



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**IMPROVING THE EFFICIENCY OF ISOCHRONAL
MAINTENANCE SCHEDULING FOR THE KC-130J**

by

Quentin Biles

June 2021

Thesis Advisor:
Second Reader:

Roberto Szechtman
W. Matthew Carlyle

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, 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 Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC, 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE June 2021	3. REPORT TYPE AND DATES COVERED Master's thesis		
4. TITLE AND SUBTITLE IMPROVING THE EFFICIENCY OF ISOCHRONAL MAINTENANCE SCHEDULING FOR THE KC-130J			5. FUNDING NUMBERS	
6. AUTHOR(S) Quentin Biles				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) This thesis explored the potential readiness benefits that could be gained from centralizing the Marine Corps' KC-130J isochronal (ISO) maintenance special inspections. ISO inspections are scheduled for aircraft based solely on elapsed calendar time, and organic squadron maintenance personnel conduct the inspection in-house. Increases in squadron workloads combined with limited personnel have resulted in these ISO inspections taking considerably longer than expected to complete, which directly impacts aircraft readiness. By consolidating the maintenance effort at a central location, contract workers can be employed to handle the ISO inspections for the entire KC-130J fleet. The Air Force has employed a similar system for its special operations C-130 fleet, and the inspection timeline has improved dramatically. The inspection process was modeled via a stochastic discrete event simulation written in Python. The model provided an estimate on the overall improvement in readiness levels across several potential scenarios, and the minimum required turnaround time based on central facility capacity can be determined.				
14. SUBJECT TERMS maintenance scheduling, KC-130J, simulation optimization			15. NUMBER OF PAGES 109	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**IMPROVING THE EFFICIENCY OF ISOCHRONAL MAINTENANCE
SCHEDULING FOR THE KC-130J**

Quentin Biles
Captain, United States Marine Corps
BS, Georgia Institute of Technology, 2012

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

**NAVAL POSTGRADUATE SCHOOL
June 2021**

Approved by: Roberto Szechtman
Advisor

W. Matthew Carlyle
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis explored the potential readiness benefits that could be gained from centralizing the Marine Corps' KC-130J isochronal (ISO) maintenance special inspections. ISO inspections are scheduled for aircraft based solely on elapsed calendar time, and organic squadron maintenance personnel conduct the inspection in-house. Increases in squadron workloads combined with limited personnel have resulted in these ISO inspections taking considerably longer than expected to complete, which directly impacts aircraft readiness. By consolidating the maintenance effort at a central location, contract workers can be employed to handle the ISO inspections for the entire KC-130J fleet. The Air Force has employed a similar system for its special operations C-130 fleet, and the inspection timeline has improved dramatically. The inspection process was modeled via a stochastic discrete event simulation written in Python. The model provided an estimate on the overall improvement in readiness levels across several potential scenarios, and the minimum required turnaround time based on central facility capacity can be determined.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	PROBLEM OVERVIEW.....	2
B.	PROPOSED SOLUTION.....	2
C.	ASSUMPTIONS AND LIMITATIONS	3
II.	BACKGROUND AND LITERATURE REVIEW	5
A.	MAINTENANCE OVERVIEW	5
1.	Maintenance Theory	5
2.	KC-130J Maintenance and Practices	6
B.	QUEUEING AND SIMULATION THEORY	12
1.	Basic Queueing Theory	12
2.	Discrete Event Simulation	14
3.	Event Graphs.....	16
C.	RELATED STUDIES AND PRACTICES	18
1.	Government Accountability Office Weapon System Sustainment	18
2.	RAND Air Force C-130 Logistics Report	19
3.	C-5 ISO Consolidation Simulation	22
4.	Air Force’s Current Situation.....	23
5.	Alternative Approaches for U.S. Navy and Marine Corps	25
III.	SIMULATION MODELING.....	29
A.	THE MODEL	29
1.	Parameters/Arguments.....	31
2.	State Variables	33
3.	Event Graph	34
4.	Statistics Collected	41
B.	THE DATA.....	43
1.	Inspection Time Data.....	43
2.	Model Initialization Data	48
IV.	RESULTS	51
A.	MODEL VERIFICATION AND BASELINE PERFORMANCE	51
1.	Deterministic Runs.....	51
2.	Simulation of Current State.....	55
B.	WORKFLOW FOR SIMULATION OF CENTRALIZED MAINTENANCE	59

1.	Input Data.....	60
2.	Model Execution.....	63
3.	Model Outputs.....	63
4.	Output Consolidation	66
C.	SUMMARY OF RESULTS	68
V.	CONCLUSION	77
A.	RECOMMENDATIONS.....	77
B.	FUTURE WORK.....	79
1.	Optimization of Maintenance Procedures	80
2.	Depot-Level Impacts	80
3.	Automate Model Initialization Data Creation	81
4.	Analyze Cause of Current Delays.....	82
5.	Special Inspection Timing.....	82
	LIST OF REFERENCES	83
	INITIAL DISTRIBUTION LIST	85

LIST OF FIGURES

Figure 1.	The KC-130J. Source: USMC (2014).....	1
Figure 2.	VMGR Location Overview. Source: USMC (2019).	7
Figure 3.	Generic Squadron Maintenance Organization. Source: CNAF (2017).	8
Figure 4.	Redacted Special Inspection Workcard. Source: NAVAIR (2018).	10
Figure 5.	KC-130J Depot Cycle Overview. Source: NAVAIR (2018).	11
Figure 6.	KC-130J Depot Maintenance Trend. Source: NAE (2020).	12
Figure 7.	A Basic Queue. Source: Gross et al. (2018).	13
Figure 8.	Queue Notation. Source: Gross et al. (2018).	14
Figure 9.	Discrete Event Simulation Time Jumps. Source: Law (2015).	15
Figure 10.	State Variable Changes During Discrete Event Simulation. Source: Gross et al. (2018).	15
Figure 11.	Discrete Event Simulation Flow Logic. Source: Law (2015).	16
Figure 12.	Basic Event Graph Concept. Source: Buss (2001).	16
Figure 13.	More Complicated Event Graph. Source: Buss (2001).	17
Figure 14.	KC-130J Operating and Support Costs. Source: GAO (2020).	19
Figure 15.	C-130 CRF Personnel Economy of Scale. Source: Roo et al. (2011).	20
Figure 16.	ISO Flow Times Versus CRF Capacity. Source: Roo et al. (2011).	21
Figure 17.	Fly-to-Fly Comparison Between Centralized and Decentralized ISO Inspection. Source: Roo et al. (2011).	22
Figure 18.	AFSOC C-130 Locations. Source: Ford (2017).	24
Figure 19.	AFSOC C-130 ISO Timeline Flow. Source: Ford (2017).	24
Figure 20.	F/A-18 P2P Readiness Comparison. Source: U.S. Navy (2020).	26
Figure 21.	F/A-18 P2P Performance Driver Tree. Source: Schell (2020).	27

Figure 22.	Basic Model Flow Chart	29
Figure 23.	Complete Model Event Graph	35
Figure 24.	Aircraft Creation Event Graph Sub-Section.	36
Figure 25.	ISO Cycle Event Graph Sub-Section.	38
Figure 26.	Depot Cycle Event Graph Sub-Section.....	40
Figure 27.	Raw DECKPLATE Data Example	44
Figure 28.	Raw 420-Day and 840-Day Mother MAF Times	45
Figure 29.	Final 420 and 840-Day Inspection Times with Fit Distributions	47
Figure 30.	420- (Top) and 840-Day (Bottom) Fitted Gamma Distributions.....	48
Figure 31.	Sample Depot Scheduling Data. Source: Friedlein (2021).	49
Figure 32.	Single-Aircraft Deterministic Run State Variables.....	52
Figure 33.	Single-Aircraft Percent Availability Calculations	52
Figure 34.	Single-Aircraft Deterministic Run Summary Statistics.....	53
Figure 35.	Example Event List Printout.....	54
Figure 36.	Current Fleet Deterministic Simulation State Variable	55
Figure 37.	Current Fleet Deterministic Simulation Summary Statistics	55
Figure 38.	Current State Simulation State Variables	57
Figure 39.	Distribution of Mean Percent Availability for Current State.....	58
Figure 40.	Normal Quantile-Quantile Plot of Mean Percent Availability for Current State	58
Figure 41.	Simulation Workflow.....	60
Figure 42.	Example ISO Service Time Cumulative Distribution Functions.....	61
Figure 43.	Gamma Distribution Shape Parameter Calculation	62
Figure 44.	Model Scenarios CSV Sample.....	62
Figure 45.	“Run the Run” Script	63

Figure 46.	Summary Statistic Output File Hierarchy	64
Figure 47.	Example Graphical Outputs	65
Figure 48.	Example Percentage Availability CSV	65
Figure 49.	Sample Combined Summary Statistic Data Frame	66
Figure 50.	Sample Tidy Output Statistic CSV	67
Figure 51.	Sample Combined Final Output Statistic Data Frame	68
Figure 52.	Mean Availability as a Function of Turnaround Time and Server Count	69
Figure 53.	Mean Availability Feasible Region (Detailed View)	70
Figure 54.	Feasible Region Model Outputs	71
Figure 55.	Feasible Region Mean Availability and Additional Days	72
Figure 56.	Feasible Region Additional Yearly Availability	73
Figure 57.	Squadron Level Baseline Availability Versus Centralized Models	74
Figure 58.	Squadron Level Availability Versus Baseline (Detail)	75
Figure 59.	Squadron Level Days of Availability Versus Baseline	76
Figure 60.	Feasible Improvements Over Baseline	78
Figure 61.	Days of Availability as a Function of Facility Size and Turnaround Time	79
Figure 62.	Pending Spike in Depot Levels and Corresponding Drop in Available Aircraft	81

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Special Inspection Time Requirements	10
Table 2.	Model Parameters	31
Table 3.	Model State Variables.....	33
Table 4.	Model Output Statistics.....	42
Table 5.	420 and 840-Day Inspection Times with Overruns Included	45
Table 6.	Final 420 and 840-Day Inspection Time Summary Statistics	47
Table 7.	Parameters for Deterministic Simulation.....	51
Table 8.	Parameters for Current State Simulation	56
Table 9.	Availability Comparison with Baseline and Max Theoretical.....	59
Table 10.	Simulation Scenarios	61
Table 11.	Model Outputs	64
Table 12.	Output Statistics Selected for Scenario Comparison	67
Table 13.	Feasible Region Highlighting Justifications	71

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AFIT	Air Force Institute of Technology
AFSOC	Air Force Special Operations Command
AICc	Akaike's Information Criterion (corrected)
AIRRS	Aircraft Inventory and Readiness Reporting System
AMC	Air Mobility Command
CAS	close air support
CMC	Commandant of the Marine Corps
CNAF	Commander, Naval Air Forces
CNO	Chief of Naval Operations
CRF	centralized repair facility
CSV	comma-separated value
DBPM	driver-based performance management
DECKPLATE	DECision Knowledge Programming for Logistics Analysis and Technical Evaluation
DES	discrete event simulation
FMC	full mission capable
FRC	fleet readiness center
FY	fiscal year
GAO	Government Accountability Office
ISO	isochronal
MAF	maintenance action form
MAGTF	Marine air-ground task force
MALS	Marine aviation logistics squadron
MAW	Marine aircraft wing
MC	mission capable
MEU	Marine expeditionary unit
MIR	multi-sensor imagery reconnaissance
MOC	maintenance operations center
MOS	military occupational specialty
MSG	maintenance steering group
NAMP	Naval aviation maintenance program
NAVAIR	Naval Aviation Systems Command
NSS-A	Naval Sustainment System-Aviation
OOMA	Optimized Organization Maintenance Activity

P2P	Performance-to-Plan
PMI	planned maintenance interval
PPM	predictive performance models
RCM	reliability-centered maintenance
SC	special control
SF	special fix
SL	special look
SP-MAGTF	special purpose Marine air-ground task force
TEC	type equipment code
USAF	United States Air Force
USMC	United States Marine Corps
VMGR	Marine aerial refueler transport squadron

EXECUTIVE SUMMARY

The KC-130J “Super Hercules” is a multirole, fixed-wing, assault support aircraft that executes a number of missions for the United States Marine Corps (USMC), including aerial refueling, personnel and cargo transport, aerial delivery, and close air support. Demand for the KC-130J is currently extremely high due to the broad array of missions it is able to support, and the platform has the highest operational tempo (time spent away from home station) of any manned USMC aviation platform. Because of this demand, the size of the KC-130J fleet is projected to grow from 57 aircraft and 3 active-duty squadrons up to 81 aircraft and 4 active-duty squadrons between now and fiscal year 2026. However, this high utilization rate, combined with manpower and parts shortages, has resulted in declining readiness levels across the KC-130J fleet.

Careful assessment of the KC-130J maintenance process has revealed that two of the major isochronal (ISO) special inspections, which are scheduled on a fixed 420-day cycle, have been overrunning their expected completion times by a significant amount. The 420-day and 840-day special inspections are designed as a form of preventative maintenance, and together they comprise over 130 individual inspections which total over 325 manhours. Squadrons are required to conduct these inspections using organic personnel, and the inspections are typically scheduled to be performed over the course of several weeks (although no official guidance is available for the scheduling of the required sub-inspections). However, recent trends have resulted in a median 420-day inspection time of 47 days and a median 840-day inspection time of 56 days. The aircraft is completely unavailable during these inspection periods and reducing the turnaround time would return valuable availability back to the squadron.

The proposed solution to reduce these inspection times is to outsource the maintenance to a consolidated facility operated by a contract maintenance force and specifically designed to conduct these inspections. A similar solution was implemented by the Air Force Special Operations Command (AFSOC) to conduct the 540-day ISO inspection for its C-130 fleet. The AFSOC facility conducts approximately 75 inspections per year, and the fly-to-fly time (time between the first flight prior to ISO and the first flight

after ISO) decreased from 35 days (standard deviation of 22 days) to 19 days (standard deviation of 9 days) by using the centralized facility. While the Air Force maintenance practices do not directly correspond to those of the Marine Corps, consolidating maintenance efforts for these inspections would relieve the squadrons of a significant maintenance burden and allow for the inspections to be conducted with a greatly improved learning curve.

The potential benefits of this solution were examined using a discrete event simulation (DES) model implemented using the Python package DESpy. Python was chosen as the platform for the model due to its open-source nature and ability to run on USMC network computers. The model provides discrete-time analysis down to the individual aircraft level and operates in single-day time increments. This allows for a highly accurate representation of aircraft availability. The model is a closed network that simulates aircraft as they move between the operating fleet, ISO inspections, and depot inspections (a major overhaul at a centralized facility that occurs every 5 years). The model is initialized to the current state of the fleet (in terms of ISO timelines, depot schedule, and future arrivals) using current data from the USMC. Additional inputs that will be examined include the service time distribution for the ISO inspections and the number of concurrent ISO servers. However, there are many other inputs that can be used to customize the model to examine other factors in this process. The model outputs include the percent of time the aircraft are available to the fleet in addition to all summary statistics for the ISO and depot queues.

The model was initially run using ISO service time distributions that were fit to the current fleet inspection times (all runs of the model were conducted over a 15-year period and executed 100 times). The statistical analysis software JMP was used to fit a Gamma distribution to the service times that were calculated from actual maintenance data. These runs of the model produced a mean aircraft availability of 82.83%. In comparison, a run of the model using only minimum deterministic times produced a maximum availability of 91.13%. Once the current baseline was established, the consolidated facility scenarios could be examined. For the consolidated scenarios, service times were treated as a target minimum turnaround time with the potential for overruns. This reflects the fact that a

contract would rely on an agreed-up return time with the understanding that certain supply or repair issues could potentially delay the return. In addition to the variable minimum service times, the size of facility was analyzed based on the number of concurrent inspections it was able to support. In total, 19 minimum turnaround times (7 to 25 days for the 420-day inspection) and 11 facility sizes (2 to 12 concurrent aircraft) were examined for a total of 209 scenarios.

The simulation results revealed two valuable pieces of information. First, the mean availability for each turnaround/size combination could be compared to the baseline to determine the overall performance increase. Second, the maximum turnaround time for a given facility size was revealed by examining the queue performance for each scenario. Overall, it was determined that optimal facility size was between 3 and 5 servers. Facilities with 2 servers did not produce stable queues, and facilities with more than 5 servers did not see a significant increase in turnaround times relative to smaller facilities. In this server range, the best-case scenario (4 servers with a 7-day turnaround) produced a 90.3% availability (+7.5% over baseline) and the worst-case scenario (5 servers with a 25-day turnaround) produced an 86.4% availability (+3.5% over baseline). These percentage increases can be converted directly to days of additional availability. In the best-case scenario, an 18-aircraft squadron would expect to see an additional 491 days of availability per year, and the worst-case scenario produces 232 additional days of availability per year.

This study demonstrates that improving the turnaround time for the ISO inspections has the potential to produce significant readiness improvements. The use of contract maintenance is a proven process that has been utilized successfully by other branches of the military, and the Marine Corps should seriously assess the potential of consolidating its KC-130J 420- and 840-day inspections.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First and foremost, I would like to thank my wife, Desiree, for her continued support and incredible parenting of our two wonderful children, Ava and Jayden. She has been my rock over these last 14 months of pandemic life, and I could not imagine writing this thesis or graduating from NPS without her support. I'm sure you're ready for me to finally get out of the house and go back to work, but I want you to know that your support means everything to me!

My advisor, Dr. Roberto Szechtman, provided me with the opportunity to pursue this research when I approached him with my loosely defined and unfunded research topic. Thank you for believing me and providing your expert guidance along the way. As I struggled to comprehend DESpy in your 8 AM class during the first quarter of all-digital learning, I never imagined that I would be spending countless hours with it for my thesis. You have been a fantastic professor and advisor over the past year, and I could not have done this without you.

LtCol James Friedlein, CWO3 Nate Lamkin, and Erik Werhner were my connection back to the KC-130J community, and I learned more about the maintenance process over the last 6 months than I did in 3 years in the fleet. Thank you, gentlemen, for setting aside your valuable time to answer my naïve questions and providing me with your unmatched subject-matter expertise. I hope that we are able to make an impact for our community!

LtCol Mark Sykes, the Marine Corps' senior Operations Research representative here at NPS, encouraged me to pursue a thesis that interested me even when I may have been behind the timeline. I cannot thank you enough for putting me in contact with all of the motivated individuals who helped me grow "I want to do something for Marine aviation" into this thesis.

There are many others who contributed to this work, and even though I am unable to list you all by name, know that your assistance was greatly appreciated.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The KC-130J “Super Hercules” is a multirole, fixed-wing, assault support aircraft designed by Lockheed Martin and operated by the United States Marine Corps (USMC). It was introduced in early 2004 to replace the aging KC-130F/T/R fleet, some of which had been in service for nearly 40 years. In the time since its introduction, the KC-130J has supported combat operations in Afghanistan and Iraq, assisted with humanitarian aid around the globe, and has been continuously deployed in support of Marine Expeditionary Units (MEUs) and special purpose Marine air-ground task forces (SP-MAGTFs). Typical missions include aerial refueling (both fixed-wing and rotary), cargo and troop transport, aerial delivery, and close air support.



Figure 1. The KC-130J. Source: USMC (2014).

Years of highly demanding operations have begun to impact the readiness levels of the KC-130J fleet. Marine Aerial Refueler Transport Squadrons (VMGRs) have struggled to meet the minimum prescribed mission capable (MC) percentage. Typical in aviation, maintenance efforts are preventative in nature so that issues are discovered and repaired before they present safety-of-flight issues. The Marine Corps accomplishes this primarily through the use of special inspections, which are conducted by the squadron once certain

criteria are met. The two broad categories for special inspections are isochronal (occurring every specified number of days) and conditions-based (occurring only when a certain event occurs or is about to occur). These inspections also vary highly in terms of requirements, ranging from simple visible inspections to multi-day endeavors that involve all components of the maintenance department. This thesis will focus on the special inspections that occur based on a 420-day cycle.

A. PROBLEM OVERVIEW

The 420-day special inspection is a highly intensive inspection that involves over 120 individual inspection items with over 270 manhours of work. Additionally, the 840-day special inspection adds 20 inspection items and over 50 manhours of work on top of the regular 420-day inspection. These totals also do not include the time required to correct any deficiencies that are discovered during the inspection. While these inspections occur, the aircraft are unable to fly and contribute negatively to the squadron's readiness levels. Recent trends have seen both the 420- and 840-day inspections take upward of 2 months to complete, which leads to multiple simultaneous inspections and even fewer aircraft available on the flight line. Improving the time required to conduct the 420/840-day special inspection would have an immediate impact on the readiness levels of the KC-130J.

B. PROPOSED SOLUTION

This thesis will explore the prospect of consolidating the 420/840-day inspections at a centralized facility instead of requiring each squadron to conduct them. The centralized facility would be staffed by a contract workforce and service the entire fleet of KC-130Js. While it may seem unorthodox for military aviation to turn to civilian contract support, some VMGRs have already experimented with augmenting their own uniformed maintenance force with contractors in an effort to improve inspection times. The U.S. Air Force (USAF) also employs centralized maintenance locations for some of its isochronal inspections, and contractors fully staff the facility for the Air Force Special Operations Command (AFSOC) C-130s. The use of contractor maintenance is a proven system that directly benefits readiness levels.

A discrete event simulation (DES) model will be used to model the KC-130J fleet as aircraft proceed through the isochronal maintenance process. The greatest benefit of this approach is that it allows the observation of aircraft levels over time and can account for both the current state of the fleet and the planned expansion over the coming years. The service times for the inspections will be represented through random variables, and multiple runs of the model will provide a greater idea of the range of possible outcomes. The decision was made to write the model using the Python programming language due to its open-source nature and the ability to run on many platforms.

Assuming the contract workforce would have an agreed upon turnaround time (with some allowance for overruns), it is possible to fix the baseline turnaround time and simulate scenarios where the centralized facility has different maximum capacities for simultaneous inspections. Repeating this process for different turnaround times provides a range of feasible time and capacity combinations that can be used to evaluate the increase in performance versus potential costs (although this thesis will not cover the cost estimation). The primary performance metric will be the “percentage availability,” which is simply the percent of the time that an aircraft is not undergoing inspections. The baseline performance will be established by running the simulation using current inspection times, and then all improvements can be referenced as an increase over the baseline.

C. ASSUMPTIONS AND LIMITATIONS

This thesis is not intended to provide an ultimate solution to the KC-130J’s special inspection problems; it is designed to provide a framework that decision-makers can use to base requirements and cost estimates off. Issues such as the supply system and organic manpower requirements are not addressed, and the baseline performance is based purely off the system as it exists today. The inspections themselves are treated as a single event, and they are not sub-divided into the various smaller inspections that comprise them. No attempt is made to optimize the order in which the sub-inspections occur. The model does not make any assessments or requirements in terms of manpower. In terms of the model, a “server” for the inspection represents the capacity to conduct work on a single aircraft without having to delay due to concurrent jobs. Finally, the model does not consider any

costs associated with the acquisition of new maintenance facilities or with the impacts of maintaining them.

Ultimately, the objective is to provide a set of time and capacity requirements with their associated performance benefits so that a decision can be made on the best way to address the issue at hand.

II. BACKGROUND AND LITERATURE REVIEW

This chapter is intended to provide the background knowledge required to understand the purpose and methodology of the model. It provides information on the organization and maintenance practices of KC-130J squadrons, discusses the basic theory behind the simulation model, and examines several studies conducted regarding similar issues within the U.S. military.

A. MAINTENANCE OVERVIEW

This section provides an overview of basic aircraft maintenance theory and the organization of the KC-130J squadrons. It provides both an overview of the structure of Marine Corps aviation maintenance and details on the specific inspections that are conducted on the KC-130J.

1. Maintenance Theory

Aircraft maintenance has been an evolving effort across both military and commercial backgrounds. Safety has always been a paramount driver of maintenance practice, and today's practices have evolved considerably over the last 70-plus years. These practices were first codified in the "Maintenance Evaluation and Program Development" handbook, known as Maintenance Steering Group 1 (MSG-1). The intent was to develop scheduled preventative maintenance procedures for the Boeing 747. This document was generalized over the years through MSG-2 in the late 1960s and eventually MSG-3 in 1978. MSG-3 was a joint effort between regulatory agencies, international airlines and engine manufacturers, and the Department of Defense. MSG-3 provides a "task-oriented approach to maintenance that analyzes system failure modes from a system level, or top down. Maintenance tasks are performed for safety, operational, or economic reasons. They involve both preventive maintenance and failure finding tasks" (McLoughlin 2006, p. 28). This formed the basis for reliability-centered maintenance (RCM), which is "a viable approach for optimizing maintenance of systems by having an optimal mix of run to failure, time-based, condition-based, and design modification maintenance tasks" (Duffuaa 2015, p. 246). Both MSG-3 and RCM form the basis for the KC-130J's preventative maintenance

program (Naval Aviation Systems Command [NAVAIR] 2020), which is a mix of scheduled and conditional inspections designed to increase availability without impacting flight safety.

