



AFRL-RQ-WP-TR-2016-0151V1

AIRFRAME DIGITAL TWIN (ADT)

Delivery Order 0001: Prognostic and Probabilistic Individual Aircraft Tracking (P²IAT)

Volume 1

Elias Anagnostou and Stephen Engel

Northrop Grumman Corporation

Northrop Grumman Aerospace Systems (NGAS)

OCTOBER 2016

Final Report

**DISTRIBUTION STATEMENT A: Approved for public release.
Distribution is unlimited.**

**AIR FORCE RESEARCH LABORATORY
AEROSPACE SYSTEMS DIRECTORATE
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542
AIR FORCE MATERIEL COMMAND
UNITED STATES AIR FORCE**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the USAF 88th Air Base Wing (88 ABW) Public Affairs Office (PAO) and is available to the general public, including foreign nationals.

Copies may be obtained from the Defense Technical Information Center (DTIC)
(<http://www.dtic.mil>).

AFRL-RQ-WP-TR-2016-0151V1 HAS BEEN REVIEWED AND IS APPROVED FOR
PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

This report is published in the interest of scientific and technical information exchange and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE					Form Approved OMB No. 0704-0188	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.						
1. REPORT DATE (DD-MM-YY) October 2016		2. REPORT TYPE Final		3. DATES COVERED (From - To) 17 December 2013 – 20 September 2016		
4. TITLE AND SUBTITLE AIRFRAME DIGITAL TWIN (ADT) Delivery Order 0001: Prognostic and Probabilistic Individual Aircraft Tracking (P ² IAT) Volume 1				5a. CONTRACT NUMBER FA8650-14-D-2413-0001		
				5b. GRANT NUMBER		
				5c. PROGRAM ELEMENT NUMBER 62201F		
6. AUTHOR(S) Elias Anagnostou and Stephen Engel				5d. PROJECT NUMBER 2401		
				5e. TASK NUMBER		
				5f. WORK UNIT NUMBER Q17D		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Northrop Grumman Corporation Northrop Grumman Aerospace Systems (NGAS) 1 Space Park Boulevard Redondo Beach, CA 90278-1001				8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Aerospace Systems Directorate Wright-Patterson Air Force Base, OH 45433-7542 Air Force Materiel Command United States Air Force				10. SPONSORING/MONITORING AGENCY ACRONYM(S) AFRL/RQVS		
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-RQ-WP-TR-2016-0151V1		
12. DISTRIBUTION/AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: Approved for public release. Distribution is unlimited.						
13. SUPPLEMENTARY NOTES PA Case Number: 88ABW-2018-5345; Clearance Date: 23 October 2018. This report contains an attached .pdf appendix file related to the research effort. Click the paperclip icon in the left vertical toolbar, and double click the attached Appendix.pdf file. See also AFRL-RQ-WP-TR-2016-0151V2.						
14. ABSTRACT Task 1 developed a methodology to automatically generate flight spectra that is statistically similar to twelve example flights provided as Government Furnished Information. Task 2 developed Version 1.0 of the P ² IAT process and identified the main sources of uncertainty. Task 3 developed a Bayesian updating procedure that fuses current-state health assessments produced by models. Task 4 developed fleet management criteria for both safety and economic strategies and a decision-making process. Task 5 demonstrated the end-to-end P ² IAT process using randomly generated and gap-filled flight data from the statistically derived flight spectrum.						
15. SUBJECT TERMS Airframe Digital Twin (ADT), digital thread (DT), Prognostic and Probabilistic Individual Aircraft Tracking (P ² IAT)						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OF PAGES 99	19a. NAME OF RESPONSIBLE PERSON (Monitor) Scott R. Wacker 19b. TELEPHONE NUMBER (Include Area Code) N/A	
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified				

Table of Contents

<u>Section</u>	<u>Page</u>
List of Figures	ii
List of Tables	v
1.0 SUMMARY	1
2.0 INTRODUCTION.....	3
3.0 METHODS, ASSUMPTIONS, AND PROCEDURES	5
3.1 Probabilistic Aircraft Usage and Loads	5
3.1.1 Process F-15C (12) Flights into Maneuvers of P-V Pairs.....	8
3.1.2 Develop Usage Statistics.....	11
3.1.3 Generate the MES	14
3.1.4 Simulate Maneuvers Using Pilot Stick Data (StS)	16
3.1.5 Develop the Baseline Spectrum and CP Stress Spectra.....	19
3.1.6 ModelCenter Spectrum Development Process	21
3.2 Probabilistic and Prognostic Individual Aircraft Tracking.....	22
3.2.1 P ² IAT Flowchart	23
3.2.2 Data Recovery.....	25
3.2.3 Current State Assessment	26
3.2.4 Future State Prediction.....	27
3.2.5 Determine Future Inspection Times.....	28
3.2.6 Project Outcomes for the Next 1,000 Flight Hours	29
3.2.7 Compile Report.....	30
3.3 Statistical Updating Process for Analysis.....	32
3.4 Decision Making With Uncertainty.....	35
3.4.1 Current Process	35
3.4.2 Probabilistic Aspect	43
3.4.3 Practical Considerations.....	46
3.4.4 When Cost Functions Are Not Available	47
3.4.5 Decision Process Flowchart.....	47
4.0 RESULTS AND DISCUSSION	50
4.1 Overview of Northrop Grumman Corporation's (NGC) P ² IAT Implementation	50
4.1.1 P2IAT File Structure.....	50
4.1.2 Overview of P2IAT Models.....	51
4.1.3 Detailed Description of P2IAT_SingleCP_Version_1.pxc Process Model	51
4.1.4 Detailed Description of P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc Process Model	62
4.2 Single Location Demonstration	70
4.2.1 Local Stress Analysis.....	71
4.3 Demonstration Results.....	78
4.3.1 Results with Negative Inspection Findings.....	81
5.0 CONCLUSIONS	87
6.0 REFERENCES.....	89
LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS.....	90

List of Figures

Figure	Page
1. P ² IAT Conceptual Approach	1
2. Probabilistic Aircraft Usage and Loads Process	6
3. Local Stress History Development Process	7
4. F-15C Exceedance Curve	9
5. Flight Data Peak-Valley (P-V) Processing Filter Study	10
6. Peak-Valley Processing Filter	11
7. Flight Data Bin Levels Consistent with StS Maneuver Suite	12
8. Maneuver Identification Algorithm	13
9. Operational Flow Diagram	15
10. Selecting Fatigue Conditions	16
11. Maximum and Minimum Loads Summary	17
12. Maximum Bending Moment vs. Symmetric Maneuvers	18
13. CP Internal Stresses for Each Fatigue Condition	19
14. CP Stress History Files	20
15. Crack Growth Life Predictions at CPs Using Baseline Spectrum	21
16. ModelCenter Flow for Spectrum Development Process	22
17. Random Variables Incorporated In P ² IAT Version 1.0	22
18. Example FDR Data	24
19. Top-Level Quarterly Update Process	24
20. Incorporate Recovered FDR Data	26
21. Perform Current State Assessment	27
22. Predict Future State	28
23. Determine Future Inspection Times	29
24. Project Outcomes to the Next Prediction Horizon	30
25. Compile Report	30
26. Output Report Header	31
27. Reported Crack Size Distributions (Quantiles)	31
28. Reported Time to Next Maintenance Action	31
29. Example SFPOF with Confidence Bounds	32
30. Adaptation for No Crack Detected Case	33
31. Adaptation for Crack Detected Case	34
32. Inspection Time for Rogue Flaw	38
33. Key Crack Sizes for Determining Inspections (4 from Reference 2)	38
34. Bi-modal Initial Flaw Distribution	40
35. SFPOF-based Inspections	41
36. P ² IAT Predicts When to Perform Damage Remediation Actions	42
37. Example Cost Function for Mitigating Small Cracks	43
38. Finding the Expected Minimum Cost Point	44
39. Finding the Ideal Time Given Its Probability	45
40. Expected Cost Reduced by Reducing Uncertainty	45
41. P ² IAT Overall Process for Each Control Point	47
42. Pre-Flight Decisions	48

43. Post-Flight Decisions	49
44. User Input Parameters for ModelCenter Model P2IAT_SingleCP_Version1.pxc	52
45. Collapsed Overview of ModelCenter Model P2IAT_SingleCP_Version1.pxc	53
46. Expanded View of Submodel FileAdminForUpdate	54
47. Component FDRDataToMES	54
48. Partially Expanded View of Submodel CreateCPStressHistoryForUpdate	55
49. Partially Expanded View of Submodel CurrentStateAssessment	56
50. Expanded View of Submodel Adaptation	57
51. Expanded View of Submodel Future_State_Assessment	58
52. Expanded View of Submodel PredictFutureInspections	59
53. Partially Expanded Overview of Submodel Project_Safety_For_Next_1000Hours	60
54. Expanded View of Submodel GenCPReport	61
55. View of FileCleanUp and CompletionAlert Components at end of P2IAT_SingleCP_Version1.pxc	62
56. CompletionAlert Dialog Presented to User When Update Has Completed	62
57. User Input Parameters for ModelCenter Model P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc	63
58. Collapsed Overview of ModelCenter Model P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc	64
59. Expanded View of Submodel FileAdminForRecovery	64
60. Expanded View of Submodel Process_Recovered_Flts	65
61. Partially Expanded View of Submodel CreateCPStressHistoryForUpdate for Recovered Data	66
62. Partially Expanded View of Submodel CurrentStateAssessment for Recovered Data	67
63. Expanded View of Submodel Adaptation for Recovered Data	68
64. View of FileCleanUp and CompletionAlert Components at end of P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc	69
65. CompletionAlert Dialog Presented to User When Data Recovery Processing Has Completed	69
66. Selected CP NG4 for P ² IAT Demonstration	70
67. Adding a Control Point to P ² IAT	71
68. CP NG4 Local Stress Analysis	72
69. NASGRO Template File	73
70. Initial Crack Size Distribution	74
71. Statistical Parameters for the Paris Regime of AA7075-T651 Plate	75
72. Fracture Toughness Statistical Parameters for AA 7075-T651 Plate	75
73. SMART DT Input File	76
74. SMART DT Crack Size Quantiles Predictions - Master Curve vs. Kriging	77
75. SMART DT Crack Size RUL Predictions - Master Curve vs. Kriging	77
76. SMART DT SFPOF Predictions - Master Curve vs. Kriging	78
77. P ² IAT Process Swim Lane Diagram for Scenario 1	79
78. Gap-Filled and No-Gap Comparative Results	79
79. Bi-modal Initial Flaw Distribution	80
80. SFPOF-based Inspections	81
81. Typical NDI POD Curve	82
82. Bimodal Crack Distribution at 600 FHs Prior to Inspection	82

83. Probability Distribution after Negative Inspection Findings at 600 FHs	82
84. Sensor Type 1	83
85. Sensor Type 2	84
86. Sensor Type 3	84
87. Inspection Outcomes at Flight 5,000 with Sensor Type 1	85
88. Inspection Outcomes at Flight 5,000 with Sensor Type 2	86
89. Inspection Outcomes at Flight 5,000 with Sensor Type 3	86

List of Tables

Table	Page
1. FDR Data Processed into StS Maneuver Probabilities	14
2. Demonstration Parameters	73

1.0 SUMMARY

Northrop Grumman Aerospace Systems (NGAS) has developed a Prognostic and Probabilistic Individual Aircraft Tracking (P²IAT) Version 1.0 process along with supporting technologies under the Task Order 0001 in Spiral 1 of the Airframe Digital Twin (ADT). The conceptual approach with key elements is shown in Figure 1.

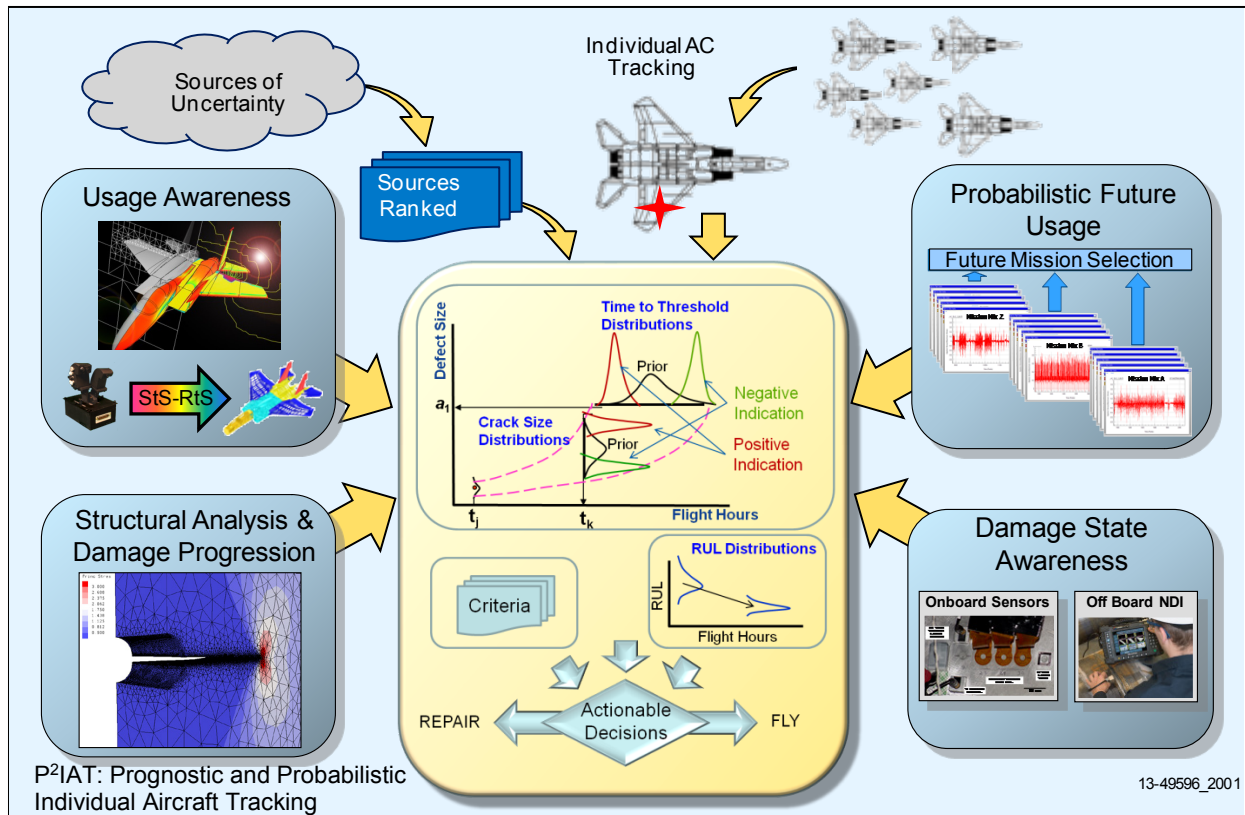


Figure 1. P²IAT Conceptual Approach

Task Order 0001 had five main tasks summarized below:

Task 1 developed a methodology to automatically generate flight spectra that is statistically similar to twelve example flights provided as government-furnished information (GFI), and the corresponding stress sequence at a selected control point (CP) in the F-15 outer wing panel. This methodology utilized Air Force Research Laboratory's (AFRL) Stick-to-Stress Real-Time Simulator version 1 (StS-RtS v1) software to the maximum extent possible¹. A baseline flight spectrum and corresponding stress sequence were created and used to demonstrate the methodology and serve as the basis for a full-scale test. Additional details are provided in Section 3.1.

¹ Certain aspects of StS were found not to operate as expected. However, in keeping with ADT Program requirements and the central premise that the uncertainties in loading can dominate uncertainties in both current and future health assessments, a method was devised to utilize the distributed loads produced by StS-RtS v1 in our process.

Task 2 developed Version 1.0 of the P²IAT process described in Section 3.2. Task 2 also identified the main sources of uncertainty, selected the most appropriate random variables to propagate forward, created a process to update damage assessments and flaw geometry (crack length and depth) within the models using evidence from state awareness while accounting for the uncertainty in both models and inspection evidence, and predicted distributions of defect size at key points in time, as well as flight hours to reach user-specified defect sizes and safety thresholds.

Task 3 developed a Bayesian updating procedure that fuses current-state health assessments produced by models that generate joint distributions of crack size (length and depth) with data from sensors for nondestructive inspection (NDI) and structural health monitoring (SHM) while accounting for the uncertainties in models and sensor data.

Task 4 developed fleet management criteria for both safety and economic strategies and a decision-making process based on comprehensive assessment of quantified risks and costs rather than deterministic treatment of uncertainties in data.

Task 5 demonstrated the end-to-end P²IAT process using randomly generated and gap-filled flight data from the statistically derived flight spectrum as input to StS-RtS v1 that then produced distributed loading that was then converted to stresses at a selected control point using the GFI finite element model (FEM)². Stress histories were then processed by crack-growth code to produce crack size distributions for the beginning and end of the update period and at the time of the next predicted inspection. The Single Flight Probability of Failure (SFPOF) prediction for the next 1,000 flights was also plotted. The results demonstrated a considerable reduction in the number of inspections needed to account for rogue flaws while maintaining a SFPOF below a user-specified threshold as well as a reduction in inspections used for crack remediation in non-safety critical control points.

To ensure openness, our P²IAT process was implemented using a commercially available tool called ModelCenter produced by Phoenix Integration. ModelCenter provided a framework to build a process flow comprised of software modules that execute in the desired sequence to propagate inputs and outputs with uncertainties from one step to the next in the flow from aircraft usage data to the required report.

² The GFI FEM required modifications to be usable for the intended purposes.

2.0 INTRODUCTION

This report documents the objectives, approach, methodologies and results of the five main tasks performed under the ADT Spiral 01 Task Order 0001 contract FA8650-14-D-2413.

The most succinct statement of objectives of the program is reiterated as a quote from BAA-13-03-RQKP: *“The Airframe Digital Twin Spiral 1 Program goal is to develop and demonstrate a probabilistic, risk-based, flight-by-flight Individual Aircraft Tracking (IAT) framework to replace the baseline deterministic IAT framework currently used for legacy aircraft. The program seeks to identify key uncertainty drivers and promising areas for future research and assess the feasibility of the approach for application to legacy fleets.”*

When fully adopted by the Aircraft Structural Integrity Program (ASIP), the community of ADT practitioners will have the capability to be more proactive in their maintenance and fleet management decisions. This is achieved as stated in the BAA *“...by increasing the accuracy of the structural prognoses developed within the Individual Aircraft Tracking (IAT) program for each Mission Design Series (MDS). More accurate prognoses can only be developed with higher fidelity flight loads data, more realistic structural analysis models than currently used and a concerted effort to quantify and reduce modeling uncertainty over time.”* However, a methodology that merely quantifies uncertainty without accuracy does not provide meaningful data for decision making. In keeping with the philosophy stated in the BAA, prediction accuracy and uncertainty reduction were central to the NGAS approach to ADT.

In prior work under the Defense Advanced Research Projects Agency (DARPA) Prognosis Program, NGAS developed a thorough understanding and quantification of the uncertainties associated with structural life predictions due to various unknowns in material microstructure. It has been shown that material microstructure is a major contributor to uncertainties in life prediction and this aspect of the problem is well-understood by NGAS. As suggested by the BAA, the key to advancing the state of the art was to extend this understanding to accurately characterize flight loads and their effect on life predictions. The key innovation in this area was purported to be the StS-RtS v1 provided by AFRL as government-furnished software. Although StS-RtS v1 was developed and verified for the F-15C, there were problems using it for ADT purposes. Consequently, NGAS devoted considerably more time in Task 1 than originally planned working with AFRL co-developer Ned Lindsley on this critical element of ADT. By working closely with Lindsley, we were ultimately able to use the distributed aero loads produced by StS-RtS v1 for critical maneuvers and more accurately determine stresses at control points rather than use conventional methods that only approximate the distribution of loads and their combined effect on stresses. Additionally, there were problems with the FEM provided as GFI.

The NGAS approach began with an analysis of data from twelve test F-15 flights provide as recordings of key kinematic parameters sampled at 80 Hz. A maneuver ID algorithm was developed to classify these data into mil-spec maneuvers according to key Mach, Nz, altitude, roll rate and roll acceleration temporal feature associated with 110 mil-spec maneuvers provided by StS-RtS v1. The algorithm produced maneuver statistics that were used to synthesize new flights for gap filling and future usage prediction (see Section 3.1 for details). Per AFRL guidance in May 2016, an abbreviated version of this maneuver ID method was created to process lower-fidelity flight data recorder (FDR) data as would be available from event recorders (see Section 3.2 for details). This version became the first step in the P²IAT process. The next

step in our approach related the identified maneuvers to local stresses by processing the distributed aero loads through the government-furnished FEM. Unanticipated effort was also expended to fix the FEM as provided because of inadequacies in the model fidelity. The resulting stress histories were then formatted for NASA Fatigue Crack Growth software (NASGRO) and Fatigue crack Structural Analysis software (FASTRAN) and processed by the probabilistic methods in the Small Aircraft Risk Technology | Damage Tolerance (SMART | DT) engine created by our teammates at the University of Texas at San Antonio (UTSA). Using parametric and/or numerical characterizations of the salient random variables (see Section 3.2), SMART | DT provides five main functions that calculate: 1) SFPOF with confidence bounds as a function of time, 2) time distribution Remaining Useful Life (RUL) to reach a user-specified safety threshold, 3) RUL distribution to reach a user-specified crack size, 4) crack-size distributions expressed as quantiles for user-specified points in time, and 5) joint length and depth probability distributions of crack size (crack geometry).

Probabilistic methods were also created to incorporate crack-state information that might be provided by onboard SHM sensors, nondestructive inspection (NDI) or nondestructive evaluation (NDE), including characterizations of the uncertainties associated with these measurements. These methods utilize a numerical joint distribution (crack length and depth) provided by fatigue/crack growth models as a priori probabilities and generate an a posteriori numerical joint distribution using Bayesian methods (see Section 3.3 for more details).

Decision-making criteria and methods were developed for safety and economic strategies and a decision-making process based on comprehensive assessments of quantified risks and costs. Predictions for the expected time to reach a user-specified SFPOF threshold (nominally 10^{-7}) are adjusted by an optional lead time and ASIP-specified maintenance boundaries (nominally 200 FHs) to produce a recommended time for inspections to maintain safety. For non-safety-of-flight locations, predictions can be made for a user-specified crack size appropriate for remediation. The distribution in time to reach the specified crack size can then be manipulated to choose the most appropriate time while considering costs and uncertainties (see Section 3.4 for more details).

On September 19, 2016, the entire end-to-end P²IAT process was demonstrated for a single DADT CP and described in more detail in Section 4.1. This computational exercise verified that the P²IAT approach is viable, and that it advances the prognostic capability of Individual Aircraft Tracking (IAT) as compared to the current U.S. Air Force (USAF) approach. The outputs of this process were formatted in accordance with specifications provided by AFRL in May 2016.

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

Section 3.0 of this Final Report will be focused on Tasks 1 through 4 of the ADT Statement of Work (SOW). Section 3.1 focuses on SOW Task 1 where the team developed a methodology to automatically generate flight spectra that is statistically similar to twelve example flights. Section 3.2 focuses on SOW Task 2 where Version 1.0 of the P²IAT process was developed. Section 3.3 covers SOW Task 3 in which a Bayesian updating procedure that fuses current-state health assessments produced by models with data from sensors for NDI and SHM is developed, and Section 3.4 discusses SOW Task 4 in which is developed a fleet management criteria for both safety and economic strategies and a decision-making process based on comprehensive assessment of quantified risks and costs rather than deterministic treatment of uncertainties in data.

3.1 Probabilistic Aircraft Usage and Loads

Probabilistic calculation of damage evolution will critically depend upon the stochastic loading applied to components. Representative flight histories have been developed that contain a variety of flights and maneuvers flown in order to capture their variability. A process has been developed to convert actual Flight Data Recorder (FDR) data to flight spectra at various fatigue critical locations (FCL) for use in P²IAT as well as for the full-scale demonstration experiments. Methods have been developed that capture uncertainty and variability associated with flights, maneuvers within flights, and missing flight data. Our approach utilized the F-15 recorded data for the 12 flights provided by AFRL in conjunction with the StS maneuver simulator to define the test loading time history and consequently the local stress histories at FCLs to be tracked.

The overall process in capturing usage uncertainty is depicted in Figure 2. A maneuver identification algorithm processes actual flight data consistent with the StS suite of maneuvers and provides usage statistics associated with symmetric, asymmetric, and roll conditions. A database of loads was developed for each StS maneuver based on stick profiles that have been provided by AFRL. Internal stresses at FCLs are obtained by applying the aerodynamic surface loads associated with the StS maneuvers to the F-15C outer wing FEM. The maneuver probabilities and internal stresses per maneuver at specific FCLs are combined to generate multiple missions with a random occurrence of maneuvers within each mission. The next step assembles the test flight spectra to be used for the full-scale experiment and allows AFRL to remove missions to simulate missing data. A gap filling procedure has been developed to gap fill the missing data statistically accounting for variability in missions and generate the control point stress histories to be used as input to P²IAT.

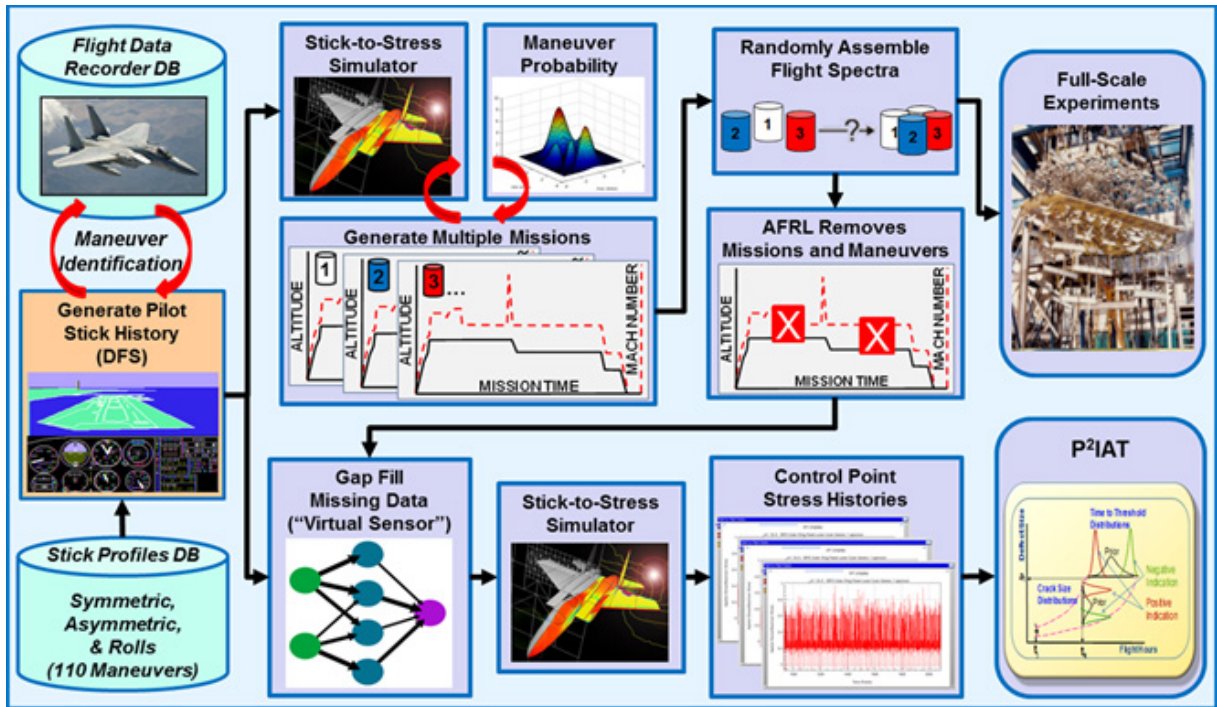


Figure 2. Probabilistic Aircraft Usage and Loads Process

Figure 3 depicts the spectrum development process for developing the outer wing panel test spectrum and the CP local stress histories. F-15 Flight data are processed into tables that relate occurrences/probabilities of P-V bins consistent with StS maneuvers based on N_z , altitude, Mach number, roll rate, and roll acceleration.

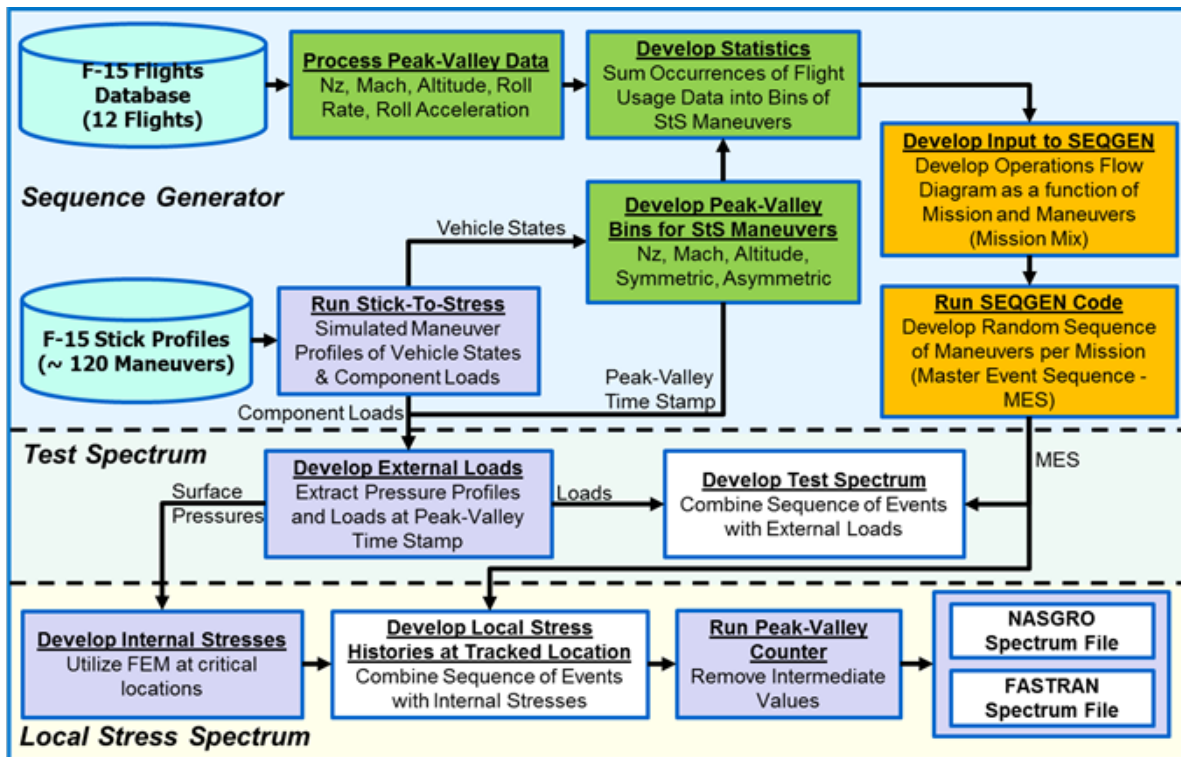


Figure 3. Local Stress History Development Process

AFRL-provided F-15 stick profiles are processed through the StS software to simulate the maneuvers and obtain the vehicle states and component loads. Simulated vehicle states are obtained for the same flight parameters (Nz, Mach, altitude, roll rate, and roll acceleration) as processed from the flight usage data. Bin levels are then defined to sum occurrences (develop statistics) of flight usage data into bins of StS maneuvers.

The maneuver statistics are used to develop the required input for the NGAS-developed SEquence GENerator (SEQGEN) code. SEQGEN generates a random sequence of events called a Master Event Schedule (MES) that is used to perform fatigue and crack growth analysis. The MES is produced by cycling through an operational flow diagram that represents the various scenarios that an aircraft may experience in terms of mission types and load sequences during takeoffs, maneuvers, and landings under various aircraft configurations. SEQGEN randomly selects a load sequence (P-V pair) from the maneuver statistics table and inserts the sequence into the MES.

As depicted in Figure 3, the next step converts the MES spectrum events to the test load spectrum and to the local stress histories. This was accomplished by extracting pressure profiles and loads at peak-valley time stamps. The bending moment, torsion moment, and shear loads (MVTs) corresponding to the StS maneuvers were combined with the MES to develop the test spectrum. The aerodynamic surface pressures associated with the StS maneuvers are used in conjunction with the F-15C FEM to obtain the internal stresses at selected FCLs on the outer wing. The internal stresses were combined with the MES to develop the local stress histories at the tracked locations.

By constructing the test spectrum from the F-15C StS suite of maneuvers, we were able to associate the loads to be applied on the test article as well as the local stresses, with the corresponding pilot stick inputs that generated those maneuvers. Our approach also related the test spectrum to flight parameters (Nz, Mach, altitude, roll rate, and roll acceleration) and aircraft configuration associated with the StS suite of maneuvers.

The following development effort was accomplished to develop loading time histories that are related to the type of data recorded during actual flight and be consistent with the StS suite of maneuvers:

- Process F-15C (12) flights into maneuvers of P-V pairs
 - Develop P-V identification processing threshold (filter FDR data)
 - Match 12 flights to Nz exceedance curve provided in DTA report
- Develop Usage Statistics
 - Develop occurrences utilizing flight parameters: Nz, altitude, Mach number, roll rate, and roll acceleration
 - Sum occurrences of flight data into P-V bins consistent with StS maneuvers
 - Develop bins of Nz occurrences/probabilities vs. Mach number and altitude for symmetric, asymmetric maneuvers, and rolls
 - Develop maneuver identification algorithm
- Generate the MES
 - Develop operations flow diagram
 - Run SEQGEN to generate flights with random load sequences
- Simulate maneuvers using pilot stick data (StS)
 - Obtain Vehicle states and component loads
 - Obtain component loads and CP stresses at P-V time-stamps
- Develop the baseline spectrum and CP stress spectra
 - Combine the MES with the internal stress at CPs

3.1.1 Process F-15C (12) Flights into Maneuvers of P-V Pairs

The process of maneuver P-V pair identification is intricate due to the continuous recording of data at 30 times per second. An optimal data processing threshold was established by matching Nz exceedances per 1,000 hours for the 12 flights to one shown in the DTA report

(F15C_DTA_Report_Vol_1.PDF). The curve utilized operational data through the fourth quarter of 1995 including 30,499 flight hours of signal data recorder (SDR) data and is shown in Figure 4.

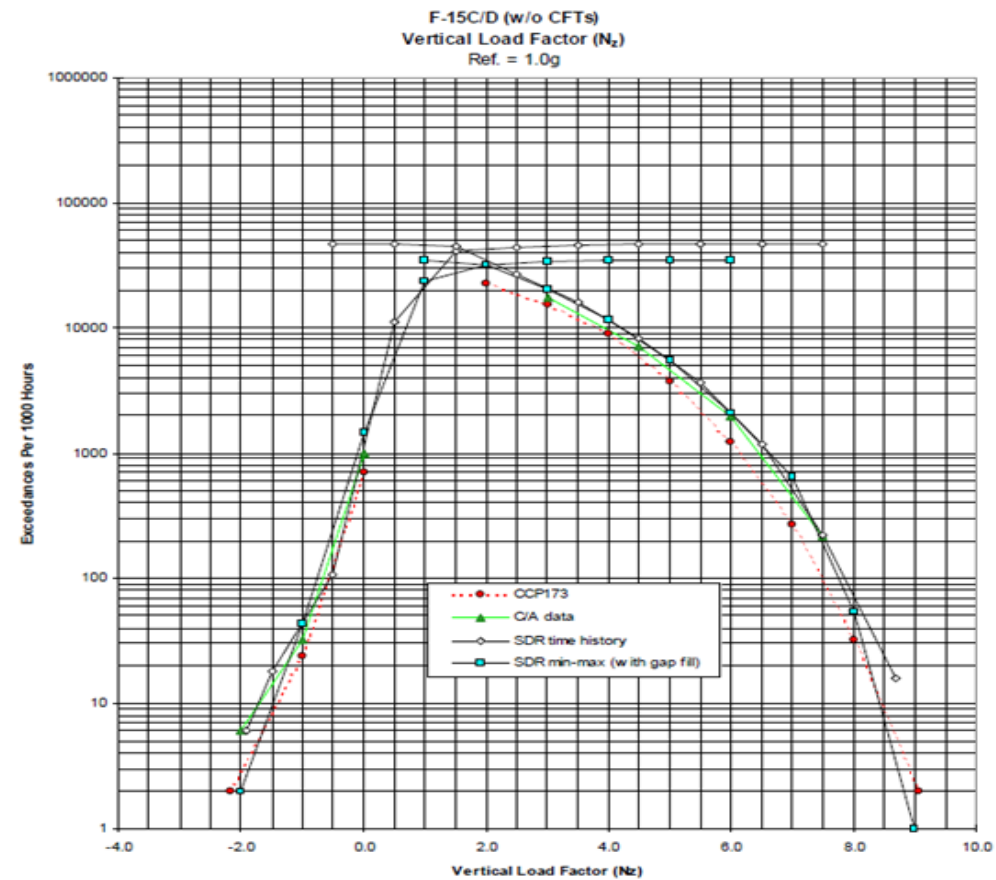


Figure 4. F-15C Exceedance Curve

A MATLAB algorithm was developed to identify N_z peak-valley data for abrupt pull-ups (APU), rolling pull-outs (RPO), and 360° rolls for all $N_z > 1$ and N_z P-V data for abrupt push-overs (APO) and 180° rolls for all $N_z < 1$. Figure 5 shows the results of the filtering process using various thresholds for only a portion of one flight.

