A Multi-Agent Simulation System for Prediction and Scheduling of Aero Engine Overhaul

Armin Stranjak,
Partha Sarathi Dutta
Strategic Research Centre
Rolls-Royce
Derby DE24 8BJ, UK.
{armin.stranjak,
partha.dutta}@rolls-royce.com

Mark Ebden
Dept. of Eng. Science
University of Oxford
Oxford OX1 3PJ, UK.
mark.ebden@eng.ox.ac.uk

Alex Rogers,
Perukrishnen Vytelingum
School of ECS
University of Southamtpon
Southampton SO17 1BJ, UK.
{acr, pv}@ecs.soton.ac.uk

ABSTRACT

The Aero Repair and Overhaul industry is facing an increasing challenge of prediction and scheduling of engine overhauls to remain competitive in a complex business arena. An appropriate technology solution is required to achieve efficient schedules while satisfying multiple opposing constraints in a highly dynamic environment. In this paper, we describe *Overhaul Prediction and Scheduling*, an agent-based simulator developed to tackle this challenge. Using negotiation strategies, it deals with the multi-dimensional scheduling optimisation problem by trading off repair costs, capacity and capability of overhaul bases, among others, in light of in-service unforseen events. It supports effective strategic decision-making via business scenario modelling.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Multi-Agent Systems

General Terms

Experimentation

Keywords

Scheduling, negotiation, risk estimation, simulation, aero repair and overhaul, multi-agent systems

1. INTRODUCTION

The challenge on the Aero Repair and Overhaul (AR&O) industry to remain competitive is growing with the increasing demands of airline businesses to manage larger fleets of aero engines. One of the most important AR&O functionalities is scheduling aero engines for repair at overhaul bases. To meet the growing demand effectively, it is desirable for AR&O scheduling to: (1) optimise the timing of aero engine repair to satisfy the cost, risk and revenue trade-offs; (2) minimise instances of aircraft-on-ground (AoG) waiting for engine repair completion; (3) have robust re-scheduling in light of unforseen events and continuously changing information; and (4) achieve effective visibility of the long-term effects of scheduling decisions. However, the above requirements are challenging because AR&O operations are highly complex, characterised by: (a) large numbers of parameters (e.g. airline operators, aircraft, engine fleets, repair bases, transportation and logistics, to name a few), each with a different and complex representation; (b) intricate interactions between these parameters; (c) large number of critical run-time process constraints; and (d) very high monetary stakes. Although current technology helps AR&O to generate acceptable schedules, assessing the long-term effects of scheduling decisions still remains a key challenge. Such decisions can be *operational* (finding a suitable time slot at an overhaul base to repair an engine) and *strategic* (investing in additional overhaul capacity and spare engines). Due to long service lives of engines (several decades), it is crucial to effectively address both. To do so, long-term simulations of AR&O operations are necessary. Using simulations of the business scenarios, alternative scheduling methods can be implemented and evaluated, and the consequences of different operational and strategic decisions can be investigated.

Against this background, we have developed an agent-based simulation tool — Overhaul Prediction and Scheduling (OPS) — capable of accurately simulating the complex AR&O operations. In OPS, different stakeholders of AR&O, such as fleet managers and overhaul owners, are modelled as autonomous agents who negotiate to decide the most *desirable* repair time of an engine that best satisfies the risk, cost, and revenue trade-offs. To do so, they use robust mechanisms for prediction and scheduling of engine overhauls. The reactiveness of OPS agents ensures quick and adequate scheduling (and re-scheduling, if required) responses to unforseen events that might affect engines and overhaul bases. The specific contributions of OPS, that make it a highly desirable tool for AR&O, are:

(1) Reliable estimation of component risk and their cumulative effect on whole-engine composite risk.
(2) Negotiation mechanisms for fleet managers and overhaul bases to schedule shop visit appointments.
(3) Adaptive re-scheduling to effectively mitigate the effects of unforeseen (in-service) events.
(4) Simulation of fleet operations, overhaul bases, and fleet managers over extended time periods.
(5) Flexible and effective ontology-based business modelling.
(6) Capability to log system variables and events that are critical for strategic decision-making via scenario modelling and "what-if" analysis.

Cite as: A Multi-Agent Simulation System for Prediction and Scheduling of Aero Engine Overhaul, A. Stranjak et al., *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)- Industry and Applications Track*, Berger, Burg, Nishiyama (eds.), May, 12-16.,2008,Estoril,Portugal,pp.81-88.

(7) Responsiveness to operational data (potentially located remotely) through Web Services interfaces.
(8) An advanced simulation engine and user interface allowing changes to simulation parameters flexibly and easily.
(9) Ability to operate in both real time and in simulation mode making it both an operational and a strategic tool.

In the rest of the paper, Section 2 discusses the AR&O domain; Section 3 presents the case for an agent-based simulation in AR&O; Section 4 describes our OPS simulator; Section 5 analyses the results from evaluating OPS; Section 6 highlights conclusions and future work.