2. KC-130J Maintenance and Practices

This section is intended to provide an overview of KC-130J operations, squadron composition, and maintenance practices. It will provide a background on how KC-130J maintenance is accomplished.

a. KC-130 Overview

KC-130Js are operated by Marine Aerial Refueler Transport Squadrons (VMGR) and execute a wide variety of mission sets. The official VMGR mission statement is “to support the [Marine air-ground task force (MAGTF)] commander by providing air-to-air refueling, assault support, [close air support (CAS)] and [multi-sensor imagery reconnaissance (MIR)], day or night under all weather conditions during expeditionary, joint, or combined operations” (United States Marine Corps [USMC] 2019). This broad array of missions has led to the platform being in high demand across the fleet Marine forces (FMF), and the personnel tempo (the number of days a Marine is deployed or unable to return home) has been the highest among all of Marine aviation (USMC 2019). As a result of this demand, the KC-130J was identified as the only manned aviation platform planned to increase in the number of active-duty squadrons as part of the Commandant of the Marine Corps’ Force Design 2030. However, readiness levels have struggled, and the Marine Corps assessed that “the biggest factor in readiness and KC-130 availability is lack of aircraft on the flight line” (USMC 2019).

b. VMGR Squadron Composition

There are currently three active duty VMGR squadrons (with a potential fourth planned in the coming years) and two reserve squadrons in the Marine Corps. Squadrons have a baseline of 15 aircraft with an eventual plan to increase to 18 once all J-models are delivered, and the transition from the legacy T-model is complete. Figure 2 provides an overview of their locations and composition as of 2019, and it should be noted that the T-

model has been phased out as of 2021. Each VMGR is assigned to a different Marine Aircraft Wing (MAW), the 2-star level aviation command, and executes missions for their assigned geographic areas.

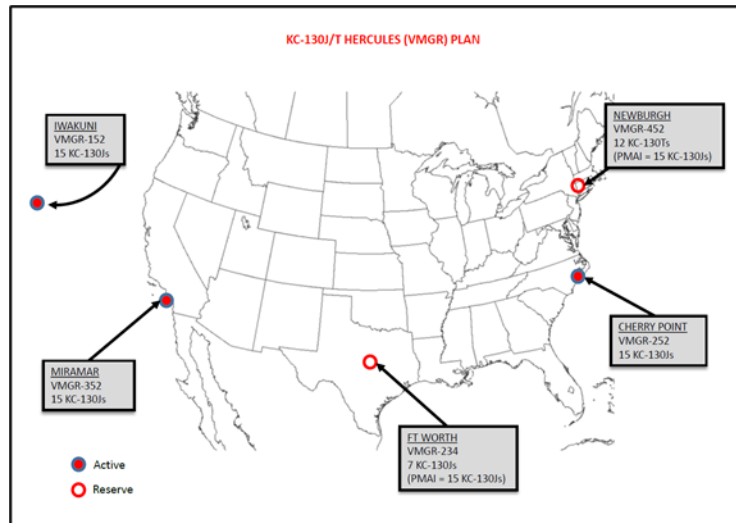


Figure 2. VMGR Location Overview. Source: USMC (2019).

Each squadron has a manning goal of 455 Marines, 298 of which comprise the maintenance department. The maintenance department is divided into 9 main divisions, each of which is specialized for a certain maintenance function. Figure 3 contains the doctrinal organization of a Marine Corps maintenance department.

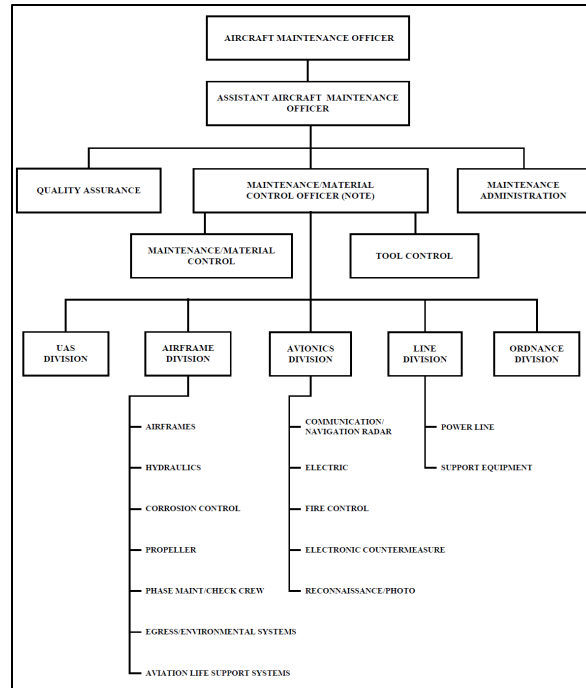


Figure 3. Generic Squadron Maintenance Organization. Source: CNAF (2017).

c. *Naval Aviation Maintenance Program*

The Naval Aviation Maintenance Program (NAMP) governs all aspects of KC-130J maintenance. The primary objective of the NAMP is to “achieve the aviation material readiness and safety standards established by the [Chief of Naval Operations (CNO)] and [Commander, Naval Air Forces (CNAF)] in coordination with the Commandant of the Marine Corps (CMC)” (Commander, Naval Air Forces [CNAF] 2017). Maintenance tasks are subdivided into three levels based on complexity and type of work required: O-level, I-level, and D-level. O-level maintenance “primarily includes inspecting, servicing, lubricating, adjusting, and replacing parts, minor assemblies, and subassemblies of aircraft” (CNAF 2017) and is typically conducted by the squadron maintenance department. These are the every-day tasks required for operation. I-level maintenance consists of more advanced maintenance actions that may require equipment that is non-organic to a squadron. It is conducted at the same base as the squadron but is performed by a third-party organization, such as the Marine Aviation Logistics Squadron (MALS). D-level maintenance is the highest echelon and is conducted at a dedicated Fleet Readiness

Center (FRC). D-level tasks include “manufacturing parts, modifying, testing, inspecting, sampling, and reclamations” (CNAF 2017). Typically, maintenance tasks are performed at the lowest level, but they may be performed by higher levels if required.

d. KC-130J Inspections

The KC-130J’s O-level preventative maintenance program is governed by the Periodic Maintenance Information Manual (NAVAIR 01–75GAJ-6). There are two primary types of inspections, special and conditional. Special inspections are conducted at a set interval, “based on elapsed calendar time, flight hours, operating hours, or number of cycles or events” (CNAF 2017). Conditional inspections consist of unscheduled maintenance tasks that must be accomplished when the aircraft meets certain requirements (such as following a bird strike or a hard landing). Time-based special inspections, also known as isochronal (ISO) inspections, are conducted at set intervals, regardless of aircraft use. These inspections range from every 35 to 1,680 days, and the reference date is the day that the squadron accepts the aircraft from the depot facility. For the 210-days-and-greater inspections, there is a plus or minus 9-day window to initiate the inspection. If the aircraft is unable to meet this inspection window due to operational requirements or an existing issue, a waiver can be granted to delay or re-baseline the inspection. Prior to 2018, the special inspections of 420 days and greater fell under a phase maintenance program and were reported as such. Phase inspections are still based on calendar day requirements but are administratively integrated with the I-level and D-level maintenance.

This thesis will focus primarily on the 420-day and 840-day special inspections. These are the largest of the ISO special inspections and delays in their completion can drastically affect aircraft availability. Due to the compounding nature of the special inspection schedule, when an 840-day inspection is due, a 420-day, 210-day, 105-day, and 35-day inspection must all be accomplished. Each inspection is defined by a “card deck” of individual inspection requirements (NAVAIR 01–75GAJ-6-3). Each *workcard* in the deck refers to a specific maintenance action, and all manpower, equipment, and time requirements are presented along with a step-by-step description of the action to be

performed. Figure 4 contains an example workcard from the 420-day inspection that has had specific information removed.

CARD NO.	NAVAIR 01-75GAJ-6-3 DATE: 1 JANUARY 2018		CHANGE NO.		SPECIAL 420 DAY	ELECT PWR: NA HYD PWR: NA
WORK AREA/ZONE	COR	RATING/MOS JOB SPECIALTY REQUIRED	CARD TIME (MANHOURS)	WORK AREA/ZONE ZONE OF AIRCRAFT	CARD TITLE	
ZONE OF AIRCRAFT		STEP-BY-STEP DESCRIPTION OF WORK HERE.				
END OF CARD						

Figure 4. Redacted Special Inspection Workcard. Source: NAVAIR (2018).

It should be noted that the workcards do not specify a specific number of maintainers required to accomplish the action, nor do they provide any guidelines for scheduling parallel maintenance actions across the aircraft. The squadron is left to determine the exact order in which the cards will be accomplished and how the parallel workload will be divided. Table 1 outlines the number of inspection cards and prescribed manhours for each of the major special inspections.

Table 1. Special Inspection Time Requirements

Inspection	Inspection Cards	Card Time (Hours)
35-Day	8	8.1
105-Day	14	69.0
210-Day	31	55.6
420-Day	70	139.1
840-Day	20	53.4

If the inspections were conducted sequentially as defined in the inspection card deck, without any parallel actions by the squadron, it would take over 40 days of 8-hour

shifts to accomplish the cumulative 840-day inspection. This time includes only the card items and does not include any discrepancies that are discovered during the inspection process.

The D-level, or depot, inspections for the KC-130J are also based on a fixed cycle of calendar days. This planned maintenance interval (PMI) cycle begins with a 95-month initial operational interval then reverts to a 60-month interval for the remainder of the aircraft's life. A brief overview of the timeline can be found in Figure 5.

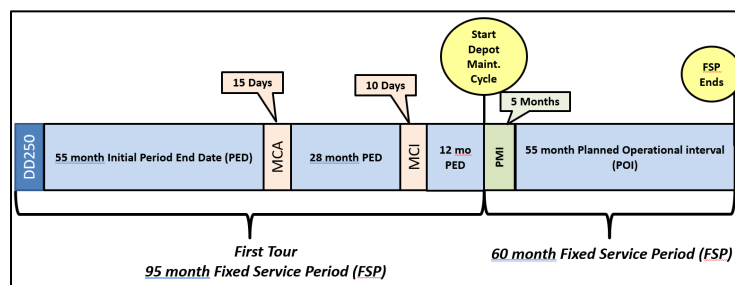


Figure 5. KC-130J Depot Cycle Overview. Source: NAVAIR (2018).

The depot maintenance is conducted at one of several civilian-manned contract facilities, the largest of which is located at Hill Air Force Base in Utah. While the prescribed timeline is approximately 5 months, or 150 days, the average time to completion has been steadily increasing, as shown in Figure 6. Steps are being taken to reduce this time, including increasing the number of available contract sites. For this reason, this thesis will only focus on the special inspections and assume that depot completion times will return to the prescribed baseline.

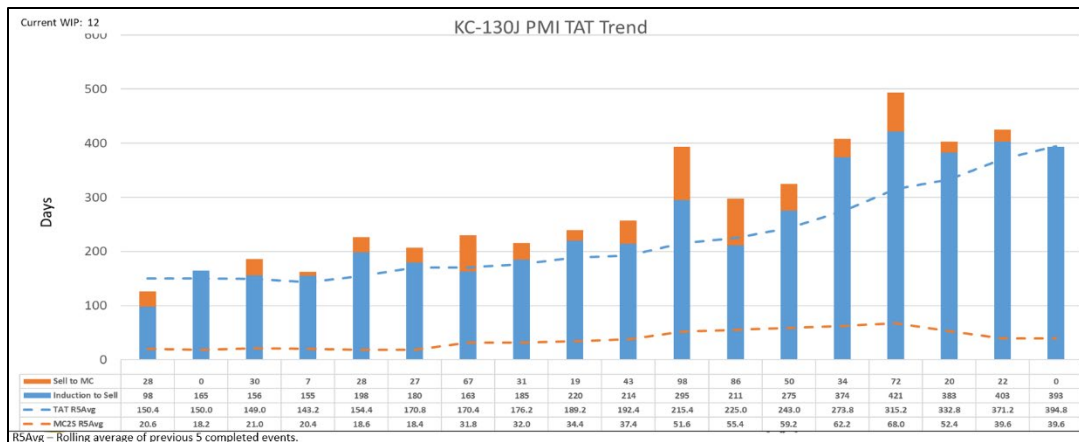


Figure 6. KC-130J Depot Maintenance Trend. Source: NAE (2020).

The most critical issue with the depot service overruns is the fact that the 60-month cycle is fixed. The scheduled date of the next depot inspection does not shift based on the amount of time taken to complete the previous depot cycle. When the depot cycle times start exceeding a year, as they have in recent instances, the time that the aircraft is available to the fleet is drastically reduced. Similar to the special inspections, a waiver process exists to potentially extend or change these dates, but it is a tool that is only relied upon in extreme circumstances.

B. QUEUING AND SIMULATION THEORY

This section covers the fundamentals associated with basic queuing theory and discrete event simulation.

1. Basic Queuing Theory

Queuing theory, simply, is the study of “customers” as they are processed through a system. Waiting in line at the grocery store, answering incoming phone calls, and even automobile manufacturing are all systems that can be modeled and analyzed using queuing theory. These customers are processed by “servers” through some event, and we typically seek to analyze the performance metrics of the system. Typical measures of interest would include the time spent in the system, the time spent waiting to start service, and the number of customers in the system. Figure 7 is the most basic representation of a queuing system.

Customers arrive from the left side, enter a queue, are processed by some number of servers, and exit to the right once processing is complete. The downward arrows indicate that customers may exit the system without ever entering the queue (balking) or after spending some amount of time in the queue (reneging).

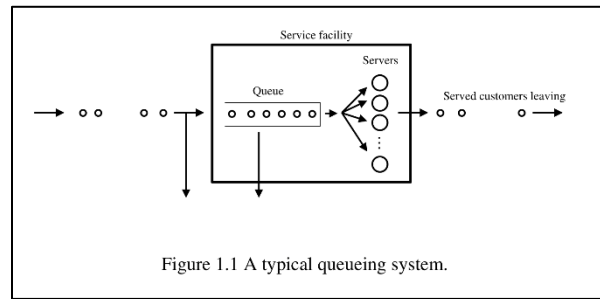


Figure 7. A Basic Queue. Source: Gross et al. (2018).

Queues are typically categorized by the arrival time process, service time process, number of servers, and the size of the queue. The notation for this method of categorization was formalized by David Kendall in 1953 (Gross et al. 2018, p. 7) and is defined by a series of letters: A/B/X/Y/Z. A represents the distribution of interarrival times. B represents the service time distribution. X is the number of servers that can process customers in parallel. Y is the maximum number of customers that can be in the system. Z is the queue discipline, or the order in which the customers are processed when they arrive. Y is typically omitted when the queue has infinite capacity, and Z is omitted when the queue is first come, first served. Figure 8 contains a summary of the basic notation used. All queues discussed for this model will be either D/G/c or G/G/c, where “G” is a generalized probability distribution that does not fit into the other categories. The model itself is a closed queueing network (entities do not exit) with both stochastic arrival and service times. It consists of two primary service stations with unlimited queue capacities and fixed server counts.

Table 1.1 Queueing notation $A/B/X/Y/Z$		
Characteristic	Symbol	Explanation
Interarrival-time distribution (A)	M	Exponential
	D	Deterministic
	E_k	Erlang type k ($k = 1, 2, \dots$)
Service-time distribution (B)	H_k	Mixture of k exponentials
	PH	Phase type
	G	General
Parallel servers (X)	$1, 2, \dots, \infty$	
System capacity (Y)	$1, 2, \dots, \infty$	
Queue discipline (Z)	FCFS	First come, first served
	LCFS	Last come, first served
	RSS	Random selection for service
	PR	Priority
	GD	General discipline

Figure 8. Queue Notation. Source: Gross et al. (2018).

Little's Law is typically used to calculate long-run parameter estimates for queueing systems. It states that the average number of customers in the system, L , is equal to the product of the average customer arrival rate, λ , and the average time the customers spend in the system, W (Gross et al. 2018, p. 10). Little's Law can be extended to very general circumstances but, given certain interarrival and service time distributions, can provide closed-form solutions for given quantities. However, for $G/G/c$ queues there are very few closed-form solutions, and simulation will be heavily relied on to observe the behavior of the system. It is important to note that a queue will only remain stable if the average arrival rate is strictly less than the average service time of the system.

2. Discrete Event Simulation

Discrete event simulation (DES) "concerns the modeling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time" (Law 2015, p. 6). Essentially, events are scheduled at fixed points in the simulation time, and events occur one at a time in the order of execution. This simulation time is referenced by a global clock that is advanced incrementally to the next scheduled event (the time of which was determined by a previous state). This is known as next-event time-advance, and it allows the model to omit periods of inactivity and focus only on the discrete event occurrences. Figure 9 demonstrates how the simulation time

jumps to discrete times determined by arrivals (As) and service times (S). No time elapses during the occurrence of the individual scheduled events.

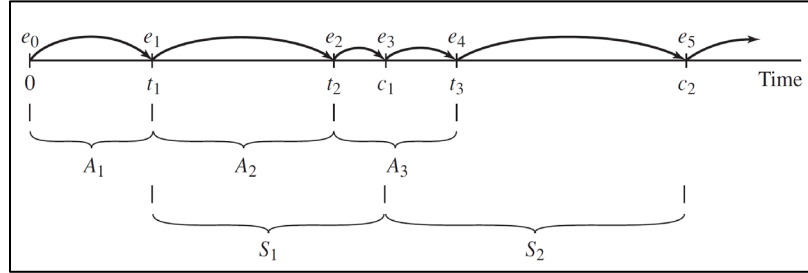


Figure 9. Discrete Event Simulation Time Jumps. Source: Law (2015).

These scheduled events are maintained on an event list, which is simply a listing of all scheduled events and the simulation time at which they occur. To provide further resolution to the model, individual entities can be created and passed among the existing states. These entities possess their own individual attributes which can be modified and recorded during a scheduled event. The model itself may also have global state variables that are used to characterize the state of the system over time. State variables are represented as piecewise constant functions that only have transitions at the scheduled event executions. Figure 10 demonstrates how a state variable, N , can vary over time. The jumps in the value are instantaneous and only change when events are executed.

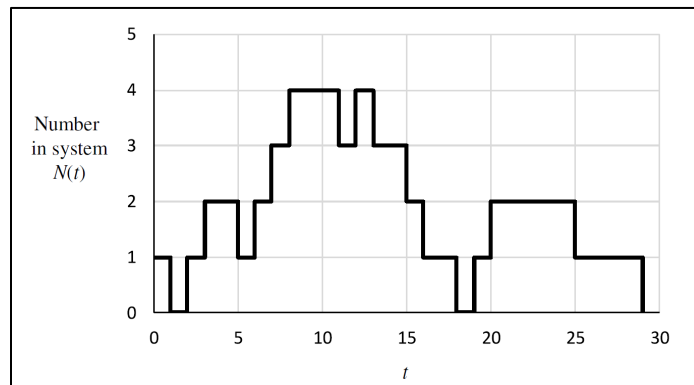


Figure 10. State Variable Changes During Discrete Event Simulation. Source: Gross et al. (2018).

The basic flow of a discrete event simulation can be found in Figure 11.

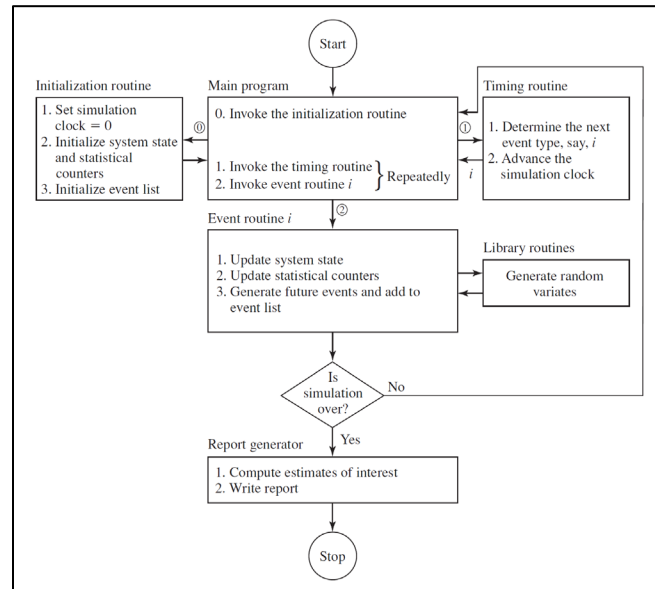


Figure 11. Discrete Event Simulation Flow Logic. Source: Law (2015).

3. Event Graphs

Event graphs are the graphical method through which discrete event simulation models are represented. They can be used to fully characterize a DES model and provide the reader with an easy method to visualize the flow of the system. Event graphs depict the individual states as circular nodes, and the nodes are connected through a series of directed edges that represent the scheduling of said events. Figure 12 represents the most basic event graph concept. Event A will schedule the new event in time t if the conditions in (i) are met. Typically, the t will be omitted if the new event is scheduled without delay.

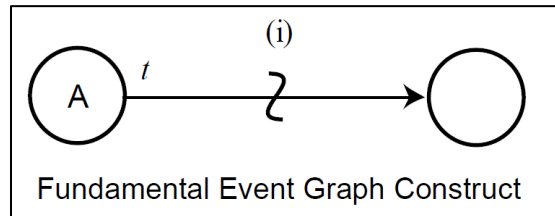


Figure 12. Basic Event Graph Concept. Source: Buss (2001).

Event graphs also depict the individual state variables that are modified with the execution of each event. These state variables appear below the state in which they are modified. During the DES process, some states may pass individual parameters between states. These parameters are depicted by their parameter name in a box on the scheduling edge, and the receiving state will list the names used for input parameters in parenthesis following the state name. Figure 13 depicts a more complicated event graph with two state variables, Q and S , and 4 states. The parameter “ i ” is passed between the states and retains its naming convention in each state. Note that this system has a self-scheduling Arrival state to represent the continuous arrival of customers to the system. Customers in this system will complete the service process a total of n times.

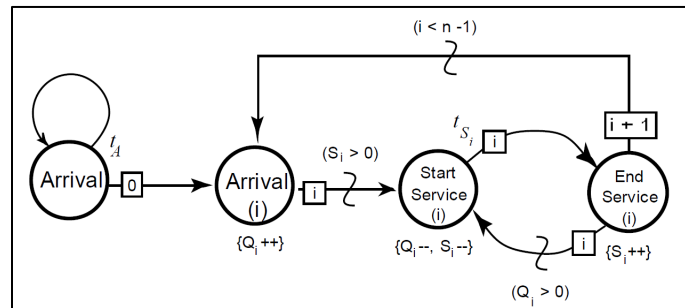


Figure 13. More Complicated Event Graph. Source: Buss (2001).

The only functions of DES that are not depicted in an event graph are the scheduling priority of events and the collection of specific output statistics not directly tied to a state variable. Due to the discrete nature of the scheduling process, it is possible that two events could be scheduled for simultaneous execution (even with using a continuous random variable for the scheduling time). In this event, the event that was scheduled first will be executed first unless the user has specified a different priority in the software. This is a common feature of DES software and should not be considered a problem. The collection of output statistics is another feature built into DES software that allows the user to measure specific actions and state variables as the model runs. For example, the time spent in a queue could be recorded by noting the time at which a customer enters the queue and

subtracting from the time at which the service is started. Individual state variables can also be recorded in this manner so that they may be analyzed following the model run.

C. RELATED STUDIES AND PRACTICES

The following studies address issues currently impacting the Department of Defense that are directly related to the problem covered by this thesis.

1. Government Accountability Office Weapon System Sustainment

In 2020, the U.S. Government Accountability Office (GAO) released report GAO-21-101SP to Congress to address the mission capable rate of aircraft within the U.S. military. Out of 46 platforms analyzed, only three were able to meet their mission capable goals in a majority of the years between fiscal year (FY) 2011 and FY 2019, and the KC-130J met its mission capable goal in zero of these years (GAO 2020). For the purpose of the study, “mission capable” meant that the aircraft was able to fly and conduct at least one of its prescribed missions (even if it was unable to perform others due to some maintenance impact). Despite the KC-130J having an average airframe age of only 11.2 years, the study found that “mission capable goals [were not met] because more aircraft were in need of age-related repairs and those repairs took longer to perform” (GAO 2020, p. 26). It was assessed that this was most likely due to the fact that the logistics support transitioned from contractor support to organic support in FY 2017. This changeover also resulted in depot maintenance costs nearly doubling over the period of the study, as shown in Figure 14.

