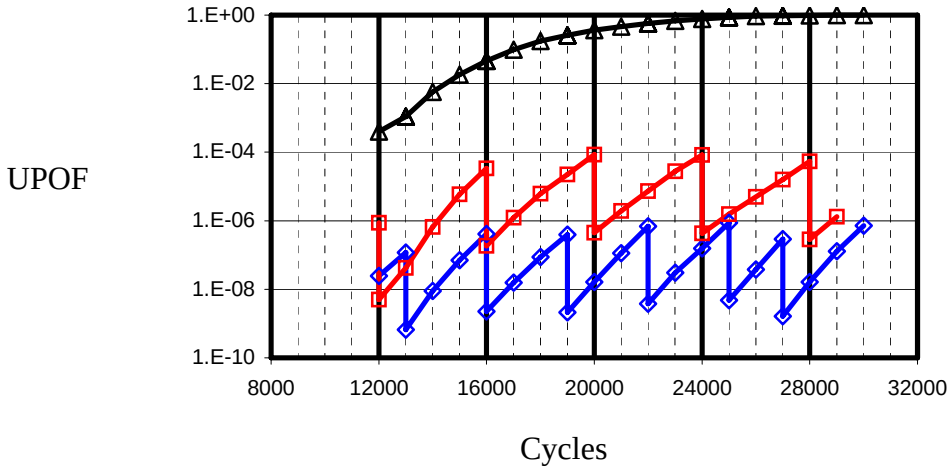

Figure 14. Unfailed Probability of Failure With Fixed and Reliability-Based Inspection Intervals

Case 3:

The same structural detail and uncertainty consideration as for case 2 were used in case 3. However, the failure criterion is based on net section yield. POD curve 4 was used to represent the inspection method. Inspections were conducted every 4000 cycles. This case was to demonstrate the generality feature of the new concept in computing the CPOF, UPOF, and hazard functions. Crack growth histories generated from case 2 were used again in case 3 without lengthy computation since the *a*-versus-*t* database is not affected by the failure mechanisms, inspection methods, or inspection intervals.

The results are shown in figures 15 and 16. In figure 15, the square markers represent the CPOF, and the triangle markers represent the UPOF. As can be seen, CPOF is significantly different than UPOF. The difference is due to the repair action being taken on the components that have failed prior to inspection for UPOF calculation. For CPOF calculation, components that have failed prior to inspection will remain failed throughout the analysis.

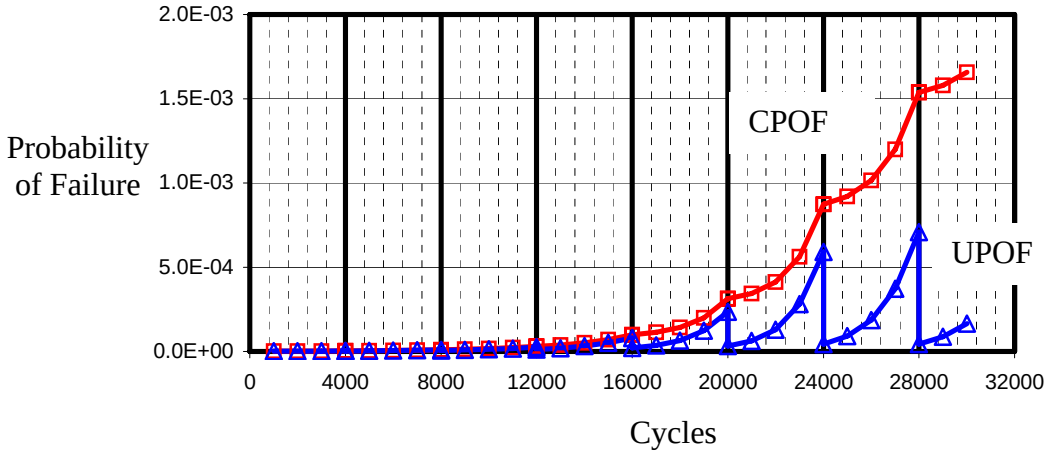

Figure 15. Comparison of UPOF and CPOF

For the reliability-based design assessment, CPOF and hazard function h(t) are useful design parameters to assess reliability-based design criteria. Paul White[7] described a reliability-based design methodology using CPOF and hazard function h(t) in his paper presented at the 6th Joint FAA/DoD/NASA Aging Aircraft Conference. This methodology requires the CPOF as well as the instantaneous failure rate (single flight probability of failure) be less than the allowable probability level, respectively. Figure 16 shows the CPOF again and the hazard function h(t) in a log scale. The square markers represent the CPOF, and the diamond markers represent hazard function h(t). For a simple demonstration, let's set the allowable probability for CPOF to 10^{-3} and set that for the instantaneous failure rate to 10^{-6}. From figures 15 and 16, one can determine that the design life is about 26,000 cycles.