2. AERO REPAIR AND OVERHAUL

Typically, an AR&O business works with multiple airline operators, involving multiple fleets of aero engines and multiple geographically distributed overhaul or repair bases (OHB) with limited resources and capabilities. AR&O is responsible for the planning and logistics required for coordinating the operational support to in-service engines. Scheduling aero engines for repair is one of the main AR&O activities. However, productive scheduling is a hard challenge due to the following critical constraints.

First, revenue earned from engines should be maximised by allowing them to fly for as long as possible whilst bounding whole-engine reliability. This can be achieved by precisely assessing engine repair dates through robust reliability estimation. Notably, the whole-engine reliability is a function of the individual reliabilities of its most critical modules, each of which has multiple reliability modes characterised by a Weibull probability distribution.[1] Second, disruption caused to the operator by AoG should be minimised. The high costs incurred due to AoGs can be mitigated by one or more of the following: (1) an efficient scheduling strategy that minimises the time an engine stays "off wing"; (2) maintaining an optimum number of spare engines; and (3) effectively predicting overhaul resource usage. Third, the limited resources of overhaul bases should be efficiently managed to allow maximum repair throughput and minimum latency. An efficient scheduling strategy can satisfy this requirement. Fourth, unforseen events affecting in-service engines and overhaul bases should be handled by contingency mechanisms. This can be achieved by re-scheduling shop visits in a robust and flexible manner. Finally, and most importantly, the long-term effects of different scheduling decisions should be assessed for making effective strategic decisions like investing in additional repair resources and / or additional spare engines.

While current AR&O operations generate acceptable repair schedules routinely, there is a clear requirement to extend existing technology to address the above requirements. However, optimal solutions for the above issues are computationally very hard and, for all practical purposes, not a stringent requirement. Rather, a competitive solution is desired that can demonstrably extend current repair policies and ensure improved productivity of AR&O operations.

To meet this requirement, we have developed a simulator of AR&O operations that performs reliable prediction of engine repair dates and efficient scheduling. It models the AR&O stakeholders as autonomous agents with operational policies incorporated. It uses high fidelity models of AR&O entities like overhaul resources and capabilities, engine fleets and aircraft types, flight patterns, and the process dynamics that relate these. These features ensure effective strategic decision-making by running, evaluating and comparing various operational scenarios and generating insights into the long-term effects of alternative business strategies.

3. AGENT-BASED SIMULATION OF AR&O

Simulation has been a powerful vehicle for rapid prototyping of complex systems in the aerospace arena [4, 11]. It provides visibility into the projected outcomes of intricate processes that is key for developing competitive business strategies while avoiding the risks and costs associated with building operational systems. The success of simulation-aided studies in aerospace is exploited through large-scale research programmes like VIVACE [2], funded by the European Commission and participated by sixty five partners across eleven countries. In this, various aerospace processes, models and methods are simulated within a distributed virtual enterprise. The partners work in this shared synthetic environment to perform design, implementation and validation of components, engines and whole aircraft. Such a flexible and integrated approach significantly reduces both the time and cost of engine and aircraft development.

Notably, simulation of complex systems can be achieved using different techniques — discrete event models, systems dynamics, Monte Carlo methods and agent technology being some of the most prominent ones. Among these, agents have been shown to provide simulation platforms with the most desirable properties [12]. It is a natural candidate for modelling open distributed systems that are too formidable to be treated using analytical methods, efficiently incorporating the resources, objectives, and policies of various process participants, and ensuring a high degree of design modularity and model execution clarity to enhance end-user confidence on the simulation outcomes. These have generated an ever-increasing interest in exploiting agent technology for developing robust simulators [6, 9, 10].

In light of the above discussion, the AR&O business, which is a complex system in its own right (see Section 2), can be effectively modelled using agent-based simulations. Indeed, a number of these have been developed in the aerospace domain. For example, the value chain of an engine development process is modelled as a multi-agent system in [5]. It uses a purpose-built simulator to create business scenarios and couples it with agent models of the value chain partners to understand potential future performances. In [7], inter-agent communication is used for creating integrated scenarios by combining individual simulation objects representing mission level interactions of defense systems. Although these applications address some requirements of the aerospace market in general, simulators for the AR&O domain are very limited in number. This is a significant deficit because the requirements of AR&O are unique and addressing them calls for bespoke technology. The most significant requirements are: estimation of engine reliability, negotiation between fleet managers and OHB owners for selecting engine repair dates, and robust scheduling mechanisms that can cope with multiple constraints (identified in Section 2). The Aerogility system [3], developed by Lost Wax, is the most relevant simulation tool in this context. It uses agent technology to simulate the aerospace aftermarket business which provides service for managing engine fleets and en-