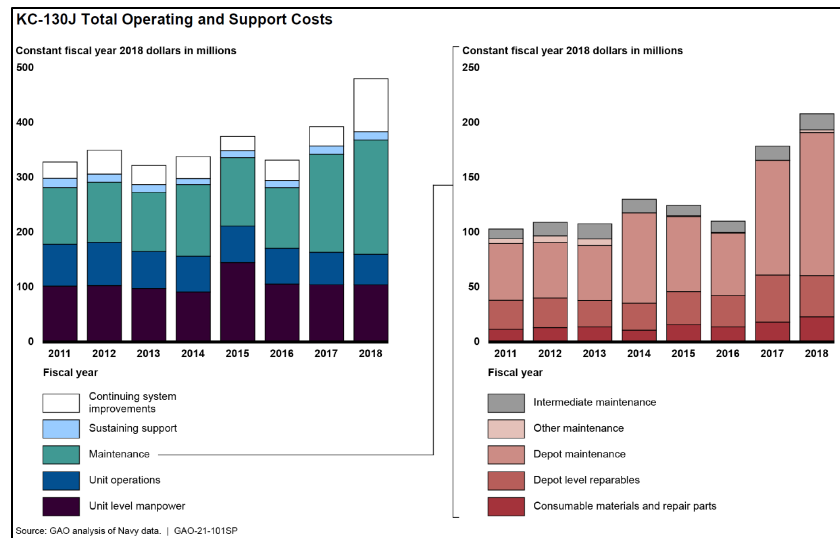


Figure 14. KC-130J Operating and Support Costs. Source: GAO (2020).

The transition to the organic supply system has had the largest impact on both maintenance costs and readiness levels. Due to the changeover, the Navy and Marine Corps have faced the challenge of obtaining technical data for the aircraft so that maintenance sustainment strategies can be updated for new trends. The biggest takeaways from this study are that the readiness issue is affecting the entire Department of Defense equally, and the Marine Corps must improve its preventative maintenance on the KC-130J.

2. RAND Air Force C-130 Logistics Report

In 2011, the RAND Corporation published *Analysis of the Air Force Logistics Enterprise: Evaluation of Global Repair Network Options for Supporting the C-130* as a part of the Project AIR FORCE analysis in order to “identify and rethink the basic issues and the premises on which the Air Force plans, organizes, and operates its logistics enterprise” (Roo et al. 2011, p. 5). This analysis was focused on alternative approaches to C-130 maintenance with options ranging from fully centralized to fully decentralized. Unlike the Marine Corps, the Air Force operates several variants of C-130, including the MC-130, AC-130, EC-130, WC-130, and HC-130, and their total inventory is several times larger than that of the Marine Corps. Air Force ISO inspections are also comprised of different maintenance items and occur every 450 days with a shorter inspection at 225 days

instead of the 420/210-day cycle employed by the Marine Corps. For these reasons, the results of the study cannot be directly translated to the Marine Corps, but they do provide significant insight into the effects of conducting centralized maintenance.

Figure 15 illustrates the manpower requirements based on the capacity of the centralized repair facility (CRF). For facilities conducting more than approximately 100 ISO inspections per year, the manpower requirement stays steady at just under 2 persons per ISO per year. However, facilities conducting fewer than 50 inspections per year see almost exponential growth for the number of personnel required. The dot on the graph portrays the actual manpower number of the maintenance facility in Little Rock, Arkansas that conducts 75 dedicated ISO inspections per year. This behavior can be explained by two reasons: First, a larger shop can better allocate its manpower in accordance with workload. Second, having a higher volume of work reduces the variations associated with large fluctuations in workloads. This prevents the long queues that can form when utilization is extremely high (Roo et al. 2011, p. 19).

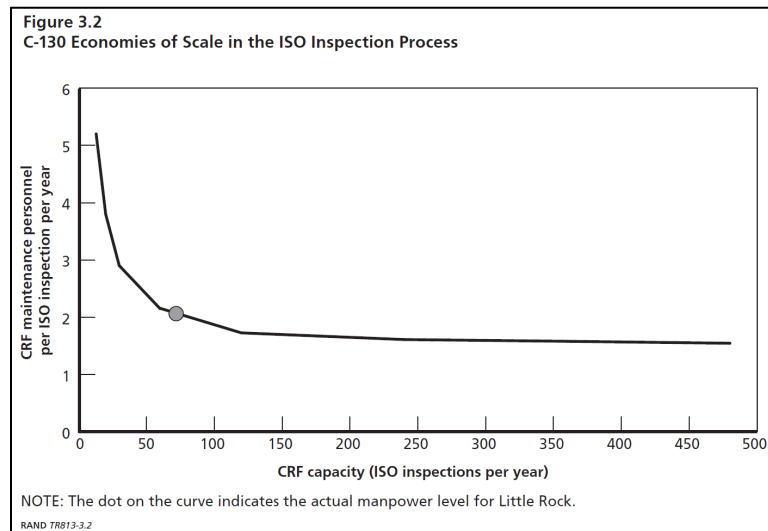


Figure 15. C-130 CRF Personnel Economy of Scale. Source: Roo et al. (2011).

The study also revealed that flow times of aircraft through the ISO process are directly related to the size of the CRF. Larger facilities are able to fill more shifts, which

allows for more continuous maintenance on the aircraft. Figure 16 shows this relationship. The square on the graph represents the output of the Air Force Special Operations Command (AFSOC) CRF located at Hurlburt Field, Florida, which is staffed almost exclusively by contractors and employs three-shift operations. This data, along with the Little Rock example in Figure 16, serves to validate the assumptions of these models.

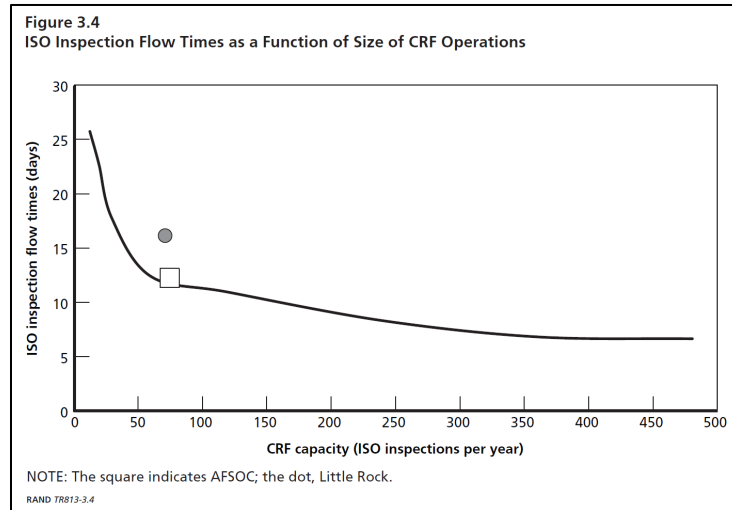


Figure 16. ISO Flow Times Versus CRF Capacity. Source: Roo et al. (2011).

The AFSOC CRF was created in 2007 through a contract with L-3 Communications to perform 75 ISO inspections per year across the five C-130 variants operated by AFSOC. RAND examined the performance benefit seen at this CRF in comparison with the historic rates produced by AFSOC ISO inspections. The results are found in Figure 17 and serve as a real-world example of the proposed RAND models. The average fly-to-fly time (time between the last flight prior to ISO and the first flight following) decreased from 35 days (standard deviation of 22 days) to 19 days (standard deviation of 9 days) when the CRF was implemented (sample size for the centralized model was 20). The RAND model's estimates were also extremely close to reality in terms of total cost (\$12 million actual vs \$11.4 million predicted) and average number of aircraft in ISO (4.0 actual vs 3.4 estimated) (Roo et al. 2011, p. 39).

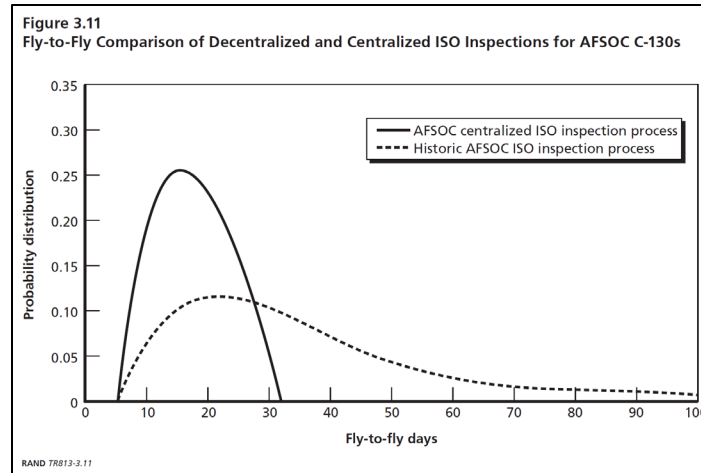


Figure 17. Fly-to-Fly Comparison Between Centralized and Decentralized ISO Inspection. Source: Roo et al. (2011).

Overall, the study concluded that the “Air Force can maintain its C-130 fleet using significantly fewer resources or can increase operational unit maintenance capabilities at a cost comparable to that of the current system by reallocating maintenance resources from unit backshops to a centralized network” (Roo et al. 2011, p. 47). Again, this study cannot be used to infer any quantitative benefit for the Marine Corps to adopt this model, but it does provide strong evidence for the benefits of centralized operations.

3. C-5 ISO Consolidation Simulation

Like the C-130, the C-5 is a military transport aircraft operated by the U.S. Air Force (USAF). As of 2008, its ISO inspections were being conducted in five centralized locations, and the Air Force Institute of Technology (AFIT) was asked to examine the effects of downsizing this to three locations (Johnson et al. 2008, p. 1). The C-5 operates on a 420-day ISO cycle, and each of the ISO locations (referred to as “docks”) was able to process one aircraft at a time. The USAF’s Air Mobility Command (AMC) determined prior to the study that a three-dock ISO system would require an estimated completion time of 14.25 days in order for the fleet of 111 aircraft to operate without excessive delays (down from the existing 18 days) (Johnson et al. 2008, p. 1). The goal of the AFIT research was to determine what could be done to improve the ISO timelines to achieve this goal and how are wait times impacted by the ISO timeline.

The discrete event simulation model crafted by the AFIT team was able to estimate both the ISO inspection process and the arrival of aircraft to the process. This allowed the team to measure both queue times and total times in the process. This model was similar to the KC-130J model in that it employed a “buffer” for depot level maintenance, utilized a 160-day depot time requirement, and the depot cycle was 5 years for the C-5A models (7 years for the C-5Bs) (Johnson et al. 2008, p. 2). The C-5s also had a very similar maintenance hours requirement for their ISO inspection (302 hours vs the approximately 325 hours for an 840-day KC-130J inspection). These similarities, while not exact, still allow insightful connections to be made between this model and the KC-130J model.

Models were run for both 3 and 4 docks utilizing 3 different scenarios: deterministic requirements for each ISO phase, a fixed 14.25-day total ISO time, and using random ISO component times based on fitted probability distributions. One of the biggest revelations from the simulation is that even for a fixed “optimal” 14.25-day ISO time, the total time spent in the ISO system was still nearly 16 days due to time spent in the queue (Johnson et al. 2008, p. 6). This indicates that the proposed 3-dock system was inadequate for the proposed minimum time. The main conclusion from the study was that if the time in ISO could not be reduced, the number of docks had to be increased to at least 4. The results of this study are useful because they provide insight on the performance metrics that could be evaluated in the KC-130J model. Namely, a balance must be found between service capacity and expected ISO time.

4. Air Force’s Current Situation

As discussed earlier, the USAF operates a CRF to perform all ISO inspections for the AFSOC C-130 models. This CRF is located in Hurlburt Field, Florida and is operated through a contract with L3. Like the USMC KC-130J fleet, the AFSOC C-130s are geographically dispersed (located in New Mexico, Pennsylvania, Florida, and RAF Mildenhall in the United Kingdom) and comparable in number (77 total aircraft serviced by the CRF) (Ford 2017). An overview of the locations can be found in Figure 18.



Figure 18. AFSOC C-130 Locations. Source: Ford (2017).

The Air Force utilizes a 540-day ISO cycle, which is broken down in Figure 19. The detailed planning cycle begins 120 days prior to ISO and allows for workload management at the CRF. It also allows for coordination between squadrons to ensure effective flow into the CRF and reduce downtime. Once in the ISO dock, the complete inspection process is scheduled for 9 days, including 3 days each of the look and fix phases. The triage meeting is used to identify any major maintenance tasks that may extend beyond the ISO inspection window, and the CRF retains responsibility for the aircraft until it can be returned in mission capable status.

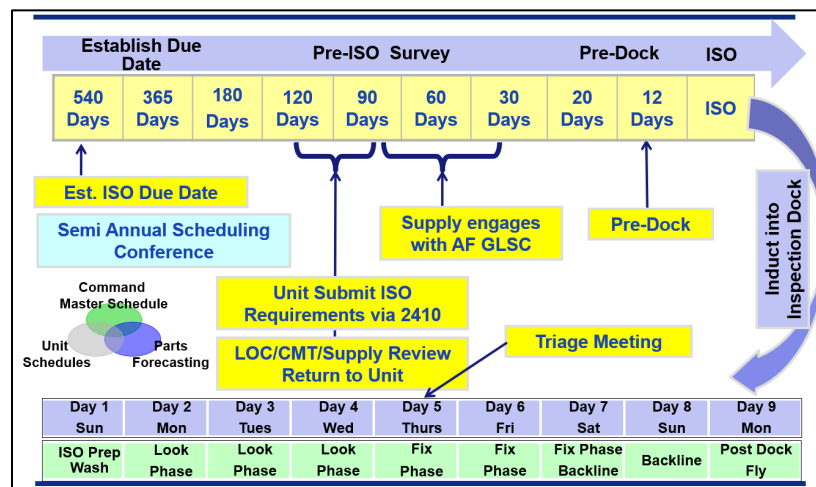


Figure 19. AFSOC C-130 ISO Timeline Flow. Source: Ford (2017).

Overall, the use of the CRF has greatly improved AFSOC aircraft availability. In a two-year period, 941 days of aircraft availability were returned to the squadrons (Ford 2017). By relieving the squadrons of the scheduled inspection responsibility, they were able to better focus on mission generation. The process has also been subject to continued refinement, which has allowed the Air Force to incorporate more concurrent maintenance tasks within the CRF and further reduce ISO times.

5. Alternative Approaches for U.S. Navy and Marine Corps

The Department of the Navy currently oversees two major programs intended to improve maintenance practices, Naval Sustainment System-Aviation (NSS-A) and Performance-to-Plan (P2P). NSS-A is an initiative that “focuses on bringing industry into the service’s maintenance centers to view the work firsthand and offer feedback” (Katz 2019). It was initiated in 2019 through a contract with the Boston Consulting Group. While no official reports have been released on the effects, the Commander, Naval Air Systems Command, Vice Admiral Dean Peters, has indicated “substantial improvements in workspace layouts, turnaround times for maintenance, backorders of high-priority requisitions that are missing from the supply shelf and planning for the future” (Peters and Taylor 2020). The NSS-A initiatives are primarily focused on the 6 depot-level Fleet Readiness Centers (FRC), which service Navy and Marine Corps aircraft across the fleet.

P2P is also a new program that was fully implemented in 2019. It is “a management approach combined with analytics to provide Commanders with a high-leverage opportunity to engage senior leadership in a data-driven discussion about performance gaps, execution barriers, and potential solutions to achieve enterprise-wide success” (Schell 2020). It accomplishes these tasks by utilizing a Driver-Based Performance Management (DBPM) system that seeks to identify the quantifiable drivers that impact the organizations desired goals. This allows leadership to directly focus on the areas that are impacting readiness. Predictive Performance Models (PPM) are incorporated into this process to allow analysis of historical trends and forecast future performance. It is important to note that this is a platform-specific system, and it must be implemented individually for each aircraft type. Currently, only the F/A-18E/F has undergone the full

P2P process, but the results have been extremely positive. The number of mission-capable aircraft pre and post P2P can be found in Figure 20.

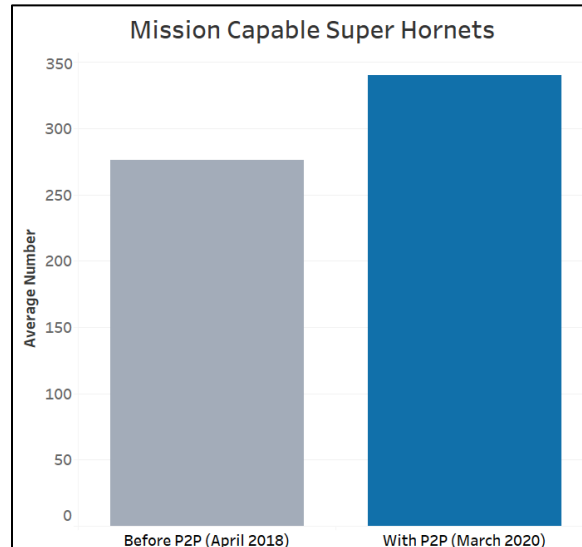


Figure 20. F/A-18 P2P Readiness Comparison. Source: U.S. Navy (2020).

The P2P process outputs identify Tier 2 and 3 Drivers that lead to the desired outcomes. Figure 21 shows the output of the P2P analysis for the F/A-18E/F. The output metrics are mission capable rate (MC) and full mission capable rate (FMC%), and each of these has several associated drivers. These drivers span the entire range of aircraft maintenance, including supply chain reform, O-level reform, and the Maintenance Operations Center (MOC). The P2P system requires detailed integration between all identified areas to be effective, but it has been shown to produce successful results.

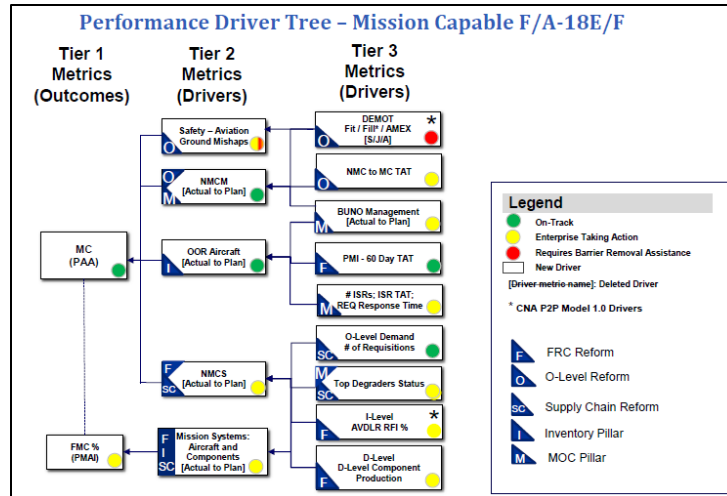


Figure 21. F/A-18 P2P Performance Driver Tree. Source: Schell (2020).

There are currently initial plans by the Center for Naval Analysis to bring the P2P process to the KC-130J platform, and it has been demonstrated to be a capable method of increasing readiness levels if implemented properly. The greatest barriers to this implementation are the high level of coordination required among all aspects of the maintenance process and the in-depth initial research required to identify the drivers.

THIS PAGE INTENTIONALLY LEFT BLANK

III. SIMULATION MODELING

This chapter provides the details of how the data was analyzed and how the model was formulated using discrete event simulation in Python. The model’s inputs and outputs are broken down, as are the methodologies used for various stages of the maintenance process. The use of current maintenance data allows the model to simulate the “worst-case” scenario of no optimization in the inspection process. This chapter is broken into two sections, covering both the model itself and the collection of the data used for inputs.

A. THE MODEL

The discrete event simulation model was written in Python utilizing the open-source DESpy package. There are a total of 15 input parameters and 8 primary output statistics. There are 11 states in the event graph. Tables 2 and 3 provide a description of each of the parameters and state variables used in the model. Due to time constraints, the model resolution is focused on the individual aircraft level and time is measured in terms of days. Aircraft are represented by a unique ID number that corresponds to their indexed location in the input lists.

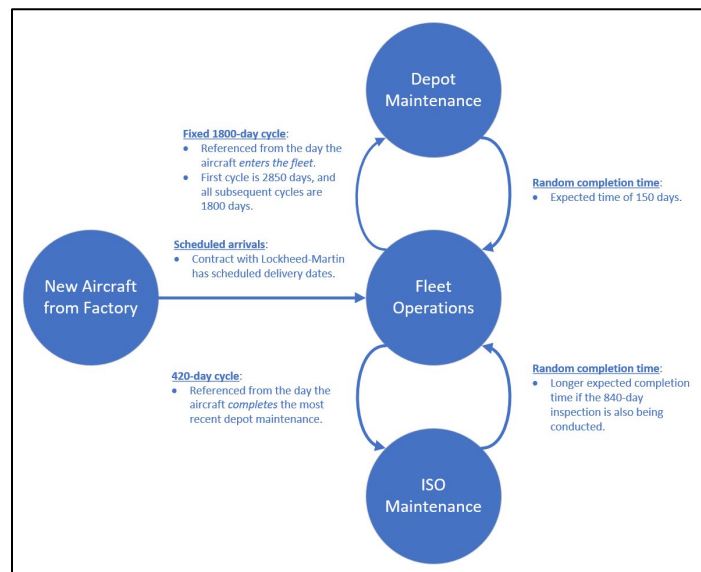


Figure 22. Basic Model Flow Chart

As shown in Figure 22, the model can be sub-divided into three primary areas: aircraft creation, ISO cycle, and depot cycle. Each of these areas will be described in-depth in their respective sections below. Once aircraft entities are created, they enter the fleet state, which represents any time the aircraft would be available to the fleet while not in scheduled depot or ISO maintenance. From the fleet, aircraft cycle between the depot-level maintenance cycle and the ISO maintenance cycle. Arrivals to the system occur at scheduled deterministic times based on contract dates for new purchases. The depot and ISO cycles represent two multi-server, first-in-first-out queues with random service times. A single server in each cycle represents the ability to conduct one inspection on one aircraft. It should be noted that there is not a strictly linear relationship between manpower requirements and inspection capacity, and the manpower utilization decreases as capacity increases. Scheduling of ISO inspections is a deterministic process based on a fixed 420-day interval since the previous depot inspection was complete. Between depot inspections, the ISO inspections will always be scheduled to begin at these set intervals, and the only exception will be if a 420-day interval is within the depot buffer of the next scheduled depot inspection. Depot inspection scheduling is also deterministic and is based on a set 60-month (1800-day) cycle baselined by the date that the aircraft arrived from the factory. The length of the service time at depot does not impact the date of subsequent depot inspections (due to maintenance contracts currently in place).

Certain assumptions were made to simplify the model. Most importantly, it was assumed that the depot-level service times would return to their contracted requirement of approximately 150 days. Since the focus was on the impact of the ISO inspection changes, the additional variability from the depot service times had to be reduced. The ISO service times are also intended to account for transit to and from the centralized facility. In reality, there could be weather or maintenance delays associated with transit to the centralized facility (especially for aircraft coming from Japan), but a fixed 2-day transit period was deemed adequate to account for this time. Since there are not prescribed parallel maintenance procedures to conduct the special inspections, each inspection was treated as a single entity that took a random amount of time to complete. It was not feasible to break the inspection into phases or manpower requirements due to the unoptimized nature of the

special inspection requirements, and the data itself was representative of the overall start and end times. The supply system was also not considered as a constraint in the model. It was assumed that a centralized facility would have access to a greater parts warehouse and would not be limited by major wait times for parts arrival. Finally, the decision was made to base the initial state of the model on the known state of the fleet opposed to using randomized depot dates for existing aircraft. Since all future and past depot dates are known, it is possible to calculate the exact state of the model for the current fleet, which serves to both reduce the “warm up” period of the simulation and to provide a more accurate representation of the system. Disregarding this known data would negatively impact the quality of the output.

A summary of all model parameters and the state variables is below.

1. Parameters/Arguments

These are passed into the model as input from the user, and remain constant throughout the simulation. They provide flexibility to explore various scenarios and examine the impact of changes to specific portions of the maintenance cycle. The names are presented exactly as they appear in the model along with a description and the Python class of the variable.