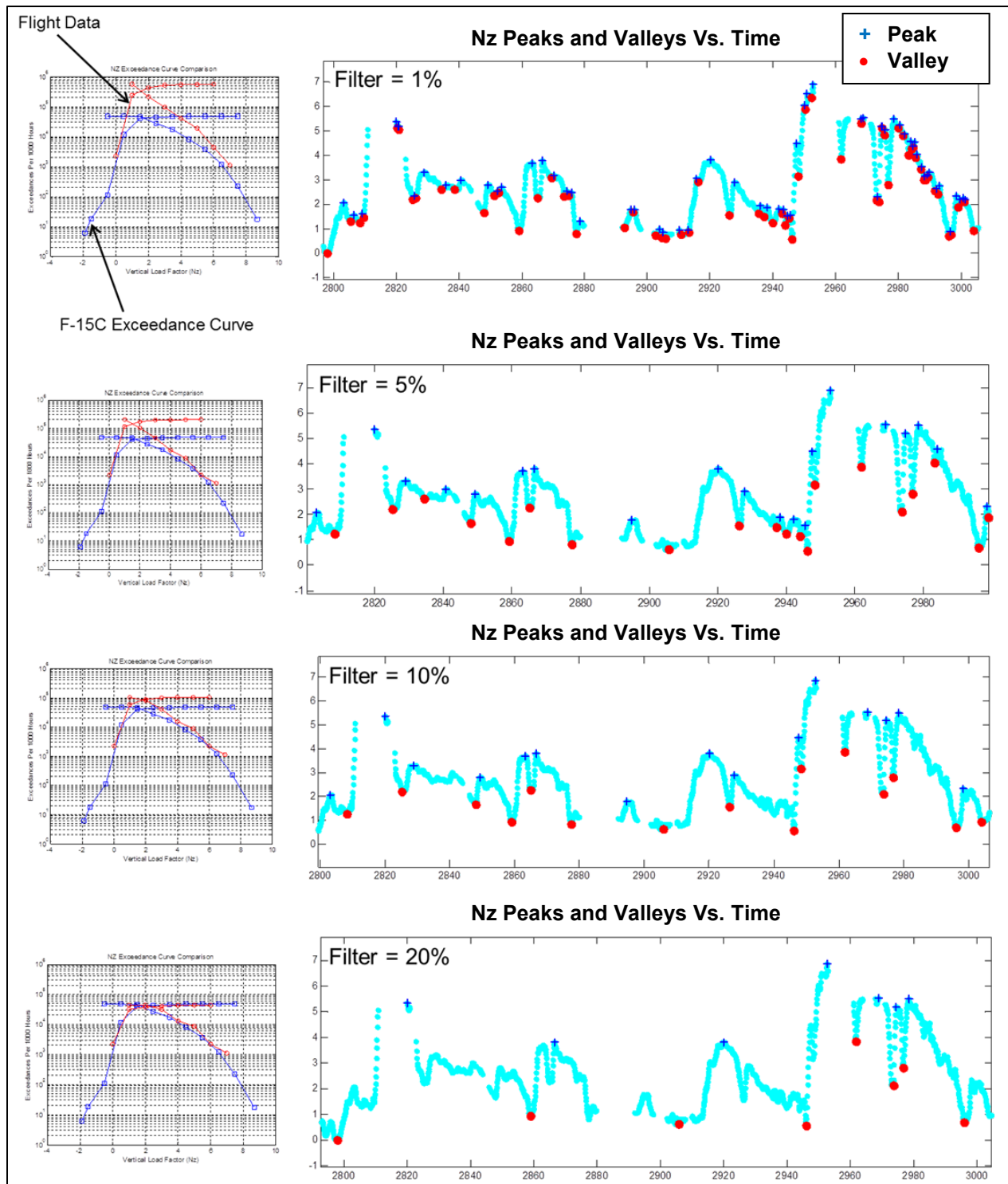


Figure 5. Flight Data Peak-Valley (P-V) Processing Filter Study

Exceedance curves were generated using all 12 flights and compared to the F-15C exceedance curve. As shown in Figure 6, a filtering parameter of 20% correlated very well to the F-15C curve and was selected to process the flight data and obtain statistics of maneuvers.

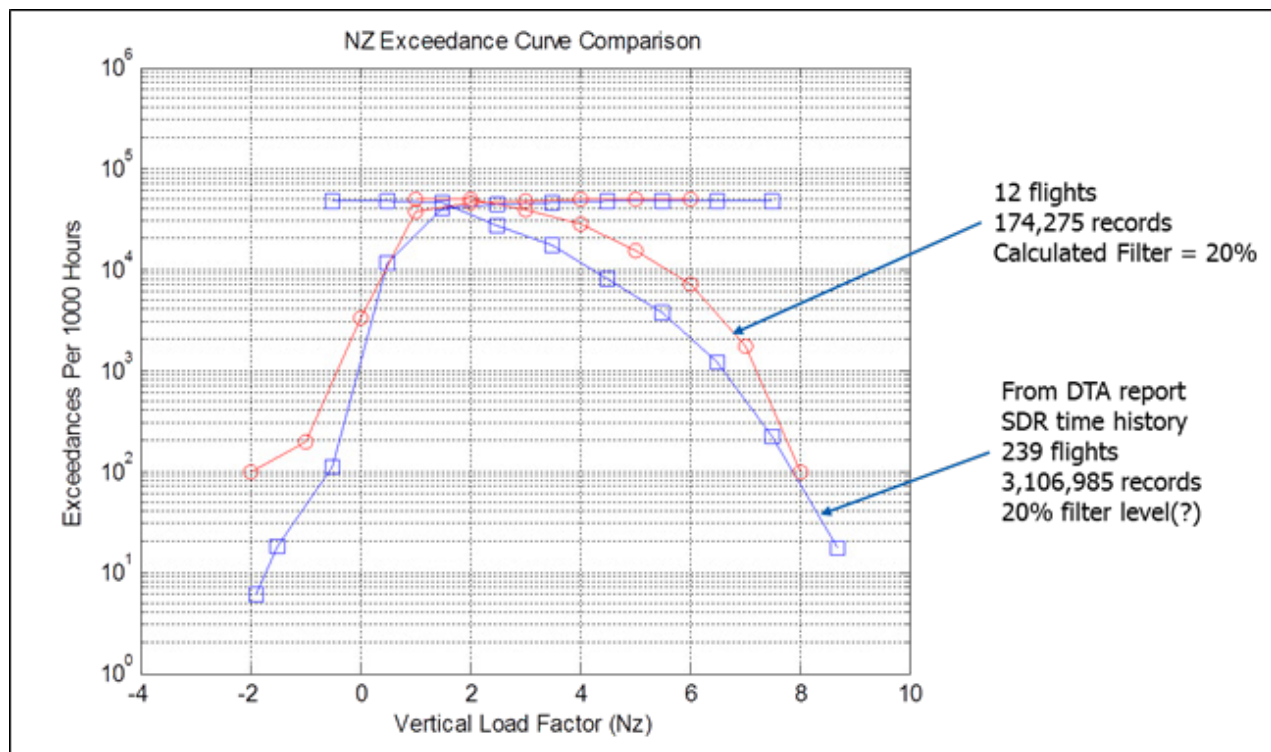


Figure 6. Peak-Valley Processing Filter

3.1.2 Develop Usage Statistics

Utilizing the filtering parameter, the F-15 flight data (FDR data) are processed into tables of occurrences/probabilities of StS maneuvers.

The first step was to identify the relevant flight parameters in identifying StS maneuvers. The flight parameters are: Nz, altitude, Mach number, roll rate, and roll acceleration. Nz is utilized to identify the symmetric maneuvers and roll rate and acceleration are used to identify the asymmetric maneuvers.

The second step was to develop bin levels to sum occurrences that are consistent with StS maneuvers. Figure 7 shows the flight data bin levels. As shown bin levels are determined for Altitude and Mach numbers and are consistent with the StS point-in-the-sky (PITS) flight conditions. Bin levels are also identified for the Nz values associated with the StS suite of maneuvers.

The third step was to develop a maneuver identification algorithm to process flight data into StS maneuver occurrences/probabilities. Two tables are generated, one for symmetric and one for asymmetric maneuvers. The symmetric maneuvers are simply identified by the Nz peak and placed in the appropriate bin level corresponding to the PITS.

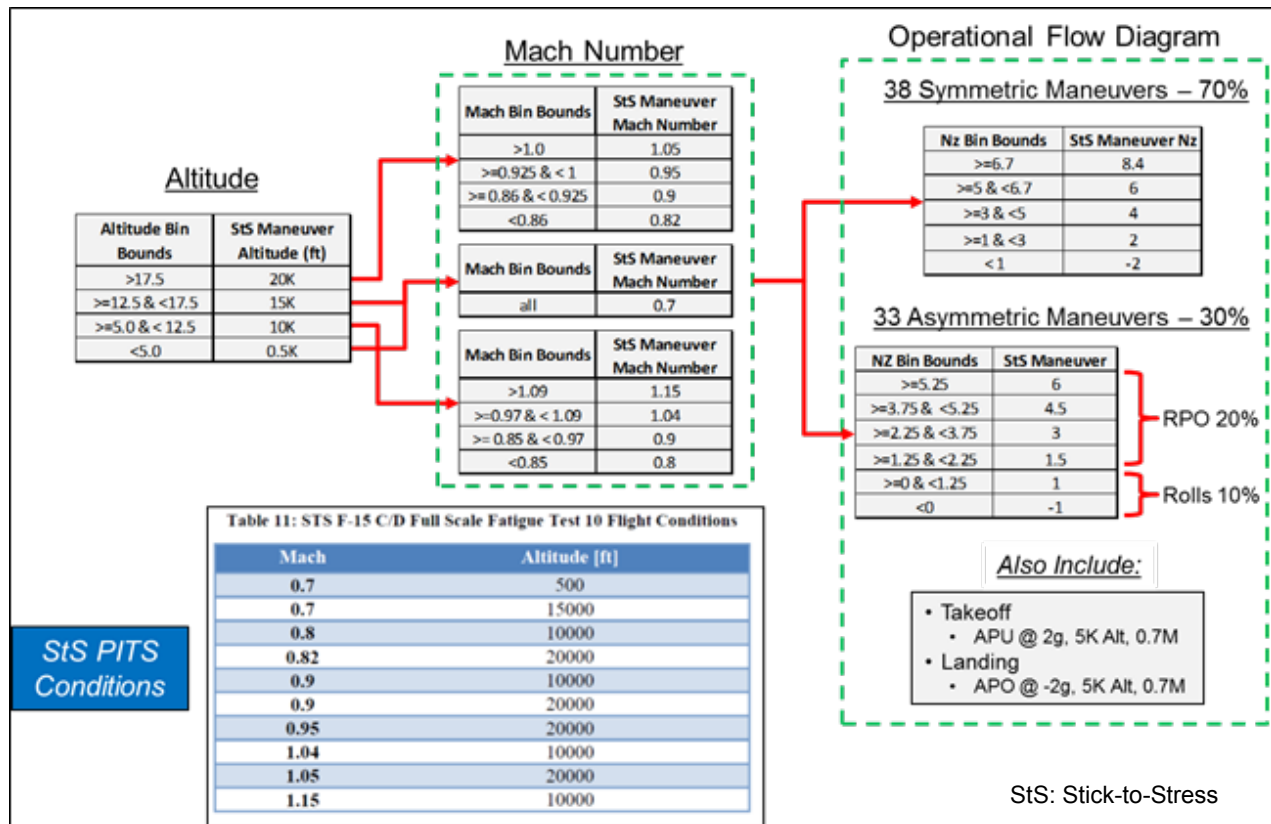


Figure 7. Flight Data Bin Levels Consistent with StS Maneuver Suite

The procedure for identifying asymmetric maneuvers was conceptualized by observing the StS maneuver state profile resulting in the following rules (reference):

1. Find the peak roll rate (P_{peak}) when $P \geq \pm 25^\circ/\text{s}$.
2. Find $P = 0$ by going back in time no more than five seconds. $P = 0$ defines the initial state.
3. Check that $\dot{P}$ is $\geq \pm 1.5 \text{ rad/s}^2$ at the interval between $P=0$ and P_{peak} .
4. If 1 and 3 are satisfied then the asymmetric maneuver is identified.
5. Define initial maneuver state by obtaining Nz, altitude, and Mach number at the $P=0$ time stamp.

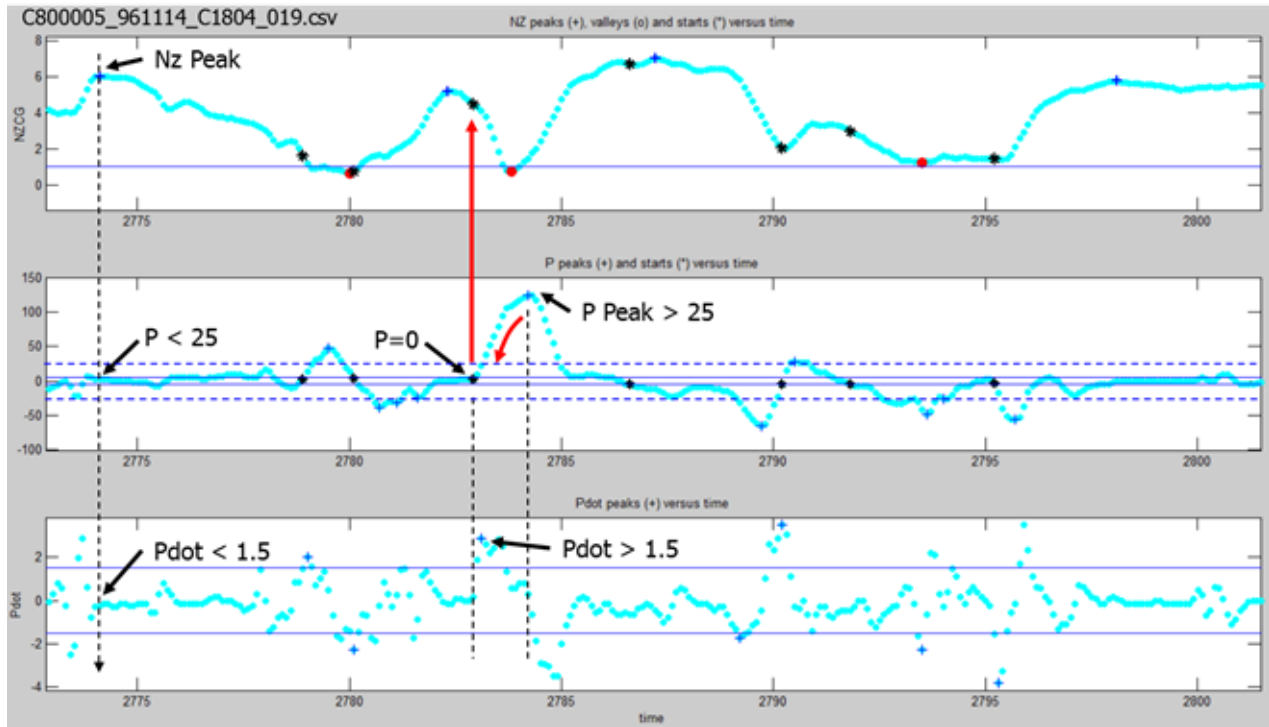


Figure 8. Maneuver Identification Algorithm

The resulting probabilities of StS maneuvers based on 12 F-15C flights are shown in Table 1. 40 symmetric and 30 asymmetric maneuvers were identified. Two inconsistencies were uncovered in the StS vehicle state data: 1) All 15 Kft altitude maneuvers (see Figure 12) are really 1.5 Kft conditions. The bin levels were updated to reflect 1.5 K altitude maneuvers and 2) All RPOs initiate at an Nz level of 6 g and loads at each PITS are all equal to the 6 g maneuver. A StS trim parameter has been uncovered that set the initial Nz to match the maneuver Nz. The probability tables as well as the asymmetric loads were updated to reflect the corrections.

Table 1. FDR Data Processed into StS Maneuver Probabilities

FDR = Flight Data Recorder

Symmetric Maneuvers (40)

	Altitude Bounds -->	<1000	>=1000 & <5750	>=5750 & <15000				>=15000			
	Mach Bounds -->	all	all	<0.85	>=0.85 & <0.97	>=0.97 & <1.09	>1.09	<0.86	>=0.86 & <0.925	>=0.925 & <1	>1
	StS Altitude -->	500	1500	10000				20000			
	StS Mach -->	0.7	0.7	0.8	0.9	1.04	1.15	0.82	0.9	0.95	1.05
NZ bounds	StS Nz	Probabilities									
>=6.7	8.4 (APU)		0.0018	0.0101	0.0018			0.0083	0.0037		0.0009
>=5 & <6.7	6 (APU)		0.0037	0.0359	0.0203			0.0295	0.0129	0.0092	0.0028
>=3 & <5	4 (APU)	0.0028	0.0157	0.0461	0.0138	0.0009		0.0590	0.0341	0.0286	0.0046
>=1 & <3	2 (APU)	0.0009	0.0065	0.0387	0.0037	0.0018		0.0323	0.0018	0.0092	0.0037
<1	-2 (APO)	0.0018	0.0147	0.0700	0.0212	0.0018		0.0829	0.0240	0.0267	0.0138

Asymmetric Maneuvers (30)

	Altitude Bounds -->	<1000	>=1000 & <5750	>=5750 & <15000				>=15000			
	Mach Bounds -->	all	all	<0.85	>=0.85 & <0.97	>=0.97 & <1.09	>1.09	<0.86	>=0.86 & <0.925	>=0.925 & <1	>1
	StS Altitude -->	500	1500	10000				20000			
	StS Mach -->	0.7	0.7	0.8	0.9	1.04	1.15	0.82	0.9	0.95	1.05
NZ bounds	StS Nz	Probabilities									
>=5.25	6 (RPO)				0.0037			0.0028		0.0018	
>=3.75 & <5.25	4.5 (RPO)			0.0055	0.0055			0.0157	0.0083	0.0018	
>=2.25 & <3.75	3 (RPO)		0.0009	0.0175	0.0028			0.0101	0.0037	0.0046	0.0037
>=1.25 & <2.25	1.5 (RPO)		0.0046	0.0175	0.0046			0.0175	0.0138	0.0157	0.0083
>=0 & <1.25	1 (360° roll)		0.0138	0.0286	0.0147	0.0028		0.0230	0.0166	0.0221	0.0055
<0	-1 (180° roll)										

StS RPO maneuvers with trim option

3.1.3 Generate the MES

The ModelCenter Spectrum Development process reads the FDR data and generates a combined probability table which is utilized to develop the MES. A NGAS-developed SEQUENCE GENERATOR (SEQGEN) code was utilized to generate a random sequence of events called a Master Event Schedule (MES). The MES is produced by cycling through an operational flow diagram shown in Figure 9 that includes the probability tables. SEQGEN randomly selects a StS maneuver which defines a load sequence (a P-V pair) and inserts the sequence into the MES. Takeoff and decent maneuvers are included for the beginning and end of each flight as shown in Figure 9. Additionally, a 1g level condition is included at the end of each asymmetric maneuver. This improved the compression/compression R (min/max) ratio of the control point stress histories.

Utilizing the operational flow diagram the input file to SEQGEN was developed and the MES was generated. The resulting baseline spectrum that can be used for both the left and right side outer panel wings includes:

Based on actual usage data: one hour/flight, 102 maneuvers/flight, includes takeoff and decent

Baseline Spectrum: 1,000 FHs = 1,000 flights = 131,770 Mx-Mn cycles = 263,540 EPs

Simulated usage: 8,000 FHs = 2,108,320 EPs

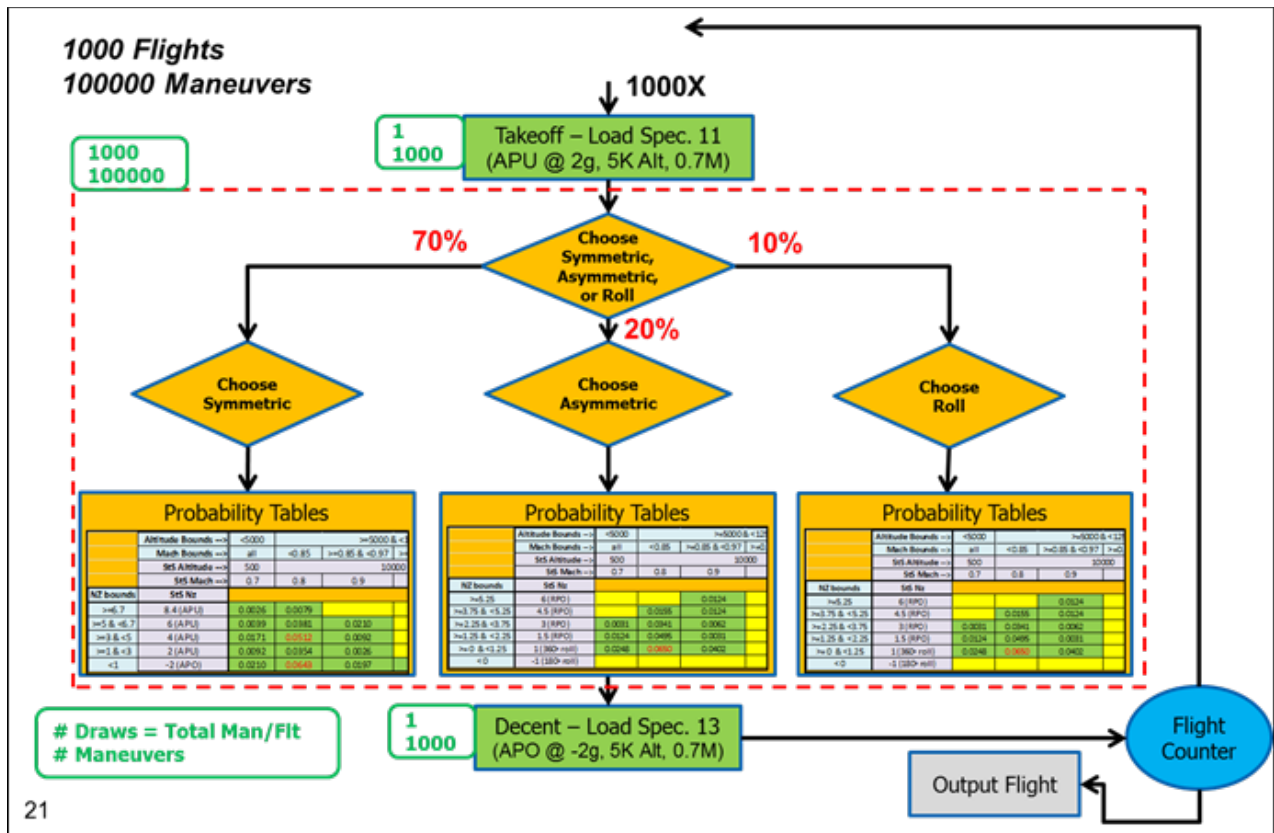


Figure 9. Operational Flow Diagram

3.1.4 Simulate Maneuvers Using Pilot Stick Data (StS)

The next step is to convert the MES spectrum events to the test load spectrum and to the local stress histories. The StS software with ROMS ON option was utilized to obtain the surface pressures for each maneuver. A procedure was developed using the NASTRAN monitor points function to sum the surface pressures at the 40% chord at LRS₂@24.3 (Figure 10) and obtain the load time histories for each maneuver. Load time histories were obtained for bending moment, torsion moment, and vertical shear. The time histories were not as expected (due to a StS software issue) thereby necessitating a manual approach in developing the fatigue conditions. Maneuvers were processed individually to define the fatigue conditions (max-min time stamps) corresponding to the maximum and minimum loads of the time history. Figure 10 shows the selection of the max-min time stamps for a single maneuver, 70 maneuvers were processed resulting in 140 fatigue conditions. The curves shown in blue and labeled as StS History are with ROMS OFF.

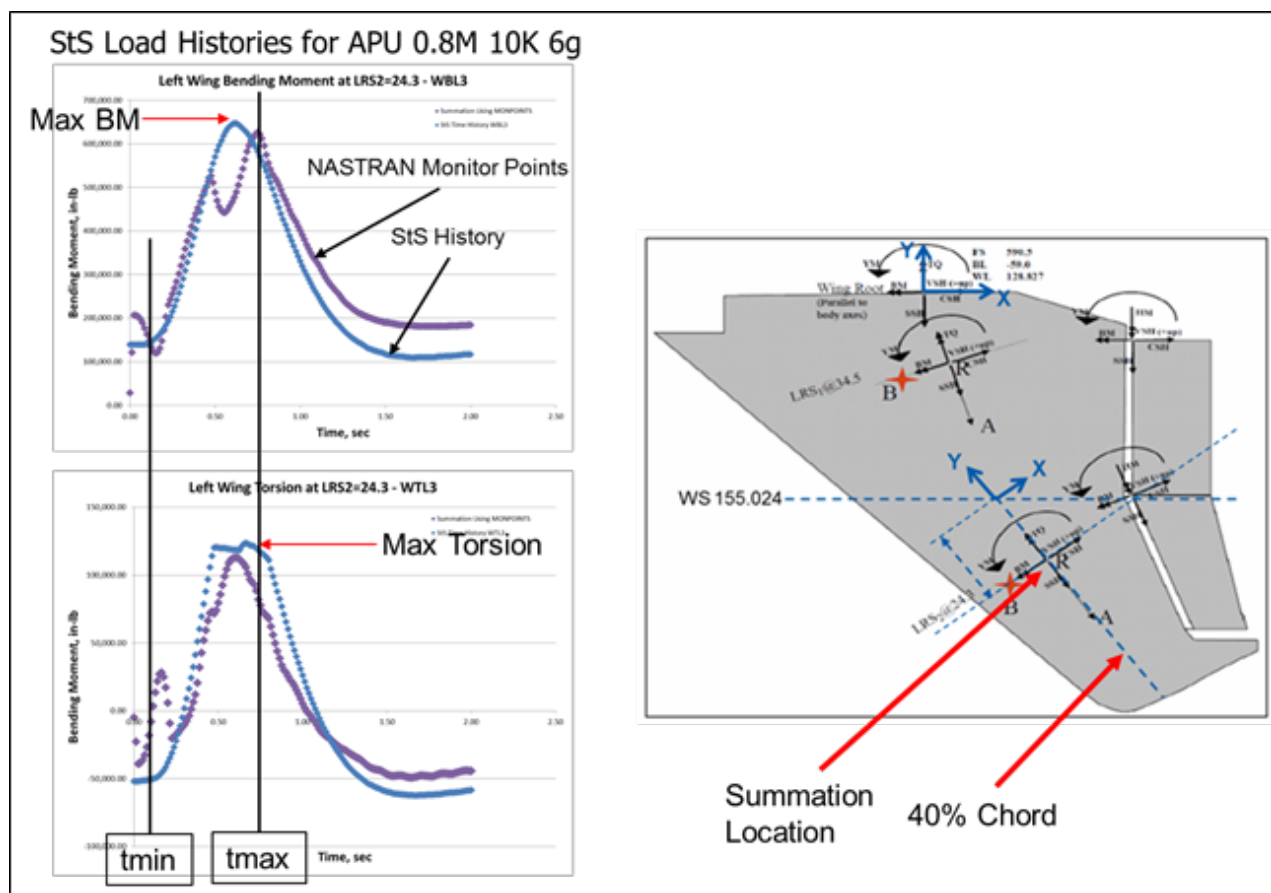


Figure 10. Selecting Fatigue Conditions

A summary of the maximum and minimum bending moments, torsion moments, and vertical shear loads for all 70 maneuvers is shown in Figure 11. Figure 12 shows a summary of the bending moments for the symmetric maneuvers.

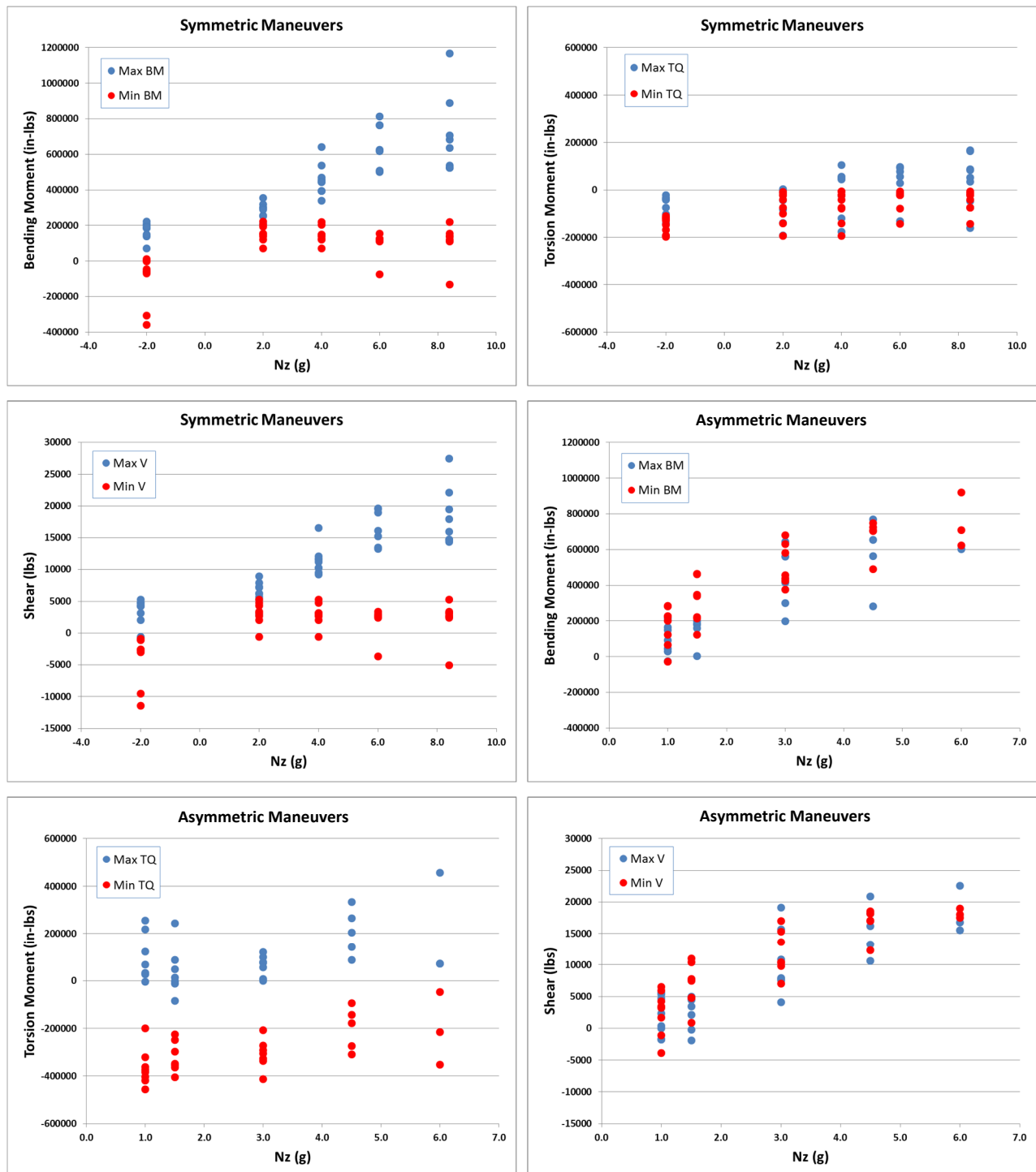


Figure 11. Maximum and Minimum Loads Summary

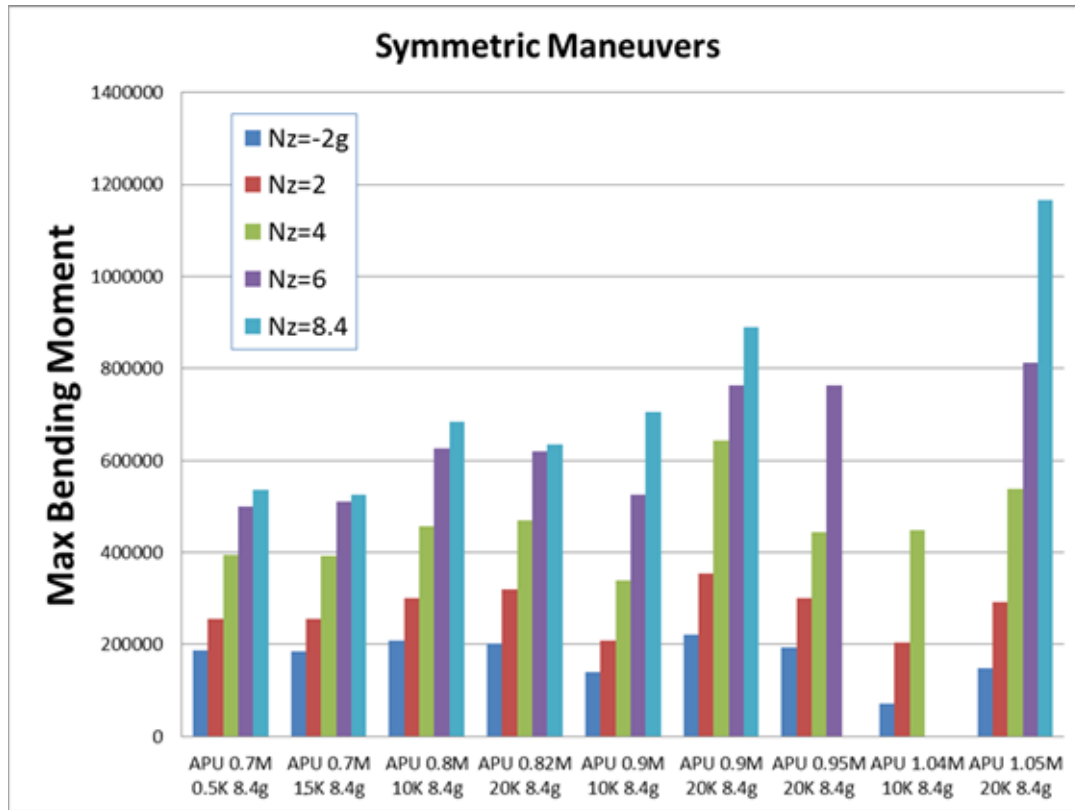


Figure 12. Maximum Bending Moment vs. Symmetric Maneuvers

The max-min time stamps for 70 maneuvers were documented in an Excel file. The ModelCenter Spectrum development process utilizes the excel file to automatically develop a single NASTRAN input file to analyze the outer wing for 140 fatigue condition and obtain the internal stresses at each CP, shown in Figure 13 for CP NG1. The internal stresses per maneuver are reformatted into the internal stress file to be consistent with the MES and combined with the MES to develop the local stress histories at all the CPs. The internal stress file in Figure 13 shows 144 fatigue conditions, the additional four conditions are for the takeoff and decent maneuvers.

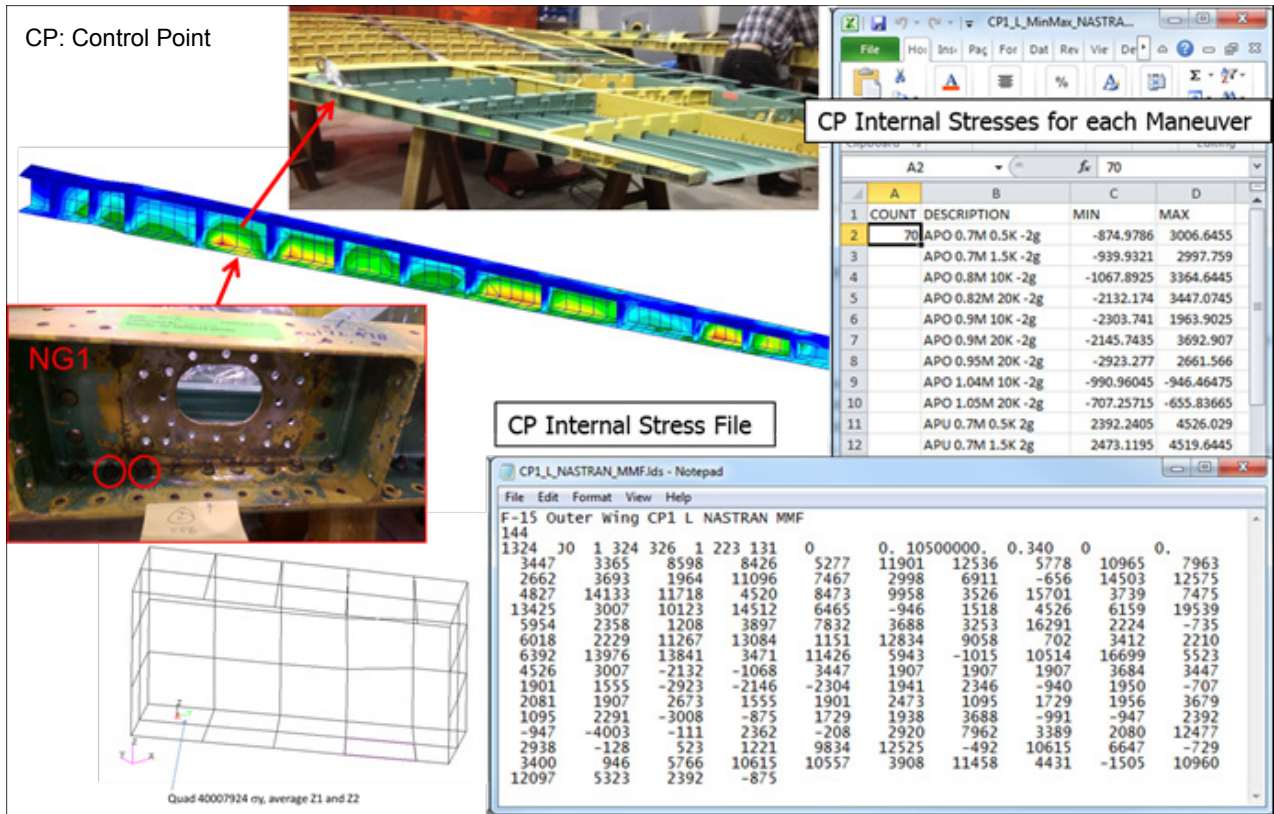


Figure 13. CP Internal Stresses for Each Fatigue Condition

3.1.5 Develop the Baseline Spectrum and CP Stress Spectra

The internal stresses at each CP and the MES are combined to develop the CP stress histories using a NGAS-developed code FATLIFE2. FATLIFE2 reads the output of SEQGEN and the stress file shown in Figure 13 and outputs a flight-by-flight file as shown in Figure 14. The flight-by-flight-file is then converted to FASTRAN and NASGRO formatted files. FATLIFE2 also eliminates the intermediate values resulting in peak-valley values. The resulting CP stress histories can be directly used as input for fatigue crack growth life predictions.