[1] A Weibull CDF is of the form: $F(t) = 1 - e^{-(\frac{t}{\alpha})^{\beta}}$, where α and β are *scale* and *slope* parameters respectively.

suring consistency and reliability of operators. The issues of the AR&O domain that we aim to address using our OPS simulator are similar to those in Aerogility. Furthermore, like OPS, Aerogility can evaluate alternative scenarios to perform "what-if" analysis for assessing future outcomes of business strategies. However, while Aerogility deals with only high-level system performances, OPS is capable of capturing additional and more fine-grained aspects like (1) accurately estimating whole-engine reliability by combining component reliabilities; (2) dynamically updating reliability using environmental factors and operational specificities; (3) determining repair dates dynamically using reliability estimates; (4) modelling individual OHB capacities and capabilities; and (5) implementing robust and detailed scheduling algorithms for OHBs. These features of OPS complement and extend the contributions of Aerogility to AR&O.

4. SIMULATOR FOR OVERHAUL PREDICTION AND SCHEDULING

The key technical capabilities of OPS are: (1) Multi-agent negotiation for scheduling and adaptive re-scheduling, (2) Modelling whole engine reliability, (3) Response to unforseen events, and (4) Post-analysis of stored performance data.

4.1 Multi-agent negotiation for scheduling and adaptive re-scheduling

4.1.1 *Agent Types*

The main functional entities of OPS are overhaul bases, fleet managers and a fleet planner. OHBs perform repair and maintenance of engines according to the engine types. An OHB contains multiple repair lines capable of repairing several engine types. A Fleet Manager (FM) is responsible for recording engine events (performance, vibrations, irregularities, etc.) during service and for simulation of engine utilisation. It also records statistics such as the number of cycles and hours of service, Weibull estimates of individual reliability modes and whole-engine reliability. A Fleet Planner (FP) determines shop visit schedules of engines which require servicing due to several reasons. The first one is when it is used for a certain number of cycles. Based on an engine's utilisation rate and this limit, it is possible to estimate the *Date of Zero Life* (DOZL) — the projected date in the future before which an engine has to be serviced. The second reason is based on a prediction (via continuous monitoring) about when the conditions of engine components reach certain limits. This predicted date is the *Date of Zero Margin* (DOZM). The third reason is the estimation of an engine's *composite* risk of disruption during service, which is a combination of all individual component risks (see Section 4.2). Given a risk threshold and a time-varying composite risk function, the estimated date when the risk exceeds the threshold is the *Date of Zero Confidence* (DOZC). An engine would require servicing on or before any of these dates, whichever is the earliest — its *removal date*. Other factors that the FP considers are engine usage pattern and OHB capacity and availability. The usage pattern helps to determine the best match with an engine's physical location at any time and that of the destination OHB. From OHB utilisation information, the FP can generate valid, conflict-free schedules. At the same time, the FP aims to maximise revenue by keeping engines in service for as long as possible.

Each of the above functional entities are represented by individual agents in OPS. For each engine fleet, there is one FM agent. A fleet is defined by a set of engines of the same type belonging to a certain airline operator. For each overhaul base, there is an OHB agent. The functionalities of the fleet planner are implemented within a single FP agent.

4.1.2 *Negotiation Scenarios*

The FP agent creates schedules up to a certain time in the future, called the *Schedule Time Horizon* (STH). As soon as the removal date of an engine enters this period, the corresponding FM agent requests FP for it to be scheduled. An FM agent issues a similar request to the FP whenever there is any change in an engine's removal date (if that is within STH). The removal date can alter due to: changes in an engine's usage pattern or utilisation rate, fleet-wide updates of engine characteristics, or unforseen events such as a bird strike. An OHB agent requests scheduling due to OHB-specific events such as delayed delivery of equipment and maintenance operations. The interactions between FMs and OHBs and the FP are shown by the `Reschedule Request` events in Figure 1.

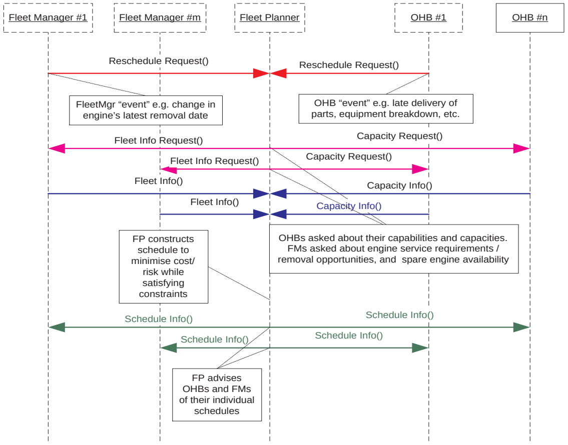
Fleet Manager #1
Fleet Manager #m
Fleet Planner
OHB #1
OHB #n
Reschedule Request()
Reschedule Request()
FleetMgr "event" e.g. change in engine's latest removal date
OHB "event" e.g. late delivery of parts, equipment breakdown, etc.
Fleet Info Request()
Capacity Request()
Fleet Info Request()
Capacity Request()
Fleet Info()
Capacity Info()
Fleet Info()
Capacity Info()
FP constructs schedule to minimise cost/ risk while satisfying constraints
OHBs asked about their capabilities and capacities. FMs asked about engine service requirements / removal opportunities, and spare engine availability
Schedule Info()
Schedule Info()
Schedule Info()
Schedule Info()
FP advises OHBs and FMs of their individual schedules