Table 2. Model Parameters

Parameter Name (in model)	Description
<code>iso_server_count</code>	ISO Server Count – The total number of servers available to conduct ISO inspections. One server can service one aircraft. (<i>integer</i>)
<code>depot_server_count</code>	Depot Server Count – The total number of servers available to conduct depot inspections. One server can service one aircraft. (<i>integer</i>)
<code>next</code>	Next Depot Date – The date (in days from the start of simulation) of the next scheduled depot inspection. These dates account for the fact that newer aircraft have a longer operational period before their first depot date. (<i>list</i>)
<code>lastDepot</code>	Last Depot Date – The date (in days from the start of simulation) of either the last completed depot inspection

Parameter Name (in model)	Description
	or the time that an aircraft entered the fleet. All values will initially be negative, except for aircraft starting the simulation in depot. <i>(list)</i>
at_depot	Starting In Depot – The days to completion of aircraft that start the simulation in depot. The values in this list will be used to schedule the “Leave Depot” event for the aircraft that start there. <i>(list)</i>
new_arrivals	New Arrival Dates – The time, in days, between each new aircraft arrival from the factory. Referenced to schedule the “New Aircraft” event. <i>(list)</i>
service_time_gen_420	420-Day Inspection Service Time Generator – A random variable used to represent the time required for a 420-day inspection. <i>(RandomVariate)</i>
service_time_gen_840	840-Day Inspection Service Time Generator – A random variable used to represent the time required for an 840-day inspection. <i>(RandomVariate)</i>
depot_time_gen	Depot Inspection Service Time Generator – A random variable used to represent the time required for a depot inspection. <i>(RandomVariate)</i>
depot_buffer	Depot Scheduling Buffer – The time, in days, used to provide separation between ISO and depot scheduling. <i>(integer)</i>
lb420	420-Day Inspection Lower Bound – The lower-bound for the 420-day inspection. Allows the model to accept service time probability distributions that produce values of 0 or lower. <i>(integer)</i>
ub420	420-Day Inspection Upper Bound – The upper-bound for the 420-day inspection. Allows the model to accept service time probability distributions that produce values up to infinity. <i>(integer)</i>
lb840	840-Day Inspection Lower Bound – The lower-bound for the 840-day inspection. Allows the model to accept service time probability distributions that produce values of 0 or lower. <i>(integer)</i>
ub840	840-Day Inspection Upper Bound – The upper-bound for the 840-day inspection. Allows the model to accept service time probability distributions that produce values up to infinity. <i>(integer)</i>
transit	Transit Time – Used to account for the total time that would be required to transit to the centralized facility and back. <i>(integer)</i>

2. State Variables

The state variables are all internal to the model. They are created upon initialization, and they are used to track performance statistics and provide functionality to the model. After the model completes its run, the final status of each can be examined to provide additional insight.

Table 3. Model State Variables

Variable Name (in model)	Description
S	Servers – Number of available servers to perform ISO inspections. Decrementd when an aircraft enters service and incremented upon completion. <i>(integer)</i>
SD	Servers Depot – Number of available servers to perform depot inspections. Decrementd when an aircraft enters service and incremented upon completion. <i>(integer)</i>
FQ	Fleet Queue – Contains all aircraft IDs for aircraft currently available to the fleet. <i>(deque)</i>
IQ	ISO Queue – Contains all aircraft IDs for aircraft currently waiting to begin ISO. All IDs are added and removed based on first-come, first-served logic. <i>(deque)</i>
DQ	Depot Queue – Contains all aircraft IDs for aircraft currently waiting to begin depot. All IDs are added and removed based on first-come, first-served logic. <i>(deque)</i>
inDepot	In Depot – Logs the sim time that the aircraft enters the depot process. Used to calculate the time in queue and time in system statistics for the depot inspection. <i>(list)</i>
inISO	In ISO – Logs the sim time that the aircraft enters the ISO process. Used to calculate the time in queue and time in system statistics for the ISO inspection. <i>(list)</i>
inFleet	In Fleet – Logs the sim time that the aircraft enters the fleet. Used to determine the total time spent in the fleet when the aircraft enters ISO or depot. <i>(list)</i>
totalInFleet	Total Time In Fleet – Used to store the elapsed time that the aircraft has spent in the fleet. Updated upon entry into the depot or ISO processes. Used to determine total percentage of time the aircraft was available. <i>(list)</i>
id_number	ID Number – The ID number of each individual aircraft. All lists are indexed based on the order in which the aircraft are created. Incremented after each aircraft is added. <i>(integer)</i>

Variable Name (in model)	Description
F	Currently In Fleet – The total number of aircraft currently available in the fleet. <i>(integer)</i>
I	Currently In ISO – The total number of aircraft in any stage of the ISO process. <i>(integer)</i>
D	Currently In Depot – The total number of aircraft in any stage of the depot process. <i>(integer)</i>
record_stats	Record Statistics – A flag used to signal when to begin recording the “F” statistic. Flipped to “True” after all initial fleet aircraft are added. <i>(logical)</i>
lateToDepot	Late to Depot Flag – Used to count the number of times an aircraft’s ISO inspection over-runs its depot start date. <i>(integer)</i>
start_fleet_count	Starting Fleet Count – The number of aircraft that start the simulation in the fleet. <i>(integer)</i>
start_depot_count	Starting Depot Count – The number of aircraft that start the simulation at the depot inspection. <i>(integer)</i>
starting_total_ac	Starting Total Aircraft Count – The total number of aircraft currently available when the simulation starts. The sum of starting fleet count and starting depot count. <i>(integer)</i>
ending_total_ac	Ending Total Aircraft Count – The total number of aircraft that will be created for the model. The starting total plus the number of new arrivals. <i>(integer)</i>
nextISO	Days Until Next ISO – The days until the next ISO inspection from the current sim time. Calculated when aircraft enter the fleet. <i>(integer)</i>
nextDepot	Days Until Next Depot – The days until the next depot inspection from the current sim time. Calculated when aircraft enter the fleet. <i>(integer)</i>

3. Event Graph

The discrete event simulation model can be completely characterized by the event graph in Figure 23. Each circular state corresponds to an event that can be scheduled during the discrete event simulation process. The directed arcs running between events represent scheduling edges, indicating the scheduling relationship between events. When an event is scheduled, the call to execute the event will be placed onto the future event list at a scheduled time along with any arguments passed into the state. The blue statements in parenthesis above the scheduling edges represent logical checks that determine whether to schedule the event.

a. Aircraft Creation

The Aircraft Creation sub-group consists of the Run, Create Fleet, Start In Depot, and New Arrival states. The primary purpose of this section is to initialize the model based on the input parameters and to populate all variables based off the id number of each aircraft. The portion of the event graph depicting this sub-group is found in Figure 24.

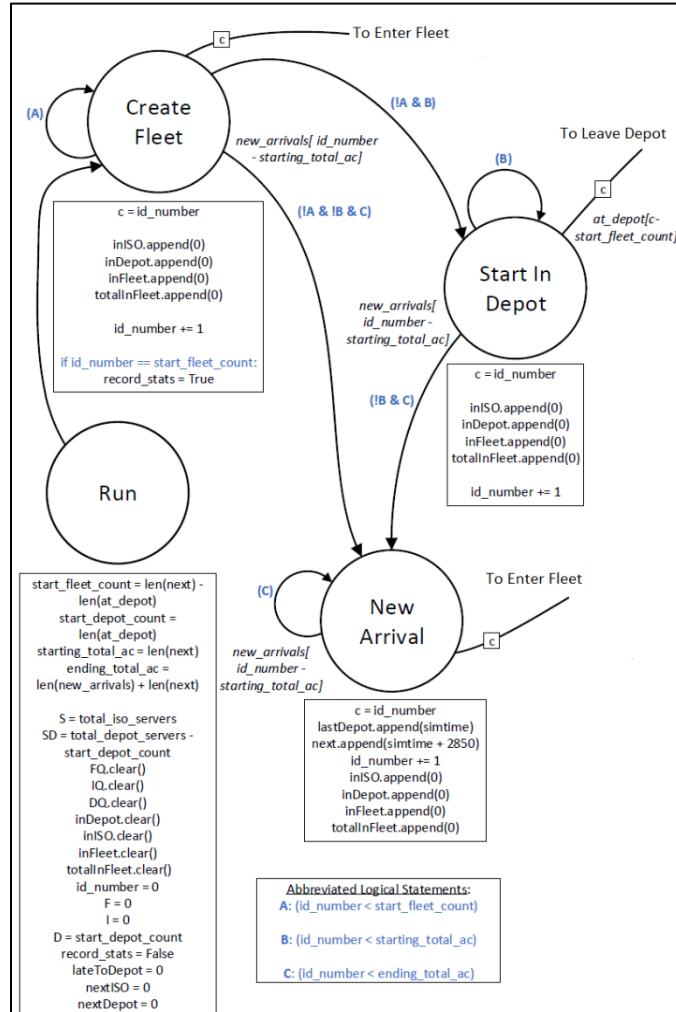


Figure 24. Aircraft Creation Event Graph Sub-Section.

The Run state initializes all state variables and schedules the initial Create Fleet event. All 4 of the count variables (`start_fleet_count`, `start_depot_count`, `starting_total_ac`, and `ending_total_ac`) are calculated based on the length

of the input lists to remove any need to enter redundant counts. These variables are used exclusively in the Aircraft Creation portion to determine which state should be scheduled. SD (the number of depot servers) will be set to the `depot_server_count` input minus the number of aircraft starting in depot, which could produce a negative number of servers. This is due to the fact that current depot levels may be elevated, and the demand may not always be reflected in future depot availability. However, once the starting aircraft exit depot, the count will return to a non-negative value.

Create Fleet will initialize all aircraft that are not currently in the depot process. Since the input lists, `next` and `lastDepot`, are already populated with scheduling data, the ID number will correspond to the index of each element. The lists used to track time spent in each state are then created by appending a new entry for each aircraft. Create Fleet will immediately schedule arrivals into the Fleet and pass the ID number as the parameter `c`. Until the ID number meets the number of aircraft starting in the fleet, Create Fleet will be continuously scheduled with zero delay. Once the initial fleet count is met, the `record_stats` flag will trip to True to all the recording of certain statistics in the Enter Fleet state, and either Start In Depot or New Arrival will be scheduled.

Start In Depot is used to initialize all aircraft that begin the simulation in depot. The state is almost identical to Create Fleet, but instead of scheduling Enter Fleet it will schedule the Leave Depot state based on the `at_depot` list. This list is indexed by the difference between the current ID number and `start_fleet_count`. If there are remaining aircraft starting in depot, the state will self-schedule with zero delay until all are created. From there, the New Arrivals state will be scheduled if aircraft will be arriving in the future.

The New Arrival state is used to create a “new” aircraft as it would exist from the factory. Both the `lastDepot` and `next` lists are expanded in this state. The current simulation clock time is appended to `lastDepot`, and the clock time plus 2850 days is appended to `next`. This represents the fact that aircraft from the factory have a 95-month initial operating period, as opposed to the 60-month period following a completed depot inspection. Subsequent arrival times are stored in the `new_arrivals` list, which is

indexed by the difference in the id number and `starting_total_ac`. The Enter Fleet state will be scheduled with no delay for each new arrival created. Once the last new aircraft is created, the Aircraft Creation process will be complete.

b. ISO Cycle

The ISO cycle is represented by a first-in-first-out queue with unlimited queue capacity and fixed server counts. The applicable portion of the event graph can be found in Figure 25.

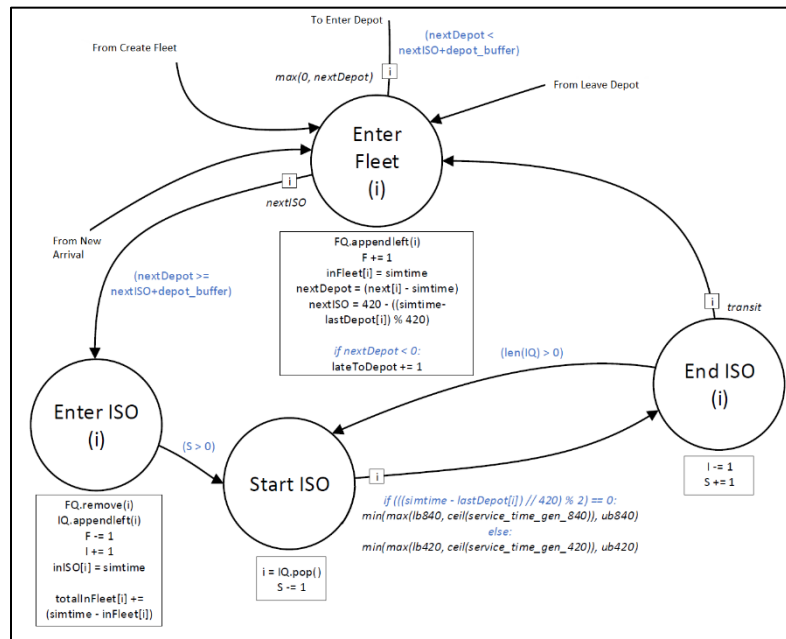


Figure 25. ISO Cycle Event Graph Sub-Section.

Aircraft enter the ISO cycle from the Enter Fleet state based on a fixed 420-day period that begins on the day the aircraft is returned from depot maintenance. Aircraft arrive to the Enter Fleet state from depot, the factory, or a completed ISO cycle. When aircraft arrive to the Enter Fleet state, the days until the next ISO and next depot inspections are calculated. The time until the next ISO inspection is calculated by the following equation: $420 - (((\text{current sim time}) - (\text{last complete depot time})) \% 420)$, where “%” is the modulo operator. This allows the aircraft to maintain the fixed 420-day interval even

when the time spent in ISO is random. A logical check is conducted to determine if the next depot inspection is due before the next ISO plus a buffer, and, if false, an ISO inspection will be scheduled.

Aircraft enter the Enter ISO state at the scheduled interval, at which time their index is removed from the fleet queue (`FQ`) and added to the left side of the ISO queue (`IQ`). The `inISO` list is updated to the current time so that it can be used to calculate the time that the aircraft spends waiting to begin ISO and the total time spent in ISO. The time that the aircraft spent in the fleet is recorded in the `totalInFleet` list to reflect the time that the aircraft was available to the fleet. If an ISO server is available, the aircraft immediately enters service via the Start ISO state.

Start ISO pops off the rightmost element of the ISO queue (recall that arriving aircraft are appended to the left), occupies a server, and schedules the End ISO state. The completion time is drawn from either the 840-day service time generator or the 420-day service time generator. This distinction reflects the fact that 840-day inspections are more involved than the lone 420-day inspection, and they require different service times to accurately reflect this difference. This determination is made by taking the difference in the current sim time and the last depot time, then seeing if the integer division result is odd or even. 840-day inspections are scheduled on even intervals, and 420-day inspections are scheduled on odd intervals. Due to the nature of the service time probability distributions, there are both upper and lower bounds for each inspection time. This allows the model to account for outliers while also staying true to the general input distribution.

The End ISO state returns the aircraft to the fleet after the specified transit time, frees up the utilized server, and schedules the Start ISO state if the ISO queue still contains waiting entities. This ensures all aircraft that enter the ISO queue are processed, regardless of the number of servers available.

c. Depot Cycle

The depot cycle behaves nearly identically to the ISO cycle. It is also represented by a first-in-first-out queue with unlimited queue capacity and a fixed number of servers. The applicable portion of the event graph can be found in Figure 26.

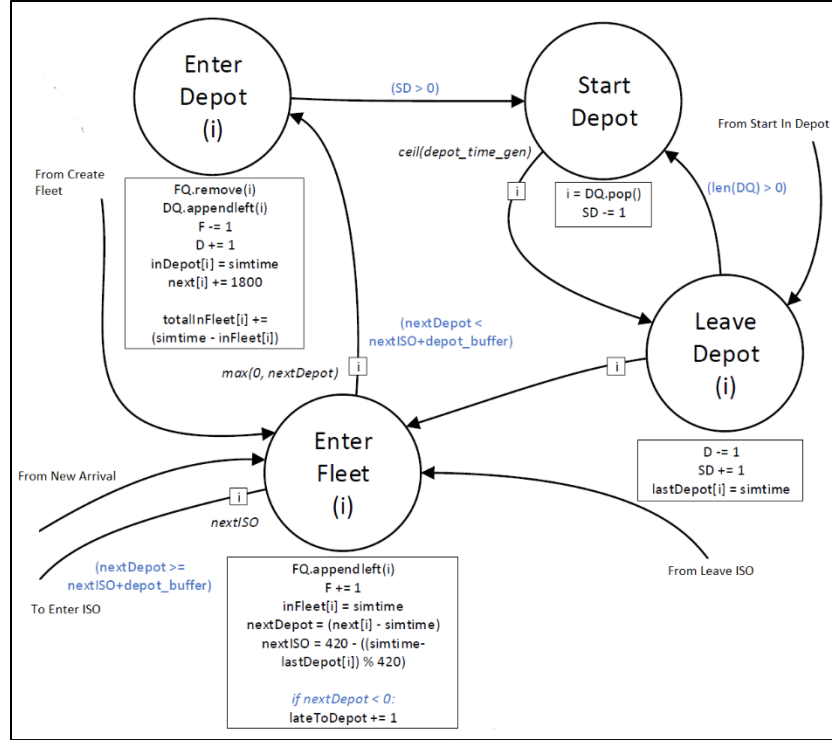


Figure 26. Depot Cycle Event Graph Sub-Section.

The arrivals are also scheduled based on a fixed 1800-day (2850-day for new aircraft) schedule that is not sensitive to the amount of time required to complete the depot process once it is started. It will always be due for a depot inspection based on this original timeline (2850 days, then 1800 days between all subsequent inspections). Each aircraft's scheduled depot start date is stored in the `next` list. Start dates are stored in a list for two reasons: First, the actual fleet depot dates are known and can be passed into the model for existing aircraft. Second, waiting to schedule the depot inspection until it is due avoids the potential situation where an aircraft spends excessive time in ISO and misses a previously scheduled depot start date.

When aircraft arrive to the Enter Fleet state, the days until the next ISO and next depot inspections are calculated. A logical check is conducted to determine if the next depot inspection is due before the next ISO plus a depot buffer. If so, the next ISO will not be scheduled, and the depot inspection will be scheduled in its place. The buffer is implemented because current fleet practice is to delay the 420-day special inspection if it

is scheduled to occur within a certain time of the depot (usually within 30–60 days) because the ISO inspection items will be handled during the depot inspection. In the rare event that an ISO inspection overruns the scheduled depot date anyways, the next depot will be scheduled immediately, and the `lateToDepot` flag will be incremented to keep track of the number of occurrences. This is not a typical event in reality, as all efforts will be made to get an aircraft to the depot by the scheduled date. Simply noting the occurrence allows the simulation to still accept any distribution of ISO completion times without leading to a conflict with the depot process.

Aircraft enter the `Enter Dept` state at the scheduled interval, at which time their index is removed from the fleet queue (FQ) and added to the left side of the depot queue (DQ). The `inDepot` list is updated to the current time so that it can be used to calculate the time that the aircraft spends waiting to begin depot and the total time spent in depot. The time that the aircraft spent in the fleet is recorded in the `totalInFleet` list to reflect the time that the aircraft was available to the fleet. The next depot date is updated by incrementing the current date by 1800. This maintains the fixed 1800-day cycle even if an ISO inspection overruns or the depot process takes longer than expected. If a depot server is available, the aircraft immediately enters service via the `Start Depot` state.

`Start Depot` pops off the rightmost element of the depot queue (recall that arriving aircraft are appended to the left), occupies a server, and schedules the `Leave Depot` state with a time generated from the depot time generator. `Leave Depot` will free up the utilized depot server, record the sim time of the completion, and immediately return the aircraft to the fleet. If the depot queue has aircraft waiting, the `Start Depot` event will be scheduled immediately in order to reduce the queue and ensure all aircraft are serviced.

4. Statistics Collected

The `DESpy` package provides the means to monitor the model's state variables through creation of set output statistics. Two different classes of statistics are utilized with this model, `SimpleStatsTally` and `SimpleStatsTimeVarying`. Tally statistics are updated each time a new observation of a desired variable occurs and provide an average of the variable taken over all the observations. For example, the time spent in a

queue is observed as entities exit the queue, and the tally statistic contains the running average, standard deviation, and extreme values. Time varying statistics, on the other hand, are utilized to track the status of a set state variable over time and report the temporal average of the state variable. For example, the number of entities in a system always has a value, and as the value changes the time varying statistic keeps track of how much time was spent at each level. In addition to observing the running total, the time varying statistic class has been modified to record the time and value of every observation in order to allow for trend visualization. These model statistics are initialized before the model runs, but the model must be told when to “collect” them through a `notify_state_change` method built into the model. There is a total of 7 of these statistics built into the model, and Table 4 provides a summary of each.

Table 4. Model Output Statistics

Statistic Name	Name in Model	Observed Value	Description
fleet_stat	fleet	F	Number in Fleet – Recorded any time an aircraft enters or leaves the Enter Fleet state after the initial fleet is created. <i>(time varying)</i>
iso_stat	iso	I	Number in ISO – Recorded any time an aircraft enters or exits the ISO cycle. <i>(time varying)</i>
depot_stat	depot	D	Number in Depot – Recorded any time an aircraft enters or exits the depot cycle. <i>(time varying)</i>
depot_TIQ	dTIQ	simtime inDepot [i]	Time in Depot Queue – Recorded when an aircraft enters the Start Depot state. Represents the time spent waiting for depot to begin. <i>(tally)</i>
depot_TIS	dTIS	simtime inDepot [i]	Time in Depot – Recorded when an aircraft enters the Leave Depot state. Represents the total time spent in the depot cycle (queue and service time). <i>(tally)</i>
iso_TIQ	iTIQ	simtime inISO[i]	Time in ISO Queue – Recorded when an aircraft enters the Start ISO state.

Statistic Name	Name in Model	Observed Value	Description
			Represents the time spent waiting for ISO to begin. <i>(tally)</i>
iso_TIS	iTIS	simtime inISO[i]	Time in ISO – Recorded when an aircraft enters the End ISO state. Represents the total time spent in the ISO cycle (queue and service time). <i>(tally)</i>

In addition to the above statistics, the `totalInFleet` state variable is used to keep track of the total time an aircraft spends in the fleet. It is updated any time an aircraft exists the fleet, and it contains a cumulative total of each aircraft's time in the fleet. Following the completion of the model run, this variable is used to calculate the percent of time that an aircraft spends in the fleet (after accounting for the time an aircraft entered the model and how long the aircraft currently in the Enter Fleet state had been there when the model ended).

B. THE DATA

Data was collected for two purposes: First, active-duty maintenance data was collected to provide an accurate input distribution for the special inspection timelines. Second, the depot timelines were collected for all KC-130J aircraft in order to create the model input parameters.

1. Inspection Time Data

Current maintenance data was pulled from Naval Air Systems Command's (NAVAIR) DECision Knowledge Programming for Logistics Analysis and Technical Evaluation (DECKPLATE) Aircraft Inventory and Readiness Reporting System (AIRRS). DECKPLATE consolidates all maintenance reports from the squadron level into a single database and is "the authoritative source for monthly flight summarized data, [type equipment codes (TECs)], Organizational Codes, and aircraft inventory of the Navy and Marine Corps" (NAVAIR 2019). At the squadron level, any maintenance action has an associated maintenance action form (MAF) submitted through the Optimized Organization

Maintenance Activity (OOMA) software. These MAFs contain data about the maintenance action including aircraft serial number, initiation and completion dates, manhours, work center involved, a brief description of the maintenance action, and several organizational codes. The MAFs are uploaded from OOMA to DECKPLATE, where they can be aggregated and filtered.

Data was pulled from May 2018 (the first instance of the 420-day Special Inspection being implemented) through February 2021 for all active-duty KC-130J squadrons. The following “Type MAF Codes” were used to identify special inspections: SC (Special Inspection Control Work Order), SL (Special Inspection Look Phase Work Order), and SF (Special Insp Fix Phase Maintenance Action). Each special inspection has an overall SC MAF, henceforth known as the “mother” MAF, used to initialize the inspection. SL MAFs are created for each “look” portion where the aircraft is inspected for issues by each work center. Finally, SF MAFs result from any discovered discrepancies that need to be repaired. From these MAFs, the 420-day and 840-day inspections can be filtered by their individual type codes (030000N and 030000Q, respectively). Overall, 16,999 individual MAFs with 13 data fields were pulled and exported as a tab-delimited text file. A sample of the raw data format is included in Figure 27.

	Type MAF Code (OOMA)	WUC	Action Org Code	WC Code	TEC	Bu/SerNo	Rcvd Date	Comp Date	Manhours	Descrpt Narr	In Work EOC Code	Up/Down Ind	Check Flight Repl
1	SC	030000B	FD3	020	ACMJ	166762	2019-06-03	2019-06-04	0	PERFORM SPECIAL 28 DAY MAU-12/A INSPECTION	NA	U	N
2	SC	030000B	FD3	020	ACMJ	166762	2019-07-01	2019-07-01	0	PERFORM SPECIAL 28 DAY MAU-12/A INSPECTION	NA	D	N
3	SC	030000B	FD3	020	ACMJ	166762	2019-07-25	2019-07-26	0	PERFORM SPECIAL 28 DAY MAU-12/A INSPECTION	NA	U	N
4	SC	030000B	FD3	020	ACMJ	166762	2019-08-22	2019-08-23	0	PERFORM SPECIAL 28 DAY MAU-12/A INSPECTION	NA	U	N
5	SC	030000B	FD3	020	ACMJ	166762	2019-09-24	2019-09-24	0	PERFORM SPECIAL 28 DAY MAU-12/A INSPECTION	NA	D	N
6	SC	030000C	FD3	020	ACMJ	165737	2018-05-21	2018-05-21	0	PERFORM SPECIAL 35 DAY ACFT INSPECTION	NA	U	N
7	SC	030000C	FD3	020	ACMJ	165737	2018-06-27	2018-06-28	0	PERFORM SPECIAL 35 DAY ACFT INSPECTION	NA	U	N
8	SC	030000C	FD3	020	ACMJ	165737	2018-08-02	2018-08-02	0	PERFORM SPECIAL 35 DAY ACFT INSPECTION	NA	U	N
9	SC	030000C	FD3	020	ACMJ	165737	2018-08-21	2018-08-22	0	PERFORM SPECIAL 35 DAY ACFT INSPECTION	NA	U	N

Figure 27. Raw DECKPLATE Data Example

The 420-day inspections often had separate reports for each engine’s inspection, and after removing these duplicates, there were 72 420-day mother MAFs and 31 840-day mother MAFs. The completion times, shown in Figure 28, follow a long-tailed distribution

that is typical of repair times. However, there are many extreme outliers that may not be a result of actual inspection items.