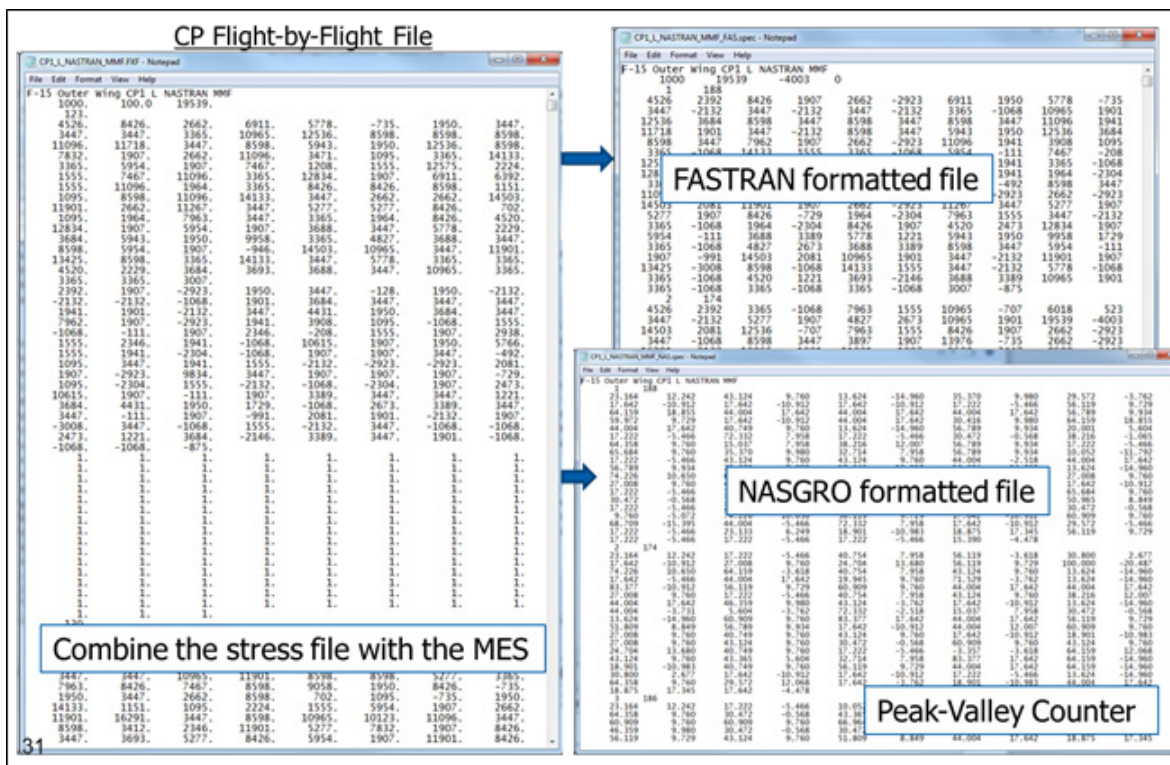


Figure 14. CP Stress History Files

The development of the baseline spectrum required an iterative process utilizing the NASGRO developed crack growth models at CPs and refining the fatigue conditions to obtain a desirable amount of crack growth within the simulated usage of 8,000 FHs. The NASGRO crack growth life predictions utilizing the baseline spectrum for the selected 10 CPs are shown in Figure 15.

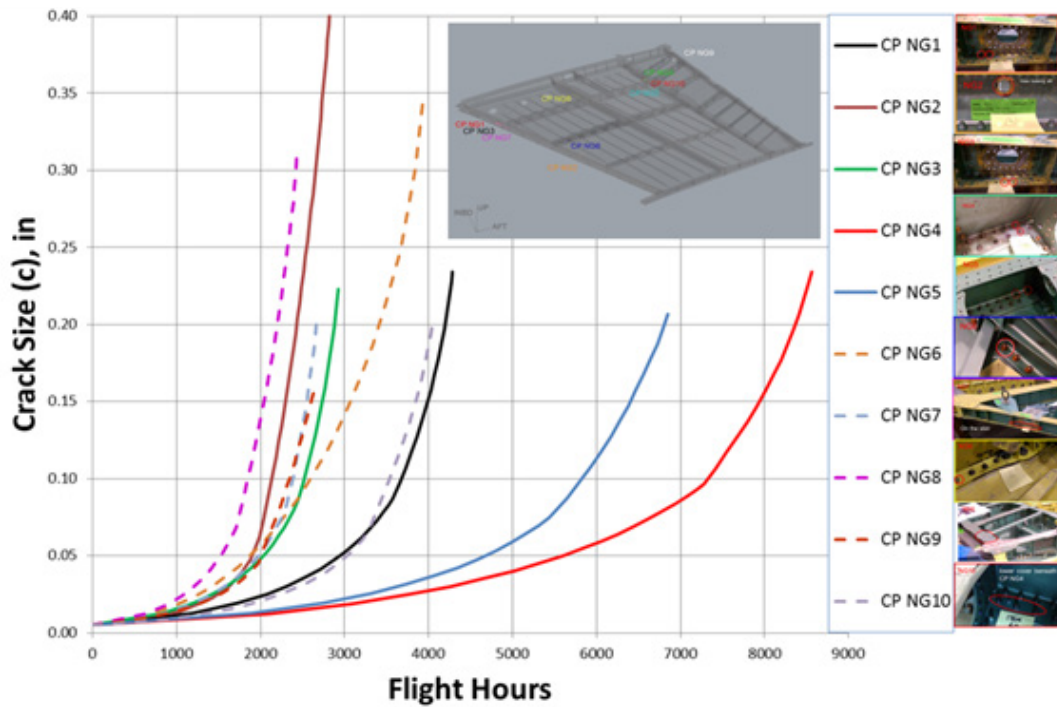


Figure 15. Crack Growth Life Predictions at CPs Using Baseline Spectrum

3.1.6 ModelCenter Spectrum Development Process

A ModelCenter flow (depicted in Figure 16) has been developed to automate the entire process in developing the baseline spectrum and the stress histories at the selected 10 CPs. The process reads the actual FDR data and generates the probability tables based on the maneuver identification algorithm consistent with the StS suite of maneuvers. The MES is then generated which includes a number of flights with a random sequence of maneuvers per flight. The user specifies the number of flights and number of maneuvers within the flights. An automated procedure then executes the StS software to generate the surface pressures for each maneuver and the component MVT (Bending-Shear-Torsion) time histories. Utilizing the time histories, the user defines the fatigue conditions by manually selecting max-min time stamps. The process automatically generates the NASTRAN input file for all fatigue conditions and at all CPs, executes NASTRAN, and produces the internal stresses at all CPs. The CP stress histories are automatically generated by combining the MES with the internal stresses at the CPs. The process generates FASTRAN and NASGRO formatted stress histories files and also generates the Extreme Value Distribution (EVD) to perform SFPOF calculations.

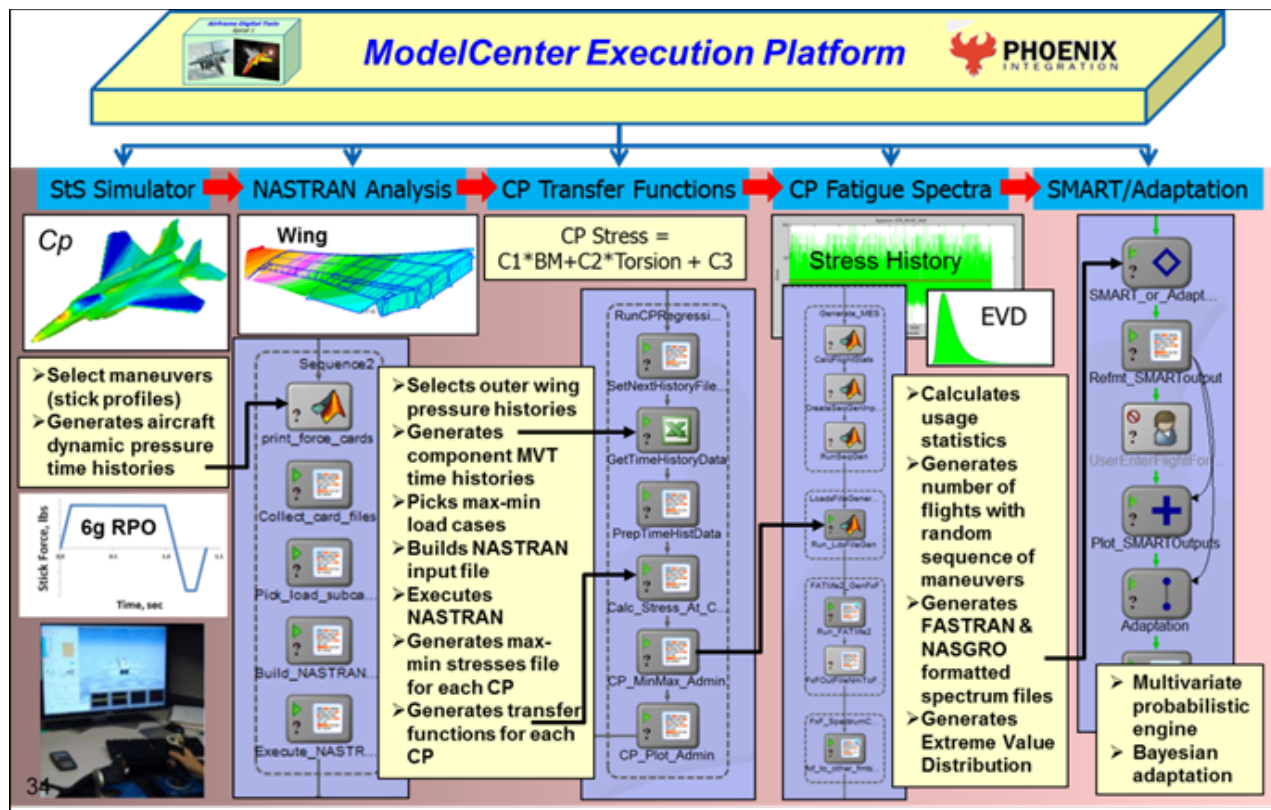


Figure 16. ModelCenter Flow for Spectrum Development Process

3.2 Probabilistic and Prognostic Individual Aircraft Tracking

An effective P²IAT process must consider the uncertainty and variability in flight records, structural configuration, material properties and many other random variables, since their contributions are important in maintaining the veracity of probabilistic health forecasts. The salient variables whose uncertainty was considered for the initial P²IAT release (Version 1.0) are shown in Figure 17.

Variables	Default Values in Template File	Units	Description	Distribution Type	Param 1	Param 2	Param 3
ai	0.005	in	Initial Crack size - Shifted from SIPS	Weibull	scale=0.00019843	shape = 2.3	
aovc	1.00	-	Aspect ratio - a/c	uniform	min=0.75	max=1.25	
Paris_C	1.3620E-08	7075-T651 Plate	Crack-growth coefficient - C	binormal	mean of log10c = -7.865783	std = 0.03705	rho=-0.8646
Paris_n	3.405	7075-T651 Plate	Crack-growth power term - n	binormal	mean = 3.405	std = 0.07494	
Fract_Tough	50.0	ksi-in ^{0.5}	Plain strain fracture toughness - K1c	normal	mu = 28.7294	sigma = 1.8034	
Sy	75.4	Ksi	Yield stress	constant	min=71.63	max=79.17	
Sult	83.4	Ksi	Ultimate tensile strength	constant	min=79.23	max=87.57	
E	1.040E+07	Ksi	Elastic Modulus	constant			
S0	18.40	Ksi	Peak bypass stress for EVD	uniform	min=17.48	max=19.32	
S3	19.10	Ksi	Peak bearing stress	constant			
Hole_Diam	0.199	in	Diameter	constant			
Width	1.120	in	Width	constant			
Thickness	0.190	in	Thickness	constant			
Hole_offset	0.430	in	hole offset	constant			

Figure 17. Random Variables Incorporated In P²IAT Version 1.0

Parameters set to constant were not exercised as random variables during the demonstration in Task 5, but are listed here for completeness since these were considered under the Task 2 analysis.

Several factors must be considered when determining which variables are the “most important” contributors to the overall uncertainty. The first step selects a quantity of interest against which the uncertainty effect can be measured. For this program, SFPOF was chosen as the most fundamental quantity of interest. All other parameters including initial/current crack state, usage, material parameters etc. are ultimately reflected in the probability of failure (POF) and, since safety is the primary consideration for individual aircraft tracking, SFPOF was chosen as the quantity of interest. It is also important to understand that the identification of parameters having critical influence over uncertainty is a dynamic function of circumstances. Parameters that are important in one stage of crack growth may not be important in other stages. Finally, characterizations made regarding the uncertainty of each random variable will control that variable’s influence over the quantity of interest. For example, if the mean or standard deviation of a hole diameter is well within the design constraints, small variations will not adversely affect the POF. However, if the characterization of a hole diameter mean is large compared to design limits, even a small variance can dominate the uncertainty in the POF. Consequently, there is no universal priority of random variables since they are generally codependent and affected by their characterization and operational circumstances.

It is also important to understand that assumptions made regarding hyper-parameter distributions on random variables can dominate the outcome of learning with field data. It is incorrect to assume that arbitrary values can be asserted for hyper-parameters and field data will eventually correct those assertions. If the hyper-parameter distributions are chosen incorrectly, the outcome cannot necessarily be corrected by Bayesian methods unless the training data is directly relevant to the parameter in question. This is typically not the case since field data is usually related to input random variables by implication (e.g., a crack size found during an inspection only implies information about initial flaw size. Many other factors are inextricably related). In view of the guidance in the BAA to consider the practicality of characterizing various random variables in the choice of incorporating them in version 1.0, a conscious decision was made based on prior experience to avoid the use of the hyper-parameter approach since such distributions are neither available nor measureable within the scope of this program. Further, incorporating them in the approach using unfounded assumptions is dangerous since it can dominate calculations and not necessarily be correctable by sensor data or field findings.

3.2.1 P²IAT Flowchart

The P²IAT process starts with the receipt of FDR data. Figure 18 shows an example of FDR data as could be recorded by an event type recorder. Each maneuver in this example is comprised of five events at which snapshots of all parameters are captured.

	A	B	C	D	E	F	G	H	I	J	K
	Flt #	TIME (sec)	ALT (ft)	MACH	NNZCG (g)	NXCG (g)	NYCG (g)	Roll Rate (PRATE, deg/sec)	Pitch Rate (QRATE, deg/sec)	Roll Acc (PDOT, deg/sec^2)	Pitch Acc (QDOT, deg/sec^2)
1	1	0	500	0.70	1.00	0.03	0.00	0.00	0.00	0.00	0.00
2	1	0.4375	500	0.70	1.89	0.07	0.00	0.14	5.08	1.34	-13.83
3	1	0.55	501	0.70	2.00	0.08	0.00	0.24	3.06	0.68	-19.51
4	1	0.7125	502	0.70	1.82	0.07	0.00	0.28	0.56	-0.03	-10.93
5	1	2	525	0.70	1.04	0.03	0.00	-0.01	0.08	-0.02	0.30
6	1	35	10000	0.80	1.00	0.03	0.00	0.00	0.00	0.00	0.00
7	1	35.3875	10000	0.80	3.01	0.15	0.00	0.22	16.97	3.73	3.81
8	1	35.5125	10001	0.80	4.06	0.19	0.00	0.63	11.46	2.84	-75.87
9	1	35.6875	10005	0.80	3.59	0.18	0.00	0.87	1.87	0.07	-35.76
10	1	37	10073	0.80	1.02	0.03	0.00	-0.04	0.17	0.27	-0.39
11	1	70	20000	0.95	1.00	0.02	0.00	0.00	0.00	0.00	0.00
12	1	70.1375	20000	0.95	1.06	0.02	0.00	0.20	-2.20	3.92	-54.06
13	1	70.175	20000	0.95	0.95	0.01	0.00	0.30	-4.48	1.26	-67.17
14	1	70.525	19998	0.95	-2.02	0.03	-0.01	-1.05	-8.69	-6.26	74.64
15	1	72	19927	0.95	0.94	0.01	0.00	0.09	-0.18	0.39	-0.34
16											

Figure 18. Example FDR Data

These events are: the beginning of the maneuver, the peak of the absolute value of Nz, the peak roll acceleration, the peak roll rate, and the end of the maneuver. The first five events (shaded blue) comprise the first maneuver, followed by the second and third maneuvers shaded in green and red. There are approximately 102 maneuvers for each flight. A single excel file containing these data would normally have all the flights in a given update if none were missing. Updates typically occur every three months, so a vehicle that flies about 400 flights per year would have about 100 flights per quarterly update. It is also common that some flights in an update may be missing and provided at a later time. The P²IAT process fills in missing data with imputed data that contain random maneuvers whose probabilities match the statistics derived from example flight data.

The following sequence of figures shows the general flow of the P²IAT process. The top-level flow is shown in Figure 17.

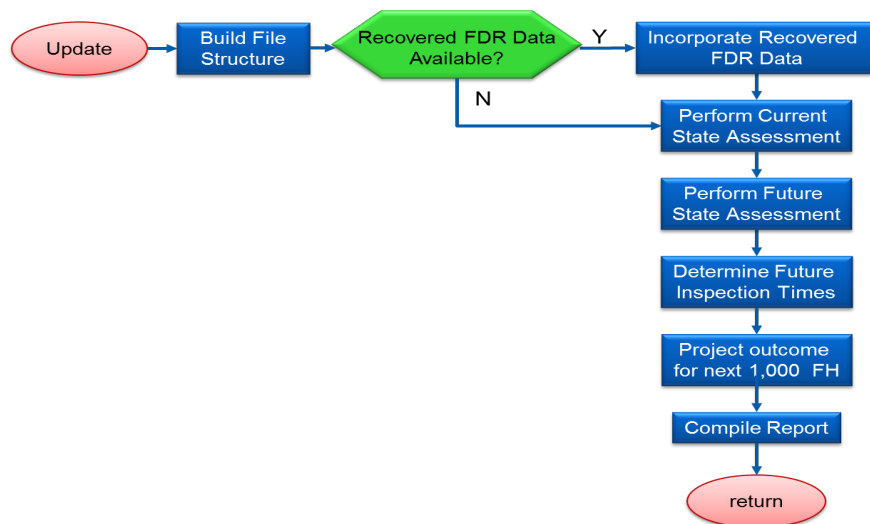


Figure 19. Top-Level Quarterly Update Process

The first step builds the file structure using the flight number in the name of the update file as an identifier. If recovered data are available, the user invokes the recovery model. Processing recovered data first established the best starting point upon which to incorporate new flights.

This avoids extra processing if not performed first. The flow for recovering data is shown in Section 3.2.1.

The next step in the top-level flow performs a current state assessment. Missing data are filled in with statistically similar flights that are randomly created using maneuver probabilities derived from past flights. FDR data are then converted to maneuvers then stress histories for fatigue models. A probabilistic assessment of the crack state at the last flight number is calculated and expressed as a joint a,c crack-size distribution. Details in this step are shown in Section 3.2.2.

The next step in the top-level flow performs an assessment of the future state. Anticipated usage is extracted from the baseline spectrum and used to grow the crack distribution from the current state out to the prediction horizon (nominally 1,000 flight hours). The time to reach various criteria is calculated for later use in decision making. Details in this step are shown in Section 3.2.3.

The next step in the top-level flow determines when to recommend inspections for both safety and remediation as explained in Section 3.4. In this step, the time to reach a user-specified crack size appropriate for the next remediation and the time to reach a user-specified safety threshold are adjusted by user-specified lead times and then fit to the standard 200 flight hour inspection opportunities. The latest 200 FH opportunity that does not exceed the adjusted inspection time is determined along with predicted crack size distributions at those points in time. Details in this step are shown in Section 3.2.4.

The next step projects the potential outcome from the present to the prediction horizon. If an inspection was recommended for safety within the horizon interval, the usage to that point is used to create an anticipated crack size distribution that is then adjusted as if an inspection with a negative finding was performed. The adjusted state is then projected out to the balance of the horizon. Details in this step are shown in Section 3.2.5.

The last step in the top-level flow compiles a report in the format specified by AFRL at the May 2016 workshop. Details in this step are shown in Section 3.2.6.

In the flowcharts that follow, blue indicates functions implemented as ModelCenter modules and purple indicated functions implemented by SMART | DT.

3.2.2 Data Recovery

Some flight usage captured by the FDR can be lost or delayed during normal operations. Missing data are imputed by the P²IAT process so that assessment can be made even when some flights are missing. If missing data eventually becomes available, the data used to fill gaps must be replaced by the actual data so that assessments and predictions are as accurate as possible. The flowchart in Figure 20 shows this process.

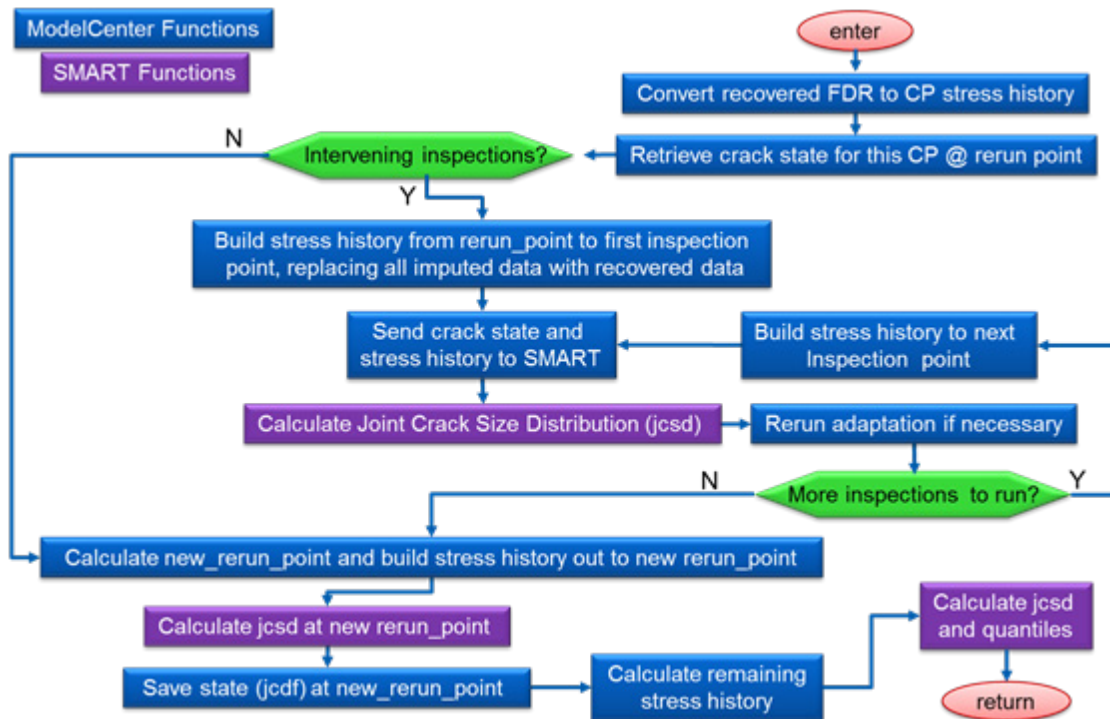


Figure 20. Incorporate Recovered FDR Data

The first step identifies maneuvers in the FDR data using key features in the Nz, Mach, altitude, roll rate and roll acceleration. The first snapshot in each maneuver identifies the altitude and Mach number. Then it declares the maneuver as an asymmetric maneuver if the maximum roll rate is greater than 50° per second. If not, the maneuver is identified as symmetric. It then uses a set of asymmetric or symmetric Nz bin boundaries and labels consistent with those used by StS-RtS v1 to find the corresponding StS-RtS v1 maneuver IDs. Mach, then altitude then Nz are matched with StS-RtS v1 maneuvers to complete the maneuver classification. In a preprocessing step, the stress sequence at each control point was determined for each maneuver using distributed loads from the StS-RtS v1 as processed through the FEM. The stress sequence is then used by fatigue/crack growth models to probabilistically determine the new crack size distribution resulting from the stress sequences.

In order to faithfully regenerate current and future state assessments that account for recovered data, all inspections that may have been performed at points in time occurring after recovered flights must be replayed with their findings. The process loop in the middle of this flowchart shows how this is accomplished. The final output from this flow is a joint crack size distribution at the end of the last full set of processed FDR data.

3.2.3 Current State Assessment

Current state refers to the crack size distribution for a given control point at the last declared flight number in a given update period. That update period begins with the previously declared last flight plus one (or one if this is the first update) and extends through the last declared flight

number in the update. The last flight must be declared by the user since flights in the FDR data may be missing³. The flow for Current State Assessment is shown in Figure 21.

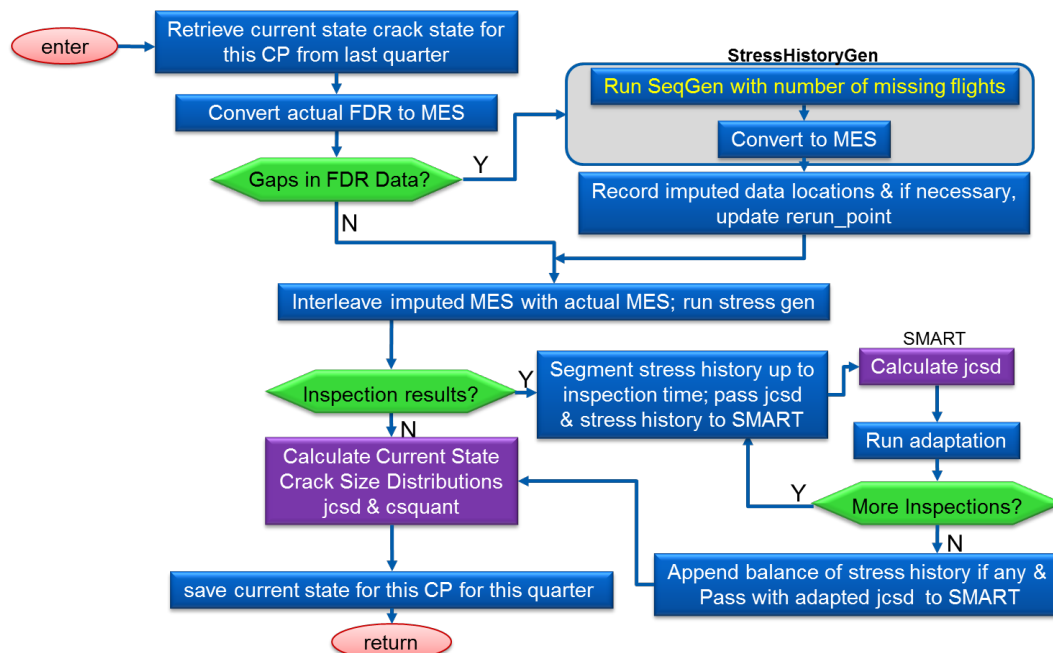


Figure 21. Perform Current State Assessment

The first step identifies maneuvers in the FDR data as previously described. If data are missing, the NCG executable named “SeqGen” is called to generate the number of flights missing using the statistics derived from previous flights⁴. These imputed flights are interleaved to fill gaps and then the composite is processed by the stress_gen code that develops stress histories for each specified control point. Stress histories are segmented between inspections so that the crack state at the time of the inspection can be matched up with the NDI results and adjusted by inspection findings using the adaptation code. P²IAT can handle any number of inspections at any times during the update period using the loop in the middle of the figure. Crack size distributions (both the joint a,c and the quantiles for c) are produced at the end of this process.

3.2.4 Future State Prediction

The future state is predicted using either the baseline spectrum or by running SeqGen to randomly generate a sequence of flights as anticipated usage. If the future mission mix is known, the statistics for that mix can be used by SeqGen to generate statistically similar flights for future predictions. The flow for this process is shown in Figure 22.

³ The algorithm recognizes that one or more flights may be missing when the last declared flight is greater than the flight number in the data record.

⁴ Previous flights from specific mission types, pilots or any combination of examples can be statistically analyzed as previously described to create the statistics for SeqGen.

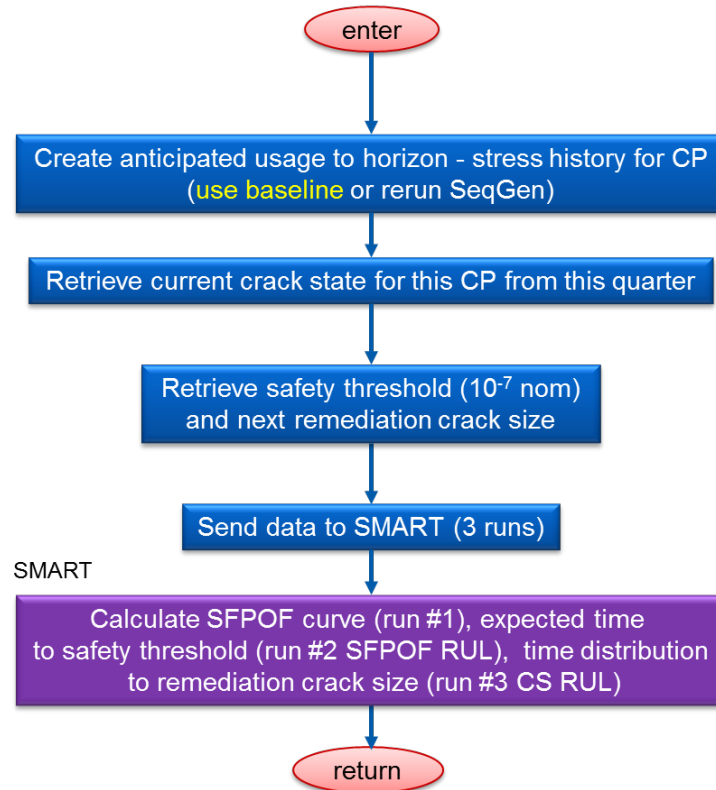


Figure 22. Predict Future State

The expected time to reach the safety threshold and the distribution in time to reach the next remediation crack size for the control point is determined by the process depicted in the figure. The anticipated usage is repeated if its length is insufficient to produce the required outputs.

3.2.5 Determine Future Inspection Times

The next process adjusts the predicted times for the safety and remediation inspections to account for optional lead times specified by the operator and to align with the normal maintenance tempo. Lead times are provided to prepare for various maintenance actions. For example if the next remediation step requires the installation of a reinforcement plate, advanced notice must be provided for the repair to ensure that materials, training and personnel are available at the time of the inspection so that the inspection and repair can be performed efficiently during the same maintenance opportunity. The normal maintenance tempo for the F-15 is field inspections every 200 flight hours with depot maintenance opportunities every 1,800 flight hours. This schedule is entered in the spreadsheet named "MaintenanceSchedule.xlsx" so that changes can be accommodated by the algorithm. The flow for this process is shown in Figure 23.

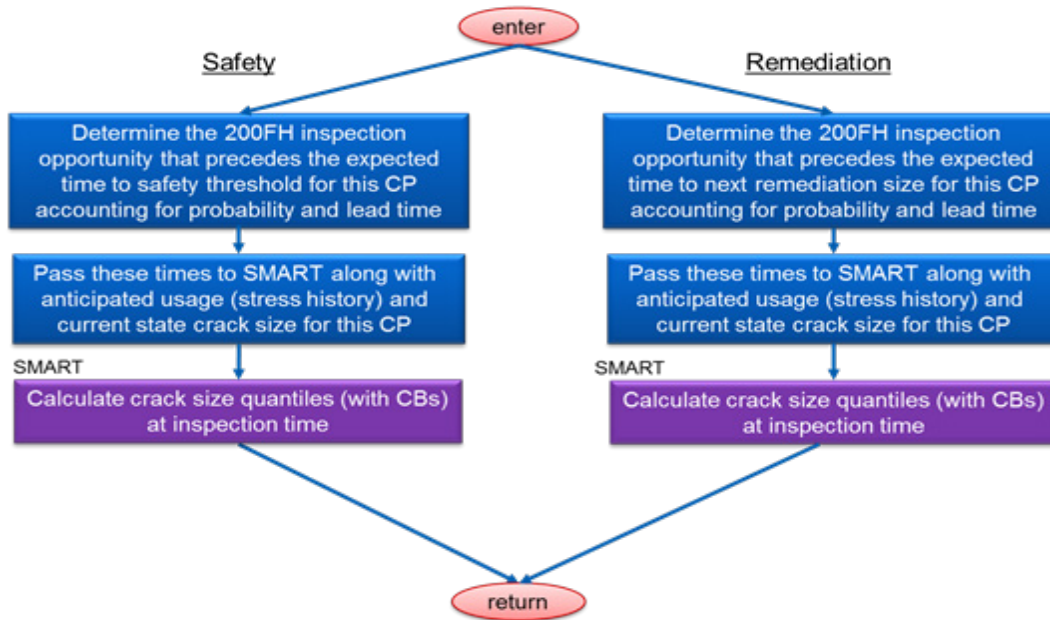


Figure 23. Determine Future Inspection Times

This process outputs crack size distributions as both joint a,c crack-size distributions and as quantiles as requested in the output report.

3.2.6 Project Outcomes for the Next 1,000 Flight Hours

This process uses anticipated usage derived as previously discussed to project the SFPOF with confidence bounds for the next horizon interval. The flow for this process is shown in Figure 24.

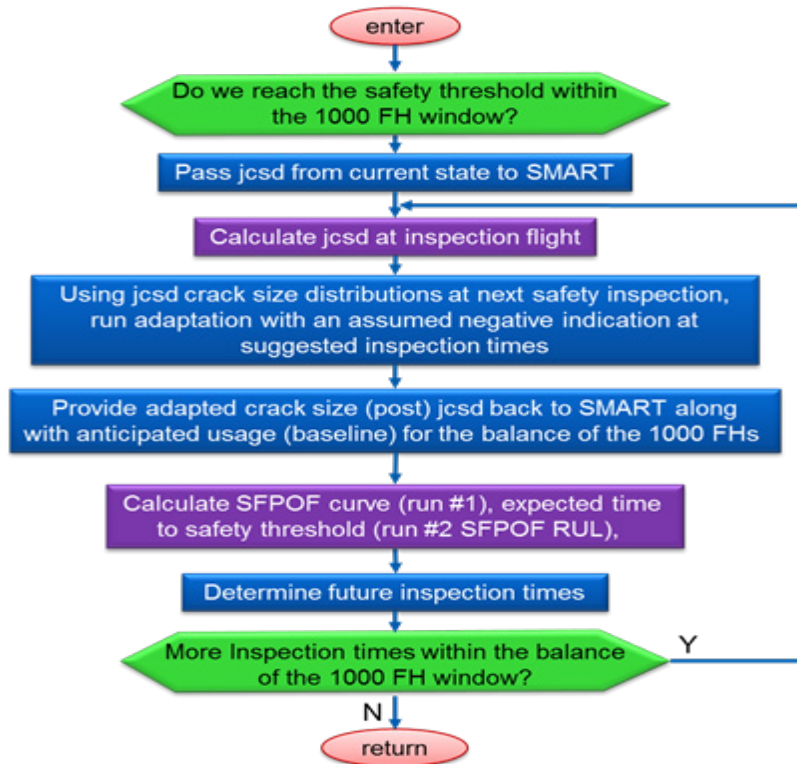


Figure 24. Project Outcomes to the Next Prediction Horizon

If an inspection was recommended for safety within the horizon interval, the anticipated usage from the current time to that point is processed and used to create an anticipated crack size distribution that is then adjusted as if an inspection with a negative finding was performed. The adjusted state is then projected out to the balance of the horizon. If more safety inspections are predicted before the horizon, the process is repeated in the loop shown. The output of this process is a plot of the SFPOF with confidence bands versus future flights.

3.2.7 Compile Report

The last step in the top-level process is to compile a report. The flow for this is shown in Figure 25.

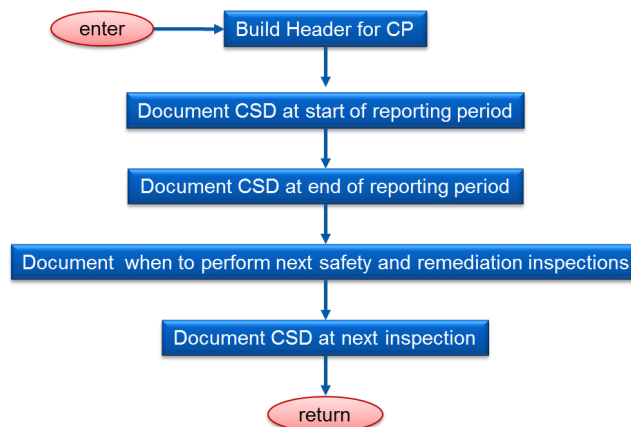


Figure 25. Compile Report

The format for the report was provided by AFRL in May 2016 and is shown in the following four figures.

Aircraft	Date of last IAT data	Cumulative Flight hours	FH increment from last report	Report Date
XXX	May 11, 2017	2,000	200	May 14, 2017

Figure 26. Output Report Header

		Start of Reporting Period				
CP No.	CP Identification	90%	75%	50%	25%	10%
NG 01	Description TBD	0.06	0.033	0.023	0.013	0.0055
	95% Confidence Band	0.05, 0.07	0.03, 0.036	0.02, 0.026	0.01, 0.016	0.005, 0.006
NG 02	Description TBD	0.06	0.033	0.023	0.013	0.0055
	95% Confidence Band	0.05, 0.07	0.03, 0.036	0.02, 0.026	0.01, 0.016	0.005, 0.006

End of Reporting Period					At Next Maintenance Action				
90%	75%	50%	25%	10%	90%	75%	50%	25%	10%
mean	mean	mean	mean	mean	mean	mean	mean	mean	mean
(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)
mean	mean	mean	mean	mean	mean	mean	mean	mean	mean
(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)	(min, max)

Figure 27. Reported Crack Size Distributions (Quantiles)

Next Maintenance Action		
Flight Hours (delta)	Date	Criterion
1800	18-Feb-21	Remediation

Number of flight from the present to the next inspection [safety or remediation, whichever comes first]

Anticipated date based on delta flight hours* FH per day

Figure 28. Reported Time to Next Maintenance Action

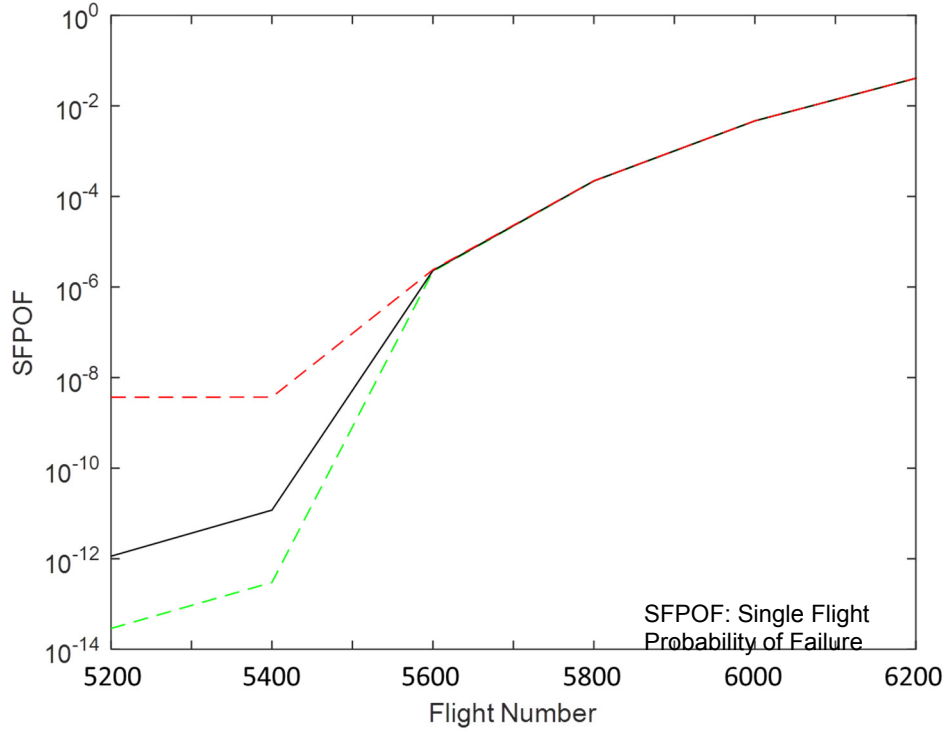


Figure 29. Example SFPOF with Confidence Bounds

The entire P²IAT process was implemented in ModelCenter and is discussed in Section 4.0.