Figure 1: Rescheduling sequence diagram

To respond to such requests from FM and/or OHB agents, the FP requires fleet-related and capacity information from FMs and OHBs respectively (`Fleet Info Request` and `Capacity Info Request` events in Figure 1). Based on the acquired information (`Fleet Info` and `Capacity Info` events in Figure 1), the FP will create a new schedule and communicate this to all affected FMs and OHBs (`Schedule Info` events in Figure 1).

Ideally, engines should be serviced at the latest opportunity before their removal dates to maximise utilisation and, hence, revenue earned. However, if an engine needs servicing earlier than anticipated (due to unforeseen events), such "compact" scheduling can lead to wide-spread disruptions of already scheduled engines. To mitigate this, the FP maintains a *contingency buffer* around all appointments to allow re-scheduling flexibility.

4.1.3 *Scheduling Algorithm*

The primary role of the FP agent is to schedule engines for servicing. After identifying those engines with their latest removal dates within the STH, FP creates the shop visit appointments using the following mechanism. First, the FP tries to schedule all engines (in no particular order) at their desired dates (i.e. within the contingency buffer distance from their latest removal date). Where this is not achieved (as the given OHB capacity is typically unable to fit all appointments at their optimal dates), the FP tries to schedule them as close as possible to their optimal removal date. The FP then identifies the *worst positioned* engine in the current schedule (one with the highest scheduling cost) and tries to swap its appointment with another only if doing so decreases the sum of the scheduling costs of both engines involved. It is important to highlight here that the cost of a schedule is a function of the deviation of its actual servicing date from the latest removal date. This is shown qualitatively in Figure 2. Here, if it is serviced before the optimal date, the cost increases because of missed opportunity of flying the engine for longer. Servicing at a time in between the optimal and the latest dates reduces the schedule flexibility and, hence, has a higher cost than the optimal. Finally, the cost after the latest date is prohibitive because, in this case, the engine will be waiting at an OHB queue (it will be removed from aircraft anyway by the latest removal date) without an immediate service appointment and losing valuable revenue. The swap that maximises this cost reduction is the one actually done. A swap changes the schedule and the algorithm starts investigating for a swap of the worst positioned engine of the new schedule. If it cannot find a swap with any other appointments, the next to the worst positioned engine is investigated. The algorithm terminates when no feasible swaps are identified for all engines.

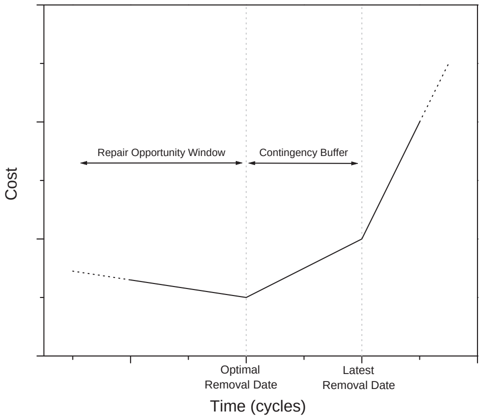

Figure 2: Scheduling cost variations

Re-scheduling appointments that are in the immediate future (e.g. in few days) would be impractical since the OHBs require a minimum lead time for service setup. A *Lock-in Time Horizon* (LTH) is used to prevent any further rescheduling of appointments that are within this time interval from their removal date. This period guarantees the eventual arrival of an engine at its destination OHB and gives flexibility to OHBs to organise the appropriate service plans.

4.2 Whole Engine Reliability Modelling

Estimation of removal dates depends on whole-engine reliability. Engine removal must happen within the time when the reliability is within an acceptable limit. However, jet engine components have widely varying event distributions, i.e. varying histograms describing at what time disruptions might occur. Whole-engine reliability is determined by combining individual component distributions, approximated by the Weibull function. By adjusting the function's so-called *scale-* and *shape* parameters, several different disruption types can be modelled:
(1) Infantile: occurrence risk decreases substantially after first few flights.
(2) Random: corresponding to constant risks.
(3) Wear-out: risk increases over an engine's life (fatigue).