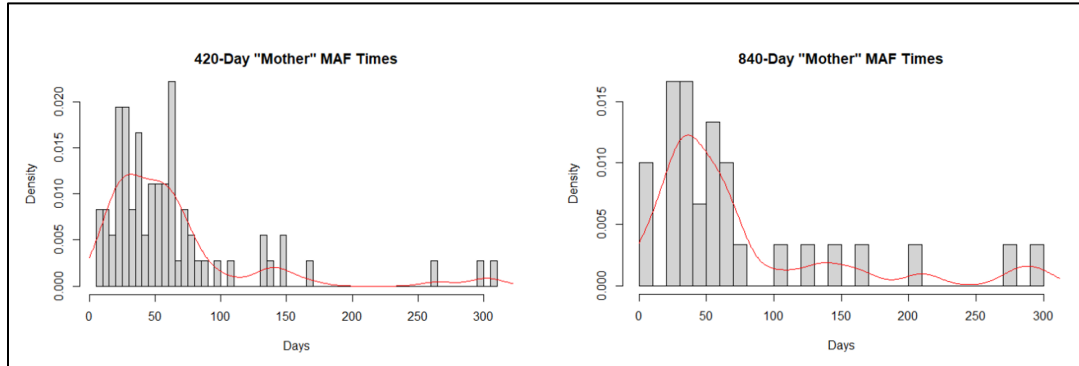


Figure 28. Raw 420-Day and 840-Day Mother MAF Times

In certain instances, downing discrepancies are discovered during the inspection but are not completed until after the “mother” MAF has been signed off as completed. These downing MAFs can extend the inspection time past what the “mother” MAF indicates, so it is necessary to examine all downing MAFs that were opened during the inspection timeframe. For each “mother” MAF, all downing PF MAFs opened were examined, and any that extended beyond the “mother” MAF completion date were recorded as the actual completion date. This reflects the fact that even though an aircraft’s inspection may have been signed off, it was not able to perform flight duties until the discovered discrepancy was fixed. A total of 20 overruns were discovered for 420-day inspections, and 13 were discovered for 840-day inspections. Table 5 shows the effects of including these MAFs (note that outliers and erroneous values are still present here).

Table 5. 420 and 840-Day Inspection Times with Overruns Included

Times (days)	Minimum	1st Quartile	Median	Mean	3rd Quartile	Max
420-Day “Mother” Times	5.00	27.00	49.50	63.26	69.50	307.00
420-Day with Overruns	5.00	29.50 (+2.70)	54.00 (+4.50)	67.40 (+4.14)	71.25 (+1.75)	307.00

Times (days)	Minimum	1st Quartile	Median	Mean	3rd Quartile	Max
840-Day “Mother” Times	5.00	30.25	50.00	74.70	70.75	297.00
840 Day with Overruns	6.00 (+1.00)	34.50 (+4.25)	58.50 (+8.50)	87.00 (+12.30)	111.80 (+41.00)	300.00 (+3.00)

The final step in cleaning the data was to identify which 420-day inspections were conducted concurrently with an 840-day inspection. Due to the cumulative nature of the special inspection process, a 420-day inspection is performed at the same time as an 840-day inspection. However, since the model treats the two inspections as separate events, we are only interested in the primary inspection that was occurring. Each of the 420-day inspections was cross-referenced with the 840-day inspections for the same airframe. If the dates overlapped, the 420-day data was omitted. This process brought the total number of 420-day inspections down to 43, which makes sense given there were 30 840-day inspections that all had a corresponding 420-day inspection.

Once the data was cleaned, it was exported to CSV format, and JMP, a statistical analysis software, was used to fit a continuous probability distribution to the data for each inspection. For the 840-inspections, the 4 highest times were removed due to being from long-term-down aircraft. The results of these fits can be found in Figure 29. For each set of data, all available continuous distributions were fit. For both inspections, the Gamma distribution provided the best fit based on the corrected Akaike’s Information Criterion (AICc).

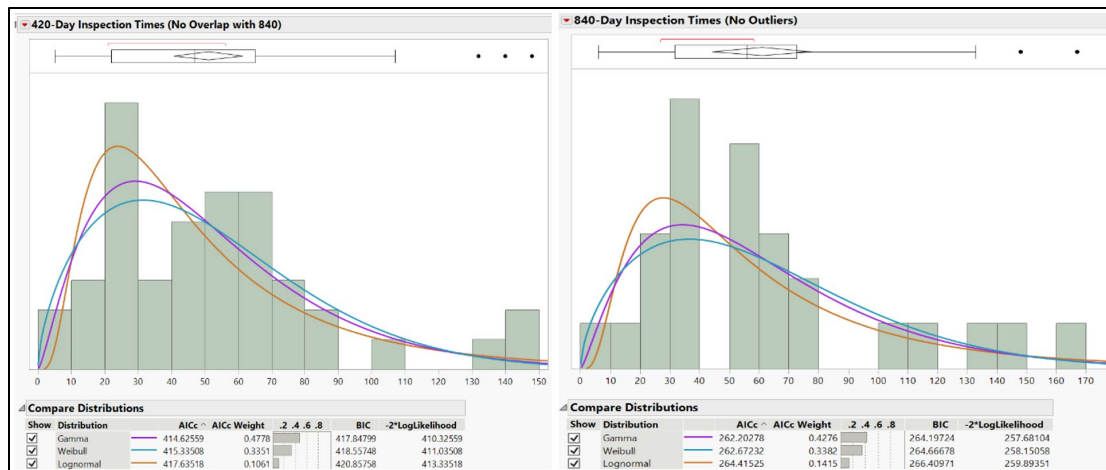


Figure 29. Final 420 and 840-Day Inspection Times with Fit Distributions

Summary statistics for the final data are below along with the parameters for the fitted distributions can be Table 6 and Figure 30, respectively.

Table 6. Final 420 and 840-Day Inspection Time Summary Statistics

	420-Day	840-Day
Observations	43	26
Mean	51.14 Days	61.04 Days
Standard Deviation	33.67 Days	41.14 Days
Lower Quartile	22 Days	31.5 Days
Median	47 Days	56 Days
Upper Quartile	65 Days	72.75 Days
Minimum	5 Days	6 Days
Maximum	148 Days	167 Days

Fitted Gamma Distribution					
Parameter		Estimate	Std Error	Lower 95%	Upper 95%
Shape	α	2.3326231	0.4715356	1.5324118	3.3902931
Scale	σ	21.923617	4.9429717	14.554559	35.464588
Measures					
-2*LogLikelihood		410.32559			
AICc		414.62559			
BIC		417.84799			

Fitted Gamma Distribution					
Parameter		Estimate	Std Error	Lower 95%	Upper 95%
Shape	α	2.2809523	0.5922086	1.3178862	3.6590728
Scale	σ	26.760078	7.7683047	15.961612	50.548189
Measures					
-2*LogLikelihood		257.68104			
AICc		262.20278			
BIC		264.19724			

Figure 30. 420- (Top) and 840-Day (Bottom) Fitted Gamma Distributions

2. Model Initialization Data

Since the true depot completion dates and future depot scheduling are available for the entire KC-130J fleet, the decision was made to utilize the data to initialize the model. These dates, along with other scheduled upgrades, intra-squadron transfers, and new arrivals, are tracked in a master spreadsheet. Figure 31 contains a small snapshot of the data, which is tracked from FY20 until FY38. The “FID” column represents the month and fiscal year of the next scheduled depot inspection (dates in parenthesis indicate that the date has been changed through the waiver process). The “BUNO” is the aircraft serial number, which is unique to each aircraft (BUNOs redacted in this sample). The last column indicates either the date that the aircraft arrived at the fleet from the factory (for new aircraft) or the date of the last completed depot inspection. Aircraft currently in depot are indicated by an orange background with the projected completion date.

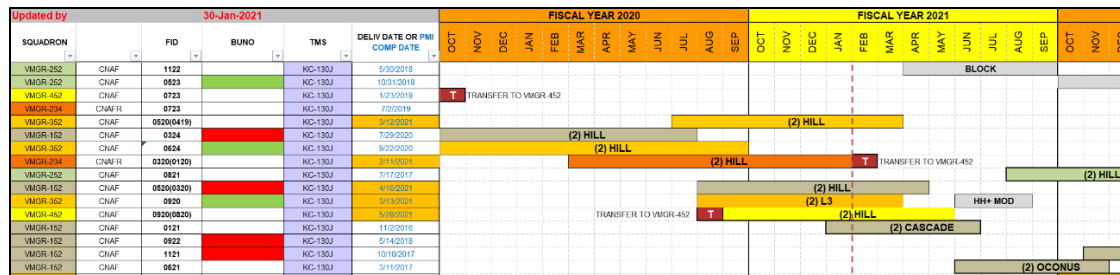


Figure 31. Sample Depot Scheduling Data. Source: Friedlein (2021).

Using the data, it was possible to break the aircraft into three groups: those currently in the fleet, those currently at depot, and the projected future arrivals. For aircraft currently in the fleet, `lastDepot` (the date of the last depot inspection, in reference to simulation time) could easily be calculated by subtracting the PMI complete or delivery date from the reference date (February 1, 2021, for the model). All values are negative for the initialization, indicating that the inspection took place before sim time starts at 0. Then `next` (the date of the next depot inspection, in reference to the simulation time) could be calculated by subtracting the reference date from the FID date. Aircraft starting in depot could have their next calculated by adding 1800 days to the FID date (which is the date they entered their current depot cycle). The `at_depot` (days until the current depot inspection is complete) was calculated for each of these by subtracting the reference date from the scheduled completion date. The `lastDepot` date for the aircraft currently in depot was simply zeroed out because it would be updated upon completion of their current cycle. Future aircraft arrival dates were utilized to populate the `new_arrivals` (days between new aircraft arrivals) by subtracting the reference date from the tentative delivery date then calculating the days between successive values. Excel was used to calculate all values with the `DAYS` function. Finally, once each list was complete, the values were exported from Excel into comma-separated value (CSV) format for easy input into the model.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. RESULTS

This chapter covers all experimental methods used to evaluate the model outputs. It also includes a detailed workflow to cover how the model outputs were consolidated and analyzed.

A. MODEL VERIFICATION AND BASELINE PERFORMANCE

Before the model could be used to evaluate the performance under experimental conditions, its functionality had to be verified and the performance baseline had to be run.

1. Deterministic Runs

Deterministic service times were used for the ISO and depot inspections so that the output statistics and the event scheduling could be verified. The output statistics were verified by running the model with a single aircraft and comparing the output statistics to the easily calculated known values. This is only possible with a very small number of aircraft due to the complexity of the model. The scheduling was verified by printing the model's event list during execution to observe the times of scheduled events. The model was run with the initial configuration in Table 7 to simulate a brand-new aircraft entering the fleet and running for 15 years.

Table 7. Parameters for Deterministic Simulation

Parameter	Value (Days)
420-Day Service Time	7
840-Day Service Time	10
Depot Service Time	150
Transit Time (transit)	2
Time Until Next Depot (next)	2850
Time of Last Depot (lastDepot)	0
Simulation Time (simtime)	5475

A graphical output of the run can be found in Figure 32. The ISO, Depot, and Fleet state variables alternate between 0 and 1.

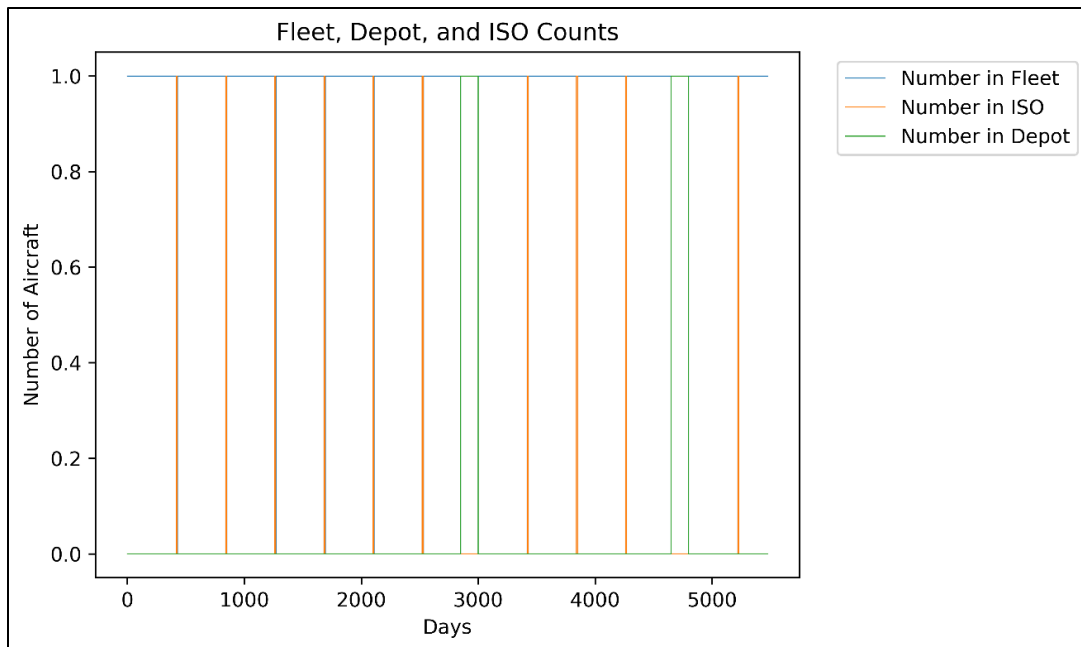


Figure 32. Single-Aircraft Deterministic Run State Variables

Figure 33 contains the calculations for the percent availability. This represents the absolute maximum availability for a new aircraft during the model's 15-year run time. 402 days were occupied by inspections or transit time, which produced an availability of 0.9266.

<u>Event</u>	<u>Value</u>	<u>Occurences</u>
420-Day	7	6
840-Day	10	4
Transit	2	10
Depot	150	2
Time NOT in fleet:		402
Time IN fleet:		5073
Model Run Time:		5475
Percent Avail:		0.9266

Figure 33. Single-Aircraft Percent Availability Calculations

The model's output statistics are saved as a comma-separated value (CSV) file for each individual run. Each of the 6 output statistics reports the number of occurrences, minimum value, maximum value, mean, variance, standard deviation, and the run number in which the observation was made. A separate CSV also outputs the percent availability for each individual aircraft during the run, and 0.9266 was the reported value. Figure 34 contains the output statistics from the run.

stat	count	min	max	mean	var	stdev	run
fleet	26	0	1	0.9266	0.068	0.2608	0
iso	22	0	1	0.015	0.0148	0.1215	0
depot	6	0	1	0.0548	0.0518	0.2276	0
dTIQ	2	0	0	0	0	0	0
dTIS	2	150	150	150	0	0	0
iTIQ	10	0	0	0	0	0	0
iTIS	10	7	10	8.2	2.4	1.5492	0

Figure 34. Single-Aircraft Deterministic Run Summary Statistics

As expected, the minimum and maximum values for depot and ISO correspond to their deterministic values. Since there is only a single aircraft in the model, the mean of the fleet statistic also corresponds to the percent availability. The queue times are also zeroed out due to the single aircraft never having to wait for service. The calculations match the model's output, as expected.

The DESpy package allows the running event list to be printed during model execution (see Figure 35). The event list contains all future events with their scheduled execution times and is updated each time an event is executed. This allows the exact scheduling times to be verified and provides an easy troubleshooting method. Below is a short selection from this run's event list. This covers the events from the fleet creation until the second ISO inspection is scheduled. The aircraft ID is indicated in parenthesis following the scheduled event name. The stopping time is indicated by the stop event that is scheduled at time 5475.

```

60 CurrentEvent: 0.0000 create_fleet <BasicModel.1> [1]
61 *** Event List ***
62 0.0000 enter_fleet (0) <BasicModel.1>
63 5,475.0000 stop <Stopper.0>
64
65 CurrentEvent: 0.0000 enter_fleet (0) <BasicModel.1> [1]
66 *** Event List ***
67 420.0000 enter_iso (0) <BasicModel.1>
68 5,475.0000 stop <Stopper.0>
69
70 CurrentEvent: 420.0000 enter_iso (0) <BasicModel.1> [1]
71 *** Event List ***
72 420.0000 start_iso <BasicModel.1>
73 5,475.0000 stop <Stopper.0>
74
75 CurrentEvent: 420.0000 start_iso <BasicModel.1> [1]
76 *** Event List ***
77 427.0000 end_iso (0) <BasicModel.1>
78 5,475.0000 stop <Stopper.0>
79
80 CurrentEvent: 427.0000 end_iso (0) <BasicModel.1> [1]
81 *** Event List ***
82 429.0000 enter_fleet (0) <BasicModel.1>
83 5,475.0000 stop <Stopper.0>
84
85 CurrentEvent: 429.0000 enter_fleet (0) <BasicModel.1> [2]
86 *** Event List ***
87 840.0000 enter_iso (0) <BasicModel.1>
88 5,475.0000 stop <Stopper.0>
89
90 CurrentEvent: 840.0000 enter_iso (0) <BasicModel.1> [2]
91 *** Event List ***
92 840.0000 start_iso <BasicModel.1>
93 5,475.0000 stop <Stopper.0>

```

Figure 35. Example Event List Printout

As expected, the first ISO inspection is scheduled at time 420, the end of the first ISO is at time 427, and the aircraft returns to the fleet following transit 2 days later at time 429. After arriving back in the fleet, the next ISO is scheduled at time 840, which is exactly what is expected. The remainder of the event list was also verified to be in accordance with the proper maintenance schedule.

After running the model with a single aircraft and verifying the correct maintenance scheduling, the model was run with the same deterministic times using the entire fleet data, as it was described in Chapter III. Depot and ISO servers were set to a high enough value so that queues would never form. Again, this creates an upper bound for the performance of the model in a “perfect” world. The graphical and performance statistics are noted in Figures 36 and 37.

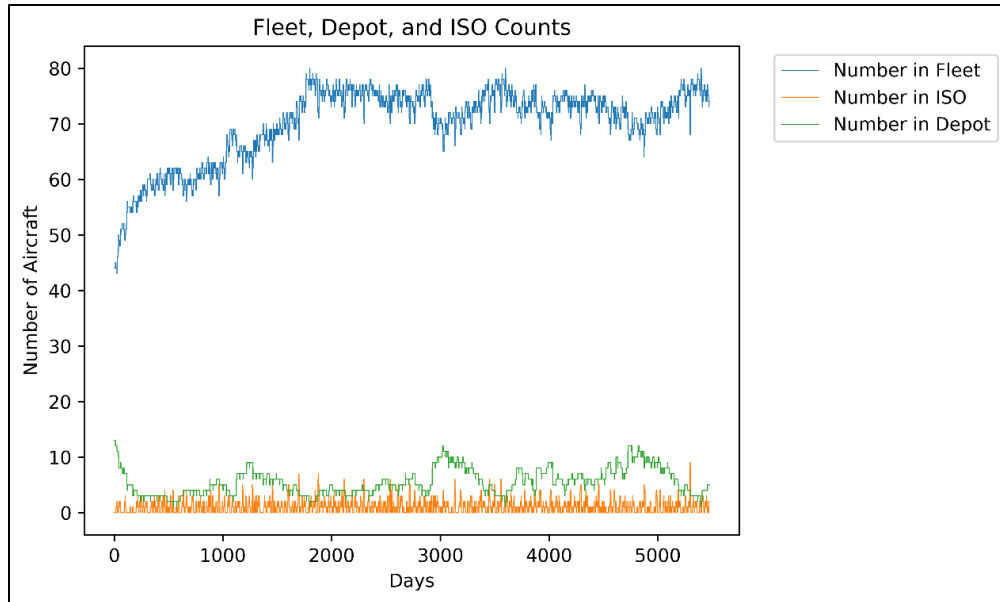


Figure 36. Current Fleet Deterministic Simulation State Variable

stat	count	min	max	mean	var	stdev	run
fleet	1912	43	80	69.9518	44.5792	6.6768	0
iso	1482	0	9	1.1006	1.3486	1.1613	0
depot	408	1	13	5.5717	5.6041	2.3673	0
dTIQ	199	0	0	0	0	0	0
dTIS	194	150	150	150	0	0	0
iTIQ	741	0	0	0	0	0	0
iTIS	739	7	10	8.1407	2.1238	1.4573	0

Figure 37. Current Fleet Deterministic Simulation Summary Statistics

Most importantly, the average percent available of the 81 aircraft was 0.9113. This represents the performance ceiling that will be observed in any of the stochastic models, which will face potential overruns and server limitations.

2. Simulation of Current State

Each squadron currently performs all special inspections at the squadron level. This results in having an “unlimited” number of servers in the model, as each squadron will conduct maintenance when it is due. The transit times will also be zero since each aircraft will go directly from the flight line to the special inspection at the squadron. The 420 and

840-day service times were represented by the fitted Gamma distributions derived from the fleet maintenance data, as detailed in Chapter III. Upper and lower bounds were established as the shortest and longest realistic inspection times and are implemented due to the fact that the Gamma distribution produces times from 0 to infinity. As mentioned in Chapter III, the depot service times are represented by a uniform distribution (between 150 and 170 days) so that some variability in the post-depot inspection process is captured. 100 runs were conducted. A summary of the main input parameters is found in Table 8.

Table 8. Parameters for Current State Simulation

Parameter	Value (Days)
420-Day Service Time	Gamma(shape=2.333, scale=21.924)
420-Day Lower Bound	7
420-Day Upper Bound	160
840-Day Service Time	Gamma(shape=2.281, scale=26.760)
840-Day Lower Bound	10
840-Day Upper Bound	160
Depot Service Time	Uniform(min=150, max=170)
Transit Time (transit)	0
Simulation Time (simtime)	5475

Output of the first simulation can be found in Figure 38. The state variables vary between runs, but the general trends are the same. It is important to note that the number in depot is very similar to the optimal deterministic runs due to the fixed scheduling, but the number in ISO hovers at a much higher level (the mean number in ISO is has a 95% confidence interval of 7.346 to 7.368, but the highest peak observed in any run is 18).

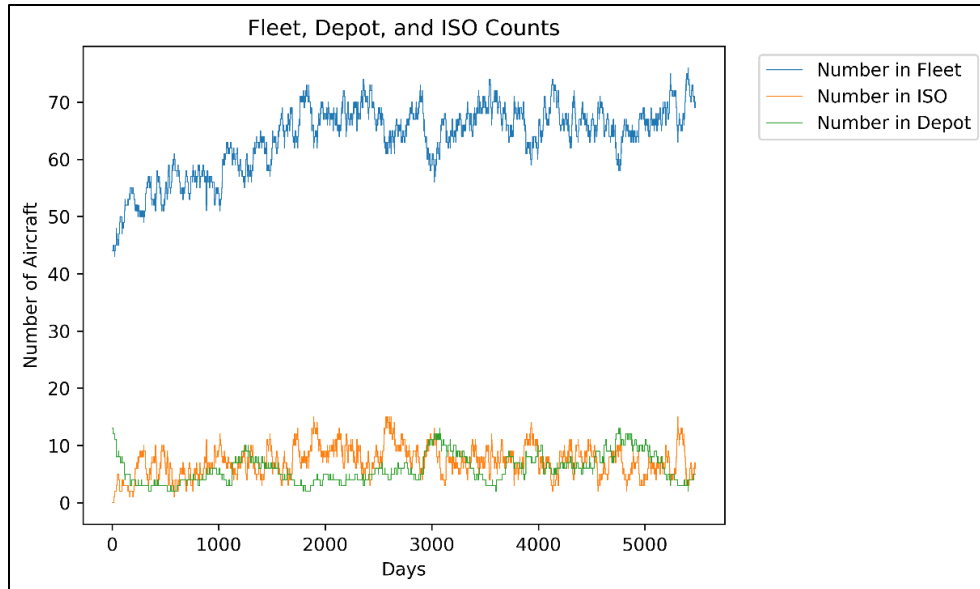


Figure 38. Current State Simulation State Variables

The overall mean percent availability was 0.8283 (95% confidence interval of 0.8282 to 0.8284), and the median was 0.8282. Figure 39 contains a histogram of all 100 overall fleet mean availabilities. The results follow an approximately Normal distribution, as shown by the Q-Q plot in Figure 40, and the 5th and 95th percentiles were 0.825 and 0.832, respectively.