3.3 Statistical Updating Process for Analysis

During the individual aircraft tracking process, evidence of a crack's state may become available during inspections and this information is incorporated along with its uncertainty in current state assessments and future state predictions through the updating process describe in this section. The evidence given by state awareness sensors (NDI) in its simplest form provides a go, no-go indication (no-detect or detect respectively). Such indications have uncertainty associated with the sensor's defect size threshold, probability of false positives and probability of detection (POD) as a function of crack size. P²IAT utilizes this evidence in a process called adaptation to update assessments and predictions. The veracity of detect/no-detect evidence is subject to uncertainties related to the quality and use of the sensor. This section describes how the reasoning process adapts predictions using state awareness evidence provided in the go, no-go form. First, the one-dimensional case (dimension a) is derived and then this is extended to two dimensions having joint probability in a and c.

Let $f_{t_1}(a)$ be the PDF of the predicted crack size at time t_1 calculated by any crack growth model (in our case, provided by SMART | DT), and the label "NC" represent the case where no crack is detected by NDI ("go" finding). The Bayesian posterior PDF of crack size at t_1 can be formulated as,

$$f_{t_1}(a | NC) = \frac{f(NC | a) \cdot f_{t_1}(a)}{\int_0^{+\infty} f(NC | a) \cdot f_{t_1}(a) da} \quad \text{Equation 1}$$

where $f(NC | a)$ is the likelihood function, and

$$f(NC | a) = \begin{cases} 1 - P_{FA} & a \leq a_{ct} \\ 1 - P_D(a) & a > a_{ct} \end{cases} \quad \text{Equation 2}$$

where a_{ct} is the crack size threshold value below which it is not considered a crack. This Bayesian updating process is illustrated in Figure 30. The posterior cumulative distribution function (CDF) can be calculated as:

$$F_{t_1}(a | NC) = \frac{\int_0^a f(NC | s) \cdot f_{t_1}(s) ds}{\int_0^{+\infty} f(NC | a) \cdot f_{t_1}(a) da} \quad \text{Equation 3}$$

Since the closed form solution to the integrals in Equation 3 is usually unavailable, effective numerical method is needed to calculate the updated CDF.

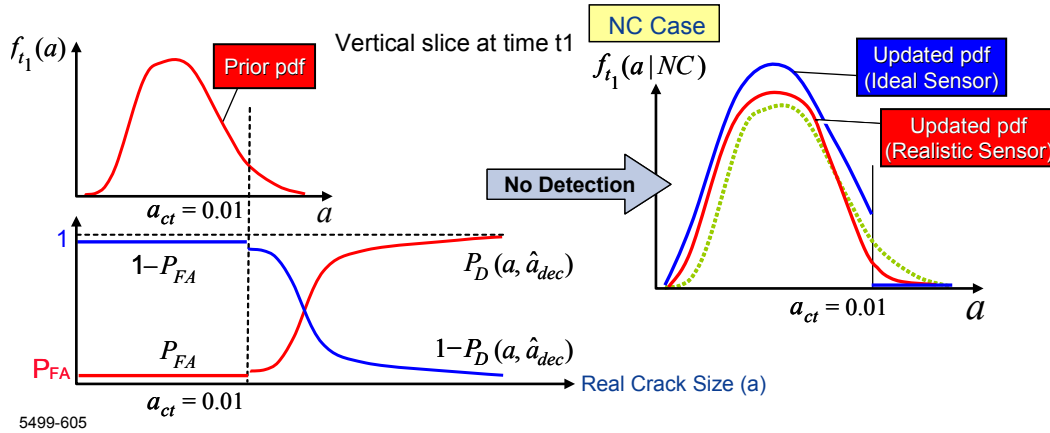


Figure 30. Adaptation for No Crack Detected Case

Similarly for a positive indication (crack detected) labeled as DC, the likelihood function is defined as:

$$f(DC | a) = \begin{cases} P_{FA} & a \leq a_{ct} \\ P_D(a) & a > a_{ct} \end{cases}, \quad \text{Equation 4}$$

and the Bayesian posterior PDF of crack size at t_1 can be formulated as

$$f(a | DC) = \frac{f(DC | a)f(a)}{\int_0^{+\infty} f(DC | a)f(a)da} \quad \text{Equation 5}$$

The Bayesian updating process for DC evidence is illustrated in Figure 31.

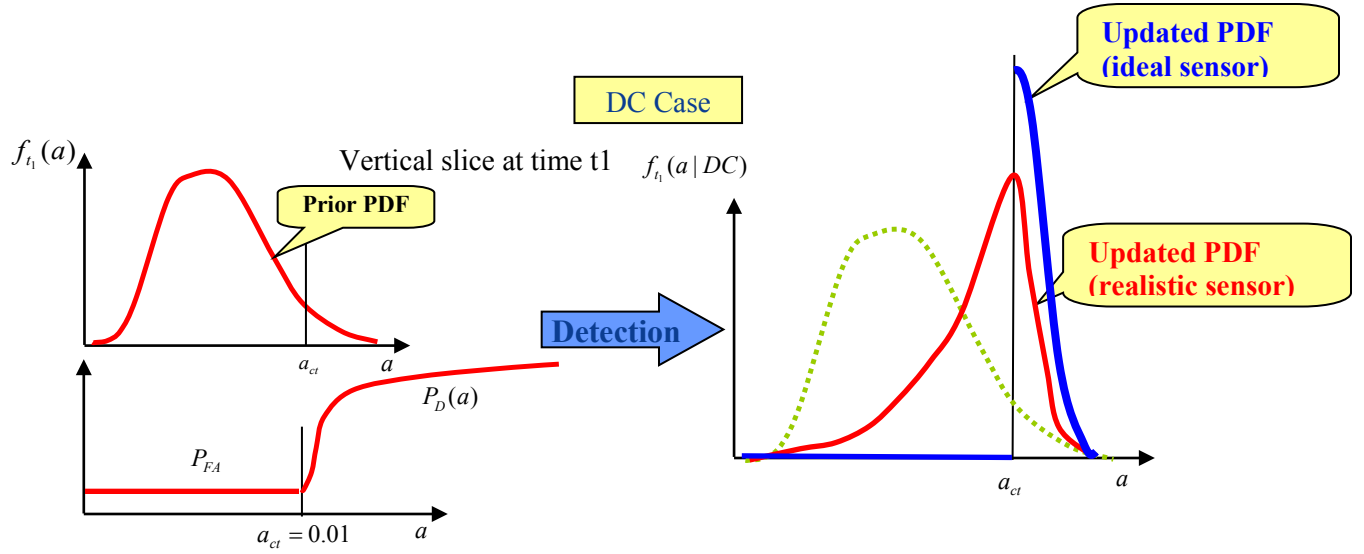


Figure 31. Adaptation for Crack Detected Case

The posterior CDF can be calculated as:

$$F_{t_1}(a | DC) = \frac{\int_0^a f(DC | s) \cdot f_{t_1}(s) ds}{\int_0^{+\infty} f(DC | a) \cdot f_{t_1}(a) da} \quad \text{Equation 6}$$

Next we extend the formulation to the two dimensional case where the crack size distribution is given as a joint crack-size distribution by SMART | DT. In this case, “a” represents the crack surface dimension and “c” represents the crack depth. In this derivation, the sensor detects crack depth c.

$$F(a_i, c_j | NC) = \begin{cases} 0 & , \quad a_i < 0 \text{ or } c_j < 0 \\ \frac{(1 - P_{FA})F(a_i, c_j)}{(1 - P_{FA})F_C(c_{ct}) + \sum_{n=M'}^{N'-1} (1 - P_D(c_n))(F_C(c_{n+1}) - F_C(c_n))} & , \quad 0 \leq c_j \leq c_{ct} \\ \frac{(1 - P_{FA})F(a_i, c_{ct}) + \sum_{n=M'}^{j-1} (1 - P_D(c_n))(F(a_i, c_{n+1}) - F(a_i, c_n))}{(1 - P_{FA})F_C(c_{ct}) + \sum_{n=M'}^{N'-1} (1 - P_D(c_n))(F_C(c_{n+1}) - F_C(c_n))} & , \quad c_j > c_{ct} \end{cases} \quad \text{Equation 7}$$

N is the number of points in the distribution. M' is that value where $c_{M'} = c_{ct}$. P_{FA} is the probability of a false positive given a crack of size a_i , c_j . P_d is the probability of detecting a crack of size c_j , and c_{ct} is the sensor's detection threshold.

The equation for the positive detect (DC) case is:

$$F(a_i, c_j | DC) = \begin{cases} 0 & , a_i < 0 \text{ or } c_j < 0 \\ \frac{(P_{FA})F(a_i, c_j)}{(P_{FA})F(c_{ct}) + \sum_{n=M'}^{N-1} (P_D(c_n))(F(c_{n+1}) - F(c_n))} & , 0 \leq c_j \leq c_{ct} \\ \frac{(P_{FA})F(a_i, c_{ct}) + \sum_{n=M'}^{j-1} (P_D(c_n))(F(a_i, c_{n+1}) - F(a_i, c_n))}{(P_{FA})F(c_{ct}) + \sum_{n=M'}^{N-1} (P_D(c_n))(F(c_{n+1}) - F(c_n))} & , c_j > c_{ct} \end{cases} \quad \text{Equation 8}$$

3.4 Decision Making With Uncertainty

This section addresses P²IAT concepts associated with decision making under uncertainty as it pertains to safety and maintenance. Two main decision topics are discussed in the context of uncertainty: safety and economics. This section addresses when to perform inspections to assure that vehicles operate with a SFPOF that is below a specified threshold especially considering the possibility of having rogue flaws. A methodology that links uncertainty reduction with economic benefits is also discussed. Although reducing uncertainty improves predictions and this can be readily linked to improving safety, there is a tacit assumption that there is also a potential economic benefit. Rather than deciding when to perform inspections to maximize information for the sake of uncertainty reduction, our methodology suggests when to perform inspections to minimize operating costs. In addition to inspections, the methodology may also be used to quantify the economic value of incorporating other sources of information so they may buy their way into the P²IAT process. Concepts presented in this section are based on fundamental economic principles as they relate to individual aircraft and individual decisions. The examples shown are intentionally simplified to best illustrate the concepts. Practical aspects of this methodology are also discussed.

3.4.1 Current Process

Current inspections per MIL-STD-1530C are driven mainly by safety as specified in Damage Tolerance Analyses (DTAs). Cracks at FCLs (FCLs) are assumed to be present from the start and grow to a critical size from an initial flaw size that could be as large as the largest crack size that could have been missed by the last inspection having negative findings. The next inspection is then scheduled to take place on or before half the time to reach the critical size as determined by this analysis. The associated FCL-specific parameters for this analysis are contained in the DTA or Force Structural Maintenance Plan (FSMP) reports including locations, critical crack sizes, NDI procedures and largest missed flaw sizes.

While this approach has served the USAF well, the P²IAT approach is intended to provide more accurate and less uncertain information upon which to base maintenance and inspection decisions. P²IAT for example, incorporates probabilistic analyses that provide a more realistic probability distribution of the expected crack size during inspections rather than simply assuming that a negative indication should use the largest undetectable size as the starting point for future predictions. P²IAT more explicitly accounts for the probability that an actual crack may be larger than the assumed “largest undetectable crack size” along with the probability that it may have been missed during inspection. By providing a probability distribution of expected crack sizes,

P²IAT avoids the worst-case “largest size” assumption that causes more frequent inspections. P²IAT thus offers potential benefits in both safety and avoidable inspection costs.

Deciding to perform an inspection simply on the basis of maximizing the information that might be gained (i.e., don’t inspect until it is likely that NDI will have a positive finding), ignores many other complicating factors that should be considered in choosing when to perform inspections and related maintenance. The following lists some of these factors:

Example Decision Factors

- Risk - failure consequences and their associated probabilities over the duration from the last to the next inspection/maintenance action including the avoidance of a single point failure during risk assessment.
- Opportunity Loss from premature actions - cost from performing maintenance including inspections sooner than actual conditions warrant due to uncertainties. Premature actions fail to extract remaining capability and thus raise the effective cost per flight hour.
- Added expense from performing maintenance too late - this is the other side of opportunity loss caused by overflying an individual vehicle creating damages requiring over and above repairs and extended loss of use.
- Opportunistic maintenance - performing service earlier than needed, but still saving cost and improving safety due to convenience and opportunity. This typically occurs when a vehicle is already in maintenance for other reasons. Delaying inspections or other actions that require a vehicle to return to the depot has less benefit if the vehicle must be in the depot for other reasons.
- Counter-productive work - inadvertently damaging or reducing reliability in the process of performing maintenance.
- Maintenance planning factors - lead time, workload, personnel availability and qualifications, prioritizations, preemptive actions, materiel availability etc.
- Unanticipated findings during maintenance - occasionally problems are found during maintenance that keeps an aircraft in depot for a longer period of time than originally scheduled. Consequently, the next aircraft coming into depot is stuck at the base with restricted flying hours forcing operators to meet daily flying hours with fewer vehicles. This motivates the need for early inspections.
- Other Metrics: Availability, Mission Capable Rate, materiel availability, scheduled and unscheduled inspection cost, maintenance costs, down time, mean time between failure (MTBF), Mean Repair Time, Type I Inherent Failures, Cost Assessment and Program Evaluation (CAPE) data (formerly Air Force Total Ownership Cost-AFTOC database)
- Information availability (e.g., Usage - in some cases flight records take 6 months to be processed), as-is vehicle state, prior maintenance etc. and immediate access to G054 Core Automated Maintenance Systems (CAMS) maintenance data and reconciled monthly

with Global Combat Support System (GCSS)

- Deployments, training, engine run certifications, weapons load certifications, daily flying schedule, and phase inspections also play a big part in the decision making.

In some cases, the inspection procedure requires the removal of the fastener at which point, the additional cost to enlarge the hole is not significant. Installation of the next larger size taper-lok fastener can have the benefit of introducing residual stress thus prolonging the time to the next inspection. Further, it may be difficult to reliably measure the crack size between fastener sizes without removing the fastener. If the fastener must be removed to determine if the crack is approaching the next size fastener, it may be advantageous to enlarge the hole and realize the benefit of extending the time to the next inspection.

This combination of factors, compounded by their uncertainties, complicates decisions that can dynamically change as a function of circumstances.

3.4.1.1 Rogue Flaws

One of the goals of P²IAT is to make better decisions about when to perform inspections. Although we expect P²IAT to minimize the number and extent of inspections, we are mindful that safety remains a priority. Current practice accounts for the possibility of having abnormal manufacturing or maintenance-induced defects that prematurely reduce life. These rogue flaws drive early inspections and may undermine potential savings afforded by P²IAT. A strategy is therefore needed to resolve the apparent conflict between postponing initial inspections for economic reasons and early inspections to maintain safety.

The current practice accounts for rogue flaws by assuming that there may be an initial crack when the structure first enters service that could have been missed by the manufacturer. The time it takes for that flaw to reach a size where the strength falls below an acceptable limit defines the time to reach the safety limit. Safety criteria dictate an initial inspection for unenhanced locations and subsequent inspections at half the time to reach the safety limit⁵. The figure below, adapted from Reference 1, illustrates the timeline for the first inspection.

⁵ Locations with life extension treatments (e.g., cold working, taper-lok fasteners, etc.) may have other criteria. Locations using taper-lok fasteners for example require the initial inspection at 95% of the time to reach the safety limit per FSMP paragraph 4.2.1. Subsequent inspections are prescribed at 50% of the remaining time to reach the safety limit.

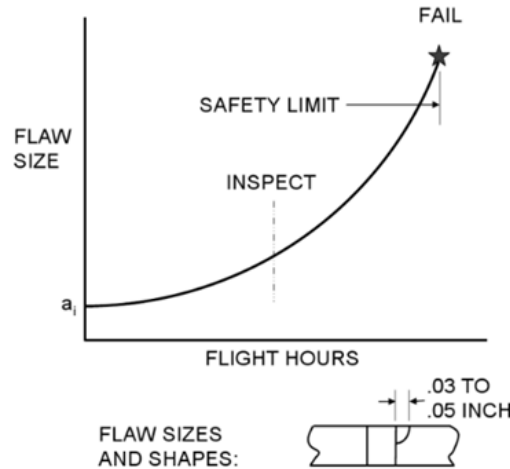


Figure 32. Inspection Time for Rogue Flaw

Since the POD for typical NDI is not 1.0, especially for NDI as performed outside of the ideal conditions in the laboratory, it is possible for a crack to be missed during the initial inspection. Evidence has shown these misses have occurred on every major class of USAF aircraft [ref 2]. Accounting for the possibility that a crack may have been missed during the first inspection, a reset size (a_{ASIP}) is used to continue the analysis to the safety limit. The $a_{90/95}$ size is traditionally used for this purpose. Note that 90% POD with a 95% confidence still leaves room for the possibility of an $a_{90/95}$ size or larger to be present. If the largest crack that could have been missed is a_{ASIP} , then repeating the strategy of inspecting at the point half way to the safety threshold would take us to a crack size of $a_{cr-miss}$ by the next inspection as shown in Figure 33 (from Reference 2).

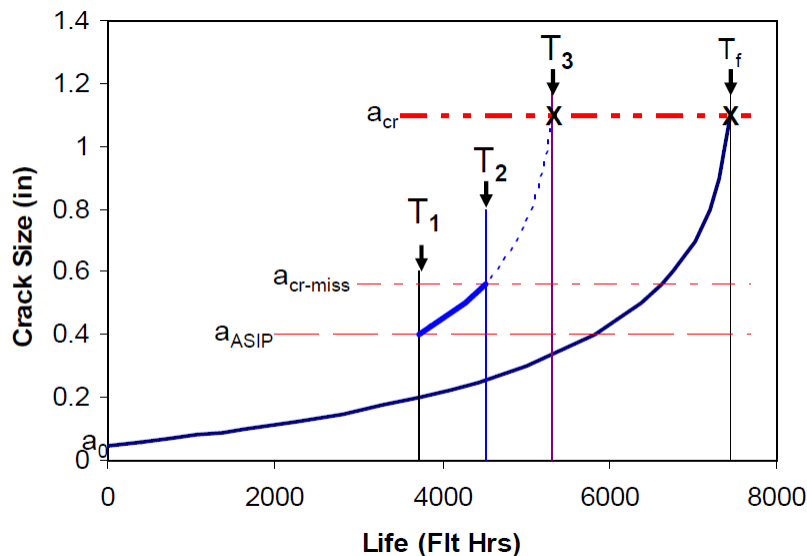


Figure 33. Key Crack Sizes for Determining Inspections (Figure 4 from Reference 2)

At time T_2 , inspection has a much better probability of detecting $a_{cr-miss}$, however there is still no guaranty that the flaw will be found in any of the three inspections (T_0 , T_1 and T_2). There is even a non-zero probability that the actual crack may be larger than the assumed largest crack that can be missed at each inspection that could result in catastrophic failure before T_1 , T_2 , or T_3 .

Since the underlying criterion is safety, we propose inspections based on probabilistic assessments of risk using mandated safety thresholds (e.g., SFPOF < 10⁻⁷), rather than a deterministic process using safety factors wherein the actual risk is not accurately quantified. The P²IAT methodology accounts for the probability that a crack is present in the entire range of sizes from zero to a_{cr} when it calculates SFPOF. It then updates the calculation with NDI findings using the probability that a crack will be missed over the entire range of sizes. The resulting posterior probability for the continuous case of crack sizes is derived directly from Bayes formula for the discrete case below:

The diagram shows the Bayes formula for the probability of a crack given a negative indication, $P(C | I^-)$. The formula is presented in two equivalent forms. Callouts identify the terms: $P(I^- | C)P(C)$ is the 'Probability of a crack given negative indication'; $P(I^-)$ is the 'Probability of miss (false negative)'; $P(I^- | C)P(C)$ is the 'A priori probability of a crack (from model prediction)'; and $P(I^- | \sim C)P(\sim C)$ is the 'Probability of a true negative'. Below the formula, it states 'Where I^- is a negative indication'.

$$P(C | I^-) = \frac{P(I^- | C)P(C)}{P(I^-)} = \frac{P(I^- | C)P(C)}{P(I^- | C)P(C) + P(I^- | \sim C)P(\sim C)}$$

Where I^- is a negative indication

**Equation
9**

A key distinction is made between the POD curves derived in a laboratory for a given NDI technique and the effective POD curve that accounts for the probability of misses in actual use on the vehicle. The former reflects how well the technique can perform while the latter reflects how badly the technique can perform in practice. When evaluating risk, the latter is more appropriate. This distinction is discussed in detail in reference 2. Rather than using the laboratory POD curve from which the probability of a true negative is inferred, it is preferable to use the effective POD curve in this analysis.

Within our P²IAT approach, we use an initial flaw size distribution that is based on the physics of failure driven primarily from the microstructure of the material [reference 3]. To be consistent with this approach, we propose to add a rogue flaw probability distribution to the microstructurally-derived initial flaw size distribution. We recognize that a rogue-flaw probability distribution may not currently exist, but discuss it here to establish the theoretical foundation for the approach. The resulting distribution is bi-modal having a rogue flaw mode responsible for infant mortality and a microstructure mode that accounts for ordinary damage evolution. We have shown that within the first two inspections, the POF associated with the rogue flaw mode rapidly diminishes toward zero with negative NDI findings allowing the balance of P²IAT specified inspections to benefit from less-frequent inspections consistent with microstructurally derived flaws.

Our approach predicts the expected time to reach the safety threshold (10⁻⁷ nominal) for SFPOF and schedules an inspection prior to that point in time. In all cases, inspections are scheduled at the normal tempo as shown in Figure 36 (maintenance opportunities at 200 flight hour boundaries for F-15).

3.4.1.2 Demonstration Criteria

For illustration purposes, we make the following assumptions:

- Probability of a rogue flaw is 10⁻³ wherein the distribution of sizes is normally distributed with a mean of 0.04 inch and a standard deviation of 0.003 (see Figure 34)

- The upper bound on the SFPOF is 10^{-7} and that criteria is the trigger point for every inspection driven by safety⁶
- The expected time is used rather than the 95% confidence point
- Inspections happen on 200 flight-hour boundaries.

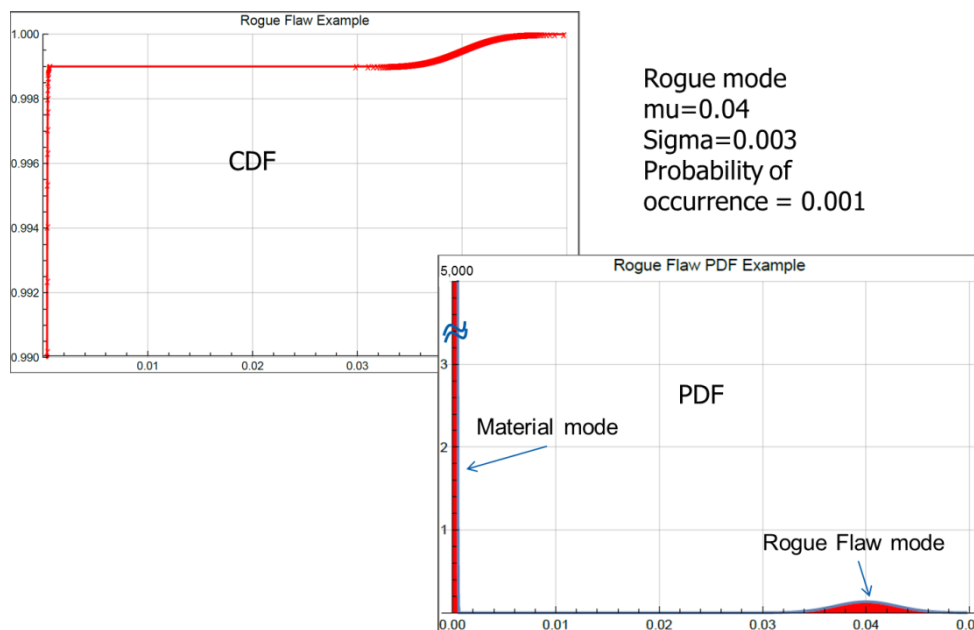


Figure 34. Bi-modal Initial Flaw Distribution

Given these assumptions, the next inspection for safety is the 200 flight hour boundary that precedes the point in time where P²IAT determines that the expected time when SFPOF for the next flight will exceed 10^{-7} as shown with red line in Figure 35 (alternatively use the green line if the 95% confidence point is preferred). This calculation includes the rogue flaw mode in addition to the microstructurally-derived initial flaw mode. With these criteria, the time for the first inspection will be determined for each individual safety-critical control point to be tracked using P²IAT. The time for subsequent inspections that are driven by safety will be determined based P²IAT predictions adapted by negative findings at each inspection. In general, the time for all subsequent inspections will be the sooner of the points determined by safety and economic criteria.

⁶ Economic criteria may drive inspections at an earlier point in time

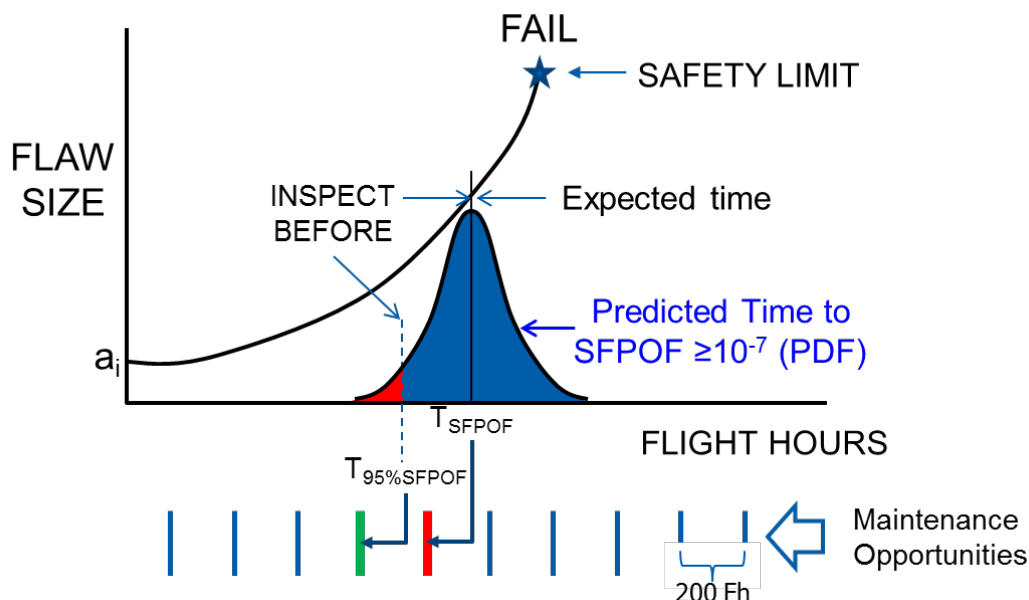


Figure 35. SFPOF-based Inspections

The expected time rather than the 95% confidence line was chosen for illustration in Section 4.2 since it is consistent with the current criteria.

3.4.1.3 Economic Decisions (for Non-Safety-of-Flight Control Points)

After the first few inspections we expect to be past the infant mortality phase caused by rogue flaws and into a safer operating phase where cost savings will play a more dominant role in decision making. At this phase, maintenance is typically focused on serviceable flaws in non-safety of flight (SOF) locations - that is to say, flaws that are sufficiently large to warrant repair, but not so large that they are dangerous. P²IAT provides information upon which to base economic decisions for various maintenance actions including inspections. Rather than a fleet-wide analysis using generalized statistics [for example see reference 4], we examine the economics at the individual control point within an individual vehicle.

Progressive maintenance actions should be taken to repair damage before it becomes problematic. Each action is appropriate at specific crack sizes that can occur at various times as shown in Figure 36. Typically, the earlier damage is fixed, the less expensive the action. Drilling a stop-arrest hole for example is likely to be less expensive than installing a reinforcement plate. Damage should ordinarily be corrected as soon as possible, however, there is a tradeoff between the cost of inspecting too often for damage that is not likely present and waiting so long that damage progresses and requires a more extensive repair. In general, a just-in-time maintenance philosophy is preferred where unnecessary inspections are minimized and key inspections are performed with sufficient lead time to take necessary actions. P²IAT can predict the time when the damage extent will reach the size that is appropriate for each remediation step so that inspections can be performed prior to repairs to corroborate the presence of damage.

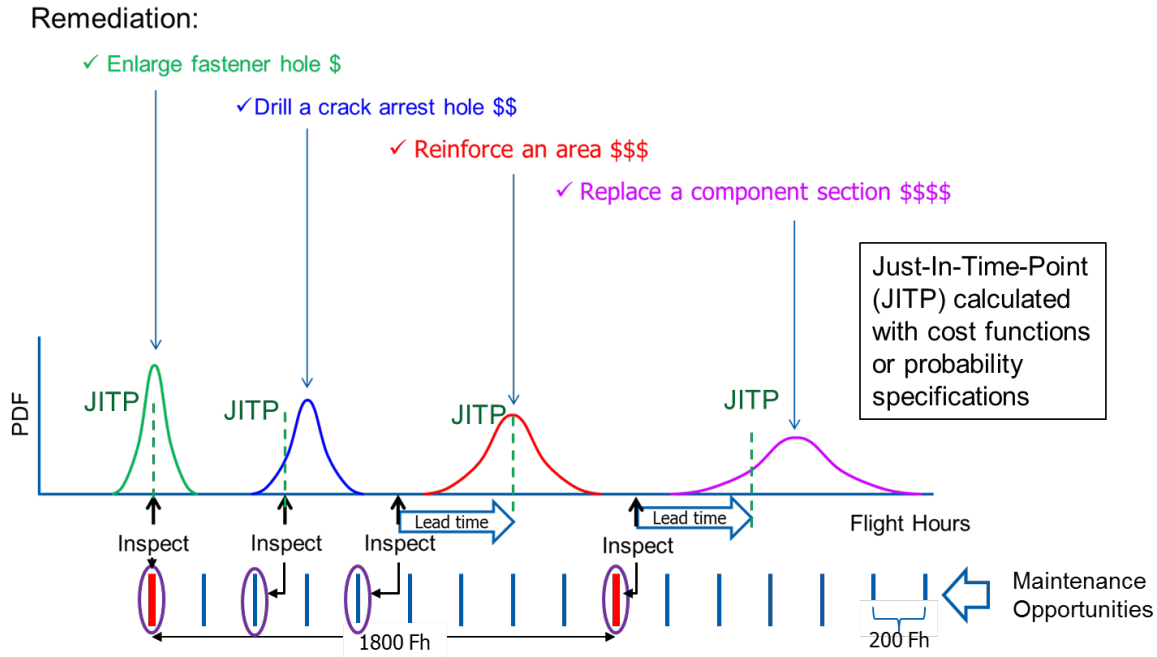


Figure 36. P²IAT Predicts When to Perform Damage Remediation Actions

Notice that the time of inspection relative to the time at each PDF mode in the figure can vary depending on many factors especially criticality of the damage and the lead time necessary to perform certain repairs. A recommendation for Prognosis Requirements that incorporates lead time, maximum allowable POF, maximum tolerable probability of premature maintenance and confidence can be found in the appendix . If an inspection is performed having negative findings, the prediction for each remediation will change according to the posterior probability calculated by P²IAT.

The optimum time for an inspection of non-safety-of-flight locations from a cost savings perspective depends on the costs of performing the inspection (which includes the loss of use of the vehicle during the inspection) and the cost of not performing the inspection - namely the cost of more extensive remediation. To further complicate matters, the time that a given remediation step is predicted has uncertainties reflected by the spread in the probability density functions (PDF) in the figure. Fortunately, fundamental economic theories can be applied to finding the optimal time for repairs and inspections given the uncertainties in the process.

To explain the economic decision methodology, we use the example of performing an inspection to determine if a potential crack around a fastener should be removed by reaming the hole and installing the next larger fastener. In practice, there may not be any lead time required for this particular action since the inspection and reaming often takes place essentially in the same maintenance visit as will be further explained later. We used this example to simplify the explanation and avoid the lead time complications discussed in the prognosis requirements discussion in the appendix.

Although the reaming process can remove a growing crack and prolong the life of the associated component, it may only be done a limited number of times if at all and should only be done when absolutely necessary. For those situations where this remediation is appropriate, a decision must be made that considers the uncertainties associated with predictions, the

probabilities of detection and false positives for NDI and the optimal time to perform such inspection/remediations. If we drill out a hole too early, we lose some of the remaining capability of the component and we take the vehicle out of service when it wasn't necessary. On the other hand, if we wait too long, the crack may grow beyond the next fastener size requiring a more expensive repair and potentially compromising safety. Ideally, we should perform maintenance when the crack reaches the largest size (a) that can be safely drilled out (at time Q in Figure 37). Unfortunately, it is extremely difficult to predict that exact time due to uncertainties. In many cases, it is also difficult to detect or measure cracks of this size. If the cost of waiting too long is greater than the cost of performing maintenance too early, it is generally better to favor early maintenance. The question then becomes - how early should actions be taken? That answer requires a probabilistic analysis.

The following figure represents a simple cost function for mitigating a small crack.

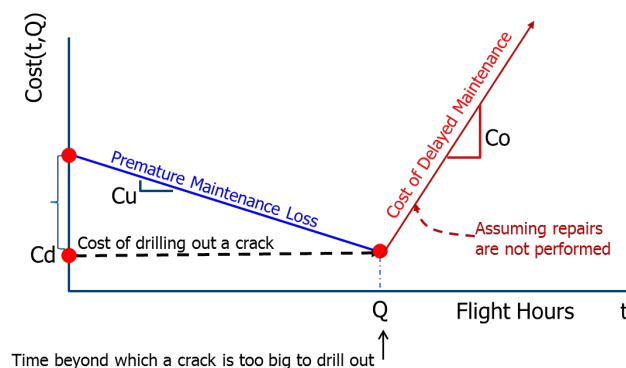


Figure 37. Example Cost Function for Mitigating Small Cracks

The cost function in this example is simplified for illustrative purposes⁷. The cost of delayed maintenance is more likely to be a step function at Q , but to simplify the mathematics in this example, we show Co , the rate at which the cost of delaying action increases as five times faster than Cu , the rate at which the cost of premature maintenance decreases (i.e., $|Co| = 5Cu$).

3.4.2 Probabilistic Aspect

If it were possible to predict exactly when the crack reaches the maximum size (a) that can be drilled out (at time Q), costs could be minimized by performing maintenance exactly at that time. Unfortunately, that exact time Q is usually unknown since crack size is dependent on many variables that are unknown. Note the cost is a function of the time t at which the repair occurs, and it is also a function of Q . Q is now considered to be a random variable which is determined by the probability distribution for when the crack reaches maximum allowable size for repair. Let t represent the flight hours when the repair is made, and t is fixed. If $Q < t$, then the cost is $Co(t - Q)$ since the repair time t is greater than Q the optimal time for repair (when the maximum allowable size (a) occurs), and if $Q > t$, then the cost is $Cu(Q - t)$. We have also shifted the minimum to zero since we are only interested in finding the best time in this analysis (see Figure 38). The actual cost will be shifted as a function of circumstances (considering all the complicating factors discussed earlier). Consequently, the expected cost function may be calculated as an expectation accounting for the probability distribution in the prediction (see

⁷ The cost of drilling out a crack is situation dependent and should account for the factors mentioned earlier including opportunistic maintenance.

Figure 38 and Equation 10) of Q . In this example, the PDF from P²IAT for the predicted time to reach Q is shown in black with its most likely time at 50 flight hours (the PDF is scaled by one thousand so it can be seen on this plot).

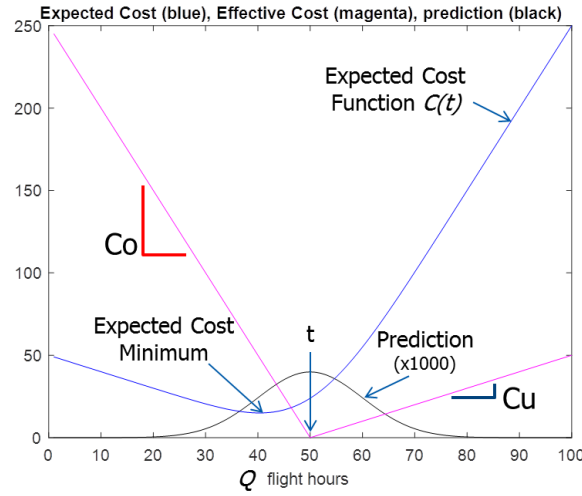


Figure 38. Finding the Expected Minimum Cost Point

The blue curve is the expected cost function that accounts for the uncertainty in the prediction. This is given by:

$$C(t) = C_u \int_t^{\infty} (Q - t)f(Q)dQ + C_o \int_0^t (t - Q)f(Q)dQ \quad \text{Equation 10}$$

To minimize this probabilistic cost function, we take its derivative, set it equal to zero and solve for the probability $F(T^*)$ - the probability for the optimized point T^* that corresponds to the point in time where the probabilistic cost is minimized (since its dC/dt is < 0 if $F(t) < C_u/(C_o + C_u)$ and $dC/dt > 0$ if $F(t) > C_u/(C_o + C_u)$):

$$\frac{dC(t)}{dt} = C_o F(t) - C_u (1 - F(t)) = 0$$

$$\text{In this simplified case, } F(T^*) = C_u/(C_o + C_u) \quad \text{Equation 11}$$

Now that we have the probability $F(T^*)$ that corresponds to the optimized point T^* , we find T^* using the inverse of the predicted CDF as shown in Figure 39.

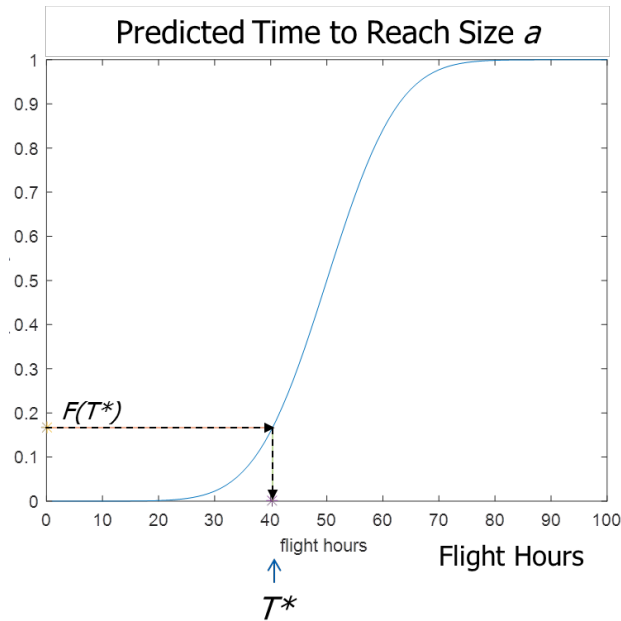


Figure 39. Finding the Ideal Time Given Its Probability

Performing maintenance at time T^* gives us the minimum expected cost.