The combination of M component risks can be represented as a finite mixture model: $p(t) = \sum_{j=1}^{M} P_j p(t|j)$, and $p(t|j) = b_j a_j^{-b_j} t^{b_j - 1} e^{-(\frac{t}{a_j})^{b_j}}$. Here, $p(t)$ is the probability of a disruption occurring at time t, and P_j is the *a priori* probability that any disruption is caused by the jth component risk. $p(t|j)$ is the likelihood that the jth component risk will lead to a disruption at time t (a Weibull function with a_j and b_j being scale- and shape parameters). The $3M$ parameters: P_j, a_j, and b_j for $j = 1, 2, \ldots, M$, are estimated using a maximum likelihood technique, and 95% confidence intervals on the parameters determined using the Fisher information matrix [8]. The data used to train the model consists of a set of engine events recorded over time.

The results of such finite mixture model analysis can enhance predictability and reliability, safely reducing the number of engine removals. In addition, it can reveal the effects of simple interdependencies among the disruption modes.

4.3 Response to Unforseen Events

A major requirement in AR&O is the creation and maintenance of flexible and robust schedules that can cope with frequent changes due to unpredictable and dynamic events, affecting one or more engines or entire fleets. If one engine is affected, it should not greatly impact other appointments in the schedule due to the contingency buffer; possibly altering those that are adjacent to the affected one. If multiple engines are affected (e.g. due to fleet-wide engine update), it can cause more widespread alternations to schedules. In addition, an OHB event, due to repair delays or maintenance activity, might affect engines on one or more repair lines.

OPS is capable of handling these events effectively and efficiently. By exploiting the contingency buffer and adaptive re-negotiation mechanism (Section 4.1.2) the FP agent can create new schedules "on-the-fly" without impacting current appointments significantly.

4.4 Post-Analysis of Performance Data

During simulation, OPS can record and store a variety of performance related data that are used for post-analysis. We can run OPS for long periods of simulated time equivalent to several decades in real time (one simulated day being roughly equivalent to a few real-time seconds). It is crucial to simulate over such time frames since normal lifetime of an aero engine can span over several decades. To achieve a reasonable visibility into all kinds of market dynamics over such extended time periods and enable strategic decision-making is an extremely challenging task. Clearly, analysing a multi-dimensional parameter space characterised by OHB capacity and capability, aircraft availability, engine risk estimation models, and their overall impact on servicing schedules is a highly complex task. The only feasible approach to explore the parameter space is via "what-if" scenario modelling and storing data such as utilisation of OHBs, turnarond time of engines, repair costs, spare levels and scheduling performance. Such data then enables exploration of various business policies and decision-making processes. Section 5.2 presents the results of extensive post-analysis using OPS.

5. EXPERIMENTAL RESULTS

5.1 Technology Overview

The OPS tool consists of multiple OHB and FM agents, one FP agent, a WS (Web Services) Bridge Agent and a Control Agent (Figure 3). The roles of OHB, FM and FP agents are already explained in Section 4.1. The WS Bridge Agent acts as the interface between the OPS tool and Web Services-based software systems, bridging the gap between asynchronous agent-based systems and the typically synchronous external applications such as databases or web-based applications. More specifically, the WS Bridge Agent allows OPS to: (1) access external (non-agent) data sources such an engine performance monitoring data-bases for performing whole-engine reliability modelling on "live" data and (2) respond to (non-agent) client requests such as rescheduling from real-life fleet managers. The Control Agent has two major roles: system control and data visualisation. As a system controller, it reads XML-based simulation and system properties files, and accordingly creates and initialises all other agent types within OPS. In addition, it can store the state of any agent at any time during a simulation via serialisation.

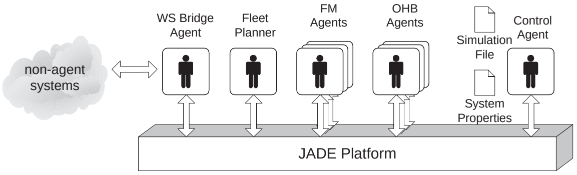

Figure 3: Architecture Overview

The OPS tool is developed using the Java programming language and is built on top of the JADE agent platform [1]. Our choice of this platform relied on several factors: FIPA compliance, open and flexible architecture, message filtering capability (supporting context- and conversation-dependent negotiations) and suitability for our requirements when compared with other considered agent platforms. An XML-based ontology is used to describe all message structures (used in inter-agent communications) and to ensure a common understanding, between all agents, about message performatives and contents. The ontology defines the agent internal data structures and determines the simulation file structure. A screen shot of the tool is shown in Figure 4. It shows four OHBs with various number of repair lines. Shop visit appointments, displayed as rectangular boxes of different colours, move to the left as time progresses. When they reach "now" (on the time-line shown at the top of the panel) an animation is launched showing an aircraft landing (on the left-hand panel) and its engines being moved to either an OHB or an OHB queue. At the same time, another animation (on the left panel) shows spare engines being mounted on the aircraft which then flies back into operation. A progress bar next to an engine in an OHB shows the number of days left in its servicing.