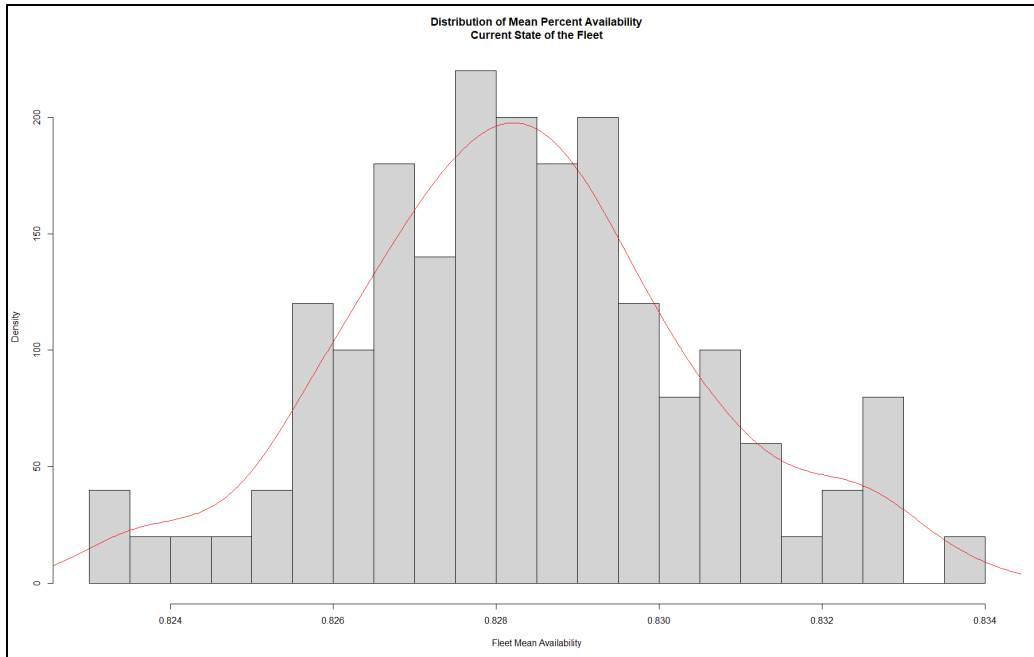


Figure 39. Distribution of Mean Percent Availability for Current State

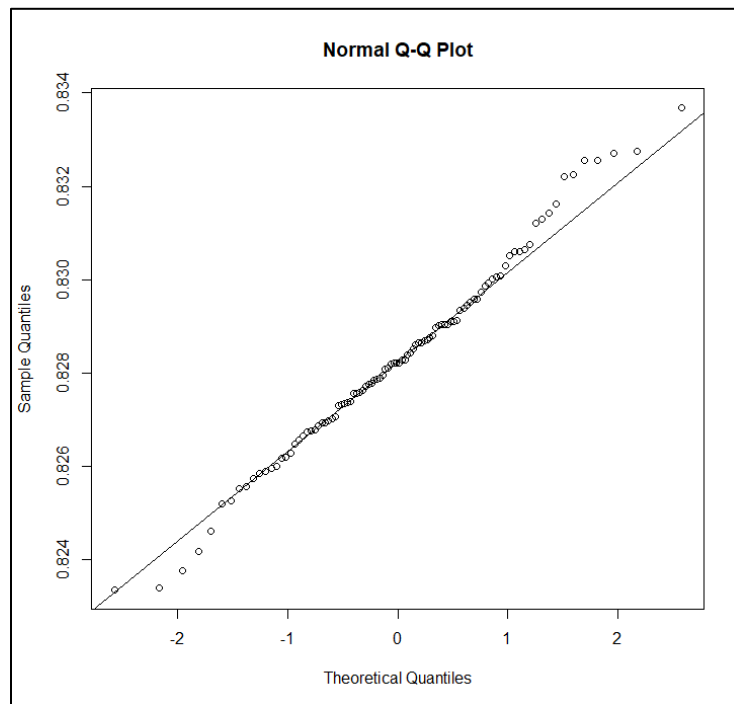


Figure 40. Normal Quantile-Quantile Plot of Mean Percent Availability for Current State

A comparison of the mean to the deterministic runs is found in Table 9. There is an 8.3% difference in current fleet performance compared to the theoretical maximum.

Table 9. Availability Comparison with Baseline and Max Theoretical

Run Type	Mean Percent Availability
Single Aircraft Deterministic (Max Possible)	0.9266
Entire Fleet Deterministic (Max Possible for Current Fleet)	0.9113
Entire Fleet Stochastic with Current Times (Baseline)	0.8283

B. WORKFLOW FOR SIMULATION OF CENTRALIZED MAINTENANCE

Once the initial baseline was established, the model could be run under a variety of circumstances to simulate scenarios with different server counts and turnaround times. Due to the number of experimental scenarios and the volume of output files, a workflow was created to automate the generation of inputs, run the model, consolidate the outputs, and analyze the results. This section will cover the steps in the workflow, and the actual results will be discussed in Section C. The overall workflow from input data to analyzing the outputs is detailed in Figure 41.

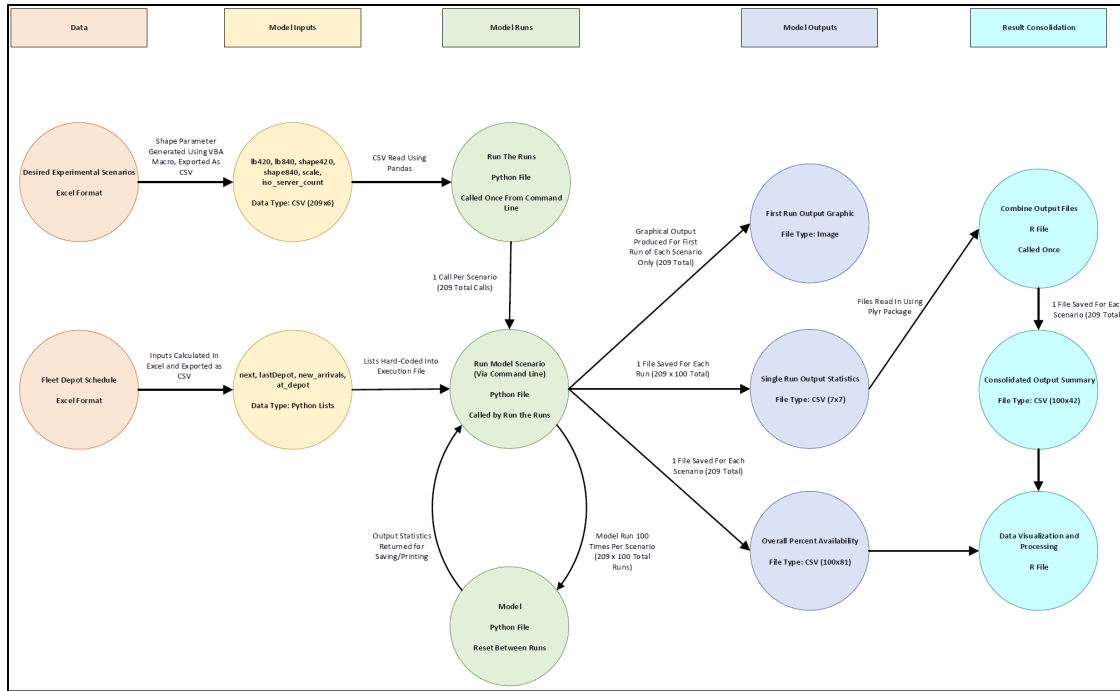


Figure 41. Simulation Workflow

1. Input Data

In order to simulate how a contract maintenance agreement would work, a minimum turnaround time was established for the special inspections with the understanding that a certain portion of the inspections will overrun due to supply or maintenance issues. Then each of these minimum turnaround times was simulated with a different number of ISO servers to represent different facility sizes. The depot time distribution, transit time, new arrival schedule, and starting state were all held constant though all runs. A single CSV file was created to define all experimental scenarios so that they could be passed into Python easily.

The model will be run with a minimum 420-day turnaround of 7 days up to a maximum of 25 days, in one day increments. The 840-day inspection will be expected to take 3 days longer, for a range of 10 to 28 days. Each of these 19 scenarios will be run for server counts of 2 through 12, for a total of 209 scenarios (each run 100 times). The scenarios are summarized in Table 10.

Table 10. Simulation Scenarios

Parameter	Range	Count	Total Scenarios
420-Day Minimum Turnaround Time	7 to 25 Days	19	209
ISO Server Count	2 to 12 Servers	11	

In each scenario, it was assumed that approximately 10% of 420-day and 20% of 840-day inspections will overrun the minimum time, up to a maximum of 90 days. These values were based off approximations from the DECKPLATE data and could be refined at a later date. They represent events such as a wait for supplies or highly intensive corrective actions, such as those for discovered corrosion. The overrun percentage was accomplished by fitting a Gamma distribution (with a fixed scale parameter) for each minimum turnaround time using Excel's Solver to calculate the shape parameter that resulted in 90% and 80% of turnaround times below the thresholds, respectively. This ensured that overruns still had a similarly shaped distribution to the current times. Example cumulative distribution functions for 15/18-day turnarounds can be found in Figure 42. Note that both the minimum and maximum turnaround times are accounted for.

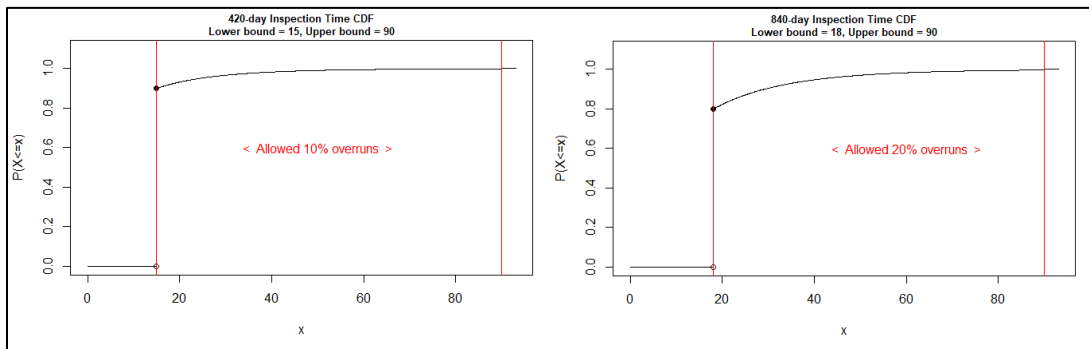


Figure 42. Example ISO Service Time Cumulative Distribution Functions

Figure 43 contains the table in Excel that was used to calculate the shape parameters for the Gamma distributions. The overrun percentages were calculated using the GAMMA.DIST function (which returns the CDF) and set as the objective in the Solver. A VBA macro was written to automate the Solver execution for each scenario.

420 Lower Bound	420 Gamma Shape	420 Gamma Scale	420 Upper Bound	420 Overrun Pct	840 Lower Bound	840 Gamma Shape	840 Gamma Scale	840 Upper Bound	840 Overrun Pct
25	0.4529	20	90	0.9000	28	0.8604	20	90	0.8000
24	0.4306	20	90	0.9000	27	0.8277	20	90	0.8000
23	0.4087	20	90	0.9000	26	0.7952	20	90	0.8000
22	0.3872	20	90	0.9000	25	0.7631	20	90	0.8000
21	0.3662	20	90	0.9000	24	0.7312	20	90	0.8000
20	0.3456	20	90	0.9000	23	0.6996	20	90	0.8000
19	0.3255	20	90	0.9000	22	0.6683	20	90	0.8000
18	0.3059	20	90	0.9000	21	0.6373	20	90	0.8000
17	0.2867	20	90	0.9000	20	0.6067	20	90	0.8000
16	0.2680	20	90	0.9000	19	0.5764	20	90	0.8000
15	0.2499	20	90	0.9000	18	0.5465	20	90	0.8000
14	0.2322	20	90	0.9000	17	0.5169	20	90	0.8000
13	0.2150	20	90	0.9000	16	0.4877	20	90	0.8000
12	0.1984	20	90	0.9000	15	0.4589	20	90	0.8000
11	0.1823	20	90	0.9000	14	0.4305	20	90	0.8000
10	0.1667	20	90	0.9000	13	0.4025	20	90	0.8000
9	0.1515	20	90	0.9000	12	0.3749	20	90	0.8000
8	0.1369	20	90	0.9000	11	0.3477	20	90	0.8000
7	0.1228	20	90	0.9000	10	0.3210	20	90	0.8000

Figure 43. Gamma Distribution Shape Parameter Calculation

Once the service time distribution parameters were known for each scenario, a CSV file was generated to contain all 209 scenarios along with the input parameters that needed to be passed into the model. The columns represented the 420- and 840-day lower bounds, the 420- and 840-day shape parameters, the scale parameter, and the ISO server count. Figure 44 contains an example of the first few lines.

lower420	lower840	shape420	shape840	scale	iso_servers
25	28	0.4529	0.8604	20	12
25	28	0.4529	0.8604	20	11
25	28	0.4529	0.8604	20	10
25	28	0.4529	0.8604	20	9
25	28	0.4529	0.8604	20	8
25	28	0.4529	0.8604	20	7
25	28	0.4529	0.8604	20	6
25	28	0.4529	0.8604	20	5
25	28	0.4529	0.8604	20	4
25	28	0.4529	0.8604	20	3
25	28	0.4529	0.8604	20	2
24	27	0.4306	0.8277	20	12
24	27	0.4306	0.8277	20	11
24	27	0.4306	0.8277	20	10
24	27	0.4306	0.8277	20	9
24	27	0.4306	0.8277	20	8

Figure 44. Model Scenarios CSV Sample

Once the scenarios were defined in a CSV, they could easily be read into Python during the model execution phase.

2. Model Execution

Running a scenario of the model using DESpy requires a separate “execution” file to import the model source file, create a model object, pass the input parameters, run the model, and save any outputs. This execution file was modified to run from the command line and to accept the 6 columns of the input CSV as arguments, and all other model parameters were hard coded into the execution file. By allowing the execution file to accept command line arguments, it could be called from another Python script (referred to as the “Run the Run” script) using the Subprocess Module. The “Run the Run” script uses the Pandas Module to read in the input scenarios CSV and run the model for each scenario. The entirety of the script can be found in Figure 45.

```
1 import subprocess
2
3 import pandas as pd
4
5 df = pd.read_csv('scenarios.csv')
6
7 N = len(df)
8
9 for i in range(N):
10     lower420 = str(df.iloc[i][0])
11     lower840 = str(df.iloc[i][1])
12     shape420 = str(df.iloc[i][2])
13     shape840 = str(df.iloc[i][3])
14     scale = str(df.iloc[i][4])
15     iso_servers = str(df.iloc[i][5])
16
17     args = lower420+' '+lower840+' '+shape420+' '+shape840+' '+scale+' '+iso_servers
18
19     subprocess.Popen('python runBasicModelCMD.py '+args)
```

Figure 45. “Run the Run” Script

By using this script to execute the model, all 209 scenarios could be executed by running a single Python file from the command line.

3. Model Outputs

The model produces 3 primary outputs for each scenario run: 1 graphical output of the fleet, ISO, and depot levels for the first run only; 1 CSV containing the percentage availability of each aircraft (81 columns total) for each run (100 rows total); and 100 CSVs, each of which corresponds to a single run, of the output parameter summary statistics (7 rows and 7 columns). Table 11 summarizes the outputs.

Table 11. Model Outputs

Scenarios	Runs	Outputs Per Scenario
209 Scenarios	100 Runs Per Scenario	1 Graphical Overview (PNG file)
		1 Percentage Availability Summary (100x81 CSV)
		100 Summary Statistic Records (7x7 CSV)

The summary statistic CSVs are saved in a folder with the following naming convention: Runs-X-Y, where X is the minimum 420-day inspection time and Y is the number of ISO servers (seen in Figure 46). Each summary statistics file name contains the number of the run. The percentage availability files and output graphics were also saved using the same “X-Y” file naming convention. As mentioned earlier in this chapter, the output statistics for each run report the number of observations, minimum, maximum, mean, variance, standard deviation, and run number.

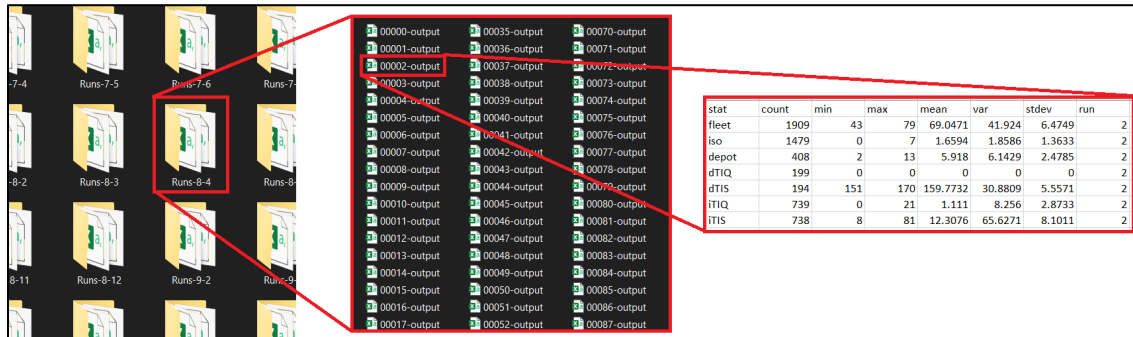


Figure 46. Summary Statistic Output File Hierarchy

Output figures identical to those seen earlier in this chapter were generated for each scenario to give a broad overview of the trends in the data (example in Figure 47). They provide quick visual trend analysis and can highlight queue behavior that may not be fully explained using only summary statistics.



Figure 47. Example Graphical Outputs

The 100-by-81 all runs percentage availability CSV contains columns corresponding to each of the 81 aircraft and the percent availability observed during each run of the model in the rows. The rows can then be averaged to determine the overall mean availability for each run. A sample is found in Figure 48.

0	1	2	3	4	5	6	7	8	9	10
0.8873	0.8873	0.8939	0.8911	0.8955	0.8906	0.8911	0.8895	0.8942	0.8932	0.8935
0.8917	0.8833	0.8933	0.8873	0.895	0.8895	0.8869	0.8908	0.8944	0.8942	0.8941
0.8725	0.8921	0.8928	0.8928	0.895	0.8915	0.8879	0.8919	0.8953	0.8888	0.8952
0.8895	0.8891	0.8922	0.8941	0.8921	0.8933	0.8921	0.8915	0.8911	0.888	0.8933
0.8917	0.8884	0.884	0.8961	0.8827	0.8882	0.8921	0.8842	0.8919	0.8897	0.8862
0.8902	0.8902	0.8933	0.8972	0.8917	0.893	0.8911	0.8917	0.8877	0.8868	0.8957
0.8924	0.8895	0.8961	0.8804	0.8961	0.8868	0.8853	0.8941	0.8928	0.8942	0.8911
0.8926	0.8908	0.8899	0.8968	0.8882	0.8904	0.8895	0.8928	0.882	0.8813	0.8959
0.8807	0.8935	0.8932	0.8942	0.8798	0.89	0.8904	0.8889	0.8911	0.8866	0.8948
0.8889	0.8928	0.8928	0.8908	0.8895	0.886	0.8913	0.8932	0.8884	0.8798	0.8935
0.8897	0.8946	0.8917	0.8864	0.8921	0.8922	0.8855	0.8827	0.8946	0.8853	0.8911
0.8831	0.8889	0.8942	0.891	0.8952	0.8893	0.8862	0.8902	0.8942	0.8833	0.8919
0.8862	0.8933	0.8911	0.884	0.8952	0.8873	0.8961	0.8922	0.8899	0.8948	0.8953
0.8922	0.8882	0.8955	0.8804	0.8897	0.8906	0.893	0.8888	0.8944	0.8941	0.886
0.8893	0.8904	0.8915	0.8844	0.8968	0.8911	0.882	0.8824	0.8959	0.8862	0.8964
0.8787	0.8835	0.897	0.8919	0.8895	0.8882	0.8809	0.8906	0.8888	0.8957	0.8952
0.888	0.886	0.8926	0.8932	0.8959	0.8886	0.8899	0.8919	0.8915	0.8919	0.8935
0.8893	0.8924	0.8935	0.8952	0.8946	0.8822	0.8908	0.8902	0.8932	0.8966	0.8906
0.888	0.8941	0.8963	0.8911	0.8747	0.8935	0.89	0.8897	0.8917	0.8957	0.8953

Figure 48. Example Percentage Availability CSV

In total, the 209 scenarios produced 21,318 output files.

4. Output Consolidation

Before the results could be analyzed, the output files had to be aggregated and formatted in a manner that allowed scenario-based comparisons. Due to its ease of use when dealing with vectorized operations, R was chosen as the tool for the data analysis. The objective was to create a single output file for each scenario consisting of the output statistics for all 100 runs. R's Plyr package allows for all CSV files in a directory to be read in as a single data frame, joined on the columns. For a given scenario, Plyr was used to read all the output statistic CSV files (100 per scenario) into a single data frame consisting of 700 rows and 8 columns, as shown in Figure 49.

	stat	count	min	max	mean	var	stdev	run
1	fleet	1906	43	79	68.6544	42.2568	6.5005	0
2	iso	1477	0	9	2.0294	1.6976	1.3029	0
3	depot	408	2	13	5.9414	6.1308	2.4760	0
4	dTIQ	199	0	0	0.0000	0.0000	0.0000	0
5	dTIS	194	151	170	160.4330	33.5110	5.7889	0
6	iTIQ	738	0	0	0.0000	0.0000	0.0000	0
7	iTIS	737	12	89	15.0638	56.4375	7.5125	0
8	fleet	1907	43	78	68.6597	42.5438	6.5226	1
9	iso	1477	0	7	2.0177	1.9217	1.3863	1
10	depot	408	2	13	5.9481	6.1193	2.4737	1
11	dTIQ	199	0	0	0.0000	0.0000	0.0000	1
12	dTIS	194	151	170	160.6237	30.6711	5.5382	1
13	iTIQ	740	0	0	0.0000	0.0000	0.0000	1
14	iTIS	735	12	64	14.8952	37.5517	6.1279	1
15	fleet	1909	43	79	68.6621	42.7578	6.5389	2
16	iso	1479	0	8	2.0080	2.0091	1.4174	2
17	depot	408	2	14	5.9543	6.2489	2.4998	2
18	dTIQ	199	0	0	0.0000	0.0000	0.0000	2
19	dTIS	194	151	170	160.7990	32.8039	5.7275	2
20	iTIQ	739	0	0	0.0000	0.0000	0.0000	2
21	iTIS	738	12	84	14.8916	49.6572	7.0468	2

Figure 49. Sample Combined Summary Statistic Data Frame

This combined data frame was then broken down by run and reformatted into a data frame with 100 rows, each corresponding to one run, and 42 columns, each corresponding to the count, min, max, mean, variance, and standard deviation of one of the 7 output statistics (using a naming format of “fleet_count,” “iso_min,” etc.). The data frame was

then saved as a CSV file for future reference, as depicted in Figure 50. A loop was used to iterate through all scenario folders and generate all output CSVs.

	fleet_cour	fleet_min	fleet_max	fleet_mea	fleet_var	fleet_stde	iso_count	iso_min	iso_max	iso_mean	iso_var	iso_stdev	depot_coi	depot_mir	depot_made	depot_mad
1	1910	43	79	68.7963	42.7196	6.536	1480	0	7	1.8953	1.6995	1.3037	408	2	13	5.93
2	1908	43	78	68.7837	43.013	6.5584	1478	0	10	1.9072	1.781	1.3345	408	1	14	5.93
3	1907	43	78	68.7189	42.4943	6.5188	1478	0	8	1.9479	1.8923	1.3756	408	2	14	5.93
4	1909	43	79	68.7951	41.9511	6.477	1479	0	8	1.893	1.8665	1.3662	408	2	13	5.93
5	1908	43	79	68.7812	43.5938	6.6026	1478	0	9	1.9425	1.9954	1.4126	408	2	13	5.93
6	1910	43	78	68.7759	42.9257	6.5518	1480	0	9	1.9136	1.9745	1.4052	408	2	13	5.93
7	1908	43	79	68.7711	42.5005	6.5192	1478	0	9	1.9202	1.9401	1.3929	408	1	13	5.93
8	1909	43	79	68.783	43.1646	6.57	1479	0	7	1.91	1.9782	1.4065	408	2	13	5.93
9	1908	43	79	68.7554	43.3654	6.5852	1480	0	9	1.9602	2.0108	1.418	408	2	13	5.93
10	1909	43	79	68.8312	42.8727	6.5477	1480	0	7	1.8694	1.837	1.3554	408	2	14	5.93
11	1906	43	79	68.7532	43.4936	6.595	1476	0	8	1.9204	1.8048	1.3434	408	2	14	5.93
12	1907	43	78	68.7962	43.1292	6.5673	1479	0	8	1.8888	1.8329	1.3539	408	2	14	5.93
13	1907	43	79	68.7324	42.4552	6.5158	1477	0	7	1.9422	1.9996	1.3786	408	2	14	5.93

Figure 50. Sample Tidy Output Statistic CSV

The final step in the data consolidation process was to combine the meaningful output statistics with the percentage availability data in a single file. The statistics listed in Table 12 were selected to compare the individual scenarios.