It is worth noting that the expected cost decreases as the uncertainty in the prediction is reduced as shown in Figure 40. The black PDF in the figure has a standard deviation of 10, whereas the magenta PDF has a standard deviation of 3. If we find the minimums of their respective cost functions (also black and magenta), we can quantify the savings on the vertical axis. As the uncertainty goes to zero, the expected cost at time T^* approaches the ideal minimum cost at time Q .

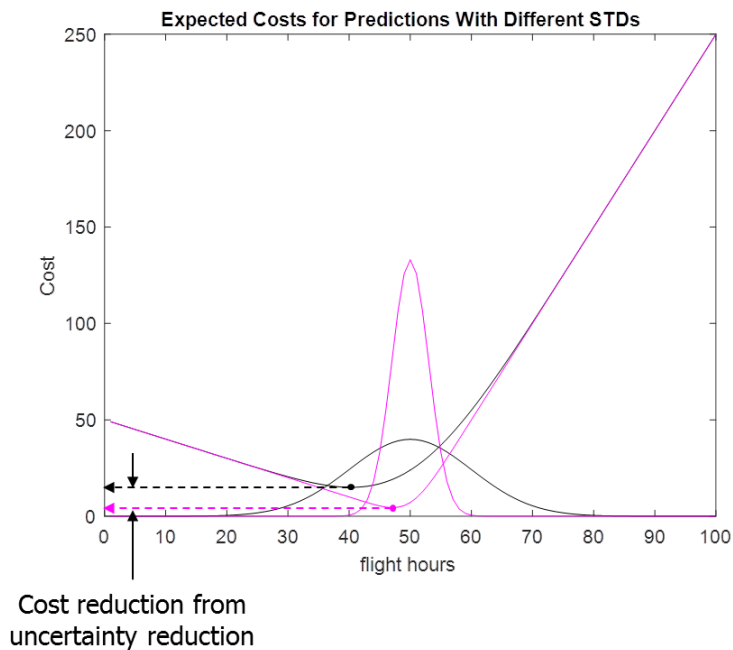


Figure 40. Expected Cost Reduced by Reducing Uncertainty

Using this analysis, it is theoretically possible to project the cost savings associated with inspections using a what-if approach. For example, suppose the initial prediction from our models produces the black PDF as shown in the figure above. We can then generate hypothetical posteriori results using adaptation with all potential results from NDI. In the simplest case, both positive and negative indications can be hypothesized and the resulting cost savings from posterior PDFs (e.g., hypothetical magenta PDFs), can be calculated before the inspection is performed. We can also use the predicted (a priori) crack size distribution at the time of inspection to determine the likelihood of both positive and negative NDI outcomes.

3.4.3 Practical Considerations

Although this methodology can be used to predict the point in time having the lowest cost considering the uncertainties in the prediction, there are some practical aspects to consider when implementing this strategy. First, the cost function depicted in Figure 37 is admittedly over simplified. The true cost function is dynamic and a function of circumstances driven by the complicating factors described earlier under the heading Current Process. As circumstances change (opportunistic maintenance, mission requirements, maintenance planning factors etc.) the true cost function changes in ways that are beyond the scope of this contract to codify.

It is also important to consider the idealized versus the practical point of remediation. For example, the control points selected for the full scale experiment typically use a 3/16-inch diameter taper-lok fastener. The next size up (aside from some smaller increments using different group sizes) is 1/4 inch in diameter (the nominal fastener increment size is 1/16 inch in diameter). This means that the largest crack that can be drilled out with one step size is 1/32 inch (about 0.03 inch). This is below the typical size that can be reliably detected, even by removing the fastener and using an in-hole NDI technique. As discussed earlier, if fastener removal is called out in the tech order for NDI, and the likelihood of finding a crack less than 1/32 inch with NDI is remote, it is reasonable to perform NDI to make sure a crack of a much larger size is not present. But then, since a smaller undetectable crack may be present, replace the fastener with the next larger size as a preventative measure. Not only does this stop an undetected small crack from growing, it also allows the maintainer to delay the next inspection by restoring the residual stress that may have been eroded during operations. For taper-loks per the FSMP, the next inspection is then postponed until 95% rather than 50% of the time to reach the safety limit.

How then does the idealized methodology fit with these practical maintenance aspects? First, realize that this is a probabilistic analysis and the fact that NDI will most likely miss a crack on the order of 0.03 inch is accounted for in the analysis. A negative indication from NDI still has value in the probabilistic analysis by significantly lowering the probability of having a large crack. It also lowers the probability of having a medium size crack albeit to a lesser extent. It even lowers the probability of having a small crack but to a much lesser extent. Nonetheless, a negative indication reduces the uncertainty in the model prediction provided the probability of a false negative is less than the probability of a true negative.

Regardless of the specific repair, an evaluation of the probability of catastrophic failure from the present to the next opportunity to make repairs for each remediation step size should be performed to best understand what level of defect can be tolerated. These predictions can be made using the prediction horizon. Similarly, the life extension benefit of each repair can be hypothesized using P²IAT. After determining which repair size offers the best benefit, the time to

reach that size can be predicted with its associated uncertainty and the best time to take action (typically beginning with an inspection) can be predicted using the method described earlier.

For example, if the most benefit can be realized by enlarging a taper-lok hole by one step size, the process would predict the time to reach that size, generate the cost function at the time of remediation, determine how much sooner to perform the action using the method described earlier, and perform in-hole NDI at that time. If the result is negative (no indication of a crack), install the next larger taper-lok to minimize costs and prolong the next inspection. If NDI returns a positive indication, rerun P²IAT to adapt the current state assessment and repeat the prediction process to evaluate the best time to perform the next appropriate remediation (e.g., reinforcing a section, replacing a section, replacing a subcomponent or entire component).

3.4.4 When Cost Functions Are Not Available

An alternative to the cost function approach is provided in the discussion of Prognosis Requirements in the appendix. With this alternative, the user still specifies lead time, but also specifies maximum allowable POF, maximum tolerable probability of premature maintenance and confidence. Maximum allowable POF is used to determine the latest time for maintenance and bounds risk. Maximum tolerable probability of premature maintenance determines how early maintenance could be performed considering the possibility that early maintenance may be unnecessary. The midpoint between the earliest and latest times is called the just-in-time point (JITP). Inspections should be performed in advance of this point as specified by the lead time.

3.4.5 Decision Process Flowchart

The overall P²IAT process is describe in the flowchart in Figure 41. Oval symbols are process entry and exit points. Blue rectangles depict functions within the P²IAT process and gray rectangles depict user inputs. This process is applied to each individual control point for the selected vehicle. The following discussion focusses only on the decision functions in the process.

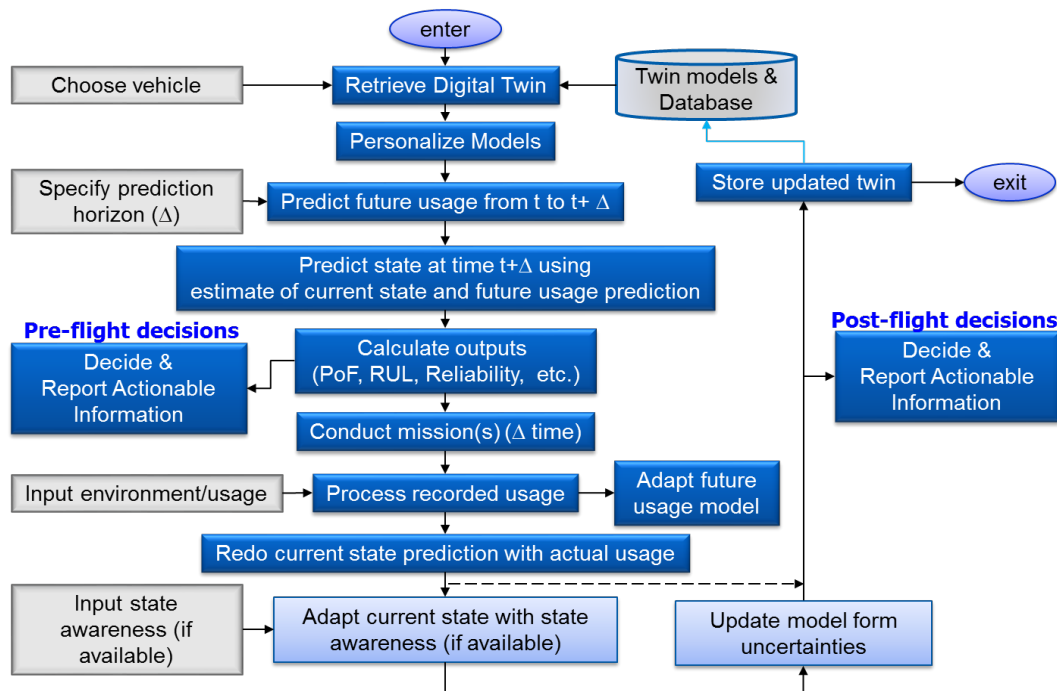


Figure 41. P²IAT Overall Process for Each Control Point

There are two places in this flow labelled “Decide & Report Actionable Information” where decisions regarding when to inspect the monitored control points are made. The function on the left provides an opportunity to assess the reliability and potential need to perform inspections over the duration extending from the present to the prediction horizon chosen two steps earlier in the process. The prediction horizon specifies the desired deployment period over which the vehicle is required to perform safely and without the need for interim inspections. This is an opportunity to determine if a particular vehicle given a specific control point is suitable for deployment over the specified prediction horizon. The process flowchart for this decision is expanded in Figure 42.

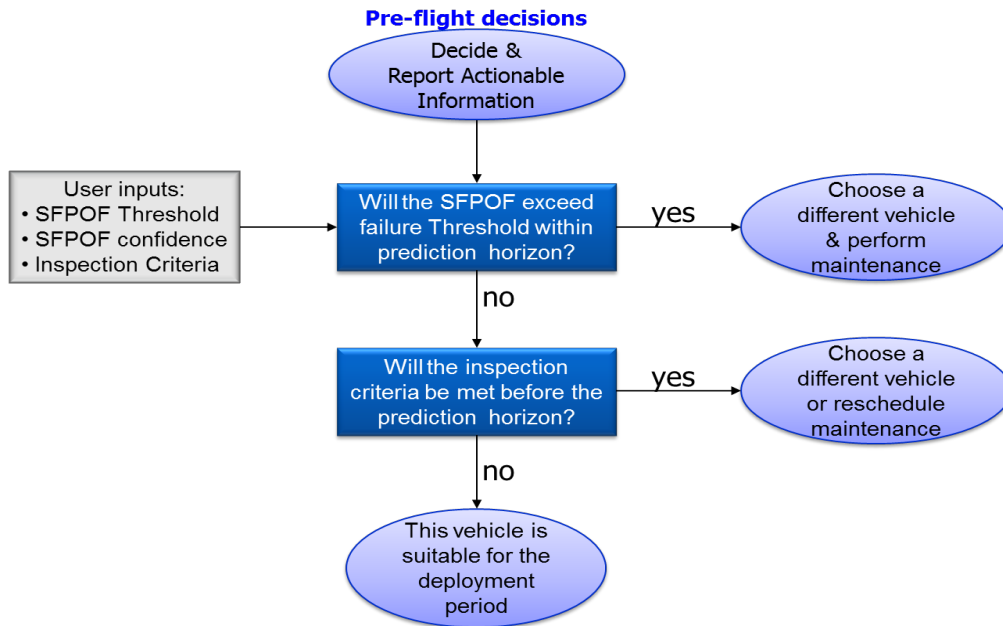


Figure 42. Pre-Flight Decisions

The user provides a threshold for SFPOF (e.g., 10^{-7}) and an associated confidence if required (e.g., 95%) for the current circumstances above which the vehicle should not be operated. P²IAT calculates the number of flight hours to reach the SFPOF threshold at the specified confidence level, and if that point occurs within the horizon, a recommendation is made to choose another vehicle and perform maintenance on the control point related to this analysis. If that is not the case, the next step checks whether or not an inspections will be necessary within the horizon. The user also provides the current cost function at an appropriate repair size (crack size where remediation will be optimum) for remediation at the control point. The user also provides other inspection criteria: lead time, maximum allowable POF, maximum tolerable probability of premature maintenance and confidence. The lead time specifies the time needed before the actual repair for the associated inspection and other planning purposes. If a cost function is not available, the user can use the maximum allowable POF and maximum tolerable probability of premature maintenance to determine the just-in-time maintenance point and move back in time from there using the lead time to determine when to do the inspection. This alternative is discussed in Prognosis Requirements in the appendix.

If the analysis determines that the optimum time for inspection is beyond the horizon, the vehicle is deemed suitable for the deployment period. If not, a recommendation is made to

choose a different vehicle and schedule this one for maintenance. This determination can only be made once the horizon is specified given an assessment of the current state of the vehicle.

Once the vehicle has flown and the actual usage and/or inspection results are incorporated, the analysis is repeated and post flight decisions can be made per the next decision flowchart shown in Figure 43.

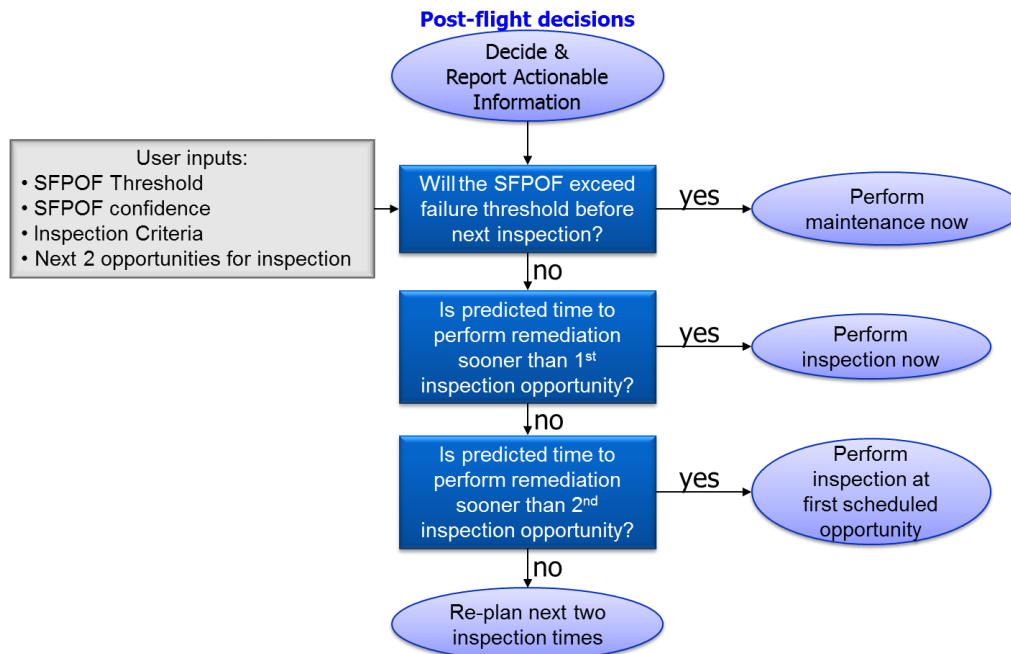


Figure 43. Post-Flight Decisions

After data from flights or inspections become available (even before the end of the previously specified horizon including data that was previously lost), it is appropriate to update predictions replacing anticipated or imputed usage with actual usage. This process creates another opportunity to reevaluate maintenance plans. As in the pre-flight decision process, the user provides SFPOF, the associated confidence required, the lead time, the cost functions (if available or the maximum allowable POF, maximum tolerable probability of premature maintenance) and the scheduled time for the next two ordinary inspection opportunities (i.e., the next two times the vehicle is expected to become available for inspection accounting for all the considerations mentioned earlier in this section). If the SFPOF exceeds the threshold with the specified confidence before the next normal opportunity for inspection, the vehicle is not safe to fly for that period and should be maintained as soon as possible. Similarly, if the vehicle will exceed the inspection criteria before the next scheduled opportunity, an inspection should take place sooner than planned. If the stated criteria will be exceeded between the first and second inspection opportunities, the vehicle should be inspected at the next scheduled opportunity. Finally, if the vehicle will not need an inspection before the second opportunity, the next two inspection times can be re-planned for a later date.

4.0 RESULTS AND DISCUSSION

Section 4.0 of this Final Report is focused on SOW Task 5 where the team demonstrated the end-to-end P²IAT process using randomly generated and gap-filled flight data from the statistically derived flight spectrum. These data were processed to extract maneuvers that were mapped to stresses at a selected control point using the government-furnished StS-RtS v1 and FEM. The first portion of this section discusses the P²IAT implementation in ModelCenter. The second portion shows some key results.

4.1 Overview of Northrop Grumman Corporation's (NGC) P²IAT Implementation

ModelCenter was chosen as the implementation platform for Version 1 of NGC's P²IAT process because it facilitates the integration of the variety of executable codes that are required to achieve P²IAT. Due to its modular nature, the process model that implements NGC's P²IAT is easily extendable to incorporate updates for future enhancements to the existing executable codes or to incorporate additional crack growth codes. It also provides the flexibility to change or add to the P²IAT process and to incorporate more sources of uncertainty in the future. NGC's P²IAT process is implemented as two separate but related ModelCenter process models:

P2IAT_SingleCP_Version_1.pxc and P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc.

There are many different executables and source codes integrated in both of NGC's P²IAT process models, some of which were developed specifically for Task Order 0001. The SMART|DT code developed by the UTSA is the core of NGC's P²IAT process and is used to generate all of the probabilistic predictions for crack size, POF and remaining useful life for both current and future states for a control point. SMART|DT was initially developed with funding from the FAA but was significantly enhanced for the P²IAT application. The models also contain VBScript components that set up the files for a new update, implement the logic necessary to prepare input files needed to call SMART|DT and other executables, process and store the final output files generated by the executables, delete unnecessary intermediate output files and perform other processing such as file merges and processing of the inspection record.

The NGC P²IAT models were designed to automate running the host of executables needed to perform P²IAT with minimal input from the user. Both models utilize and automatically populate a file structure that organizes the required input files and preserves the important output files in logical folders to maintain a record of the update information.

4.1.1 P²IAT File Structure

At the top of the file structure are two folders: *FDR Data Central Repository* and *LocalUpdateData*. The *FDR Data Central Repository* is where all data files from the FDR are stored as *.xlsx* files. The *LocalUpdateData* folder contains all the vehicle and control point specific input and output files used and generated by the software in the model. The first level of subfolders under the *LocalUpdateData* folder contains top level folders designated for each vehicle in the P²IAT Version 1.0 model. Vehicles are simply named L and R in anticipation of the full-scale test. Under each vehicle folder there are folders for each update period and also pre-populated folders that must be initialized before any updates are performed. All vehicle level subfolders also have additional subfolders for each control point as appropriate.

The initial pre-populated folders are: *Templates*, *InitialLifeData*, *InspectionData*, *FirstUpdateStarterFiles*, *Executables* and *RestoreExecutables_Version 1*. The *Templates* folder

contains the template for generating the report for the update period. The *InitialLifeData* folder contains files for the various initial crack size distributions and subfolders for each control point containing input file templates for SMART|DT as well as the baseline stress history for the control point and its associated files. The *InspectionData* folder contains the inspection record for the vehicle, *InspectionRecord.xlsm*, as well as two files containing engineering parameters that the user can modify: *InspectionParameters.txt* and *MaintenanceSchedule.xlsx*.

FirstUpdateStarterFiles contains the initial versions for four spreadsheets used by the model to record and reference the following data: missing FDR data (*L_Missing_Flt_Nums.xlsx*), re-run points (*L_ReRun_Flt_Nums.xlsx*), update flight boundaries (*L_QuarterlyUpdateFlightBoundaries.xlsx*) and recovered FDR data (*L_RecoveryRecord.xlsx*).

Executables is the folder in which all executables are stored and from which they are run when each model executes. *RestoreExecutables_Version 1* contains a clean copy of the files in *Executables* in case the *Executables* folder needs to be restored after an inadvertent execution of the model.

4.1.2 Overview of P²IAT Models

The P2IAT_SingleCP_Version_1.pxc process model performs the quarterly update process for a given control point on a vehicle. The P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc process model performs the recovery process for missing flight data that has been recovered.

The P2IAT_SingleCP_Version_1.pxc model performs a quarterly update for a specific aircraft control point at the last flight of a given update period. This consists of creating a stress history from FDR data and using it to predict a probabilistic assessment of the joint crack size. Additionally, the model uses the baseline spectrum to grow the crack distribution from the last flight of update period out to the prediction horizon (nominally 1,000 flight hours) at which point, probabilistic assessments of POF and remaining useful life are generated. These assessments are used to predict when to recommend inspections for both safety and remediation. An update report is automatically generated to capture the required probabilistic assessments and predictions for the update period.

The P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc model performs the processing for FDR data that has been recovered. This includes creating an updated stress history with the recovered data and recalculating a probabilistic assessment of the joint crack size distribution at the last flight of the previous update period.

Both models were designed to require minimal input from the user and automatically populate a file structure that organizes the files needed for archiving a record of the update. The structure and operation of model P2IAT_SingleCP_Version_1.pxc will be described first followed by a description of model P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc which utilizes some of the same components and submodels contained in the P2IAT_SingleCP_Version_1.pxc model.

4.1.3 Detailed Description of P²IAT_SingleCP_Version_1.pxc Process Model

Details of each user input required to run the P2IAT_SingleCP_Version_1.pxc model are described in the Software User's Manual but the screenshot below is a view of how the user configures the model for running an update. The parameter names are the black text and the parameter values are in green. After the first update, normally the user will only need to update

the bottom five parameters: Select_Vehicle, Select_CP, DateOfUpdate, DateOf_Last_IAT_Data and LastFlightOfUpdatePeriod. The settings for Select_Vehicle, Select_CP and LastFlightOfUpdatePeriod are used to populate the file structure as described above. The date parameters are used for generating the update period report.

FirstUpdateEver	false
CrackGrowthCodeFormat	SMART_NASGRO
CRACK_GROWTH_CODE	MASTERC
InitialFlawSize	NoRogueFlaw
Select_Vehicle	L
Select_CP	CP_004
DateOfUpdate_mmdyyyyy	1/16/2017
DateOf_Last_IAT_Data_mmdyyyyy	1/15/2017
LastFlightOfUpdatePeriod	200

Figure 44. User Input Parameters for ModelCenter Model P2IAT_SingleCP_Version1.pxc

Each time model P2IAT_SingleCP_Version_1.pxc is run to process an update for a vehicle control point, it creates the folders necessary to store the output files that must be retained for that control point. Under the appropriate vehicle folder, an update folder is created with the name *UpdFlt_#####* where ##### denotes the last flight flown prior to the current update. A copy of the FDR data file for the update period is stored in folder *UpdFlt_#####* so it can be processed locally rather than in the *FDR Data Central Repository*. A subfolder for the control point is also created by the model and named to match the control point label, for example, *CP_004*. Each time the model is run to process a different control point, a new control point subfolder is created. The control point subfolder is where the model stores all the output files that need to be retained for reference.

The screenshot below shows the collapsed version of the P²IAT Version 1 process model to highlight the main functional components of the model. Each  component expands into a sequence submodel.

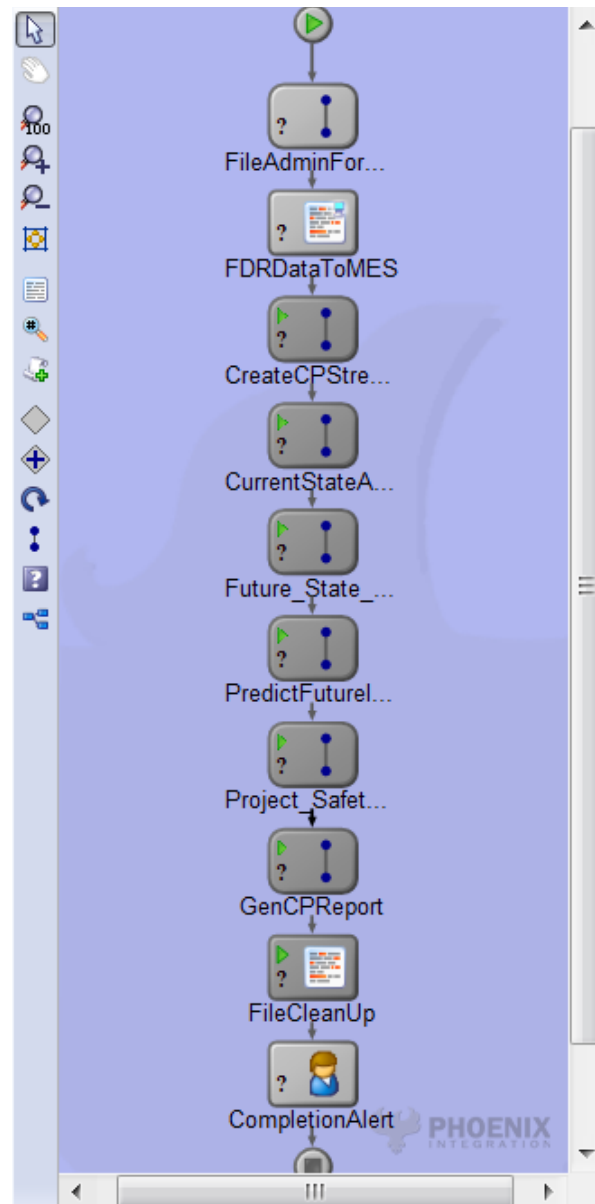


Figure 45. Collapsed Overview of ModelCenter Model P2IAT_SingleCP_Version1.pxc

Due to the graphics in ModelCenter some of the names of the collapsed modules are not fully visible. However, from top to bottom, the main modules of the model are: *FileAdminForUpdate*, *FDRDataToMES*, *CreateCPStressHistoryForUpdate*, *CurrentStateAssessment*, *Future_State_Assessment*, *PredictFutureInspections*, *Project_Safety_For_Next_1000Hours*, *GenCPReport*, *FileCleanUp* and *CompletionAlert*. Their respective functions are: perform the file structure administration for the update, translate FDR data to a Master Event Sequence file, create a stress history file from the FDR Data, run SMART|DT to perform current state assessment, run SMART|DT to perform future state assessment, predict future inspections, project safety for the next 1,000 hours, generate the report for the update period and delete unnecessary files. A detailed description of each module follows.

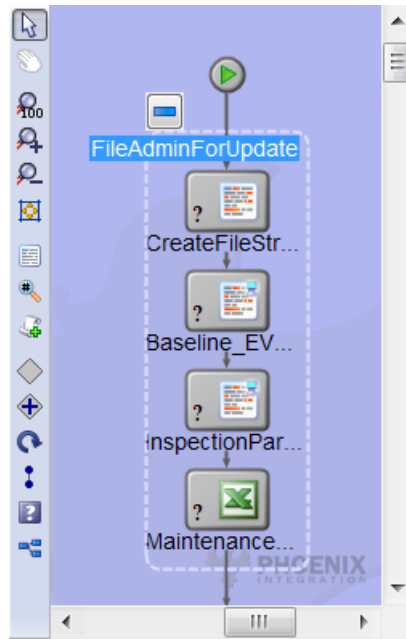


Figure 46. Expanded View of Submodel FileAdminForUpdate

This screenshot shows the expansion of the *FileAdminForUpdate* module into four components. The first component, *CreateFileStructureForUpdate*, creates the folders for the current update and copies all required input files to the *Executables* folder. The other three components instantiate the data from the three files *Baseline_EVD.EVD*, *InspectionParameters.txt* and *MaintenanceSchedule.xlsx* in the model so this data can be utilized by other components in the model.



Figure 47. Component FDRDataToMES

FDRDataToMES is a single component that instantiates the *FDRDataToMES* executable which converts data from the FDR file to a MES file. The MES file will be used by the next module *CreateCPStressHistoryForUpdate* to create the stress history file for processing the update. *FDRDataToMES* also determines if there are missing flights in the FDR data file and provides a count of the missing flights to *CreateCPStressHistoryForUpdate*.

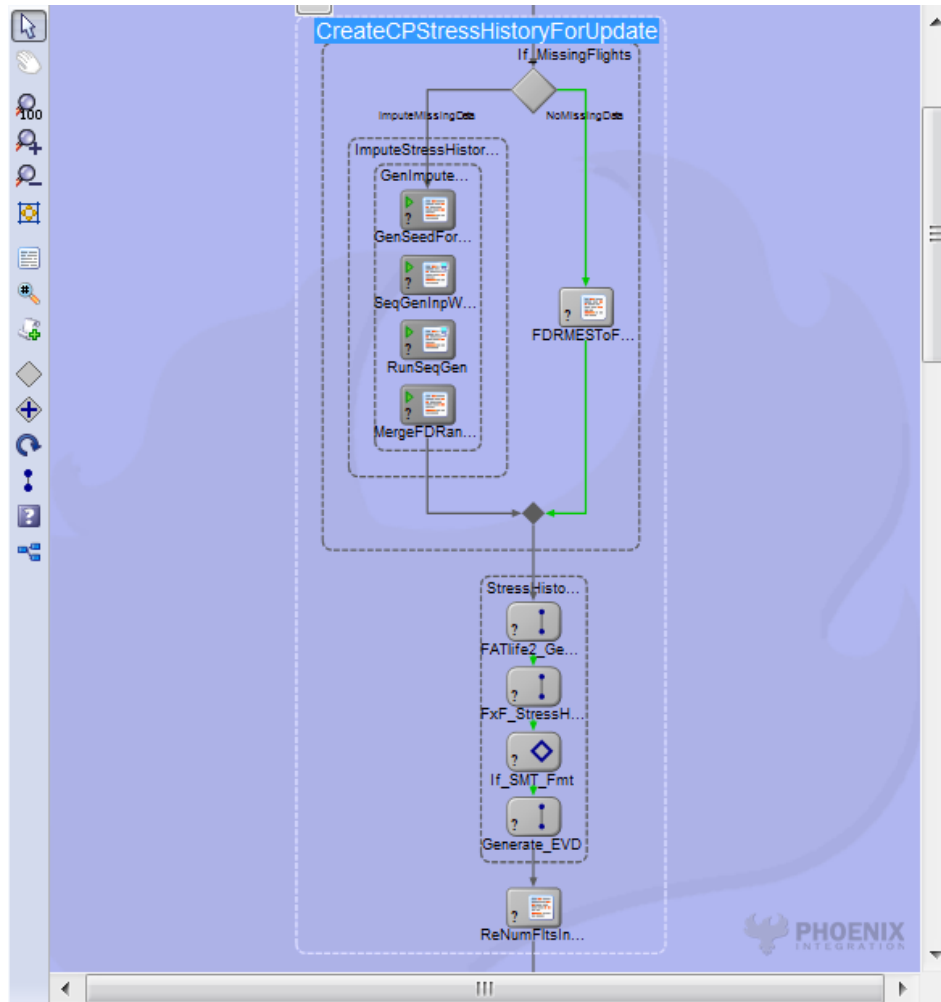


Figure 48. Partially Expanded View of Submodel CreateCPStressHistoryForUpdate

CreateCPStressHistoryForUpdate expands to a large submodel that instantiates several executables needed to create the stress history file for processing the update. Before creating the stress history file, this module utilizes the count of missing flights from *FDRDataToMES* to determine if there is a need to impute data. If there are no missing flights, the MES file from *FDRDataToMES* is used as is by the *FATlife2_GenFxF* module. The *FDRMESToFATlife2_GenFxF* component links the MES file from the *FDRDataToMES* module to the *FATlife2_GenFxF* module. If there are missing flights, the components of the *ImputeStressHistories* submodel update the FDR MES file with imputed data to replace the missing flights. The functions of the first three components of *ImputeStressHistories* are: generate a random seed, create a SeqGen input file based on the maneuver probabilities for the control point being updated and call seqgen2155.exe to randomly generate the flights needed to replace the missing flights. The fourth component, *MergeFDRandImputedMESFiles*, contains source code that merges the imputed data with the FDR data MES file to create the MES file that will be used by the next submodel, *StressHistoryGeneration*.

StressHistoryGeneration is composed of four small submodels. These submodels are not shown fully expanded but each contains a component to call the instantiated executable plus one other component to perform file administration of the outputs generated by the executable. The

first submodel, *FATlife2_GenFxF*, calls the *fatlife2I55n* executable with the final MES file from FDR data to generate a flight by flight file in FxF format. The FxF file is processed by submodel *FxF_StressHistConverter* which calls *fxf_to_other_fmcsMC.exe* to create a spectrum file in the format specified by the user input, *CrackGrowthCodeFormat*. If the spectrum file format is SMART_NASGRO then further formatting of the spectrum is performed by the *If_SMT_Fmt* module which calls *SMARTSpectrumFormatter.exe*. The final module of *StressHistoryGeneration* is *Generate_EVD* which calls *Gen_EVD.exe* to create an EVD file based on the stress history file generated. Module *ReNumFltsInStrHist* ensures that the flight numbers in the stress history have relative flight numbers that start with flight 1.

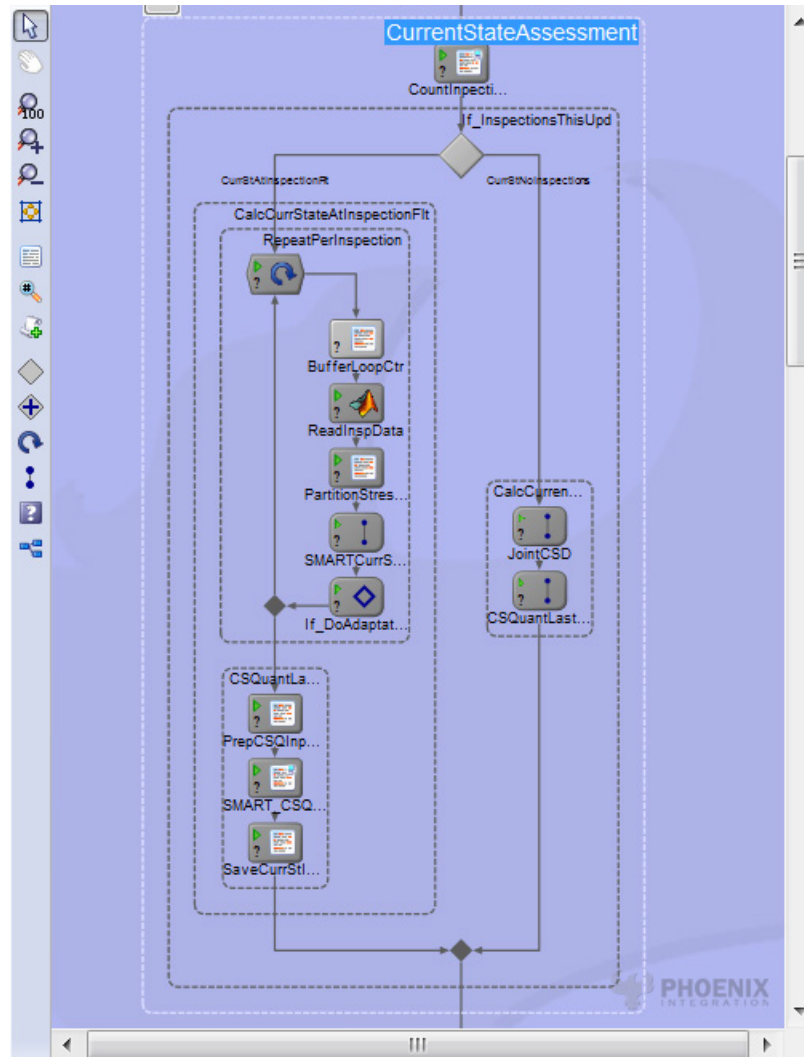


Figure 49. Partially Expanded View of Submodel CurrentStateAssessment

The *CurrentStateAssessment* module also expands to a large submodel. For legibility, the *If_DoAdaptation* module is shown collapsed but will be expanded when described below. The main function of *CurrentStateAssessment* is to predict the size of the crack in both the a and c dimensions at the end of the update period. The probabilistic methods of the SMART|DT executable are used to generate this prediction and produce both a joint crack size distribution (JCSD) as an Excel .csv file and also a textual file which is the crack size distribution expressed

as quantiles (CSQUANT). If there are any inspections during the update period, the model will loop to process the JCSD with the NGC_Adaptation executable which adjusts the JCSD based on the inspection findings.

The first component of the *CurrentStateAssessment* module is *CountInspectionsForUpdatePeriod* which reads the file *InspectionRecord.xlsx*. This file is a record of all inspections for all control points of the vehicle. *CountInspectionsForUpdatePeriod* instantiates a MATLAB executable that determines how many inspections occurred for the control point being processed during the current update period and passes this count to the decision component, *If_InspectionsThisUpd*. If there are no inspections for the control point, the model takes the *CurrStNoInspections* branch and processes the *CalcCurrentState* submodel which calls SMART|DT twice, once by the *JointCSD* module and then by the *CSQuantLastFlt* module. The first call to SMART|DT uses the stress history generated for the update to produce the predicted crack sizes as a JCSD. The second call to SMART|DT produces the crack size distribution as quantiles.

If there are inspections during the update period, the model executes the other branch, *CalcCurrStateAtInspectionFlt* which loops to process each inspection point of the current update. In this path, the component *ReadInspData* which is a MATLAB Plug-In, processes the *InspectionRecord.xlsx* file to extract the inspection findings for the inspection of the current pass of the loop. The VBScript code in the *PartitionStressHistory* component partitions the stress history file so that it ends with the flight of the current inspection point. Then *SMARTCurrStateAssessment* calls SMART|DT with the partitioned stress history to calculate the current state JCSD.

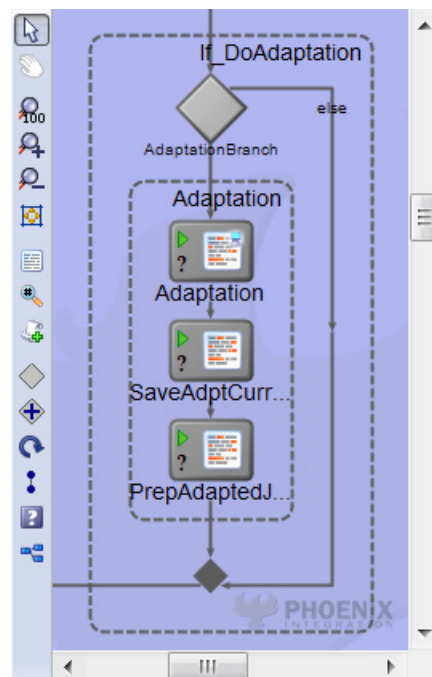


Figure 50. Expanded View of Submodel Adaptation

Next the *If_DoAdaptation* module (expanded above) determines if the last flight of the partition is an inspection point, and if so, the JCSD is passed to the Adaptation module which calls the *NGC_Adaptation* executable to adjust the current state JCSD to account for the results

of the inspection at that flight. If the last flight of the partition is not an inspection point, the *Adaptation* module will be bypassed and the *CSQuantLastFlt* module will be executed to call SMART|DT again to produce the predicted crack size distribution as quantiles. Both paths of *CurrentStateAssessment* produce a JCSD that is utilized by the next module *Future_State_Assessment*. The crack size quantiles file will be incorporated in the update report for the end of the reporting period.