The colours of the appointment boxes denote the quality of a schedule. Thus, Green represents an appointment that is at the engine's ideal removal date. If the actual removal date is much earlier than the latest, the appointment is shown in Blue. Orange and Yellow represent those that are before their latest removal dates but are very close to it (lacking re-scheduling flexibility). Finally, Red marks those that are removed on the latest removal date but cannot be serviced due to lack of OHB resources. Such an engine, after being taken off an aircraft, is stored in an OHB queue where it waits until its destination OHB becomes available. The left panel shows a world map representing aircraft in service, the spare engine pool and the OHB queue.

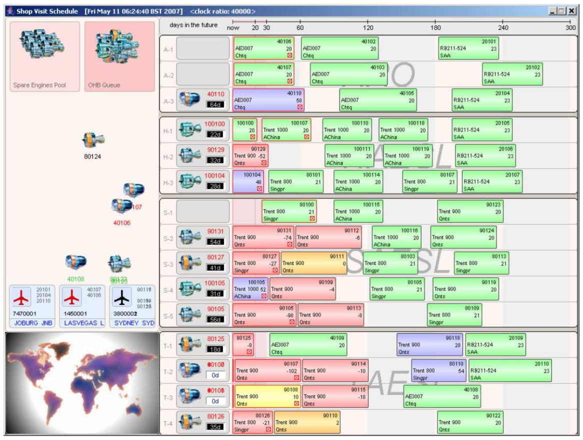

Figure 4: OPS scheduler visualisation

5.2 "What-if" Analysis

One of the major contributions of OPS is its ability to simulate AR&O operational scenarios for extended time periods and assess projected performance impacts. Although a large number of performance metrics can be measured using OPS, it is sufficient to present the results from a representative set to highlight its capability. Some of the most important metrics in AR&O are: turnaround time, number of AoG occurrences, and OHB utilisation. Note, all parameter values used for the experiments in this paper are artificial exemplars and do not represent any actual operations.

5.2.1 *Impact on Turnaround Time*

An engine repeatedly undergoes a cycle of events: (1) in service, (2) in transport from an airport to an OHB, (3) in an OHB queue, (4) in repair at an OHB, (5) in transport from OHB to airport, and (6) in spare engine pool. The total time it spends during events (2) to (5) is when it is unavailable and hence prevented from earning revenue. Turnaround time relates to this period of unavailability. Our simulations record the time an engine spends during each of the above events. Thus, if within a given simulation period, an engine has been unavailable for a total of t days and has made s repair visits, then $\frac{t}{s}$ represents its turnaround per repair (tpr). Turnaround is measured across each engine fleet. If a fleet X has N_X engines with tpr values $\{tpr_i^X \mid i = 1, ..., N_X\}$, then the turnaround per repair for the fleet is: $tprf^X = \frac{\sum_i tpr_i^X}{N_X}$.

Figure 5 plots the $tprf$ values averaged over five simulation runs, each running a certain scenario for several simulated calendar years. A scenario is defined by two factors: (1) the number of in-service engines in a certain fleet ($\mid E_{is}^X \mid$), and (2) the number of spare engines dedicated for that fleet ($\mid E_{sp}^X \mid$). Each plot in Figure 5 records the $tprf^X$ against ratio of engines to spares for different $\mid E_{is}^X \mid$ value: 10, 12, 16 and 20. For each $\mid E_{is}^X \mid$, the value of $\mid E_{sp}^X \mid$ is decremented as: $\frac{\mid E_{is}^X \mid}{2}$, 5, ..., 0.

In these experiments, the minimum value of $tprf$ is 60 days. This is because, the durations of events (2), (4) and

(5) are fixed for any repair: 5, 50 and 5 days, respectively. Event (3), however, can vary in length due to varying sizes of the waiting queue at OHBs, with a minimum value of zero. Note that, the OHB capacities are the same across all these experiments. Therefore, for any given value of $| E_{is}^{X} |$, as the number of spare engines increase, so does the contention for sharing OHB repair capacity. Thus, the average waiting time at OHBs (event (3)) increases, causing $tprf$ to increase. The same explanation holds for the increase in $tprf$ as $| E_{is}^{X} |$ increases. With an increase in the total number of engines (considering both in-service and spares), the OHB waiting time increases. There is, however, a significant benefit of having more spares — having far less AoGs (discussed shortly in Figure 7), thereby, more than compensating the effects of higher $tprf$ values.