Table 12. Output Statistics Selected for Scenario Comparison

Statistics Selected for Comparison
Mean Overall Fleet Percent Availability
Percent of Runs with Mean Availability Greater than Baseline
Fleet Percent Availability 5 th Percentile
Fleet Percent Availability 95 th Percentile
Mean Number in Fleet
Mean Number in ISO
Mean Standard Deviation of Number in ISO
Mean Time in ISO Queue
Mean Standard Deviation of Time in ISO Queue
Mean Maximum Time in ISO Queue
Mean Total Time in ISO

An R script was utilized to extract the relevant statistics from each run, calculate the means, and save the 11 output statistics into a data frame with each row corresponding to a scenario (209 rows total). The final 420-day lower bound and the ISO server count for the scenario were also embedded into the data frame so that they could be utilized for visualization purposes. A portion of the final output file is found in Figure 51.

iso420	iso_server_count	fleet_mean	iso_mean	iso_sd	ITIQ_mean	ITIQ_sd	ITIQ_max	ITIS_mean	mean_pct_avail	pct_avail_geq_baseline	fifth_quant	ninetyfifth_quant
7	2	68.67075	2.012110	1.782995	4.961568	8.308085	49.47	14.915760	0.8946156	1.00	0.8903559	0.8978580
7	3	69.23413	1.447751	1.271931	0.767226	2.354801	19.66	10.728583	0.9020147	1.00	0.9008057	0.9031617
7	4	69.31263	1.370980	1.207423	0.177113	0.964672	10.29	10.158944	0.9030705	1.00	0.9021097	0.9039172
7	5	69.33079	1.350674	1.189238	0.036780	0.380752	6.19	10.010048	0.9033191	1.00	0.9024089	0.9041092
7	6	69.33398	1.346840	1.189804	0.007573	0.134080	2.98	9.981167	0.9033649	1.00	0.9025176	0.9041580
7	7	69.33819	1.340899	1.188103	0.000613	0.014292	0.37	9.936696	0.9034245	1.00	0.9026394	0.9043241
7	8	69.33961	1.340227	1.189749	0.000028	0.000736	0.02	9.931762	0.9034363	1.00	0.9026570	0.9042541
7	9	69.33961	1.340223	1.189747	0.000000	0.000000	0.00	9.931736	0.9034363	1.00	0.9026570	0.9042541
7	10	69.33961	1.340223	1.189747	0.000000	0.000000	0.00	9.931736	0.9034363	1.00	0.9026570	0.9042541
7	11	69.33961	1.340223	1.189747	0.000000	0.000000	0.00	9.931736	0.9034363	1.00	0.9026570	0.9042541
7	12	69.33961	1.340223	1.189747	0.000000	0.000000	0.00	9.931736	0.9034363	1.00	0.9026570	0.9042541
8	2	68.13924	2.542709	2.146335	7.821898	11.246723	58.14	18.849352	0.8876288	1.00	0.8782891	0.8925940
8	3	69.04630	1.633303	1.356384	1.085599	2.898077	21.66	12.105521	0.8995552	1.00	0.8985072	0.9006323
8	4	69.16542	1.519298	1.263647	0.235712	1.144191	11.02	11.262003	0.9011329	1.00	0.9003374	0.9019207
8	5	69.18452	1.495831	1.246977	0.056473	0.507888	7.65	11.085708	0.9013918	1.00	0.9003694	0.9021831
8	6	69.19630	1.487423	1.241694	0.010285	0.163605	3.46	11.021339	0.9015620	1.00	0.9007493	0.9022986
8	7	69.19385	1.487123	1.239961	0.001290	0.028809	0.70	11.022423	0.9015305	1.00	0.9007273	0.9023014

Figure 51. Sample Combined Final Output Statistic Data Frame

With all relevant output data in a single file, the results could be visualized and analyzed.

C. SUMMARY OF RESULTS

Figure 52 contains the overall percentage availability results for all 209 scenarios. Individual lines represent a constant minimum turnaround time and are labeled in blue. The x-axis corresponds to the number of available servers so that a comparison can be made between both turnaround time and server count. The most immediately noticeable fact is that having greater than 6 ISO servers creates almost no improvement in the mean availability. The infeasible service time and server count combinations also are readily apparent as the availability harshly drops off when the inspection system becomes overwhelmed.

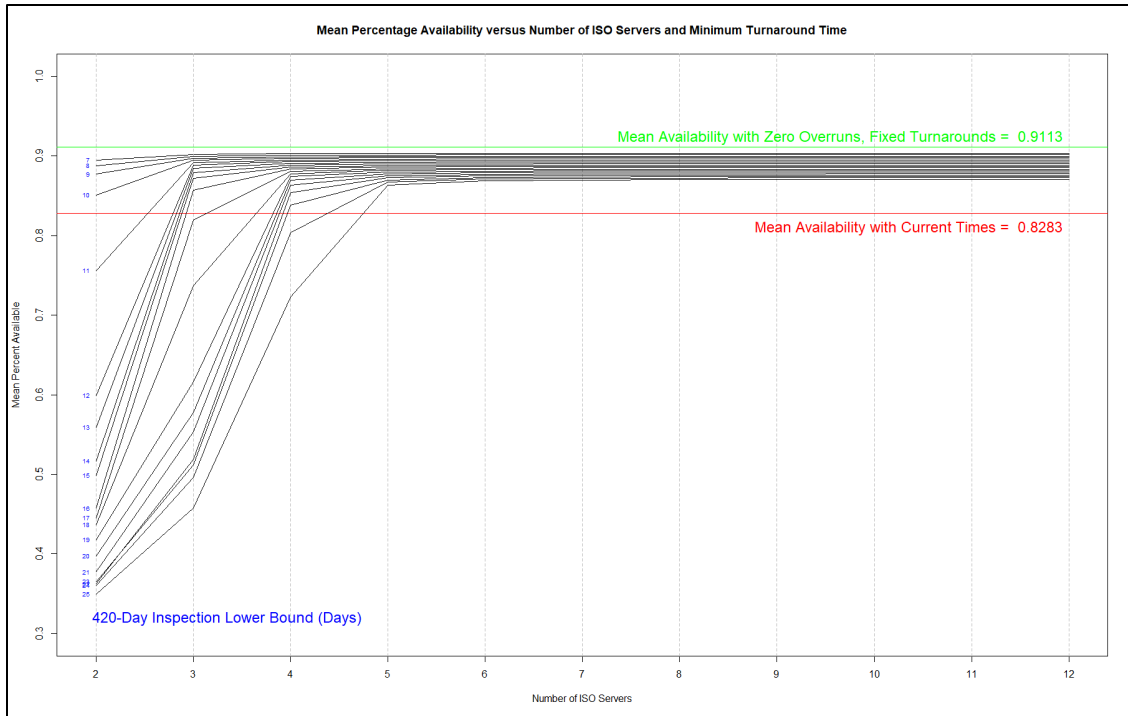


Figure 52. Mean Availability as a Function of Turnaround Time and Server Count

Figure 53 is a more detailed section of the feasible area consisting of 2 to 6 servers and above the baseline turnaround time. This results in 69 potential time/server combinations that would potentially present an improvement over the baseline.

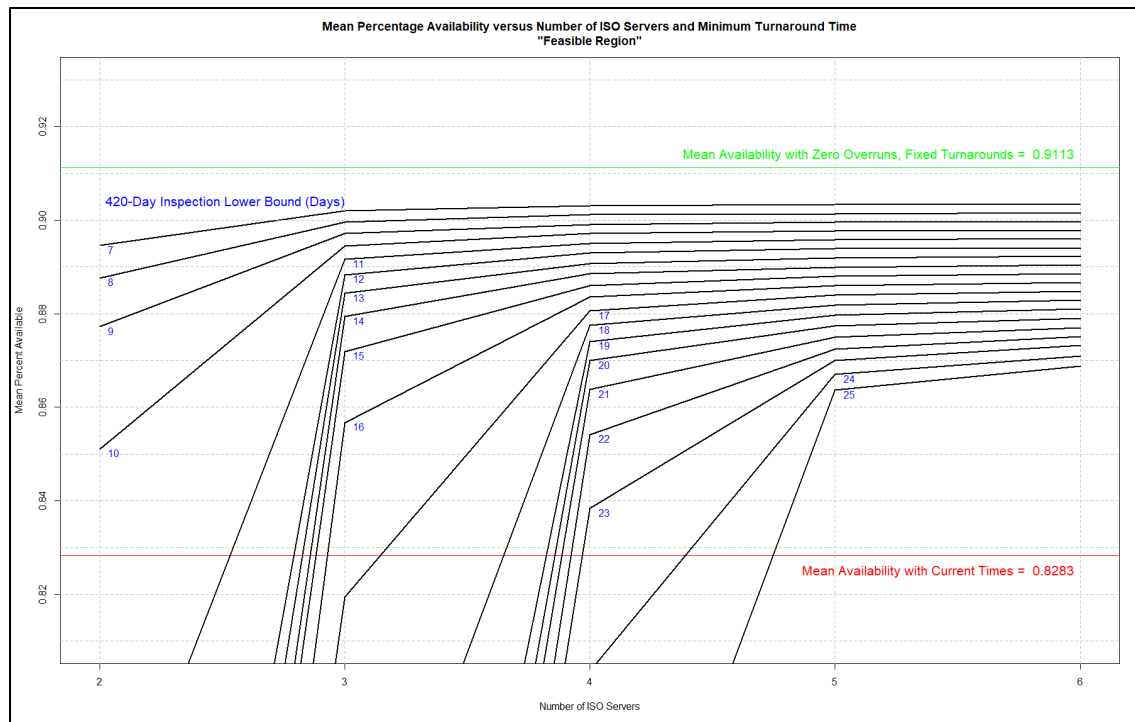


Figure 53. Mean Availability Feasible Region (Detailed View)

A summary of the key performance parameters for the scenarios in the feasible region is in Figure 54. The scenarios with 6 ISO servers were omitted because they presented only marginal improvements over the 5 server scenarios. The highlighted statistics indicate a potential issue and are covered in Table 13. Further analysis would be required before proceeding with any scenario with a potentially problematic performance statistic. The best-case scenarios provide an improvement of 7.5% over the baseline. The distribution of overall availabilities is fairly narrow for each scenario, and the difference between the 5th and 95th percentiles is typically in the range of less than 0.5%.

ISO Server Count	420-Day Lower Bound	Mean Number in ISO	Mean SD of Number in ISO	Mean Time in ISO Queue	Mean SD of Time in ISO Queue	Mean Maximum Time in ISO Queue	Mean Time in ISO	Mean Overall Availability	Improvement over Baseline	Percent of Runs with Availability Greater than Baseline	Overall Availability 5th Percentile	Overall Availability 95th Percentile	5th-95th Percentile Range
2	7	2.012	1.783	4.962	8.308	49.470	14.916	0.895	0.066	1.00	0.890	0.898	0.008
	8	2.543	2.146	7.822	11.247	58.140	18.849	0.888	0.059	1.00	0.878	0.893	0.014
	9	3.333	2.706	12.663	15.792	74.050	24.729	0.877	0.049	0.99	0.862	0.886	0.024
	10	5.302	4.326	26.259	27.801	109.230	39.376	0.851	0.023	0.89	0.815	0.876	0.061
3	7	1.448	1.272	0.767	2.355	19.660	10.729	0.902	0.074	1.00	0.901	0.903	0.002
	8	1.633	1.356	1.086	2.898	21.660	12.106	0.900	0.071	1.00	0.899	0.901	0.002
	9	1.816	1.425	1.419	3.516	25.450	13.461	0.897	0.069	1.00	0.896	0.898	0.002
	10	2.019	1.499	1.887	4.167	28.650	14.972	0.895	0.066	1.00	0.893	0.896	0.003
	11	2.240	1.573	2.471	4.938	31.680	16.611	0.892	0.063	1.00	0.889	0.893	0.004
	12	2.493	1.677	3.311	5.960	36.410	18.489	0.888	0.060	1.00	0.886	0.890	0.004
	13	2.796	1.822	4.535	7.363	41.540	20.739	0.884	0.056	1.00	0.881	0.887	0.006
	14	3.176	2.010	6.305	9.181	48.210	23.566	0.879	0.051	1.00	0.875	0.883	0.008
	15	3.747	2.361	9.508	12.129	57.680	27.803	0.872	0.044	1.00	0.863	0.878	0.015
	16	4.890	3.292	16.959	19.487	81.780	36.331	0.857	0.028	0.98	0.834	0.871	0.037
4	7	1.371	1.207	0.177	0.965	10.290	10.159	0.903	0.075	1.00	0.902	0.904	0.002
	8	1.519	1.264	0.236	1.144	11.020	11.262	0.901	0.073	1.00	0.900	0.902	0.002
	9	1.674	1.323	0.333	1.457	13.620	12.408	0.899	0.071	1.00	0.898	0.900	0.002
	10	1.822	1.367	0.425	1.679	14.570	13.504	0.897	0.069	1.00	0.896	0.898	0.002
	11	1.985	1.416	0.558	1.992	16.250	14.714	0.895	0.067	1.00	0.894	0.896	0.002
	12	2.141	1.458	0.714	2.329	17.950	15.875	0.893	0.065	1.00	0.892	0.894	0.002
	13	2.313	1.514	0.933	2.810	21.190	17.154	0.891	0.062	1.00	0.890	0.892	0.002
	14	2.477	1.556	1.173	3.216	22.670	18.374	0.889	0.060	1.00	0.887	0.890	0.002
	15	2.676	1.616	1.539	3.794	25.310	19.848	0.886	0.058	1.00	0.884	0.887	0.003
	16	2.861	1.666	1.928	4.314	27.660	21.230	0.884	0.055	1.00	0.882	0.885	0.003
	17	3.086	1.745	2.540	5.126	30.440	22.896	0.881	0.052	1.00	0.879	0.882	0.003
	18	3.322	1.827	3.265	6.049	34.990	24.656	0.877	0.049	1.00	0.875	0.879	0.004
	19	3.584	1.921	4.213	7.080	39.470	26.601	0.874	0.046	1.00	0.870	0.876	0.006
	20	3.897	2.031	5.496	8.233	42.210	28.934	0.870	0.042	1.00	0.866	0.873	0.008
5	7	1.351	1.189	0.037	0.381	6.190	10.010	0.903	0.075	1.00	0.902	0.904	0.002
	8	1.496	1.247	0.056	0.508	7.650	11.086	0.901	0.073	1.00	0.900	0.902	0.002
	9	1.640	1.302	0.078	0.620	8.470	12.153	0.900	0.071	1.00	0.899	0.900	0.002
	10	1.784	1.345	0.110	0.776	10.010	13.224	0.898	0.069	1.00	0.897	0.899	0.002
	11	1.926	1.383	0.142	0.890	10.330	14.276	0.896	0.067	1.00	0.895	0.897	0.002
	12	2.073	1.423	0.180	1.044	11.490	15.371	0.894	0.066	1.00	0.893	0.895	0.002
	13	2.226	1.470	0.255	1.316	13.340	16.510	0.892	0.064	1.00	0.891	0.893	0.002
	14	2.375	1.499	0.316	1.492	14.480	17.617	0.890	0.062	1.00	0.889	0.891	0.002
	15	2.524	1.543	0.403	1.734	15.500	18.721	0.888	0.060	1.00	0.887	0.889	0.002
	16	2.678	1.577	0.516	2.022	17.220	19.867	0.886	0.058	1.00	0.885	0.887	0.002
	17	2.835	1.612	0.646	2.312	19.000	21.032	0.884	0.056	1.00	0.883	0.885	0.002
	18	2.995	1.653	0.810	2.614	20.100	22.222	0.882	0.054	1.00	0.881	0.883	0.002
	19	3.160	1.698	1.044	3.059	22.230	23.448	0.880	0.051	1.00	0.878	0.881	0.002
	20	3.335	1.740	1.299	3.489	24.310	24.748	0.877	0.049	1.00	0.876	0.879	0.003
	21	3.519	1.789	1.637	4.037	26.800	26.114	0.875	0.047	1.00	0.873	0.877	0.003
	22	3.706	1.832	2.022	4.575	28.650	27.512	0.872	0.044	1.00	0.871	0.874	0.003
	23	3.897	1.868	2.416	5.069	30.200	28.941	0.870	0.042	1.00	0.869	0.872	0.003
	24	4.128	1.938	3.084	5.867	33.200	30.654	0.867	0.039	1.00	0.865	0.869	0.004
	25	4.379	2.028	3.951	6.868	36.710	32.525	0.864	0.035	1.00	0.860	0.866	0.006

Figure 54. Feasible Region Model Outputs

Table 13. Feasible Region Highlighting Justifications

Statistic	Highlighting
Mean Number in ISO	Greater than number of available servers, indicating potentially unstable queue.
Mean Time in ISO Queue	Greater than one week, indicating that aircraft are spending a potentially unsupportable amount of time waiting for maintenance.
Percent of Runs with Availability Greater than Baseline	Less than 100% greater than the baseline, indicating potential for regression.
Overall Availability 5 th Percentile	Less than baseline, indicating fewer than 95% of runs outperform the mean baseline.

Once the mean aircraft availability is known, it is possible to translate the performance increase in terms of additional days of availability. Multiplying the

improvement over baseline by 365 days gives an average number of additional days in a year that a single aircraft would be available compared to the current state. The same procedure can be repeated for the 5th and 95th percentile improvements to provide the range in which 90% of model outputs fall. It is important to note that these availabilities are based on an overall average over the model's 15-year run time and would not necessarily correspond to year-by-year results. Figure 55 contains the additional availabilities for a single aircraft over the course of one year in terms of the mean, 5th percentile, and 95th percentile of scenario outputs (potentially problematic scenarios from Figure 54 are removed).

ISO Server Count	420-Day Lower Bound	Mean Overall Availability	Mean Additional Days of Availability Per Year Per Aircraft	5th Percentile Additional Days of Availability	95th Percentile Additional Days of Availability
3	7	0.902	26.9	26.5	27.3
	8	0.900	26.0	25.6	26.4
	9	0.897	25.1	24.6	25.5
	10	0.895	24.2	23.7	24.8
	11	0.892	23.1	22.3	23.8
	12	0.888	21.9	21.1	22.7
	13	0.884	20.5	19.1	21.5
4	7	0.903	27.3	26.9	27.6
	8	0.901	26.6	26.3	26.9
	9	0.899	25.8	25.5	26.1
	10	0.897	25.1	24.9	25.5
	11	0.895	24.4	24.0	24.7
	12	0.893	23.6	23.3	24.0
	13	0.891	22.8	22.4	23.1
	14	0.889	22.0	21.6	22.5
	15	0.886	21.1	20.5	21.5
	16	0.884	20.2	19.7	20.6
	17	0.881	19.1	18.5	19.6
	18	0.877	18.0	17.2	18.6
5	19	0.874	16.7	15.4	17.5
	20	0.870	15.2	13.6	16.4
	7	0.903	27.4	27.0	27.7
	8	0.901	26.7	26.3	27.0
	9	0.900	26.0	25.7	26.3
	10	0.898	25.3	25.1	25.6
	11	0.896	24.6	24.3	24.9
	12	0.894	23.9	23.6	24.2
	13	0.892	23.2	22.8	23.5
	14	0.890	22.5	22.1	22.8
	15	0.888	21.8	21.4	22.1
	16	0.886	21.0	20.6	21.4
	17	0.884	20.3	19.9	20.7
	18	0.882	19.5	19.1	19.9
	19	0.880	18.8	18.3	19.2
	20	0.877	17.9	17.4	18.3
	21	0.875	17.0	16.4	17.6
	22	0.872	16.1	15.6	16.7
	23	0.870	15.2	14.7	15.9
	24	0.867	14.1	13.3	15.0
	25	0.864	12.9	11.7	13.9

Figure 55. Feasible Region Mean Availability and Additional Days

These numbers can be scaled accordingly to produce expected additional days of availability for an 18-aircraft squadron and the entire 81-aircraft KC-130J fleet. These results are located in Figure 56.

ISO Server Count	420-Day Lower Bound	Mean Overall Availability	Mean Additional Days of Availability Per Year		
			Per Aircraft	Per Squadron (18 Aircraft)	Per Fleet (81 Aircraft)
3	7	0.902	26.9	484.3	2179.4
	8	0.900	26.0	468.1	2106.7
	9	0.897	25.1	452.6	2036.9
	10	0.895	24.2	435.1	1957.9
	11	0.892	23.1	416.1	1872.3
	12	0.888	21.9	394.4	1774.7
	13	0.884	20.5	368.5	1658.1
4	7	0.903	27.3	491.2	2210.6
	8	0.901	26.6	478.5	2153.3
	9	0.899	25.8	464.9	2092.2
	10	0.897	25.1	452.5	2036.2
	11	0.895	24.4	438.6	1973.5
	12	0.893	23.6	424.9	1912.1
	13	0.891	22.8	409.9	1844.6
	14	0.889	22.0	396.0	1782.0
	15	0.886	21.1	379.1	1705.9
	16	0.884	20.2	363.0	1633.5
	17	0.881	19.1	343.6	1546.3
	18	0.877	18.0	323.2	1454.2
5	19	0.874	16.7	300.7	1353.1
	20	0.870	15.2	273.7	1231.7
	7	0.903	27.4	492.9	2217.9
	8	0.901	26.7	480.2	2161.0
	9	0.900	26.0	467.9	2105.7
	10	0.898	25.3	455.7	2050.8
	11	0.896	24.6	443.4	1995.1
	12	0.894	23.9	430.8	1938.4
	13	0.892	23.2	417.7	1879.6
	14	0.890	22.5	405.0	1822.3
	15	0.888	21.8	392.0	1764.1
	16	0.886	21.0	378.8	1704.6
	17	0.884	20.3	365.5	1644.6
	18	0.882	19.5	351.8	1582.9
	19	0.880	18.8	337.7	1519.8
	20	0.877	17.9	322.5	1451.4
	21	0.875	17.0	306.5	1379.2
	22	0.872	16.1	290.2	1306.0
	23	0.870	15.2	274.2	1233.9
	24	0.867	14.1	254.3	1144.2
	25	0.864	12.9	232.4	1046.0

Figure 56. Feasible Region Additional Yearly Availability

At the upper end, squadrons are seeing an expected increase of over 450 days per year. This would be roughly equivalent to having an additional aircraft that is never in a depot or special inspection. These results are depicted graphically against the baseline at the squadron level in Figures 57 and 58.