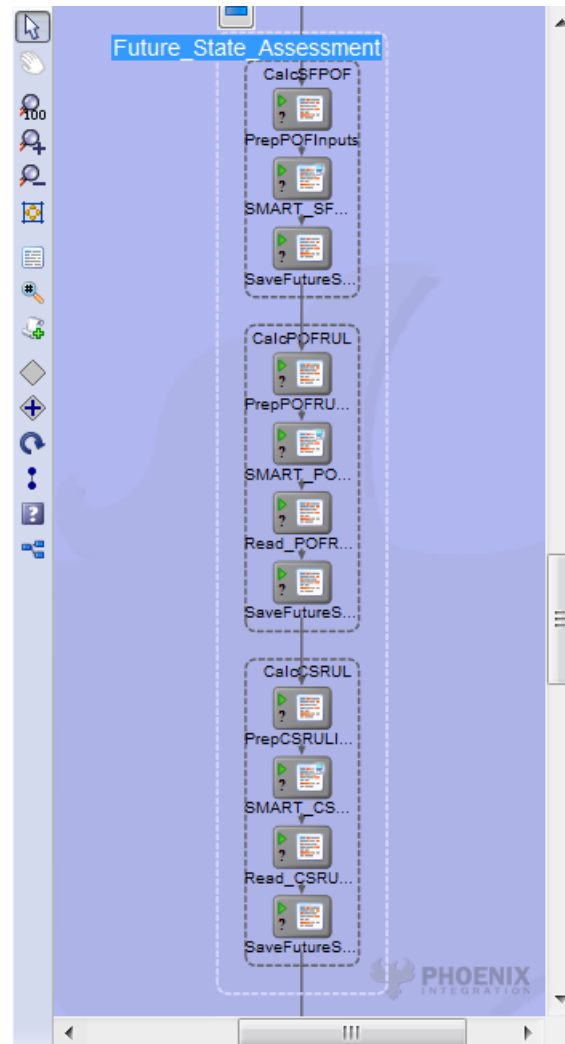


Figure 51. Expanded View of Submodel Future_State_Assessment

Another large but simpler submodel, *Future_State_Assessment* is composed of three submodels each of which calls SMART|DT for a different type of calculation. The first of these submodels is *CalcSFPOF* which like the others, *CalcPOFRUL* and *CalcCSRUL*, contains two or more components written in VBScript that support the calls to SMART|DT by preparing its input files and processing its output files to provide data to the update report and to save all necessary files in the control point specific update folder. The outputs of *CalcSFPOF* are two text files: *control point label_N_POF.cb* and *control point label_N_POF.pof*. Both contain the SFPOF for 1,000 flights into the future. The .pof file is saved for reference but is not used for the report. In addition to SFPOF, the .pof file contains Single Flight Hazard and Cumulative POF. The .cb file

contains the SFPOF values for 1,000 flights in the future plus the upper and lower confidence bounds for SFPOF at each flight number requested.

CalcPOFRUL calls SMART|DT to produce the RUL in flights based on a given POF which is configurable but defaults to 1.0E-7. The output file for this calculation is a text file with a .pofrul extension. The base filename of the .pofrul file is composed of *control point label*, “_N_POFRUL_” and the POF requested for the calculation, e.g., “1p0E-7.” The mean flight for probability of failure-based remaining useful life (POFRUL) *PredictFutureInspections* to calculate the flight at which the next safety inspection must be performed.

The output of the *CalcCSRUL* module is a prediction of remaining useful life in terms of crack size. SMART|DT is provided a requested remediation crack size and it calculates the number of future flights that must be flown to generate a crack of the size requested. The value for remediation crack size is configurable with the default set at 3.0E-02. The output file produced by SMART|DT for *CalcCSRUL* is a text file with a .csrul extension. This file contains the mean number of flights in the future that must be flown to generate a crack of the remediation crack size requested plus a PDF of the cracks per flight numbers. The mean flight for CSRUL is used downstream by the module *PredictFutureInspections* to calculate the flight at which the next inspection for remediation must be performed.

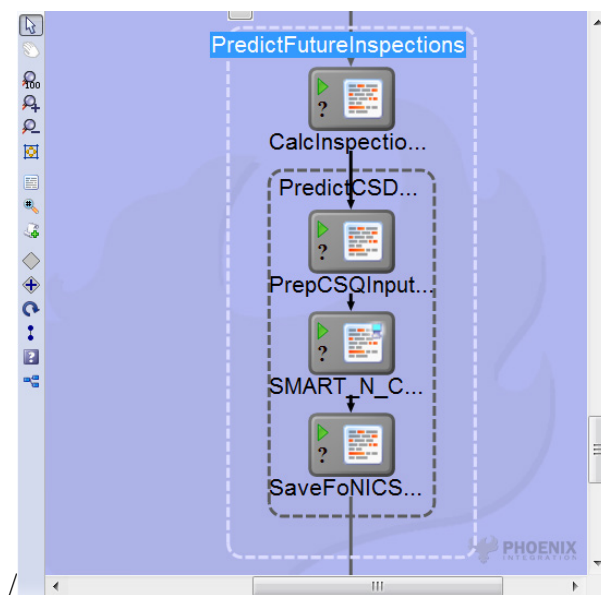


Figure 52. Expanded View of Submodel PredictFutureInspections

Module *PredictFutureInspections* has two main functions. The VBScript component, *CalcInspectionFlt*, uses the mean flight numbers from POFRUL and crack-size-based remaining useful life (CSRUL) to predict the flight for the next safety inspection and the next remediation inspection respectively. This information is used to determine the delta flights and the date for the next inspection. This information along with the reason for the next inspection (remediation or safety) is written to the update report. The second function of *PredictFutureInspections* is performed by module *PredictCSDAtNextFutInspFlt* which is similar to the *CSQuantLastFlt* modules in both paths of the current state calculations. Here it is used to calculate the crack size distribution for the next inspection as quantiles. This information is also written to the update report.

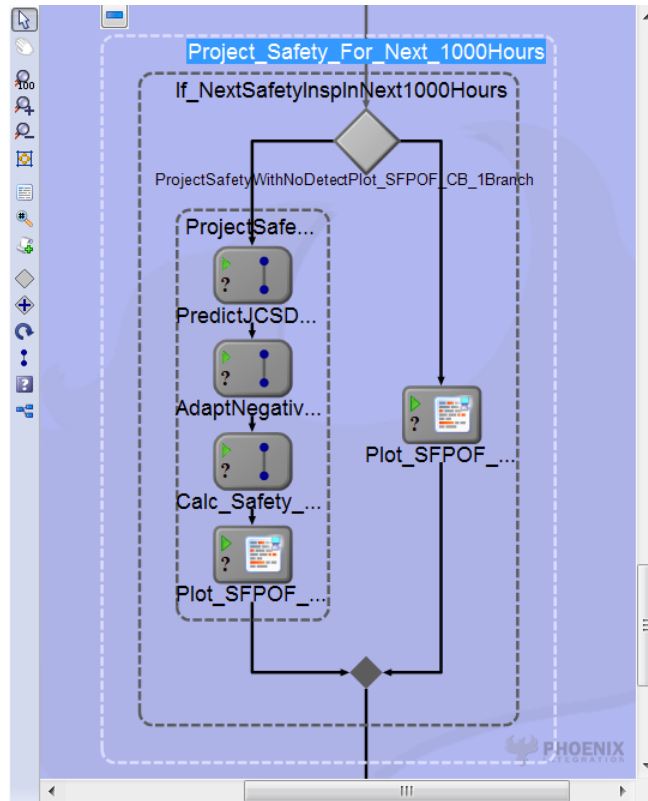


**Figure 53. Partially Expanded Overview of Submodel
Project_Safety_For_Next_1000Hours**

If the result of *PredictFutureInspections* indicates that the next safety inspection is beyond the next 1,000 flight hours, the right branch of module *Project_Safety_For_Next_1000Hours* (shown above) will be executed. Component, *Plot_SFPOF_CB_1*, will generate a plot of the POF curve with its confidence bounds for 1,000 flight hours from the last flight of the update period. This plot is part of the report for the update period.

If the next safety inspection is projected to occur within the next 1,000 flight hours, the left branch of the decision, *ProjectSafetyNoDetect*, will execute. The function of this path is to project the POF curve for the balance of the 1,000 hours from the projected flight for safety inspection using the assumption that an inspection finding was negative. The three modules in this branch that perform this function are: *PredictJCSDAtNextSafetyInsp*, *AdaptNegativeInspection* and *Calc_Safety_SFPOF*. *PredictJCSDAtNextSafetyInsp* runs SMART|DT to predict the JCSD at the flight of the next predicted inspection for safety. *AdaptNegativeInspection* calls NGC_Adaptation with a simulated negative inspection finding. The JCSD adapted for the negative inspection finding is used by *Calc_Safety_SFPOF* as the initial crack size distribution input to SMART|DT which is run to predict the POF curve for the balance of the 1,000 flight hours after the flight for the safety inspection. The fourth component, *Plot_SFPOF_CB_2*, also produces a plot of the POF curve for 1,000 flight hours from the last flight of the update period. To accomplish this it stitches together two POF curves: the original POF curve for 1,000 hours beyond the last flight of the update period and the POF curve for the balance of the 1,000 flight hours beyond the simulated negative inspection for safety.

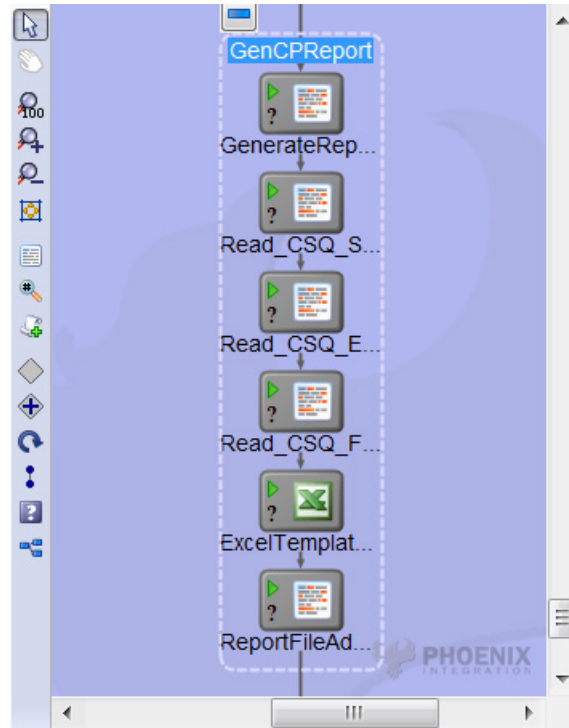


Figure 54. Expanded View of Submodel GenCPReport

As the name implies, submodel *GenCPReport* generates the report for the update period. Each of its six components creates data for the report template. The first component, *GenerateReportDataForCP*, creates the filename for the report file based on the following format: *Vehicle ID* & “_CP_” & *Control Point ID* & “_P2IAT_Report_Update_Flt_” & *Last Flight of Update Period* “.xlsx”. It also performs calculations required to generate the following information for the report: date of report, date of last IAT data, criteria for next inspection, delta flights from last flight of update period until next inspection, flight hours corresponding to the flight of the next inspection and date of the next inspection. Components *Read_CSQ_SoRP* and *Read_CSQ_EoRP* read the data from the .cscdf files from the start and end of the reporting period, *N_CSQUANT_SoRP.cscdf* and *N_CSQUANT_EoRP.cscdf*, respectively. It inserts the data into the report in the sections titled “Predicted Crack Size Distribution at Start of Reporting Period” and “Predicted Crack Size Distribution at End of Reporting Period.” Similarly, component *Read_CSQ_FoNI* reads the data from the .cscdf file calculated at the flight of the next inspection, *N_CSQUANT_FoNI.cscdf*, and inserts it into the report in the section titled “Predicted Crack Size Distribution at Next Maintenance Action.” Component *ExcelTemplateForReport* instantiates the report template in the model so the data from the other four modules can be incorporated into the template via the data links of the model. Finally, component *ReportFileAdmin* copies the populated template file to the control point specific update folder.

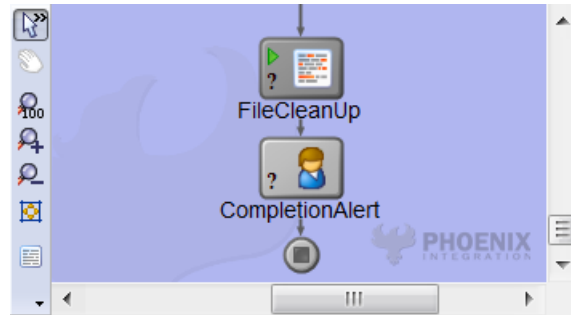


Figure 55. View of FileCleanUp and CompletionAlert Components at end of P2IAT_SingleCP_Version1.pxc

The last two components of the model are *FileCleanUp* which deletes all unnecessary files generated during the update process and *CompletionAlert* which halts the model and displays the following window to the user.

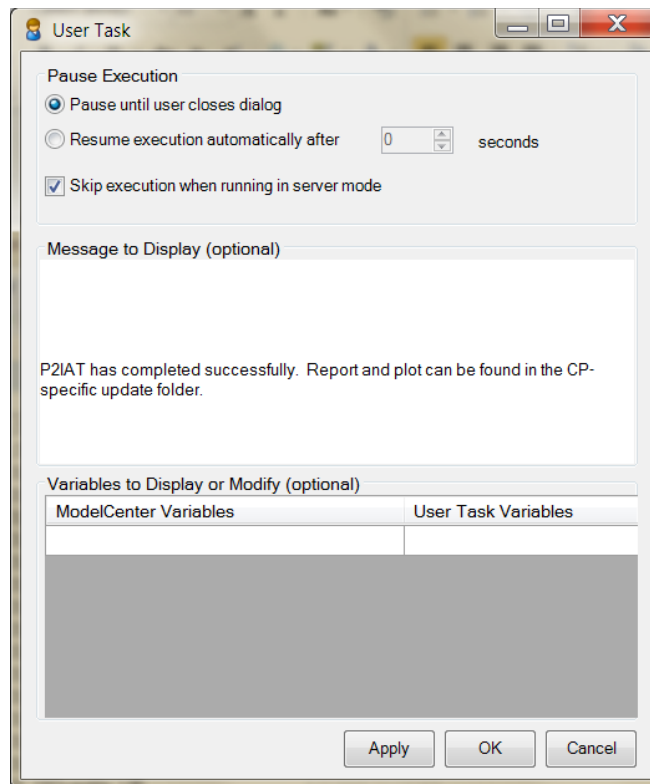


Figure 56. CompletionAlert Dialog Presented to User When Update Has Completed

The user must press one of the buttons at the bottom of the window, *Apply*, *OK* or *Cancel* to complete the execution of the model. Since the model has already finished its processing, it does not matter what button the user chooses to press.

4.1.4 Detailed Description of P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc Process Model

Details of each user input required to run the P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc model are described in the Software

User's Manual. The screenshot below shows the inputs the user must update to perform a recovery of flight data. The parameter names are the black text and the parameter values are in green. After the model has been configured the first time, normally the user will only need to update the bottom three parameters: *Select_Vehicle*, *Select_CP* and *FlightBoundaryOfLastCompletedUpdate*. The settings for *Select_Vehicle* and *Select_CP* are used to populate the file structure as described above. The value entered for *FlightBoundaryOfLastCompletedUpdate* is used to reference the correct file of recovered data. The format for this file name is L_Recovered_FDRData_UpdFlt_#####.xlsx where ##### is the value entered for *FlightBoundaryOfLastCompletedUpdate*. Although the flight number in the file name is padded to 5 digits, the value provided as input *FlightBoundaryOfLastCompletedUpdate* should not be padded with leading zeros.

CrackGrowthCodeFormat	SMART_NASGRO
CRACK_GROWTH_CODE	MASTERC
InitialFlawSize	NoRogueFlaw
Select_Vehicle	L
Select_CP	CP_004
FlightBoundaryOfLastCompletedUpdate	400

**Figure 57. User Input Parameters for ModelCenter Model
P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc**

The screenshot below shows the collapsed version of the P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc process model to highlight the main functional components of the model. Each  component expands into a sequence submodel.

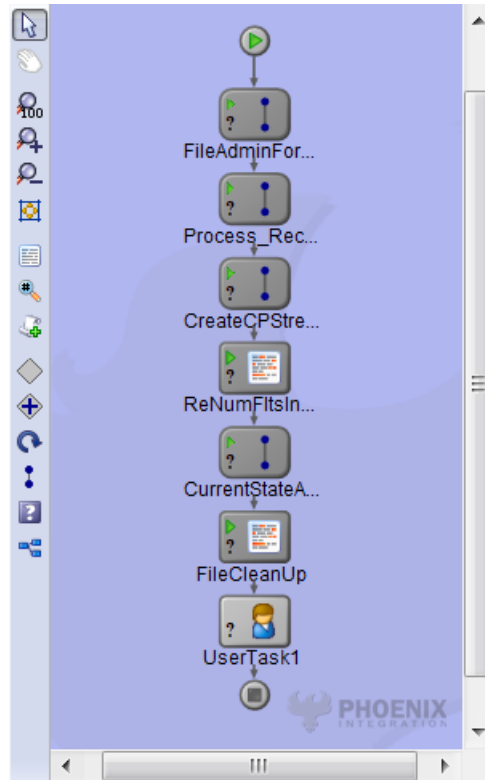


**Figure 58. Collapsed Overview of ModelCenter Model
P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc**

From top to bottom, the main submodels are: *FileAdminForRecovery*, *Process_Recovered_Flts*, *CreateCPStressHistoryForUpdate*, *ReNumFltsInStrHist*, *CurrentStateAssessment*, *FileCleanUp* and *UserTask1* (completion alert). A detailed description of each module follows.

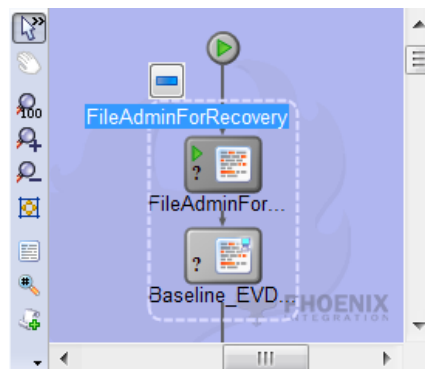


Figure 59. Expanded View of Submodel FileAdminForRecovery

This screenshot shows the expansion of the *FileAdminForRecovery* module into its two components. The first component has the same name as the submodel, *FileAdminForRecovery*. This component locates the appropriate recovered flight data file and copies all required input files to the *Executables* folder. The second component instantiates the data from the *Baseline_EVD.EVD* file in the model for use by other modules. It is a carryover from the

P2IAT_SingleCP_Version_1.pxc model and is not actually needed by the P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc model.

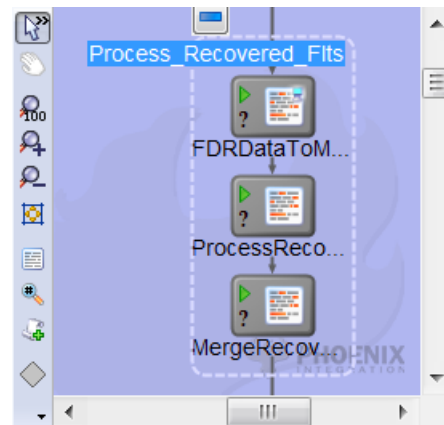


Figure 60. Expanded View of Submodel Process_Recovered_Flts

Process_Recovered_Flts is a three component submodel that processes the recovered flight data file. The first component is the FDRDataToMES executable which converts data from the recovered data from the FDR to a Master Event Sequence file. The next module *ProcessRecoveredData* is VBScript code that analyzes the MES file of recovered data as well as the following files: L_Missing_Flt_Nums.xlsx, L_ReRun_Flt_Nums.xlsx, L_QuarterlyUpdateFlightBoundaries.xlsx and L_RecoveryRecord.xlsx. It determines the first and last flight numbers in the recovered data, removes the recovered flights from the L_Missing_Flt_Nums.xlsx file and updates file L_RecoveryRecord.xlsx. It uses the rerun points in L_RecoveryRecord.xlsx to determine which FDR data files to concatenate with the current recovery data to create a stress history that encompasses the first rerun point through the FlightBoundaryOfLastCompletedUpdate. The third component, *MergeRecovMESWithLastUpdMES*, merges the MES of recovered flight data with the composite MES of all updates back to earliest recovery point, replacing the imputed data from previous update periods with the recovered data. This composite MES will be utilized by the next submodel, *CreateCPStressHistoryForUpdate*.

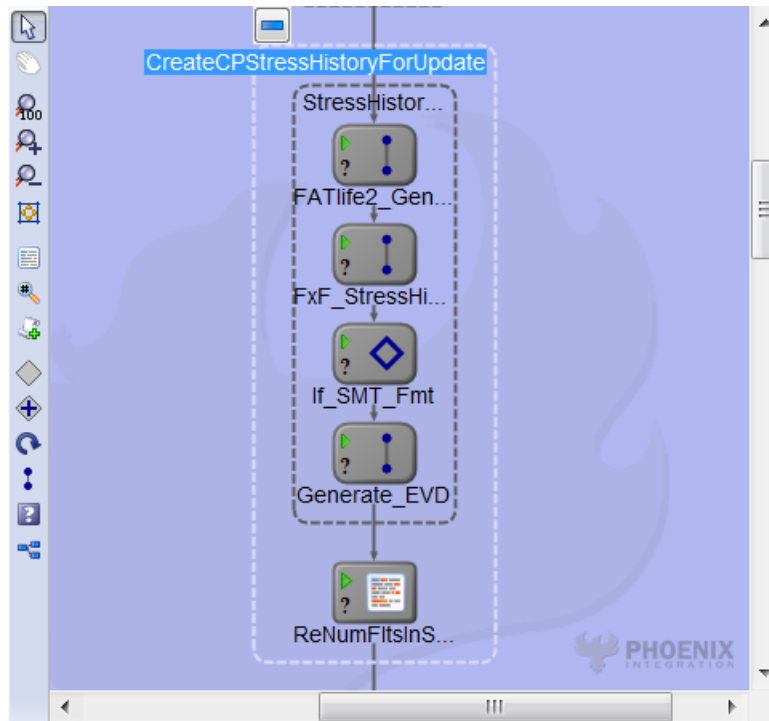


Figure 61. Partially Expanded View of Submodel CreateCPStressHistoryForUpdate for Recovered Data

CreateCPStressHistoryForUpdate is similar to the submodel of the same name in the P2IAT_SingleCP_Version_1.pxc model. The difference is that the submodel that creates imputed data if there are missing flights is not needed in the recovery model. The recovery model version of *CreateCPStressHistoryForUpdate* is composed of only the submodels *StressHistoryGeneration* and *ReNumFltsInStrHist* which are identical to those same submodels in P2IAT_SingleCP_Version_1.pxc. For continuity the description from P2IAT_SingleCP_Version_1.pxc is repeated below.

StressHistoryGeneration is composed of four small submodels. These submodels are not shown fully expanded but each contains a component to call the instantiated executable plus one other component to perform file administration of the outputs generated by the executable. The first submodel, *FATlife2_GenFxF*, calls the fatlife2I55n executable with the final MES file from FDR data to generate a flight by flight file in FxF format. The FxF file is processed by submodel *FxF_StressHistConverter* which calls fxf_to_other_fmcsMC.exe to create a spectrum file in the format specified by the user input, *CrackGrowthCodeFormat*. If the spectrum file format is SMART_NASGRO then further formatting of the spectrum is performed by the *If_SMT_Fmt* module which calls SMARTSpectrumFormatter.exe. The final module of *StressHistoryGeneration* is *Generate_EVD* which calls Gen_EVD.exe to create an EVD file based on the stress history file generated. Module *ReNumFltsInStrHist* ensures that the flight numbers in the stress history have relative flight numbers that start with flight 1.

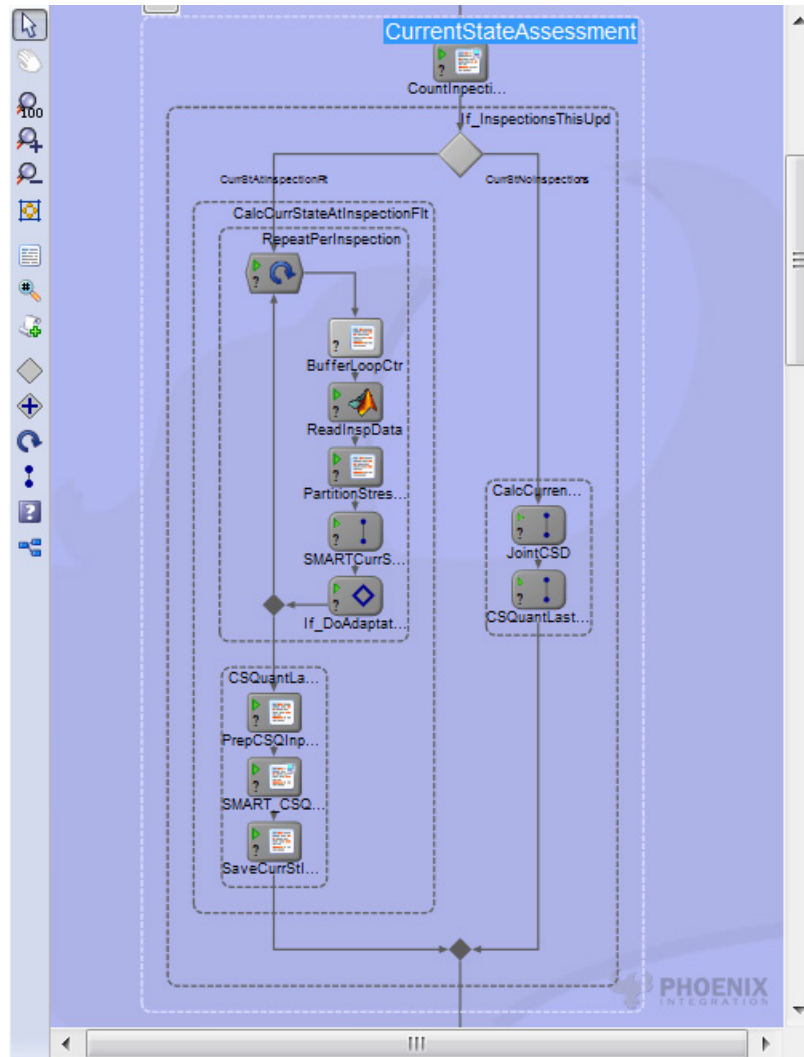


Figure 62. Partially Expanded View of Submodel CurrentStateAssessment for Recovered Data

The *CurrentStateAssessment* is identical to the submodel of the same name in model, P2IAT_SingleCP_Version_1.pxc. For continuity the description from P2IAT_SingleCP_Version_1.pxc is copied below.

The *CurrentStateAssessment* module also expands to a large submodel. For legibility, the *If_DoAdaptation* module is shown collapsed but will be expanded when described below. The main function of *CurrentStateAssessment* is to predict the size of the crack in both the a and c dimensions at the end of the update period. The probabilistic methods of the SMART|DT executable are used to generate this prediction and produce both a JCSD as an Excel .csv file and also a textual CSQUANT file which is the crack size distribution expressed as quantiles. If there are any inspections during the update period, the model will loop to process the JCSD with the NGC_Adaptation executable which adjusts the JCSD based on the inspection findings.

The first component of the *CurrentStateAssessment* module is *CountInspectionsForUpdatePeriod* which reads the file *InspectionRecord.xlsx*. This file is a record of all inspections for all control points of the vehicle. *CountInspectionsForUpdatePeriod*

instantiates a MATLAB executable that determines how many inspections occurred for the control point being processed during the current update period and passes this count to the decision component, *If_InspectionsThisUpd*. If there are no inspections for the control point, the model takes the *CurrStNoInspections* branch and processes the *CalcCurrentState* submodel which calls SMART|DT twice, once by the *JointCSD* module and then by the *CSQuantLastFlt* module. The first call to SMART|DT uses the stress history generated for the update to produce the predicted crack sizes as a JCSD. The second call to SMART|DT produces the crack size distribution as quantiles.

If there are inspections during the update period, the model executes the other branch, *CalcCurrStateAtInspectionFlt* which loops to process each inspection point of the current update. In this path, the component *ReadInspData* which is a MATLAB Plug-In, processes the *InspectionRecord.xlsx* file to extract the inspection findings for the inspection of the current pass of the loop. The VBScript code in the *PartitionStressHistory* component partitions the stress history file for the so that it ends with the flight of the current inspection point. Then *SMARTCurrStateAssessment* calls SMART|DT with the partitioned stress history to calculate the current state JCSD.

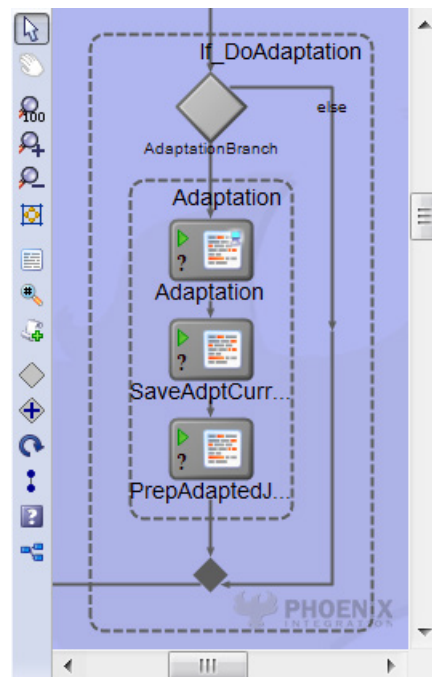


Figure 63. Expanded View of Submodel Adaptation for Recovered Data

Next the *If_DoAdaptation* module (expanded above) determines if the last flight of the partition is an inspection point, and if so, the JCSD is passed to the *Adaptation* module which calls the *NGC_Adaptation* executable to adjust the current state JCSD to account for the results of the inspection at that flight. If the last flight of the partition is not an inspection point, the *Adaptation* module will be bypassed and the *CSQuantLastFlt* module will be executed to call SMART|DT again to produce the predicted crack size distribution as quantiles. Both paths of *CurrentStateAssessment* produce a JCSD that is utilized in the next update period as the initial JCSD and a crack size quantiles file that will be used as the quantiles data for the start of the reporting period in the update report for the next update period.

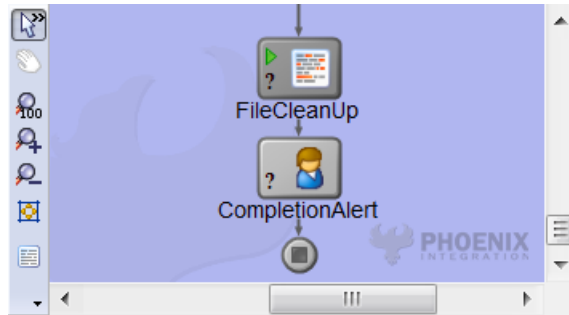


Figure 64. View of FileCleanUp and CompletionAlert Components at end of P2IAT_SingleCP_Flight_Data_Recovery_Version_1.pxc

The last two components of the model are *FileCleanUp* which deletes all unnecessary files generated during the execution of the recovery model and *CompletionAlert* which halts the model and displays the following window to the user.

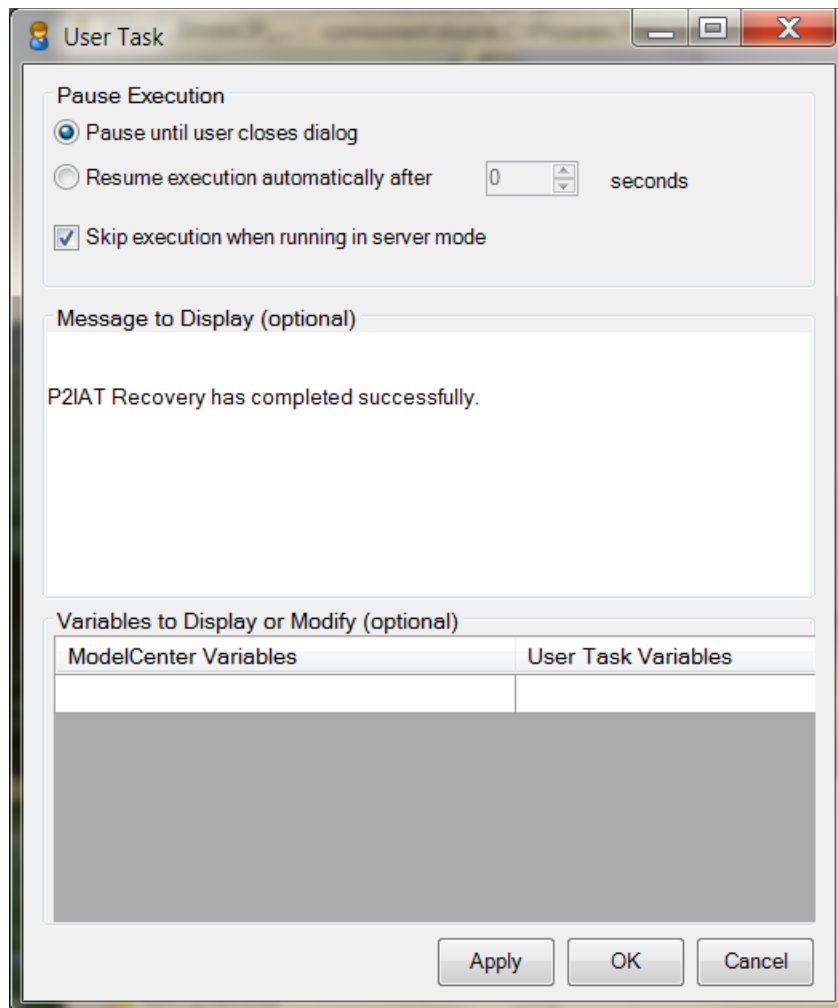


Figure 65. CompletionAlert Dialog Presented to User When Data Recovery Processing Has Completed

The user must press one of the buttons at the bottom of the window, *Apply*, *OK* or *Cancel* to complete the execution of the model. Since the model has already finished its processing, it does not matter what button the user chooses to press.

4.2 Single Location Demonstration

The innovative capabilities of the P²IAT process were incrementally demonstrated on September 19, 2016 to AFRL on a single CP. Ten control points were selected in the F-15C outer wing, under Task Order 0002. The description of the selection process will be covered under the Task Order 0002 final report. CP NG4 located on the rear spar lower flange at XW 167.3 was selected for the demonstration, shown in Figure 66. This section describes the process of adding a CP to P²IAT and the inputs associated with CP NG4 as related to the demonstration.

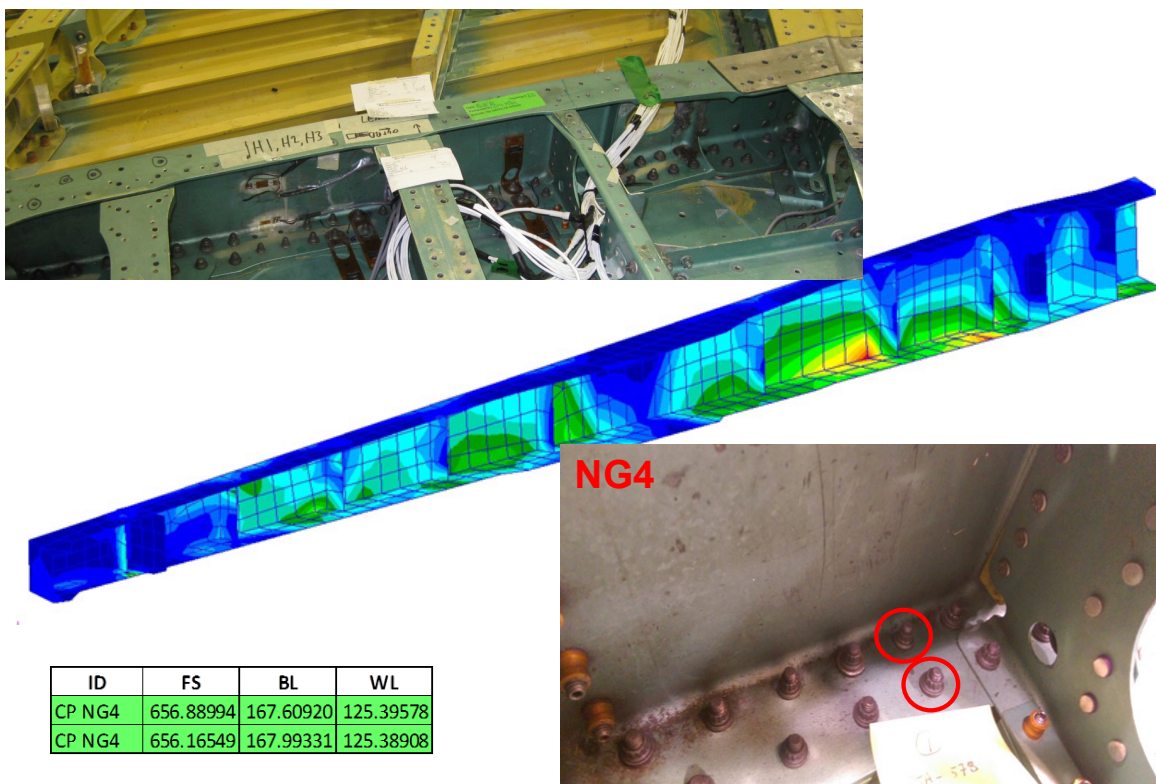


Figure 66. Selected CP NG4 for P²IAT Demonstration

The key tasks in adding a CP are:

- Develop fatigue crack growth models based on the local geometry and loading
- Develop probabilistic inputs
- Develop local stress history

The main attributes of the P²IAT framework are:

- Fatigue crack growth models are consistent with current deterministic modeling approaches
- User choice on model complexity from simple analytical solutions (conventional IAT) to 3-D

crack growth modeling

- P2IAT includes world-wide accepted fatigue crack growth codes: FASTRAN, AFGROW, and NASGRO
- Other crack growth lifing codes can be added within the SMART framework by developing input/output wrappers
- Probabilistic inputs include: geometry, crack growth rate data, material parameters, initial crack size, initial aspect ratio, peak stress, EVD
- Probabilistic framework is not depended on the choice of FCG model
- P2IAT ModelCenter framework allows local stress spectra at CPs to be defined automatically by specifying the element id and the component of stress

Figure 67 shows the overall process in adding a CP to P²IAT. The development of the CP stress histories from FDR data and the process for gap filling and recovering missing flight data is described in Sections 3.1 and 4.0.