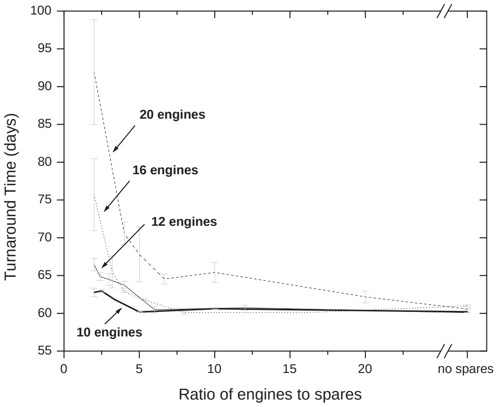

Figure 5: Turnaround time

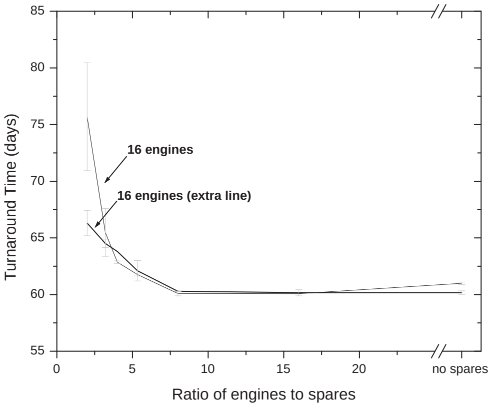

Figure 6: Comparing turnaround times

The detrimental effects on turnaround time can be reduced by having more repair capacity. This is shown in Figure 6 where an extra repair line is added to an OHB capable of repairing the particular fleet being studied. Thus, for the same $| E_{is}^{X} |$ (16 in this case), $tprf$ is significantly reduced by having one extra OHB line. Notably, this reduction is most prominent when the number of spares is $\frac{|E_{is}^{X}|}{2}$, when the contention for OHB capacity is the most severe. With fewer spares, $tprf$ is at its minimum possible value anyway; hence, extra OHB capacity bears no impact on it.

These experiments reveal information that is crucial for assessing the long-term impact that a certain number of spare engines, fleet size and OHB capacity (and their combinations) have on turnaround time. Estimating this analytically is extremely hard if not infeasible (Section 3). It is also not possible to perform such evaluations in real-time given the criticality of AR&O operational processes. Hence, simulation is an indispensable tool for generating such awareness and guides strategic decision-making.

5.2.2 Impact on AoG

An AoG is caused due to the unavailability of one or more engines required to be allocated to an aircraft with missing engine(s). Since airline operators earn revenue when the aircraft are in service, AoGs are undesirable for both operators (severe impact on service quality) and AR&O who incur penalties for disruptions caused to operators. Due to the complex nature of global fleet management, it is infeasible to precisely assess the impact that a certain resource level (fleet size, spare engines and OHB capacity) might have on the number of AoGs. Simulation offers an effective way of measuring this critical performance criterion.

Figure 7 shows the number of AoGs caused over the entire simulated period under different scenarios (defined similarly as in Figure 5). With fewer spare engines, the number of AoGs increases, thereby negating the benefit of reduced turnaround time as found in Figure 5. Also, for the same ratio of total number of engines to number of spares, AoGs increase with the total number of engines. Although the general trends in these plots are intuitive, it is vital for AR&O strategic decision-making to ascertain the *actual number* of AoGs caused in a given scenario and the *actual differences* in this number across different scenarios. This is infeasible without having a robust simulator.

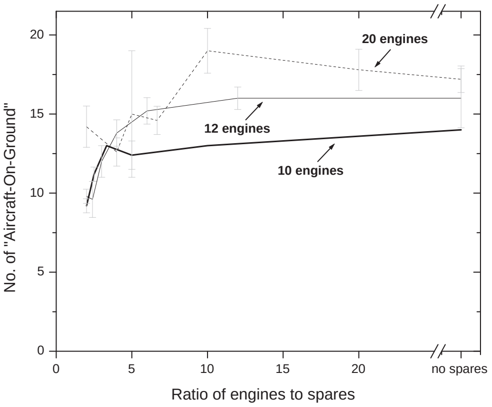

Figure 7: Aircraft on ground occurrences

5.2.3 Impact on OHB Utilisation

OHBs represent a key investment of AR&O. To ensure competitive returns, OHBs should be operating optimally. One way of achieving this is to reduce their idle times as much as possible. Again, it is a significant challenge to estimate OHB utilisation analytically in the complex AR&O business. The OPS simulator is capable of measuring OHB utilisation under various scenarios and, thus, generate vital information about the operational efficiency of AR&O.

Figure 8 shows the number of days that an OHB is idle as a percentage of the total number of days in a simulation. These plots show that in general, OHB utilisation decreases (idle time increases) as the number of engines and spares decrease. The total repair time decreases with fewer engines. For the same number of engines and spares, OHB utilisation reduces with extra OHB capacity (Figure 9).