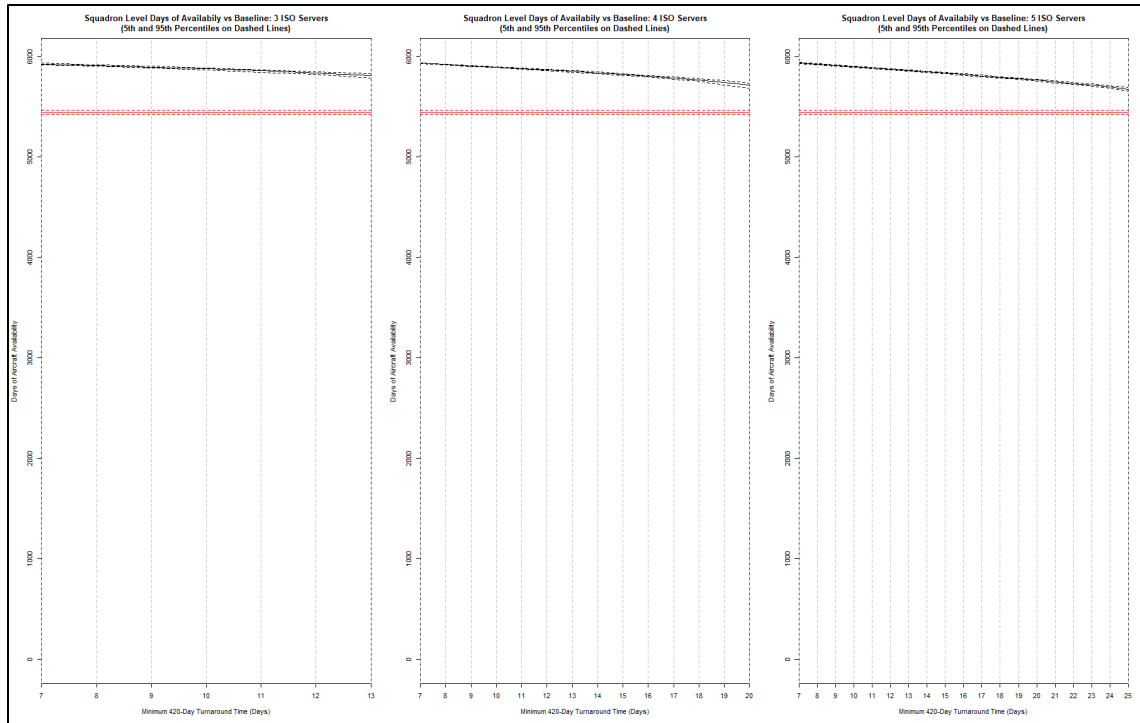


Figure 57. Squadron Level Baseline Availability Versus Centralized Models

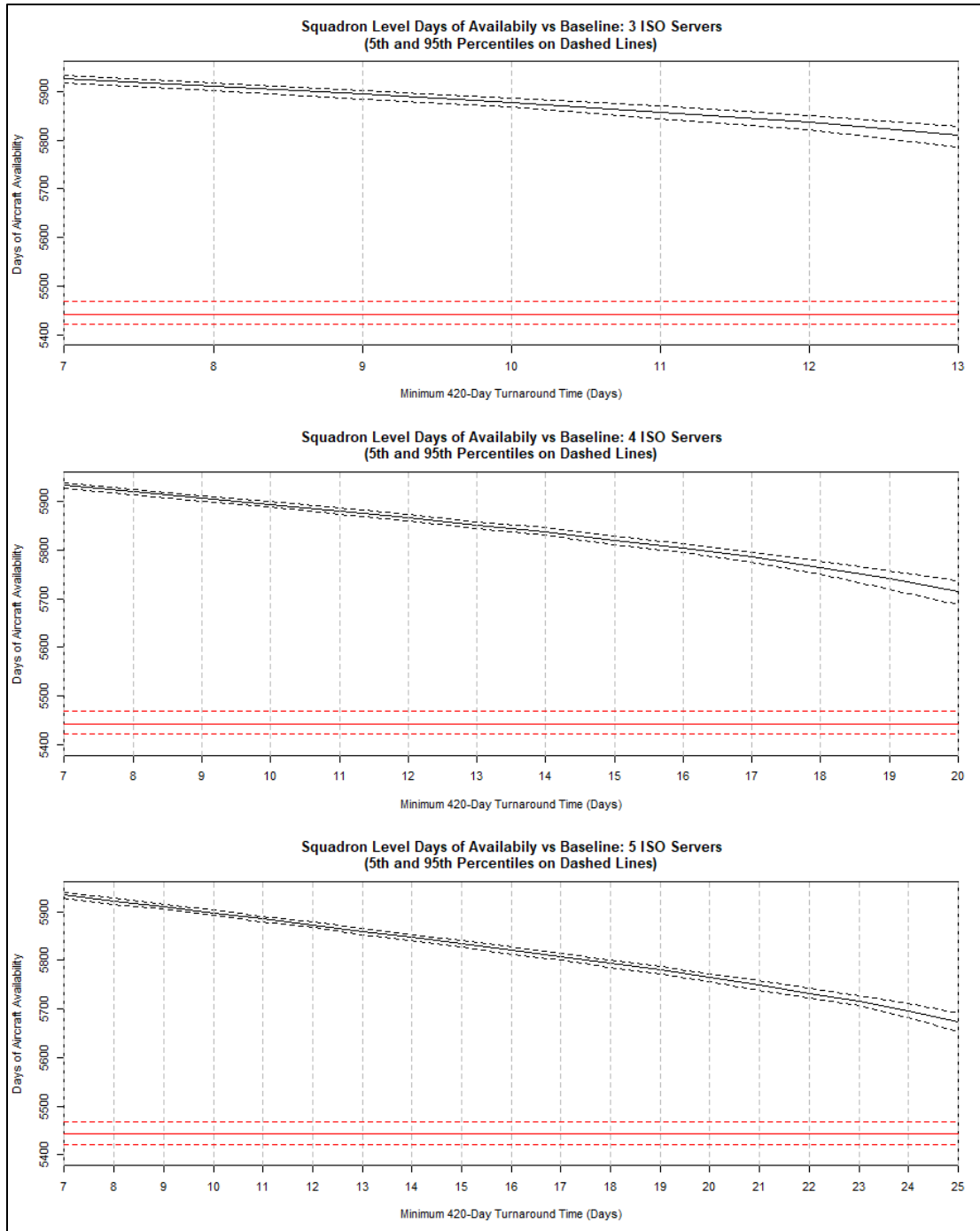


Figure 58. Squadron Level Availability Versus Baseline (Detail)

Overall, having between 3 and 5 ISO servers presents the best potential centralized facility size. Even in the worst-case scenario (25-day turnaround at the 5-server facility), the mean availability was over 230 days higher per year compared to the baseline. The

scenarios with only 2 servers produced potentially unstable queues with the current fleet state, and further analysis would need to be conducted to assess their suitability. Figure 59 below contains the prior results on a single graph for easy comparison of their means. The baseline plus an additional year of availability was highlighted for reference purposes.

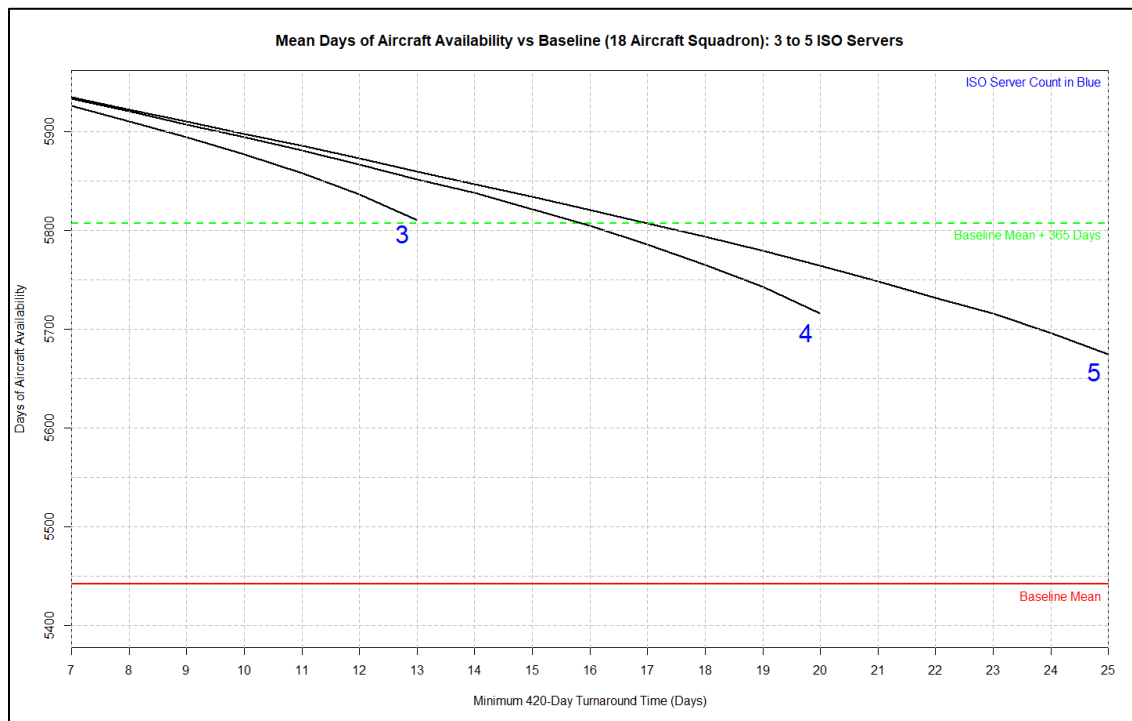


Figure 59. Squadron Level Days of Availability Versus Baseline

Smaller facilities did not handle the larger turnaround times as well as the larger facilities, but there was not a large performance difference for the shorter turnaround times. Below the 15-day turnaround time, there is only negligible difference between the 4- and 5-server facilities. This indicates that the maximum feasible turnaround time should be determined before facility size is chosen. Further considerations and assumptions will be covered in Chapter V.

V. CONCLUSION

This chapter presents recommendations based off the simulation results and provides guidance for potential future work.

A. RECOMMENDATIONS

The consolidated ISO inspection concept has been demonstrated to produce notable increases in readiness levels as long as the minimum turnaround time is appropriate for the size of the consolidated facility. Facilities with a capacity of greater than 6 simultaneous aircraft do not produce a noticeable increase in readiness levels for turnaround times of 25 days or less and should be avoided due to the excessive associated cost. Simulation scenarios that produced a mean number of aircraft in ISO that was greater than the number of servers were not considered due to being potentially unstable. This resulted in a “feasible region” of scenarios for facility sizes of 3 to 5 servers.

Days of availability was used as the primary measure of performance, and Figure 60 shows the expected increases in availability across the feasible region for different unit sizes. The baseline model run with current inspection times provided an expected readiness level of 82.83%. This is equivalent to 5,442 days of aircraft availability per year for a squadron of 18 aircraft. Worst-case scenarios in the feasible region produced an increase of 232 additional days of availability per squadron per year, and the best-case resulted in 492 additional days.

ISO Server Count	420-Day Lower Bound	Mean Overall Availability	Mean Improvement over Baseline	Mean Additional Days of Availability Per Year		
				Per Aircraft	Per Squadron (18 Aircraft)	Per Fleet (81 Aircraft)
3	7	0.902	0.074	26.9	484.3	2179.4
	8	0.900	0.071	26.0	468.1	2106.7
	9	0.897	0.069	25.1	452.6	2036.9
	10	0.895	0.066	24.2	435.1	1957.9
	11	0.892	0.063	23.1	416.1	1872.3
	12	0.888	0.060	21.9	394.4	1774.7
	13	0.884	0.056	20.5	368.5	1658.1
4	7	0.903	0.075	27.3	491.2	2210.6
	8	0.901	0.073	26.6	478.5	2153.3
	9	0.899	0.071	25.8	464.9	2092.2
	10	0.897	0.069	25.1	452.5	2036.2
	11	0.895	0.067	24.4	438.6	1973.5
	12	0.893	0.065	23.6	424.9	1912.1
	13	0.891	0.062	22.8	409.9	1844.6
	14	0.889	0.060	22.0	396.0	1782.0
	15	0.886	0.058	21.1	379.1	1705.9
	16	0.884	0.055	20.2	363.0	1633.5
	17	0.881	0.052	19.1	343.6	1546.3
	18	0.877	0.049	18.0	323.2	1454.2
5	19	0.874	0.046	16.7	300.7	1353.1
	20	0.870	0.042	15.2	273.7	1231.7
	7	0.903	0.075	27.4	492.9	2217.9
	8	0.901	0.073	26.7	480.2	2161.0
	9	0.900	0.071	26.0	467.9	2105.7
	10	0.898	0.069	25.3	455.7	2050.8
	11	0.896	0.067	24.6	443.4	1995.1
	12	0.894	0.066	23.9	430.8	1938.4
	13	0.892	0.064	23.2	417.7	1879.6
	14	0.890	0.062	22.5	405.0	1822.3
	15	0.888	0.060	21.8	392.0	1764.1
	16	0.886	0.058	21.0	378.8	1704.6
	17	0.884	0.056	20.3	365.5	1644.6
	18	0.882	0.054	19.5	351.8	1582.9
	19	0.880	0.051	18.8	337.7	1519.8
	20	0.877	0.049	17.9	322.5	1451.4
	21	0.875	0.047	17.0	306.5	1379.2
	22	0.872	0.044	16.1	290.2	1306.0
	23	0.870	0.042	15.2	274.2	1233.9
	24	0.867	0.039	14.1	254.3	1144.2
	25	0.864	0.035	12.9	232.4	1046.0

Figure 60. Feasible Improvements Over Baseline

Figure 61 contains a graphical depiction of the expected gain in availability at the squadron level as a function of turnaround time and facility size. Availability increases as facility size increases or turnaround time decreases.

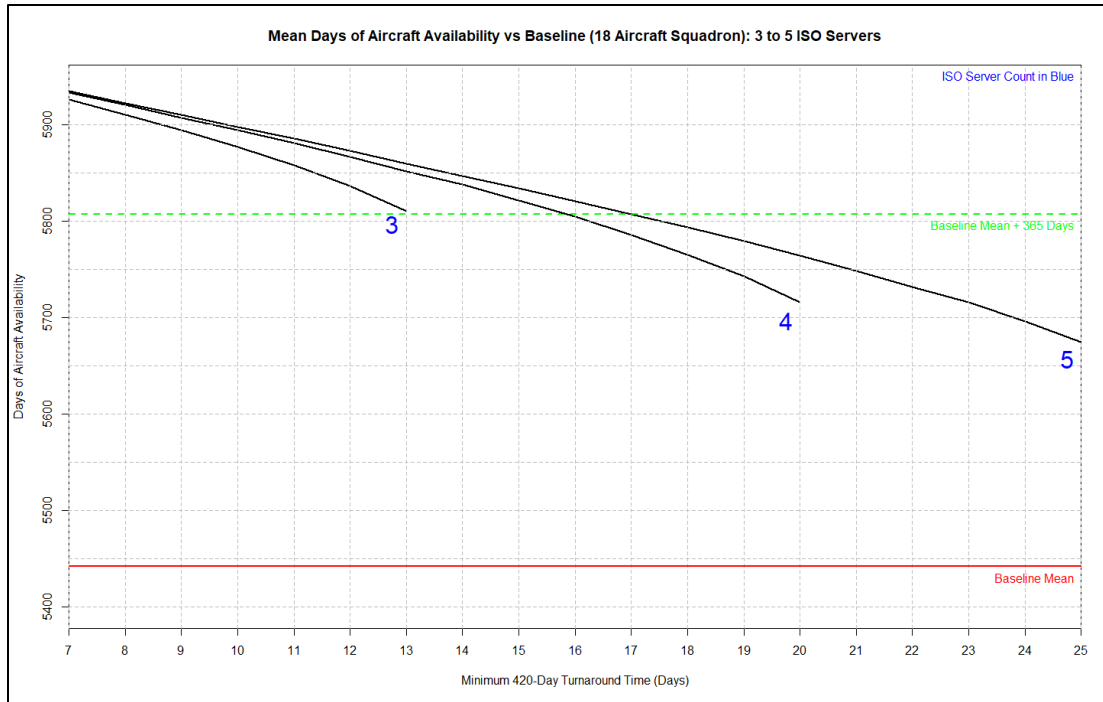


Figure 61. Days of Availability as a Function of Facility Size and Turnaround Time

Since the cost of the consolidated facility is directly proportional to the size, the smallest facility that can handle the required turnaround time should be chosen. The potential turnaround time will be a matter of optimizing the inspection and supply processes to reduce downtime, which would need to be a subject of future work. Overall, the benefits of a consolidated maintenance facility are considerable when compared to the current status quo.

B. FUTURE WORK

Conducting this research exposed a number of potential issues in the KC-130J maintenance process. In addition to scenarios that the model could be adapted to examine, there are certain aspects of the maintenance process that can be optimized to improve the special inspection process as it exists today.

1. Optimization of Maintenance Procedures

Perhaps one of the largest takeaways from this research is that the KC-130J maintenance publications are considerably lacking in terms of direction for the larger special inspections. A cumulative 840-day inspection requires over 140 individual inspections to be conducted across 10 defined aircraft zones by 9 different maintenance divisions. However, the only guidance is to perform inspections sequentially, as they are listed in the publication. There is no recommended method to conduct inspections concurrently or even a framework for how the work should be broken down over a defined timeline. Each squadron must plan and coordinate these inspections through the use of internally designed “sequence control cards” that are unofficial schedules based on experience. There is no guarantee that these squadron-level practices are optimal, and there is not a specific division to oversee their creation or update. At the very least, a framework for accomplishing the special inspections across a defined timeline should be published to assist in the planning of maintenance efforts. In addition to the scheduling issues, the manpower requirements for individual inspections are poorly defined. The planning factors provided for each inspection are only the total time to accomplish the task and the specific military occupational specialties (MOSs) required. There is not a recommended number of workers or an exact manhour breakdown. Updating the publications to provide an efficient sequence for the inspections, along with a detailed required manpower breakdown, would greatly reduce the time spent conducting these inspections. It would also assist in defining a baseline turnaround time required for the centralized maintenance facility.

2. Depot-Level Impacts

One of the major assumptions of this study was that the depot-level inspections would return to their advertised turnaround time. It was discussed in Chapter II that depot inspections are currently taking in excess of 2 to 3 times the published 150-day turnaround to complete. This is an unsustainable practice, and the backlog of aircraft currently in depot (currently at 13 aircraft) combined with the influx of new aircraft from the factory creates potential issues in the next 10 years. The aircraft currently in depot will be scheduled to return for a second time just as the majority of new aircraft are ending their 95-month break

in. This will cause a large drop in aircraft available to the fleet, and the problem will be exacerbated if turnaround times are significantly longer than intended. This behavior was detected in the stable model outputs even with the near-optimal turnaround times, and the major spike is visible in Figure 62. This spike around the 3000-day mark is consistent across simulation scenarios due to the fixed scheduling of the depot inspection. Future depot dates will need to be shifted, potentially greatly, to account for this future spike.

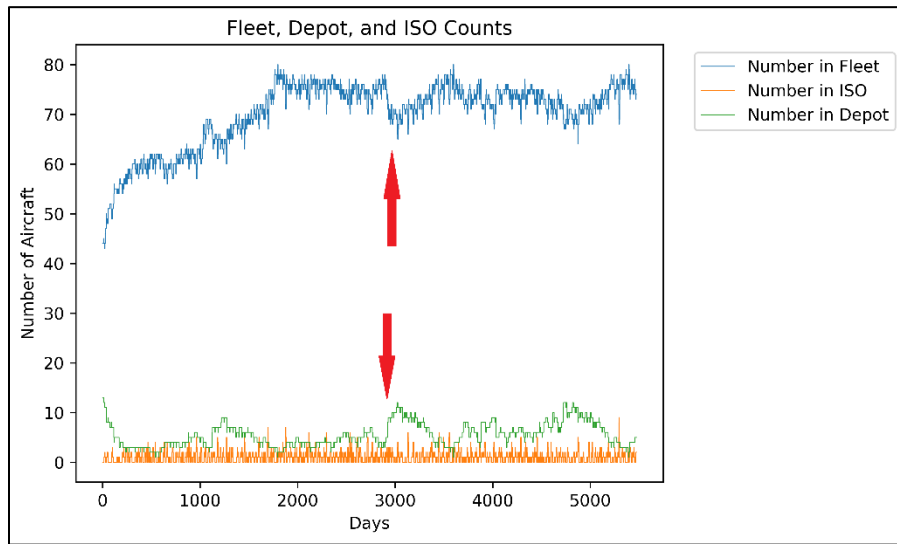


Figure 62. Pending Spike in Depot Levels and Corresponding Drop in Available Aircraft

In addition to the above considerations, the model developed for this thesis can be utilized to measure the effects of various depot completion times. Any input distribution can be accepted for the depot inspections, and the resulting impact to the fleet can be measured.

3. Automate Model Initialization Data Creation

The USMC currently uses Excel spreadsheets to track the depot and arrival schedule for aircraft. These spreadsheets provide a valuable look at the future state of the fleet, but they must be manipulated manually and are extremely cumbersome. Using the techniques designed to generate the model input data, an updated Excel file could be

designed to automatically produce the required inputs for the model. This would allow for real-time changes in the input data to be reflected in future simulations without the intensive process of generating the data manually. One of the most valuable characteristics of the model is that it can simulate the fleet exactly as it exists instead of relying on random generation, and retaining this capability would be very valuable.

4. Analyze Cause of Current Delays

The current 420- and 840-day median inspection times were found to be 47 and 56 days, respectively. However, this thesis did not determine why the inspections were taking so long. Other than examining the downing maintenance discrepancies that extended beyond the reported completion, no real analysis was conducted to see why the inspections took the time that they did. Overruns could be due to parts shortages, manpower shortages, poorly optimized inspection procedures, or additional maintenance that was discovered during the inspection. This analysis would require each individual inspection to be broken down into the sub-inspections and analyzed independently. Performing this analysis would be extremely beneficial because it could mean improving the inspection baselines without having to acquire additional resources (for example, optimizing the flow of inspections or maintaining a stock of a certain part).

5. Special Inspection Timing

The final recommendation for future work would be to explore the consequences of modifying the current 420-day cycle to something longer. The Air Force utilizes a 540-day schedule for the exact same airframe, although their preventative maintenance practices differ from the Marine Corps'. It would not be unreasonable to model the effects of a different ISO interval, and the model would be able to accommodate it with minimal modification. Of course, such a drastic change would have to be supported by engineering and maintenance checks, but the readiness impacts associated with the change could be freely examined. This research could be combined with modified depot schedules/timelines to assess a number of potential future scenarios.

LIST OF REFERENCES

- Buss A (2001). Basic event graph modeling. *Simulation News Europe*, Technical notes.
- Commander, Naval Air Forces (2017). The Naval Aviation Maintenance Program. COMNAVAIRFORINST 4790.2C. Patuxent River, Maryland.
<https://www.navair.navy.mil/sites/g/files/jejdrs536/files/2020-08/CNAFINST%204790.zip>.
- Duffuaa S, Raouf A. (2015). Planning and Control of Maintenance Systems, 2nd ed. (Springer, Switzerland)
- Ford M (2017). AFSOC ISO CRF. PowerPoint presentation provided to the author via personal communication, December 2020.
- Friedlein JR (2021) Depot scheduling data provided to the author via personal communication, January 2021.
- Government Accountability Office (2020). Weapon system sustainment. GAO-21-101SP. Government Accountability Office, Washington, DC.
<https://www.gao.gov/assets/gao-21-101sp.pdf>.
- Gross D, Shortle J, Thompson J, Harris C (2018). Fundamentals of Queuing Theory, 5th ed. (Wiley, New Jersey)
- Johnson A, Glasscock C, Little A, Muha M, O'Malley D, Bennet M (2008). C-5 isochronal inspection process modeling. *Proc. of the 40th Winter Sim. Conf.*
<https://doi.org/10.1109/WSC.2008.4736196>.
- Katz J (2019). Navy awards \$16 million contract mod for aviation maintenance efforts. *Inside the Pentagon's Inside the Navy*, 32(45), ProQuest Central.
- Law A (2015). Simulation Modeling and Analysis, 5th ed. (McGraw-Hill Education, New York)
- McLoughlin B (2006) Maintenance program enhancements. *Aero Quarterly* 06(4): 24–28
- Naval Aviation Enterprise (2020). Current readiness. PowerPoint presentation provided to the author via personal communication, December 2020.
- Naval Aviation Systems Command (2018). Card Deck: Special/Conditional Maintenance Requirements KC-130J Hercules and C-130J Aircraft. NAVAIR 01–75GAJ-6-3. Patuxent River, Maryland.

- Naval Aviation Systems Command (2019). User Manual (UM) for DECision Knowledge Programming for Logistics Analysis and Technical Evaluation (DECKPLATE) Aircraft Inventory and Readiness Reporting System (AIRRS) Ad Hoc and Reports. Document Number 6313. Patuxent River, Maryland.
- Naval Aviation Systems Command (2020). Periodic Maintenance Manual: KC-130J and C-130J Aircraft. NAVAIR 01-75GAJ-6. Patuxent River, Maryland.
- Naval Aviation Systems Command (2018). 2018 Maintenance, engineering, logistics review. Presentation.
- Peters D, Taylor B (2020). First-hand results of naval sustainment system-aviation. *Naval Aviation News*, 102(1), 4-6.
- Roo B, Carrillo M, Drew J, Lang T, Maletic A, Massey H, Masters J, McGarvey R, Sollinger J, Thomas B, Tripp R (2011). Analysis of the Air Force logistics enterprise: evaluation of global repair network options for supporting the C-130. TR-813-AF. RAND Corporation, Santa Monica, California.
https://www.rand.org/pubs/technical_reports/TR813.html.
- Schell G (2020). CNA analytics to support performance-to-plan aviation. PowerPoint presentation provided to the author via personal communication, October 2020.
- United States Marine Corps (2019). 2019 Marine Corps Aviation Plan.
<https://www.aviation.marines.mil/portals/11/2019%20avplan.pdf>.
- United States Navy (2021). Why P2P: A brief overview of P2P and its methodology.
<https://p2p.navy.mil/About-P2P/Why-P2P/>.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California