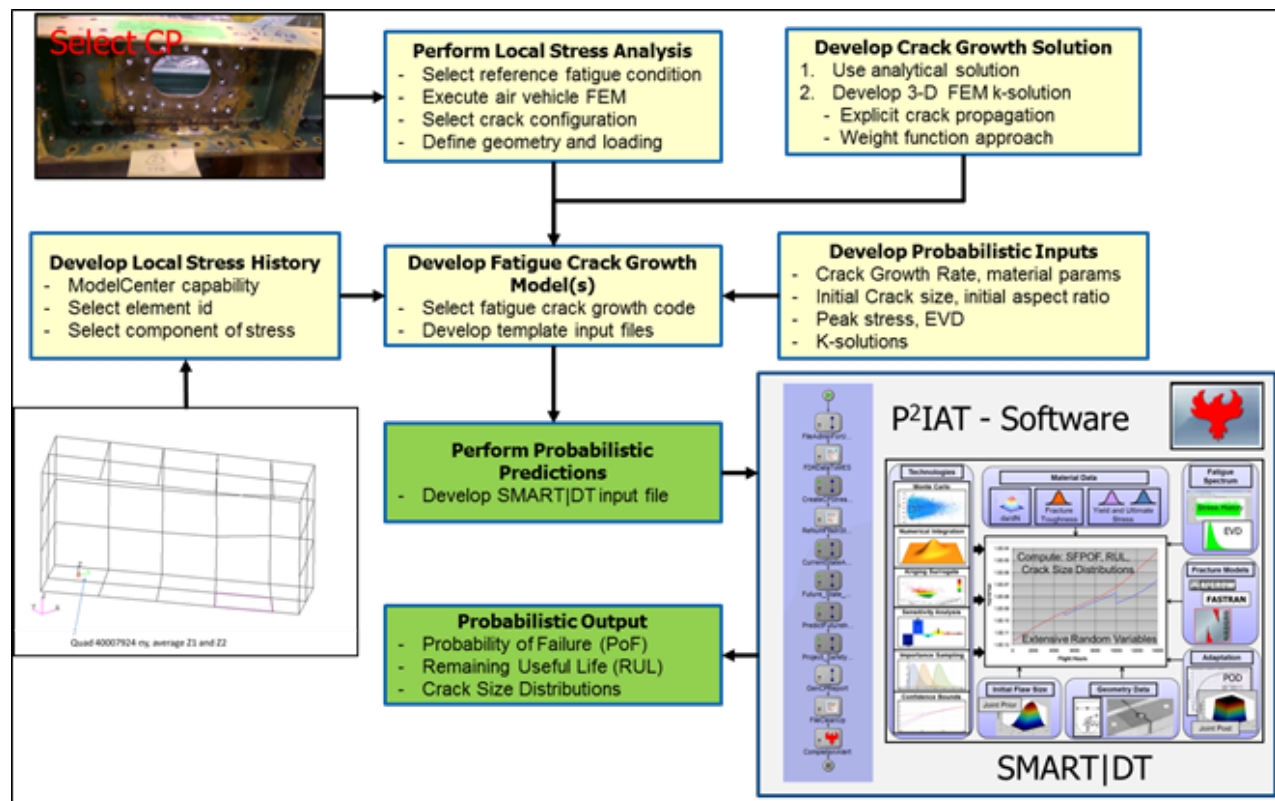


Figure 67. Adding a Control Point to P²IAT

4.2.1 Local Stress Analysis

A summary of the local stress analysis is shown in Figure 68. The NASGRO code was selected to perform the probabilistic crack growth life predictions within the P²IAT-SMART|DT

framework. The selected crack configuration corresponds to a corner crack at a hole based on the Fawaz-Anderson CC16 solution. CC16 is a new stress intensity solutions based on recent finite element results. The stress analysis was performed using a reference fatigue condition corresponding to a symmetric 20 Kft, 1.05 M, 8.4 g maneuver. The F-15C outer wing model was exercised and a local stress analysis was performed to develop the crack growth loading parameters, S0 (bypass stress), and the S3 (bearing stress). The details of the stress analysis will be documented under the Task Order 0002 final report. The stress analysis produced all the required deterministic parameters associated with the local geometry at CP NG4 as shown in Figure 68.

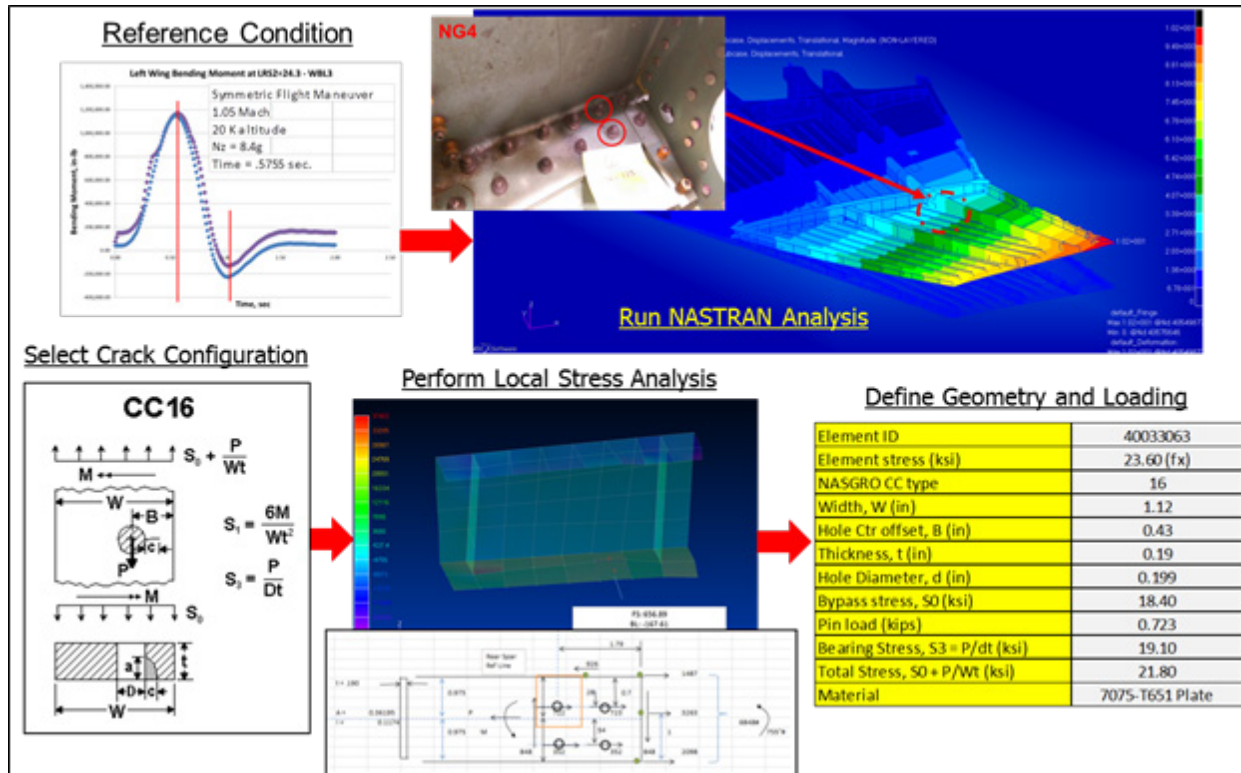


Figure 68. CP NG4 Local Stress Analysis

The NASGRO template file (Figure 69) was generated based on the values obtained from the stress analysis. The template file is utilized in the P²IAT-SMART|DT framework and is updated based on sampling the random variables. The available random variables are:

- Initial Crack Size
- Aspect Ratio
- Initial Crack in both directions
- da/dn
- Fracture Toughness
- Yield Stress, Ultimate Stress

- Peak Stress
- Hole Diameter, Hole Offset

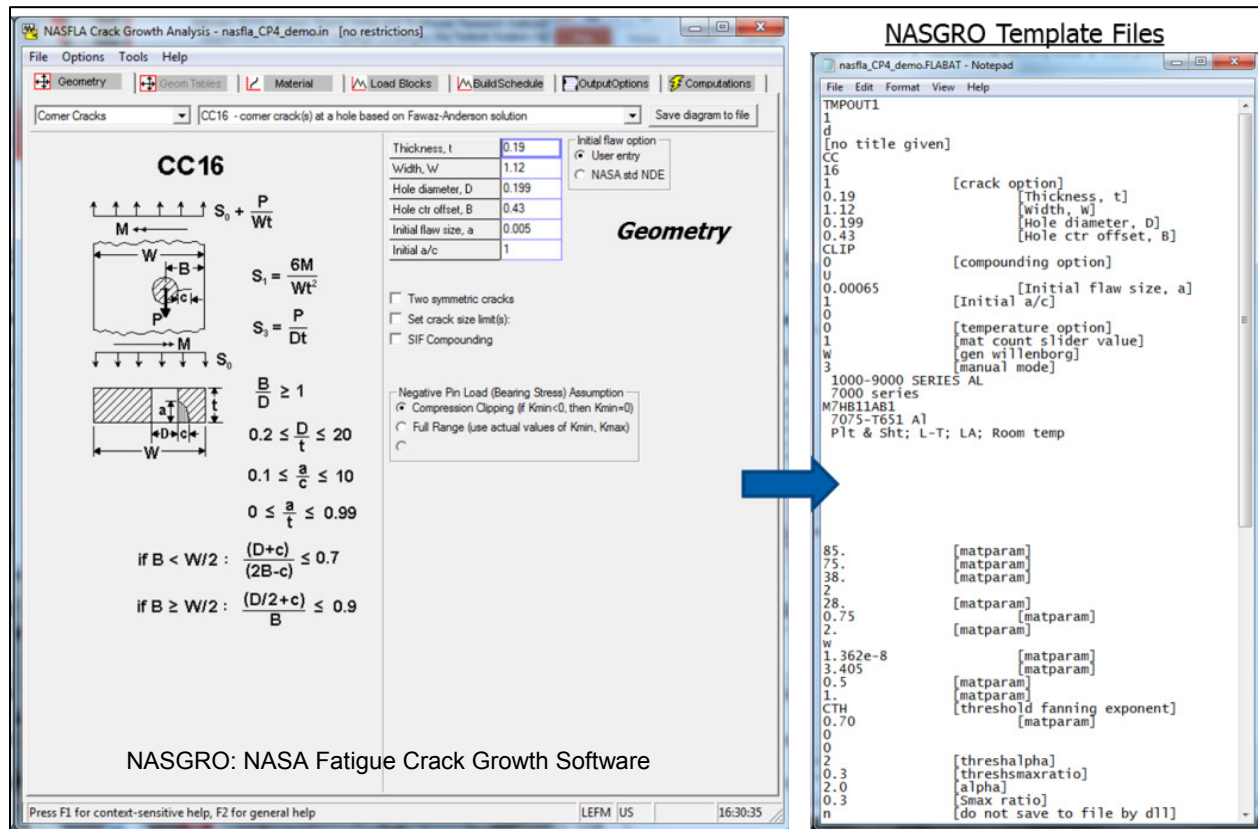


Figure 69. NASGRO Template File

The demonstration parameters are shown in Table 2.

Table 2. Demonstration Parameters

Variables	Default Values in Template File	Units	Description	Distribution Type	Param 1	Param 2	Param 3
ai	0.005	in	Initial Crack size - Shifted from SIPS Weibull distribution, tabular input	Weibull	scale=0.00019843	shape = 2.3	
aovc	1.00	-	Aspect ratio - a/c	uniform	min=0.75	max=1.25	
Paris_C	1.3620E-08	7075-T651 Plate	Crack-growth coefficient - C	binormal	mean of log10c = -7.865783	std = 0.03705	rho=-0.8646
Paris_n	3.405	7075-T651 Plate	Crack-growth power term - n	binormal	mean = 3.405	std = 0.07494	
Fract_Tough	50.0	ksi-in ^{0.5}	Plain strain fracture toughness - K1c	normal	mu = 28.7294	sigma = 1.8034	
Sy	75.4	Ksi	Yield stress	constant	min=71.63	max=79.17	
Sult	83.4	Ksi	Ultimate tensile strength	constant	min=79.23	max=87.57	
E	1.040E+07	Ksi	Elastic Modulus	constant			
S0	18.40	Ksi	Peak bypass stress for EVD	uniform	min=17.48	max=19.32	
S3	19.10	Ksi	Peak bearing stress	constant			
Hole_Diam	0.199	in	Diameter	constant			
Width	1.120	in	Width	constant			
Thickness	0.190	in	Thickness	constant			
Hole_offset	0.430	in	hole offset	constant			

The initial crack size distribution (Figure 70) was developed from one generated on a previous program for aluminum material where the failure initiates at constituent particles. The

original distribution which had a median crack size of 4 microns was shifted to the right to produce a median size of 31.7 microns. The F-15C left wing selected for the full-scale experiment was retired at 5,138 FHs. The distribution was shifted to produce a predicted SFPOF at retirement of 1.0×10^{-7} , as shown in Figure 76.

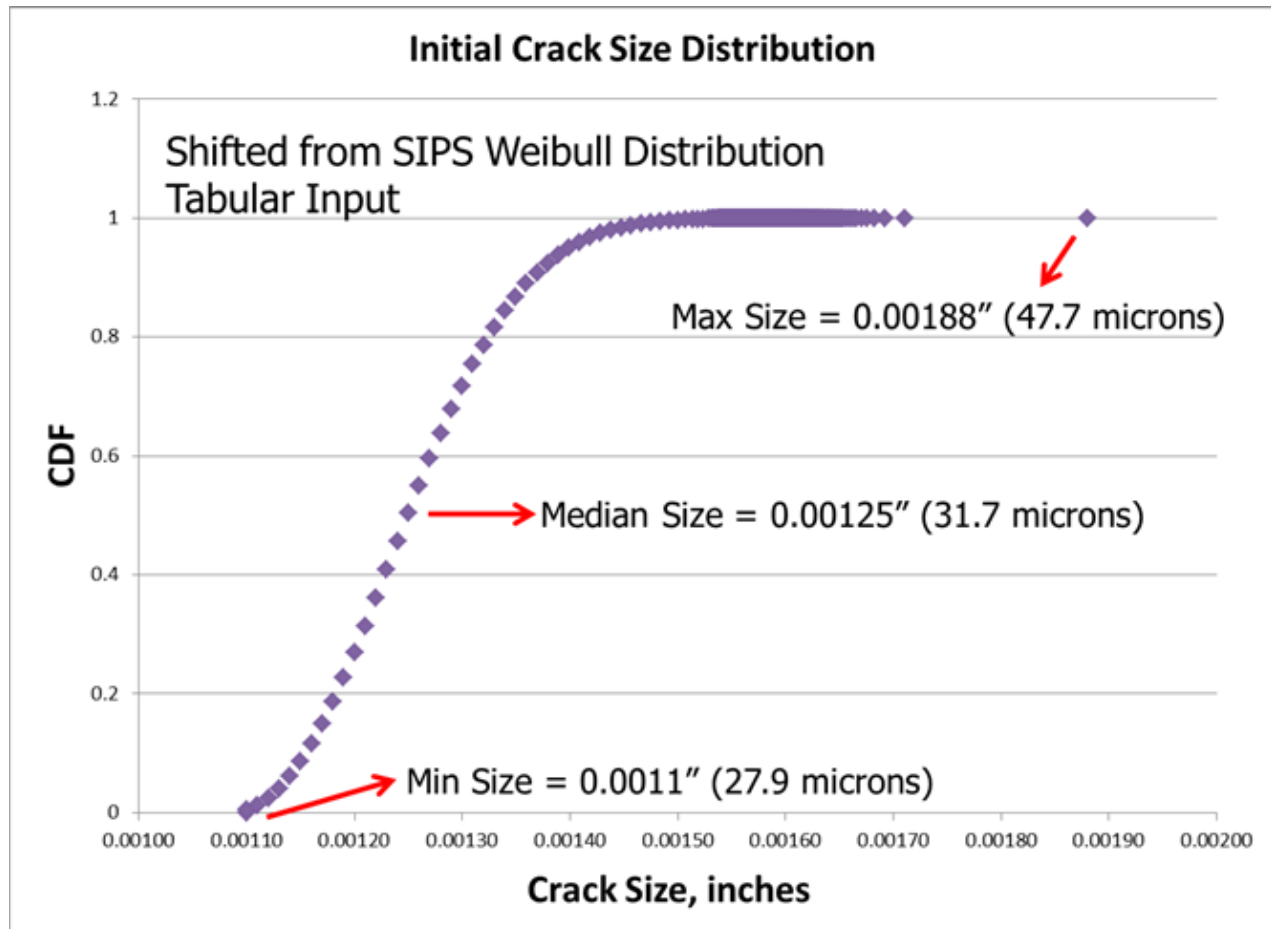


Figure 70. Initial Crack Size Distribution

Material statistical parameters were developed for AA7075-T651 plate and utilized for the demonstration. The actual material for CP NG4 is AA7075-T7352 forging but lack of crack growth rate data deemed unacceptable in performing a statistical analysis. A crack growth rate statistical model was developed to capture statistical uncertainty of crack growth rate data at the Paris, threshold, and fracture regimes. For the demonstration the statistical parameters at the Paris regime were utilized and shown in Figure 71.

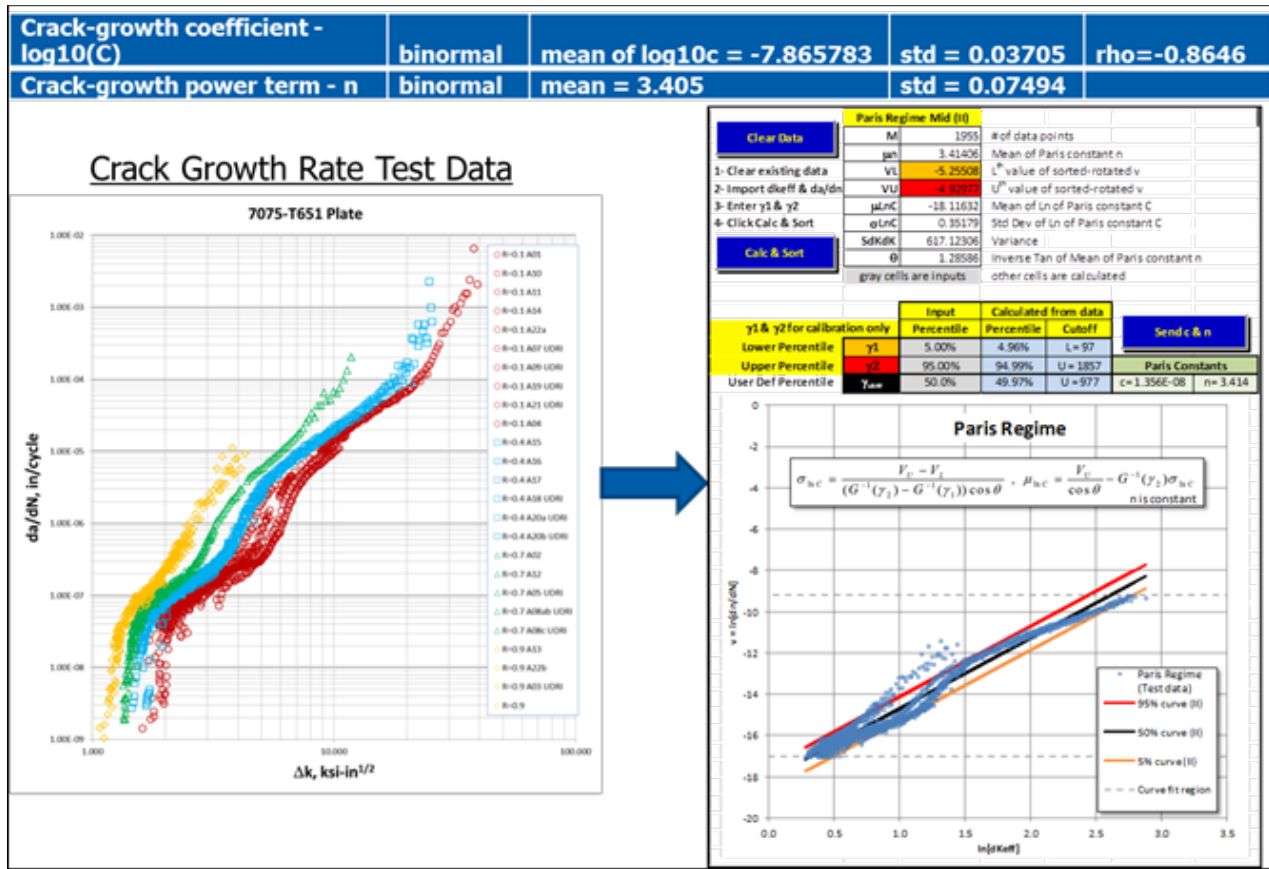


Figure 71. Statistical Parameters for the Paris Regime of AA7075-T651 Plate

Fracture toughness test data for AA7075-T651 plate material were previously generated on another program and were fitted to a normal distribution as shown in Figure 72.

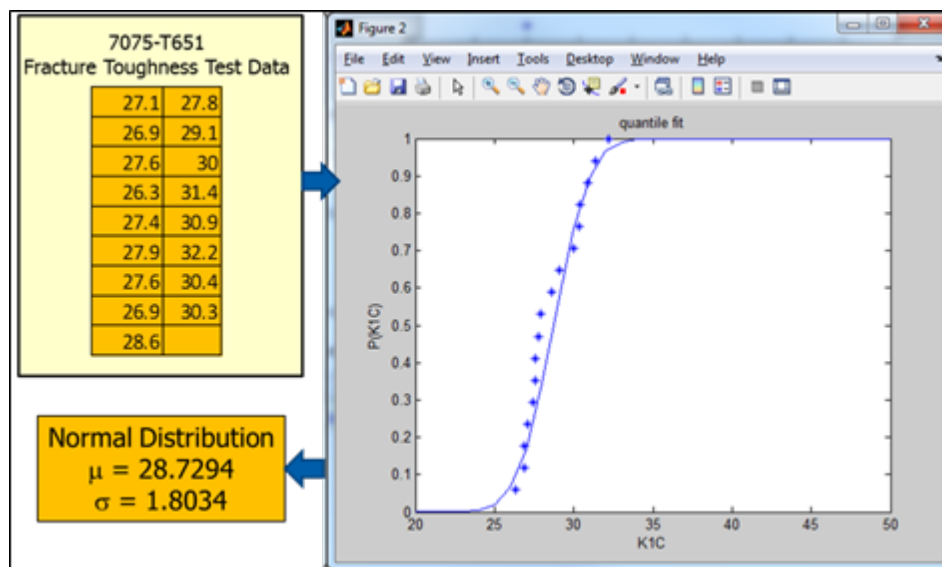


Figure 72. Fracture Toughness Statistical Parameters for AA 7075-T651 Plate

The SMART|DT input file (Figure 73) was developed utilizing all the parameters and probabilistic predictions were performed to verify the input variables and compare the SMART|DT Master Curve and Kriging probabilistic options.

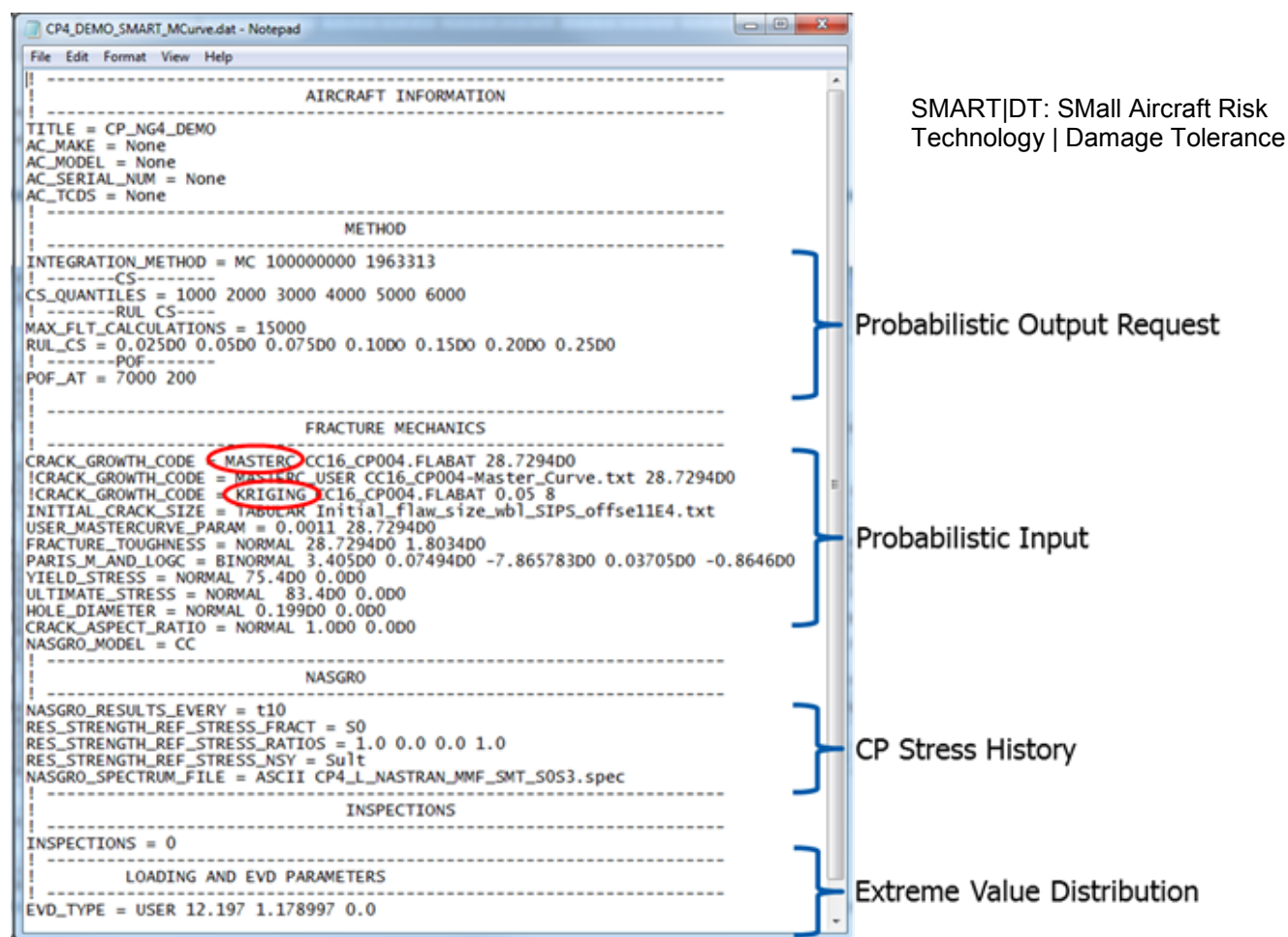


Figure 73. SMART|DT Input File

SMART|DT probabilistic predictions for crack size quantiles (Figure 74), crack size remaining useful life (Figure 75), and SFPOF (Figure 76) are compared between the Master Curve and Kriging options. The baseline spectrum at NG CP4 was utilized. The master curve uses three random variables: initial crack size, fracture toughness, and EVD. Probabilistic predictions with Kriging were performed using three, five, and eight random variables. The results in Figure 74 are expanded at each flight count for clarity. The Master curve option, which computationally is more efficient, and Kriging with three RVs produce results that are comparable. Kriging results show that as the RVs increase so does the resulting uncertainty in the output distributions.

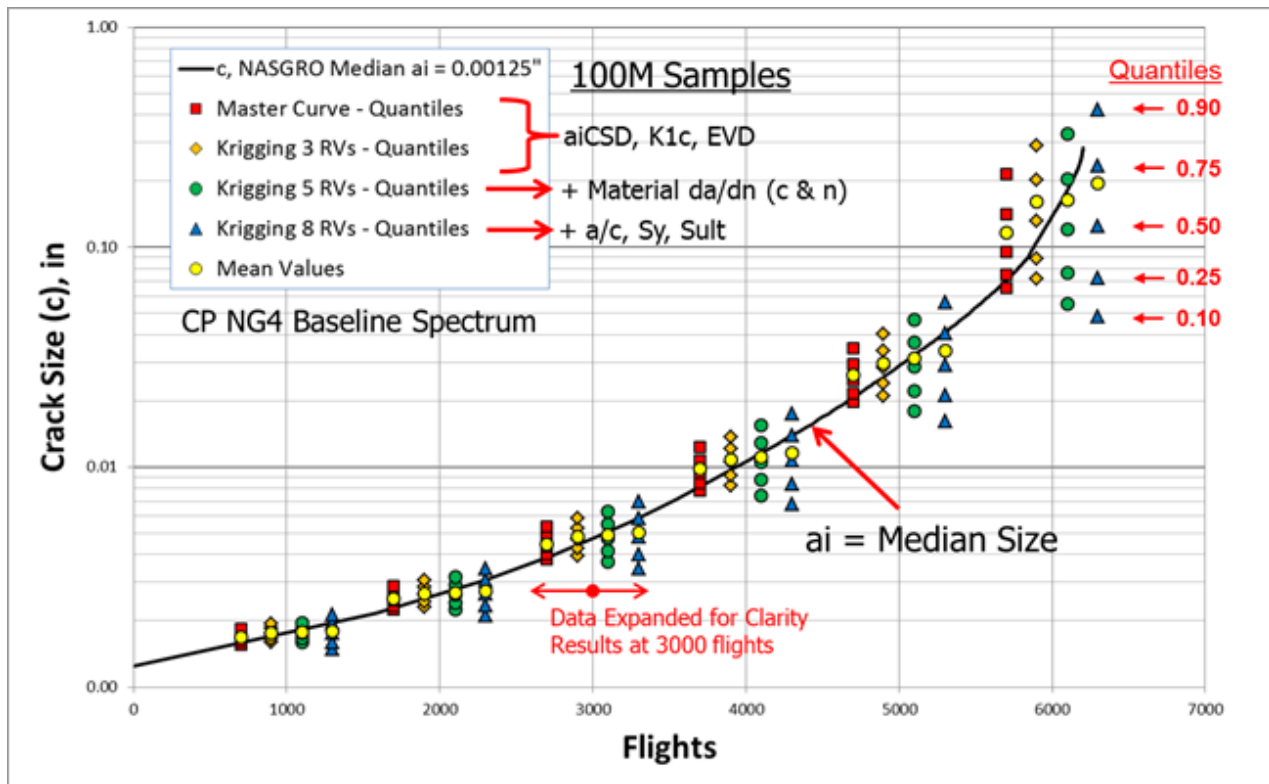


Figure 74. SMART|DT Crack Size Quantiles Predictions - Master Curve vs. Kriging

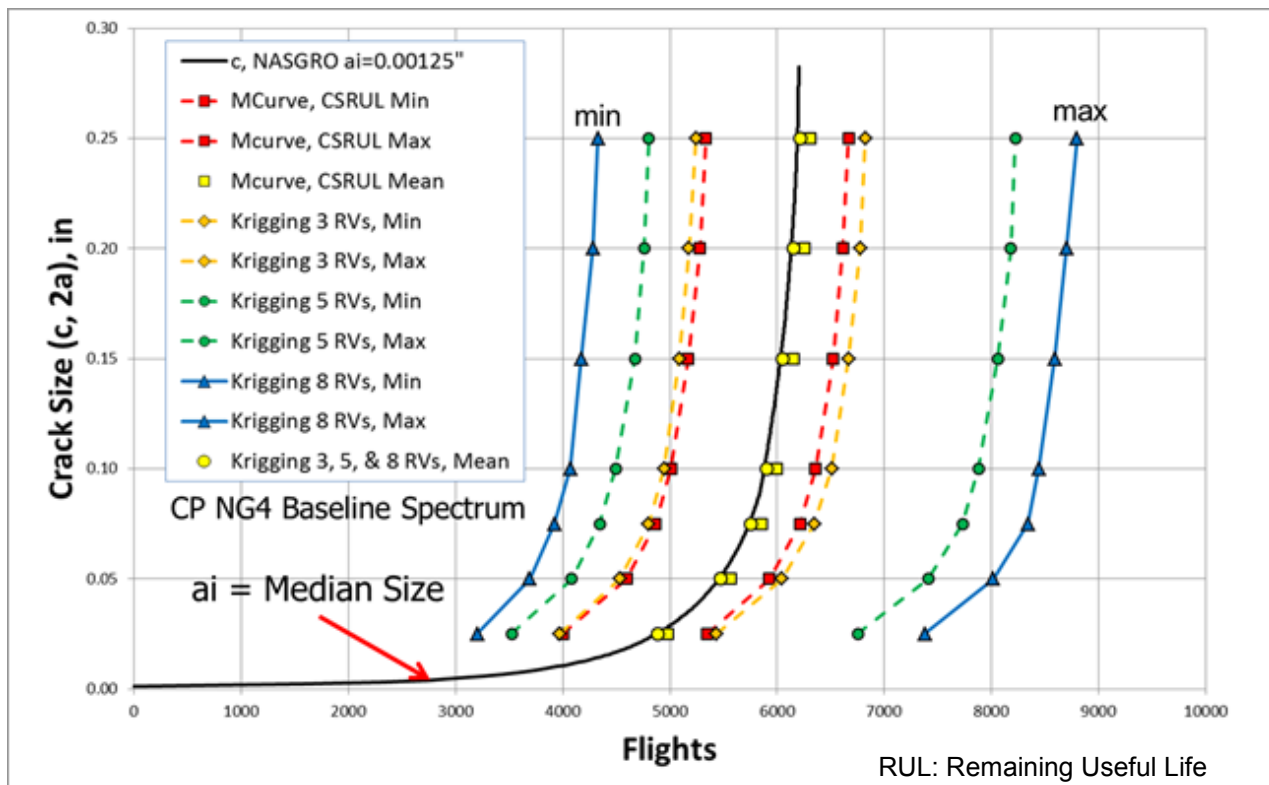


Figure 75. SMART|DT Crack Size RUL Predictions - Master Curve vs. Kriging

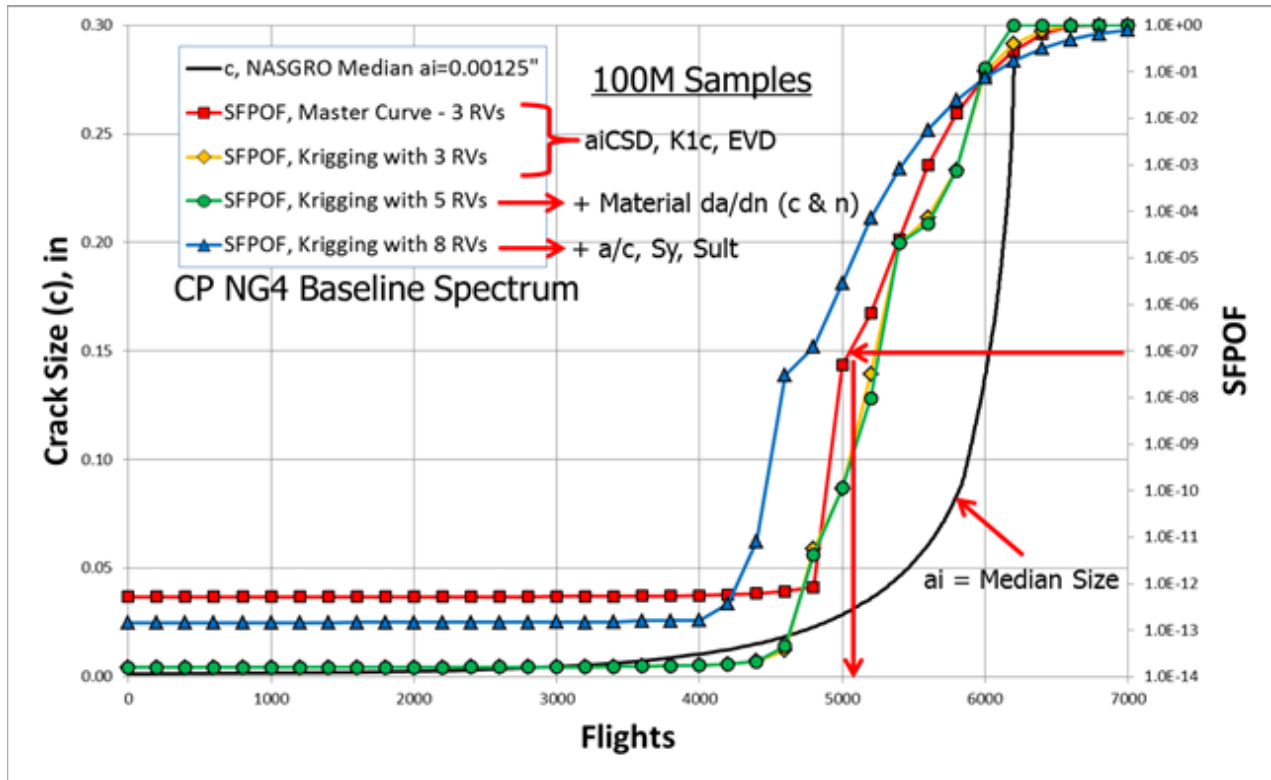


Figure 76. SMART|DT SFPOF Predictions - Master Curve vs. Kriging

4.3 Demonstration Results

On September 19, 2016, the P²IAT process as discussed above was demonstrated to AFRL. In normal fleet operations, data are received on a quarterly basis. For our demonstration, we simulated approximately 200 flights in an update to reflect the likely number of flights that would run in the Task Order 0003 full-scale test. The control point selected for this demonstration was Outer Wing Rear Spar Lower Flange XW 167.3. The following six scenarios were demonstrated:

Scenario 1: Flight numbers 1 through 200 with no gaps and no inspections comprised the first update. This scenario was used to walk through the basic ModelCenter flow from FDR data processing to the required P²IAT outputs. Simulations were paused during this scenario to explain the operational details as discussed above.

The swim-lane diagram in Figure 77 was used to explain the interaction between operator, ModelCenter and SMART | DT.