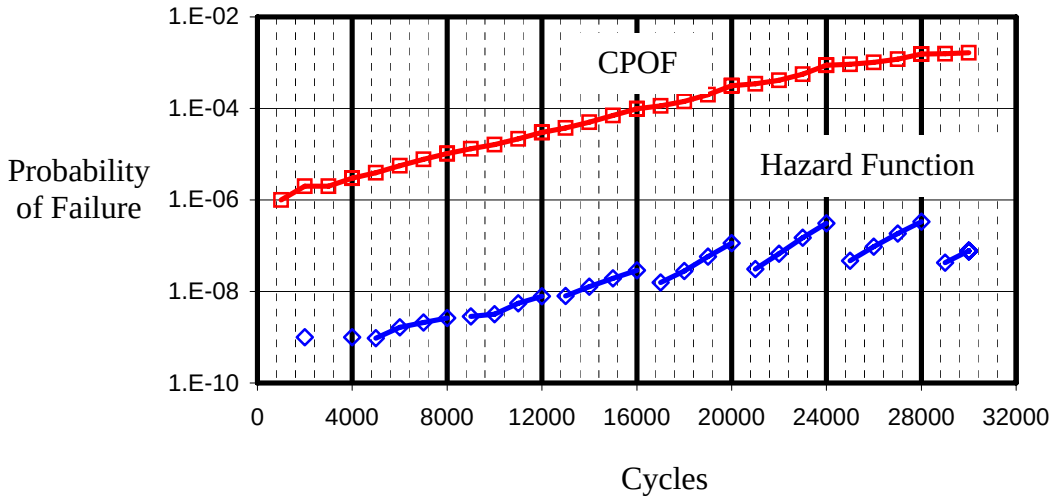

Figure 16. Cumulative Probability of Failure and Hazard Function

Summary and Discussions

This paper demonstrates a generic concept for probabilistic damage tolerance-based maintenance planning. This concept is based on the integration of damage tolerance methodology, probabilistic methods for inspection-free structures, and a recursive probability integration (RPI) method accounting for maintenance-related uncertainties.

Reduction in computational time using the RPI method relative to the conventional simulation-based approach comes from two sources: (1) The required number of Monte Carlo simulations (MCS) for the RPI method depends directly on the P_f without inspections and the larger the failure probability predicted by an MCS, the fewer samples are required. Since P_f without inspection is always no less than P_f with inspections, this leads to computational time savings, especially when the P_f is reduced significantly by an effective maintenance plan. (2) The *a*-versus-*t* database can be used repeatedly for different failure requirements; different inspection methods; and various inspection intervals, fixed or reliability-based.

Since the RPI is a simulation-based method that allows consideration for various damage scenarios, detection techniques, failure requirements, definitions of probability of failure, etc., it has a wide range of applications. As demonstrated, the unfailed probability of failure, cumulative probability of failure, and instantaneous failure rate or hazard function were calculated. A simple illustration of using the cumulative probability of failure and instantaneous failure rate for reliability-based design assessment was given. The application area could include corrosion and composite damage. It is suitable for optimal inspection scheduling with reliability and cost consideration. It can also be used to verify new probabilistic methods or provide error-checking to ensure the reliability of failure prediction.

References

1. Southwest Research Institute, "Probabilistic Structural Analysis Methods (PSAM) for Select Space Propulsion System Components," Final Report NASA Contract NAS3-24389, NASA Glenn Research Center, Cleveland, Ohio, 1995.

2. Shiao, M., Le, D., and Cuevas, E., "Feasibility Study of a Reliability-Based Damage Tolerance Analysis Method for Rotorcraft Structures and Dynamic Components," The Seventh Joint DoD/FAA/NASA Conference on Aging Aircraft, 8-11 September 2003, New Orleans, Louisiana.

3. Wu, Y-T., Shiao, M., Shin, Y., and Stroud, W.J., "Reliability-Based Damage Tolerance Methodology for Rotorcraft Structures," Proceeding of 2004 SAE World Congress, Reliability and Robust Design in Automotive Engineering, Special Publication No. 1844, March 2004.

4. Shiao, M., and Wu, Y-T., "An Efficient Simulation-Based Method for Probabilistic Damage Tolerance Analysis With Maintenance Planning," 9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, July 2004.

5. Harter, J.A., "AFGROW Users Guide and Technical Manual, AFGROW for Windows 2K/XP, Version 4.0009.12," June 2004.

6. Berens, A.P., Hovey, P.W., and Skinn, D.A., "Risk Analysis for Aging Aircraft Fleets," U.S. Air Force Wright Laboratory Report, WL-TR-91-3066, Vol. 1, October 1991.

7. White, P., Barter, S., and Molent, L., "Probabilistic Fracture Prediction Based on Aircraft Specific Fatigue Test Data," 6th Joint FAA/DoD/NASA Aging Aircraft Conference, September 16-19, 2002.

8. Kapur, K.C. and Lamberson, L.R., "Reliability in Engineering Design," John Wiley & Sons, Inc., 1997.