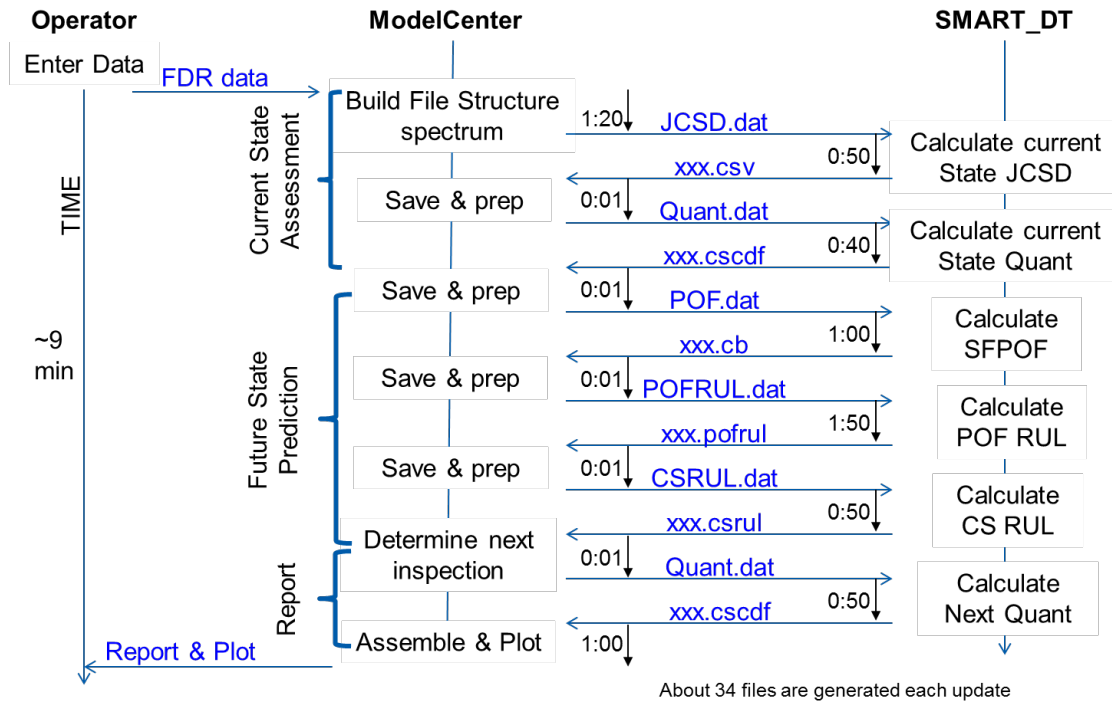


Figure 77. P²IAT Process Swim Lane Diagram for Scenario 1

Scenario 2: Flight numbers 201 through 400 with segments of missing data in the beginning, end and middle were added to the results from scenario 1. The outputs were compared to a previously run case with no gaps to examine the differences between imputed data and no missing data. Results are shown in Figure 78.

End of Reporting Period									
90%		75%		50%		25%		10%	
0.001526099		0.001464927		0.001405859		0.001356467		0.001321177	
0.001525772	0.001526519	0.001464688	0.001465172	0.00140566	0.001406061	0.00135628	0.001356646	0.00132098	0.001321343
↓ No gap result (mean) ↓									
0.001526337		0.001465103		0.001405982		0.001356687		0.001321445	
-2.38265E-07		-1.7606E-07		-1.227E-07		-2.195E-07		-2.68E-07	
Difference between no gap and imputed result									

Figure 78. Gap-Filled and No-Gap Comparative Results

Scenario 3: Flights 401 through 600 were added with no gaps. The remediation crack size was changed to 0.05 inch and the time to reach this size with a 0.95 probability was predicted. This demonstrated the ability to predict a crack would occur having a 0.90 POD with a 95% confidence. The point where this would occur was predicted to be flight 5,851.

Scenario 4: Flights that were missing in the 201 to 400 block were recovered. The resulting sequence of maneuvers was compared to the run with no missing flights. The results were identical.

Scenario 5: This scenario demonstrated the effect of combining a rogue flaw distribution with the standard initial flaw size distribution attributed to the material microstructure. For this illustration, we made the following assumptions:

- Probability of a rogue flaw is 10^{-3} wherein the distribution of sizes is normally distributed with a mean of 0.04 inch and a stand deviation of 0.003 (see Figure 79)
- The upper bound on the SFPOF is 10^{-7} and that criteria is the trigger point for every inspection driven by safety⁸
- The expected time is used rather than the 95% confidence point
- Inspections happen on 200 flight-hour boundaries.

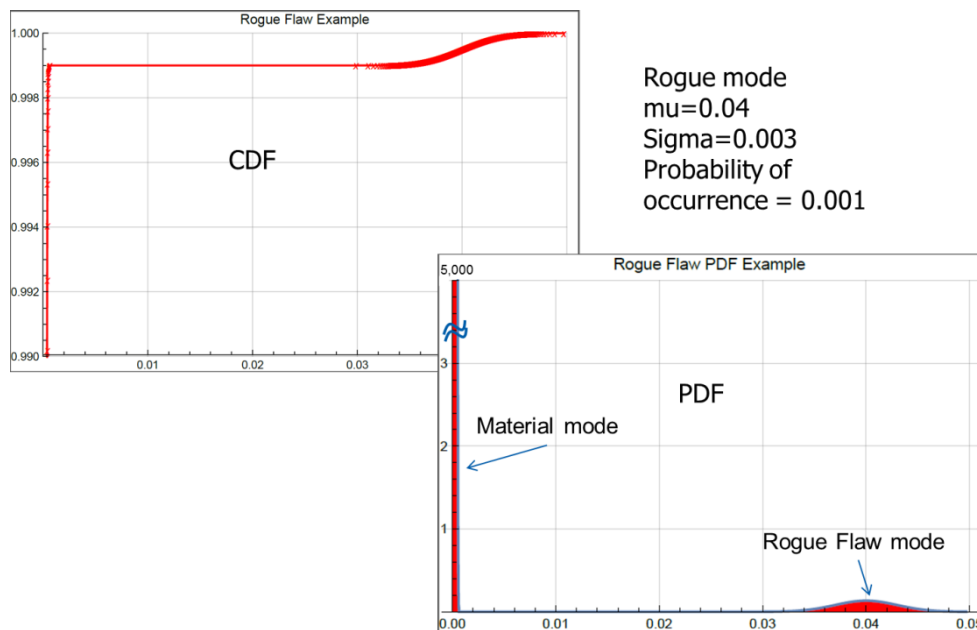


Figure 79. Bi-modal Initial Flaw Distribution

Given these assumptions, the first inspection for safety is the 200 flight hour boundary that precedes the point in time where P²IAT determines that the expected time when SFPOF for the next flight will exceed 10^{-7} as shown with red line in Figure 80 (alternatively, use the green line if the 95% confidence point is preferred). This calculation included the rogue flaw mode in addition to the microstructurally-derived initial flaw mode. With these criteria, the time for the first inspection was determined for the example control point. The time for subsequent inspections that are driven by safety was determined based on P²IAT predictions adapted by negative findings at each inspection.

⁸ Economic criteria may drive inspections at an earlier point in time

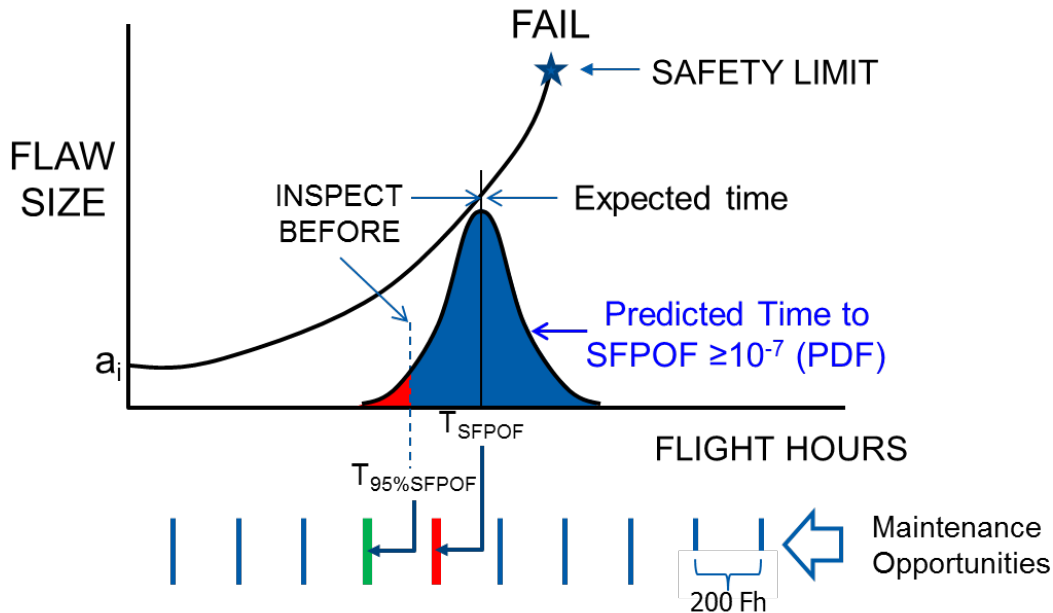


Figure 80. SFPOF-based Inspections

Note: The expected time rather than the 95% confidence line was chosen for the demonstration since it is consistent with the current criteria.

4.3.1 Results with Negative Inspection Findings

Given the aforementioned assumptions, the following results were obtained for CP NG04. Using the baseline spectrum statistically derived from the 12 example flights provided as GFI, predictions were made to reach the safety criteria given the bimodal flaw distribution shown in Figure 34. The first inspection was recommended at 600 FHs. With an assumed negative finding, the next inspection was then recommended at 800 FHs. Thereafter subsequent inspections were recommended at 2,000 FHs then 3,200 FHs. As theoretically predicted, the first two inspections (at 600 and 800 FHs) sufficiently reduced the probability of infant mortality to significantly postpone further inspections. This presumes that the inspection method does not introduce new defects (i.e., an inspection procedure that does not remove the fastener). If the fastener is removed for the inspection, we recommend that the rogue mode be added back into the current crack distribution and continue from there. This would postpone the next inspection for 600 FHs as was the case from time zero. The NDI POD curve for this example, shown in Figure 81, represents a typical NDI sensor response having a 0.95 probability of detecting a 0.05 inch crack.

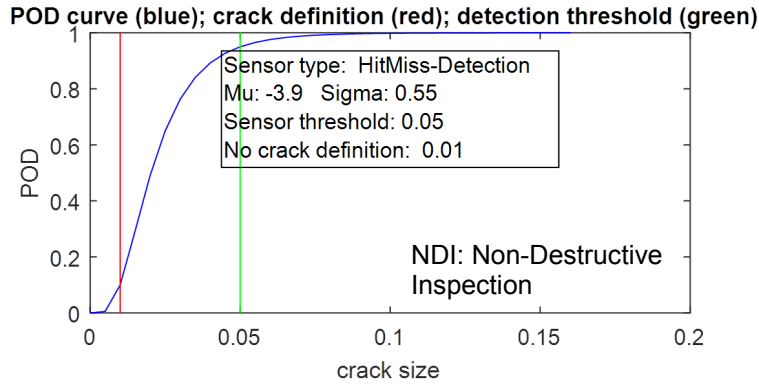


Figure 81. Typical NDI POD Curve

The crack probability distribution before and after the negative indication at 600 FHs are shown in Figure 82 and Figure 83, respectively.

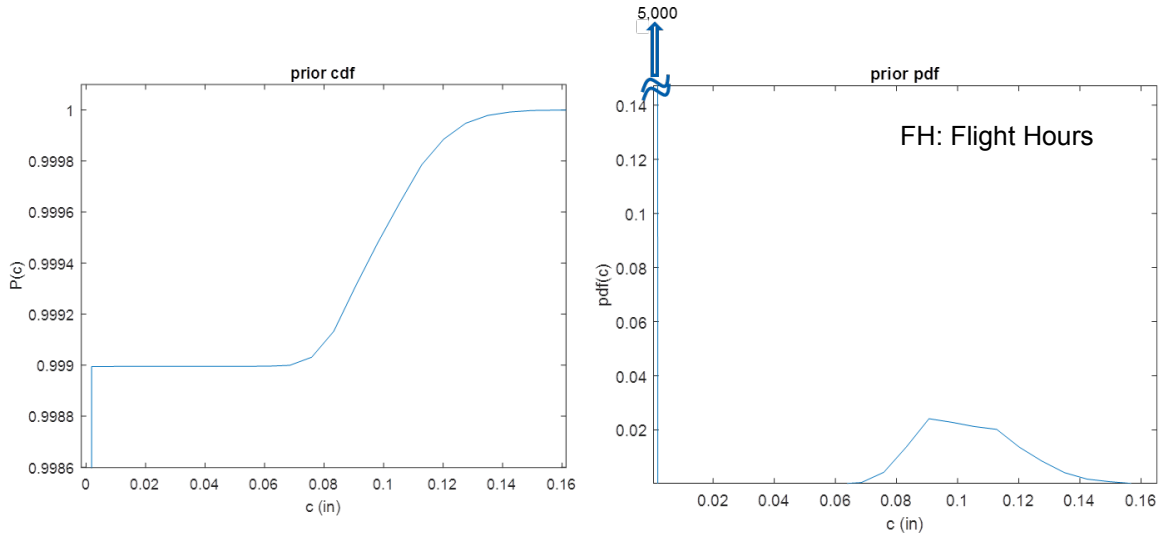


Figure 82. Bimodal Crack Distribution at 600 FHs Prior to Inspection

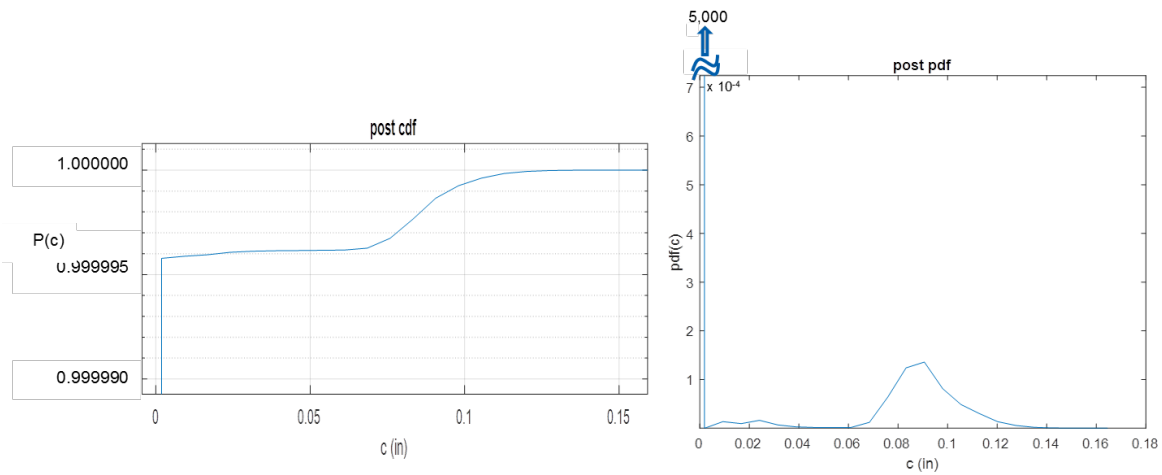


Figure 83. Probability Distribution after Negative Inspection Findings at 600 FHs

Note that even after one inspection (FH=600), the probability of the rogue flaw mode was significantly diminished (plots have different scales).

Scenario 6: This scenario simulated the conditions of a retired panel as would be used in the Task Order 0003 demonstration. To simulate this condition, 4,600 flights were run prior to the demonstration. Thereafter updates were added having 200 flights per update. Inspections were performed with simulated NDI sensors having three different probability-of-detection curves as shown in the following three figures. Sensor types 1-3 had a 0.90, 0.95 and 0.999 probability of detecting a 0.05 inch crack, respectively.

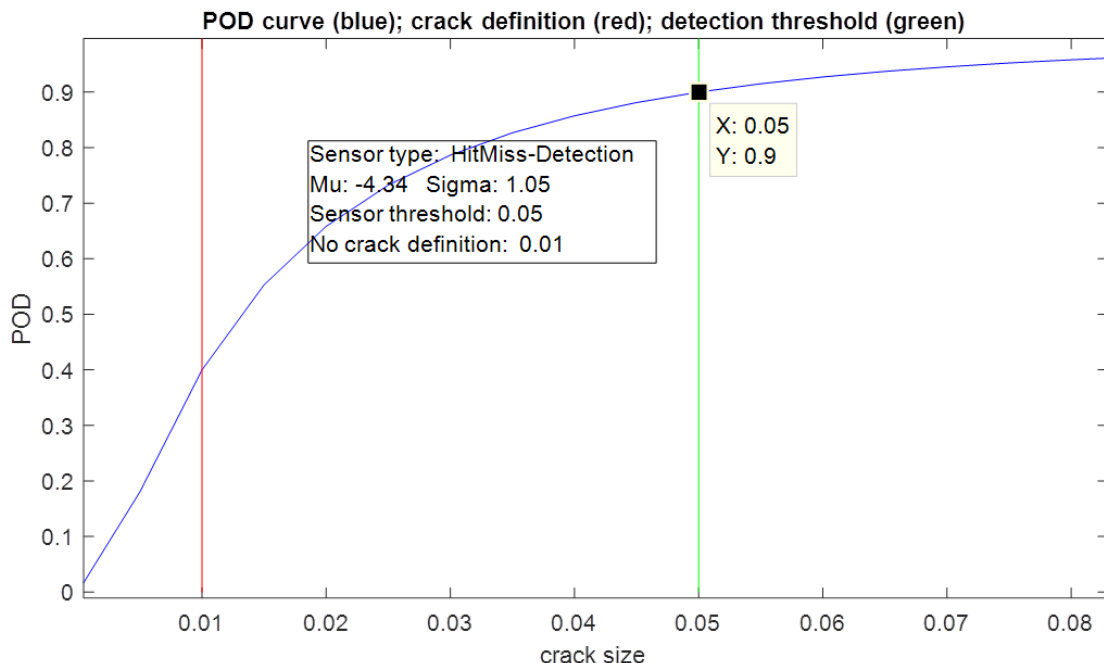


Figure 84. Sensor Type 1

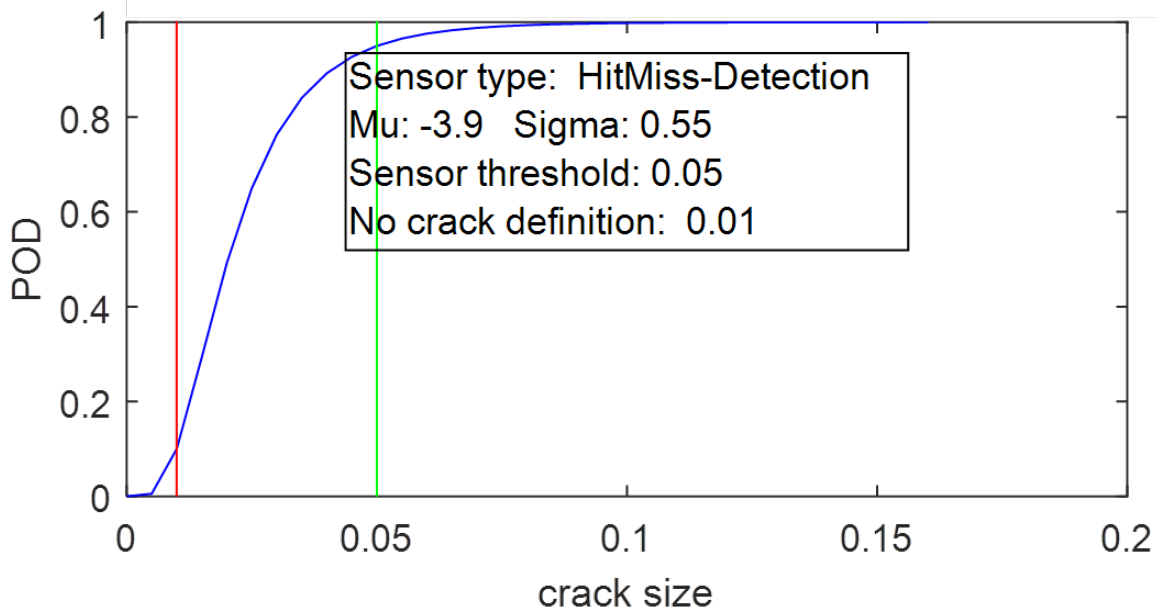


Figure 85. Sensor Type 2

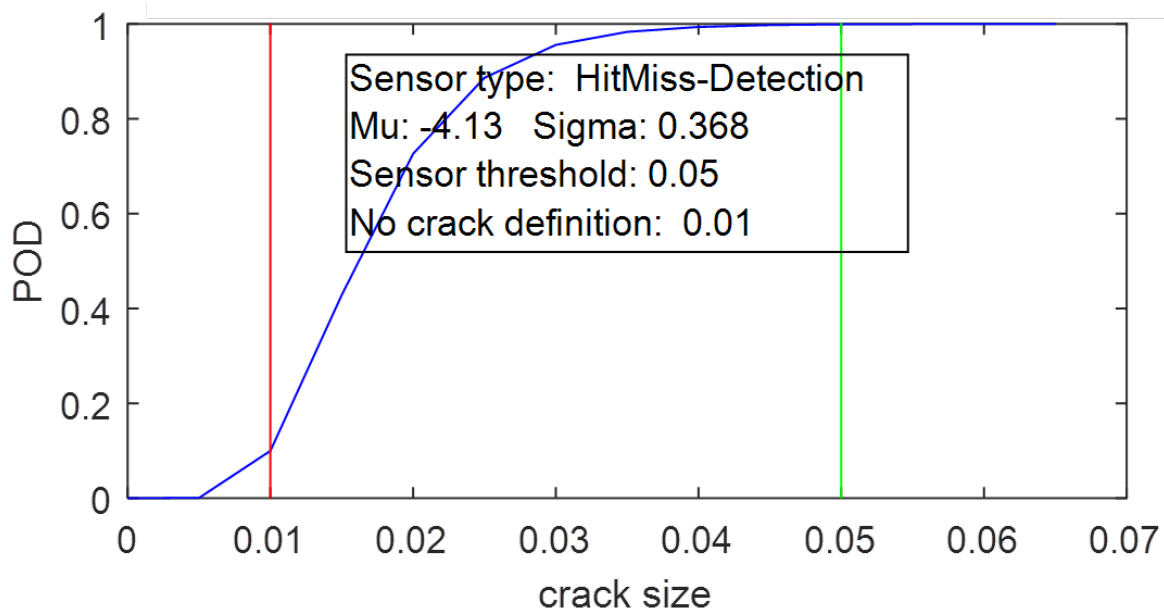


Figure 86. Sensor Type 3

Remaining useful life predictions were performed for a 0.03-inch crack. This is the maximum size that could be drilled out if the next larger fastener is installed for this control point. Note that although a crack of 0.03 inch is well below the reliable threshold of detection for most NDI instruments, the detect/no-detect finding still provides a probabilistic assessment of the crack size during inspections even when the result is “no indication found”. This concept is a significant departure from the traditional interpretation of go, no-go findings from inspections. Conversely, a positive indication from NDI does not necessarily mean that a crack exists - just as a large crack size reported by a deterministic model does not necessarily mean that a crack of that size exists. False alarms are common with such instruments. It is the probabilistic

interpretation of both fatigue models and NDI findings (both positive and negative indications) that, when properly interpreted in the context of their respective uncertainties, provides the best assessment of the crack state.

The expected time to reach a 0.03 inch crack was predicted to occur at flight 5,172. No lead time was specified so P²IAT recommended an inspection at flight 5,000 to fall in with the maintenance tempo. At flight 5,000, a simulated inspection with a negative indication was performed using sensor type 1. This is consistent with the test article findings since positive indications were not found at this control point for either test article (both have approximately 5,000 FH). Given this result, P²IAT recommended another inspection in 241 flights (at regular interval 5,200). A positive indication at either 5,000 or 5,200 would have recommended immediate installation of the next larger fastener.

A what-if scenario was performed with other sensor types to demonstrate the usefulness of P²IAT for analyzing the value of better information. If sensor type 3 was used to perform NDI at flight 5,000, the next expected time to reach a crack of 0.03 inch was predicted to be an additional 299 flights rather than 241 flights. In practice both sensors would have required an inspection at 5,200. Better sensors, including those that provide crack size estimates, prolonged inspections beyond 5,200 (not included in this report).

To better illustrate these results, we show the joint crack size distributions before and after negative NDI findings. The following figures show the joint a,c cumulative distributions on the left and the marginal distribution in the c dimension on the right. Figure 87 shows the result with sensor type 1. Figure 88 shows the result with sensor type 2. Figure 89 shows the result with sensor type 3.

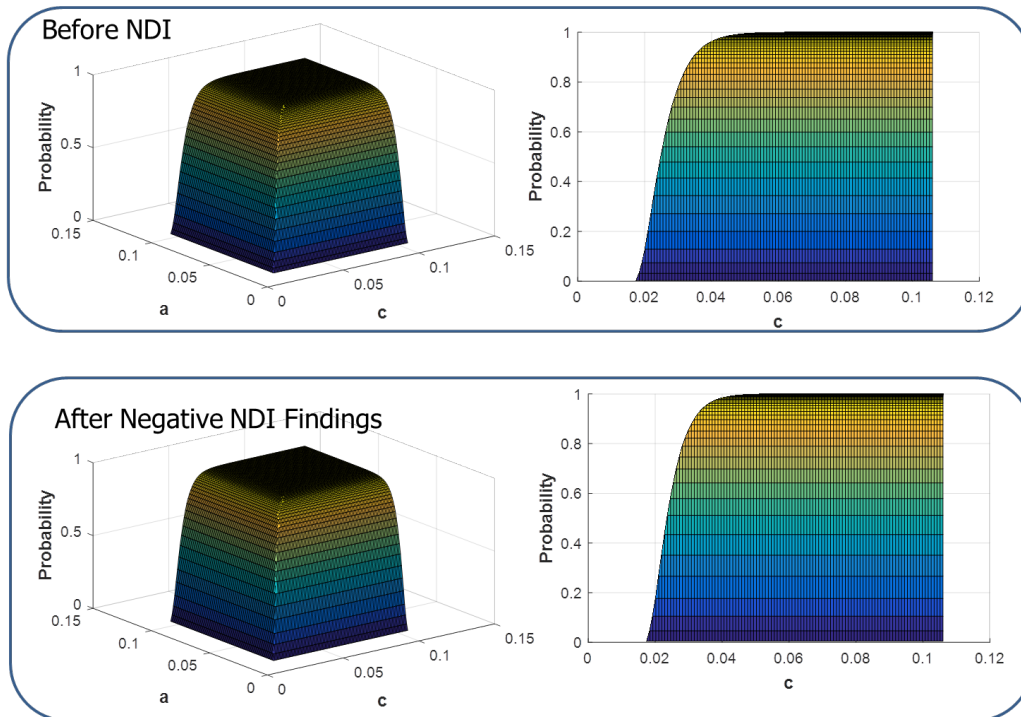


Figure 87. Inspection Outcomes at Flight 5,000 with Sensor Type 1

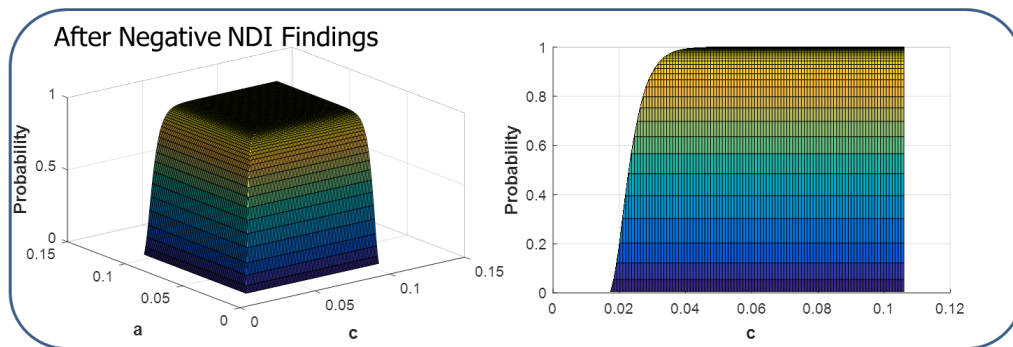
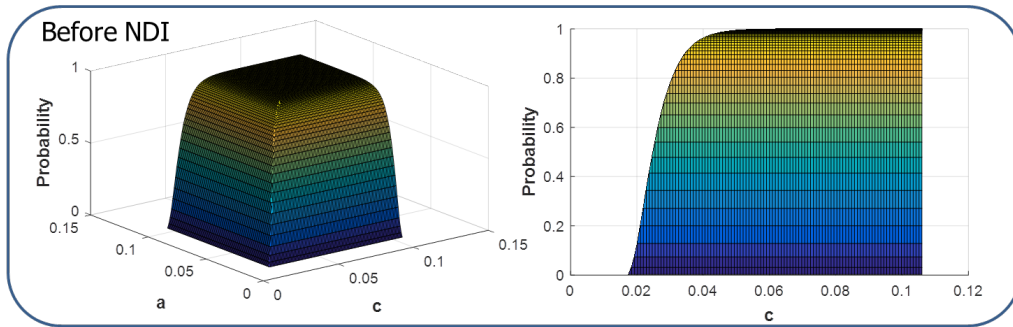


Figure 88. Inspection Outcomes at Flight 5,000 with Sensor Type 2

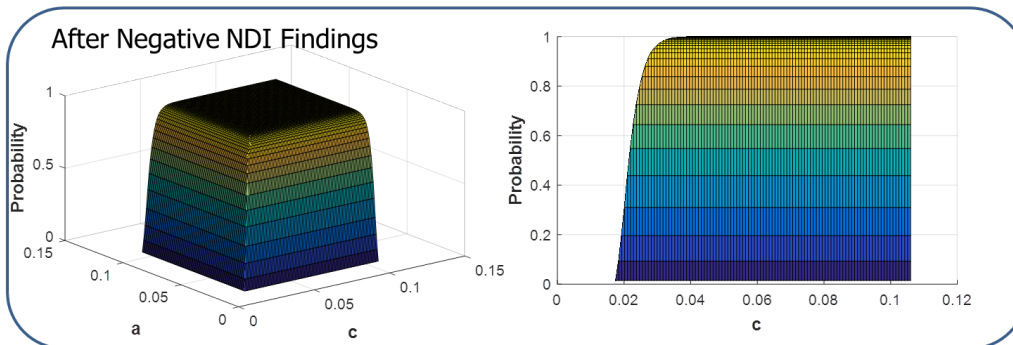
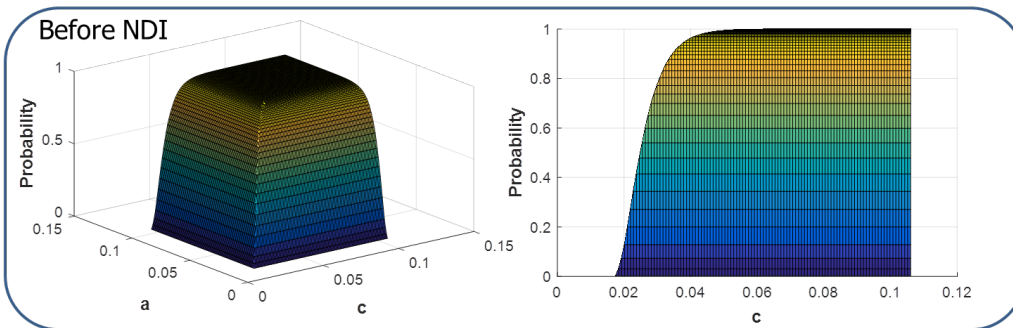


Figure 89. Inspection Outcomes at Flight 5,000 with Sensor Type 3

5.0 CONCLUSIONS

During the course of this contract, NGC has produced the following methodologies and accomplishments in support of the ADT objectives:

1. A methodology implemented in Matlab that statistically analyzes preprocessed flight recorded data (12 flights) provided as GFI to identify the probability of various maneuvers represented in the StS-RtS v1 maneuver suite.
2. A method to synthesize anticipated flights whose maneuvers reflect the same statistics as user-provided example flights (can be used to characterize mission types, individual pilot flying style, mission mixes etc. by examples).
3. Developed a probabilistic aircraft usage and loads capability that utilized FDR data, physics-based models, and a maneuver identification algorithm to capture uncertainty and variability associated with mission mix, maneuvers within missions, and missing data.
4. Software that synthesizes event-based FDR data from the dominant StS-RtS v1 maneuvers to provide inputs that drive the P2IAT process / demonstration.
5. Developed a method to impute missing data with characteristic statistics.
6. Created a work-around to compensate for software problems in the StS-RtS v1 program so that distributed loads could be used as intended in the development of stress histories.
7. Utilized StS-RtS v1 software to associate the loads to be applied on the test article as well as the local stresses histories, with the corresponding pilot stick inputs that generated those maneuvers. Our approach also related the test spectrum to flight parameters (Nz, Mach, altitude, roll rate, and roll acceleration) and aircraft configuration associated with the StS suite of maneuvers.
8. Developed a ModelCenter capability to automate the entire process in developing the baseline spectrum and local stress histories at any location on an aircraft.
9. Corrected errors in the GFI FEM so that accurate stresses could be generated for the TO1 demonstration.
10. Incorporated uncertainties from various sources including actual and anticipated usage, materials, initial and continued flaw size in both length and depth dimensions, fracture toughness, yield stress, ultimate tensile strength, elastic modulus, peak stress and structural geometry.
11. Developed a stochastic material model to capture statistical uncertainty of crack growth rate data at the Paris, threshold, and fracture regimes and construct probability curves from threshold to failure to be used with fatigue crack growth life codes.
12. Developed a framework that can be extended to more random variables.
13. Developed a two dimensional adaptation method that can adjust crack size assessments and predictions given inspection data and the corresponding POD function.
14. Developed a probabilistic engine that can process numerical distributions in addition to parametric distributions.
15. Expanded the SMART | DT implementation to handle 2-dimensional joint crack size distributions.
16. Added RUL moments, and distributions, FASTRAN interface, user-defined spectrum for NASGRO, crack size distributions (JCDF and quantiles) and peak stress to SMART | DT.
17. Developed and demonstrated cross entropy theory for importance sampling.
18. Incorporated spectrum peak stress as a random variable in SMART | DT.
19. Added a FASTRAN interface to SMART | DT.

20. Developed criteria for inspections that incorporate both safety and economic considerations for SOF and non-SOF control points.
21. Outlined a method to evaluate the economic value of uncertainty reduction in terms of repair costs.
22. Developed a methodology for managing rogue flaws with minimal inspections while maintaining safety directives.
23. Created a computational platform to perform what-if analyses involving hypothetical characterizations of random variables and inspection scenarios.
24. Prototyped NGC's P²IAT process by integrating the methods and features above as process models in ModelCenter - an open, commercially available tool. NGC's implementation allows the operator to "push a button" to perform quarterly updates and recover missing FDR data for a control point and automatically generate an update report with the desired outputs specified by AFRL in the specified formats.
25. Performed a real-time P²IAT demonstration involving six scenarios demonstrating: gap-filling, FDR data recovery, predicting time to reach a crack with 90/95 POD, adjusting assessments and predictions based on NDI results, predicting inspection times for safety and remediation for both new and aged vehicles and a considerable reduction in the number of inspections needed to manage rogue flaws.

The ADT efforts and the P²IAT demonstration to date have clearly demonstrated the advantages that the system will have for the DOD. We believe that the tools developed and integrated in ModelCenter provide an effective platform for further development.

6.0 REFERENCES

1. Eastin, R. and Bos, M., ICAF 2009, Bridging the Gap between Theory and Operational Practice, Proceedings of the 25th Symposium of the International Committee on Aeronautical Fatigue, Rotterdam, The Netherlands, 27-29 May 2009.
2. Butkus, Lawrence M., Gallagher, Joseph P., Berens, Alan P., Malas, James C., Babish IV, Charles A., U.S. Air Force Efforts in Understanding and Mitigating the Effects of “NDI Misses,” 24th ICAF Symposium - Naples, 16-18 May 2007.
3. Anagnostou, E., Engel, S., Fridline, D., Hoitsma, D., Madsen, J., Nardiello, J., and Papazian, J., “Science-Based Multiscale Modeling of Fatigue Damage for Structural Prognosis,” Proceedings of 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2010.
4. Lincoln, John W., and Meliere, Ronald A., Economic Life Determination For A Military Aircraft; ASC 97-2894, 17 December 1997.

APPENDIX – Prognosis Recommendations

Double-click the paper clip in the vertical toolbar on the left to view the attached appendix, and double-click the file name to open the related file.

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

Acronym	Definition	Acronym	Definition
ADT	Airframe Digital Twin	MES	Master Event Schedule
AF	Air Force	MTBF	mean time between failure
AFRL	Air Force Research Laboratory	MVT	bending-shear-torsion
AFTOC	Air Force total ownership cost	NASGRO	<u>N</u> ASA <u>F</u> atigue <u>C</u> rack <u>G</u> rowth <u>S</u> oftware
APO	abrupt push-over	NDE	nondestructive evaluation
APU	abrupt pull-up	NDI	nondestructive inspection
ASIP	Aircraft Structural Integrity Program	NGAS	Northrop Grumman Aerospace Systems
CAD	computer aided design	NGC	Northrop Grumman Corporation
CAMS	Core Automated Maintenance System	PDF	probability density function
CAPE	cost assessment & program evaluation	PITS	point-in-the-sky
CDF	cumulative distribution function	POD	probability of detection
CDRL	Contract Data Requirements List	POF	probability of failure
CP	Control Point	POFRUL	probability of failure-based remaining useful life
CSRUL	crack-size-based remaining useful life	P ² IAT	Prognostic and Probabilistic Individual Aircraft Tracking
DARPA	Defense Advanced Research Projects Agency	P-V	peak-valley
DC	detected crack (crack detected)	QA	quality assurance
DoD	Department of Defense	RPO	rolling pull-out
DT	damage tolerance	RUL	remaining useful life
DTA	damage tolerance analysis	SDR	Signal Data Recorder
EVD	extreme value distribution	SEQGEN	Sequence Generator
FASTRAN	FATigue crack STRuctural ANALysis	SFPOF	single flight probability of failure
FCL	fatigue critical location	SIPS	Structural Integrity Prognosis System
FDR	Flight Data Recorder	SHM	structural health monitoring
FEM	Finite Element Model	SLAP	Service Life Assessment Program
FH	flight hours	SLEP	Service Life Extension Program
FSMP	Force Structural Maintenance Plan	SMART	Small Aircraft Risk Technology
GCSS	Global Combat Support System	SOF	safety of flight
GFI	government-furnished information	SOW	Statement of Work
IAT	individual aircraft tracking	StS-RtS	Stick-to-Stress Real-time Simulator
JCSD	Joint Crack Size Distribution	TO	task order
JITP	just-in-time point	USAF	United States Air Force
IMS	Integrated Master Schedule	UTSA	University of Texas at San Antonio
MDS	mission design series		