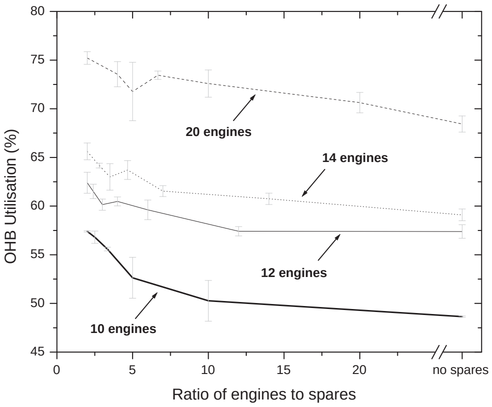

Figure 8: OHB utilisation

Note that although Sections 5.2.1, 5.2.2, and 5.2.3 show results of separate performance measures, they are in fact inter-related. This is revealed in the figures where each scenario simultaneously influences (favourably or otherwise) turnaround, AoGs and OHB utilisation. It is only by considering them together that a holistic picture about the trade-offs in performance can be obtained which can then be fruitfully exploited for effective business decisions.

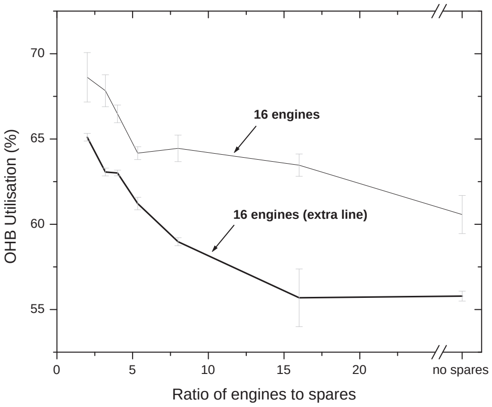

Figure 9: Comparing OHB utilisation

5.3 Responsiveness to Unforseen Events

AR&O operations require appropriate mechanisms to handle unforseen events. Such events, including technical factors and environmental circumstances, may necessitate servicing of engines immediately or within a certain time period. One example is a "campaign", which is when an entire fleet requires a certain update at the earliest opportunity before a certain deadline.[2] Given that other fleets would have to undergo their routine repair schedules and that OHB resources are shared across fleets, it is vital to understand what impact such a "campaign" might have on the system. One way of measuring the reactiveness of the system is by measuring the level of spare engines. This information is of critical importance because, as shown in Section 5.2, spare engines are valuable resources that determine a variety of other performance measures. Using robust simulation and detailed data logging, OPS can not only assess the impact on performance caused by unforseen events but also provide crucial diagnostics support.

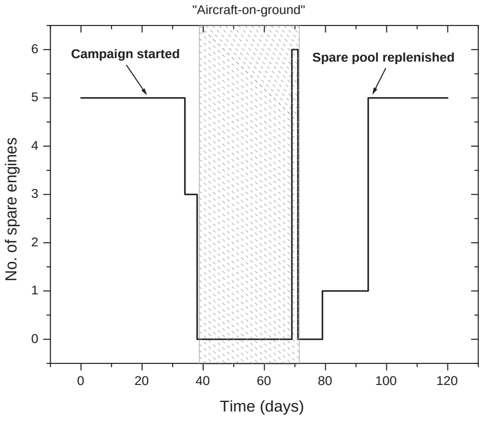

Figure 10: Spare Engines Level

Figure 10 shows the spare level of a certain fleet (with 6 engines, flying on 3 aircraft, and 5 spare engines) for which a campaign event was issued. Due to different flight patterns, one aircraft flew in for service before the rest. This is when 2 spares were fitted (day 37) on that aircraft for it go back in service. Its previous 2 engines were sent to an OHB. On day 40, the remaining 2 aircraft were grounded for service and their 4 engines were sent to OHB. However, having only 3 spares left at that time, only one aircraft could return to service. So, from day 40, there was one AoG instance. The last spare engine was also sent to repair (all engines should respond to the campaign). Subsequently, the two in-service aircraft returned between day 40 and 70 (i.e. at the earliest opportunity as required by the campaign) and 4 of their engines went to OHB. At this stage, with no spares, there were 3 AoGs and all 11 engines were at OHBs. On day 70, servicing of 6 engines (those flying at the beginning) was completed and they were sent to the spare pool. However, with 3 AoGs, they were immediately fitted to enable them to fly. Servicing of one more engine (the spare which was never fitted) was completed on day 80 and sent to spares. Finally, on day 96, servicing of 4 engines (which started between days 40 and 70) finished and they were sent to spares.

5.4 Timing Analysis of Scheduling

Although the experiments were conducted in faster-than-real time mode, we are interested in analysing the real execution time of the scheduling algorithm (Section 4.1). A fast algorithm will ensure that the time to generate a new schedule would be less than the smallest interval between consecutive scheduling requests, thereby, guaranteeing consistent schedules. Another way of evaluating this algorithm is by measuring the *quality* of schedules. Indeed, the results from the "what-if" experiments (Section 5.2) capture some indirect quality measures because turnaround time, number of AoGs and OHB utilisation are impacted by the manner in which the repair jobs are scheduled. As future work, we aim to compare these results against other benchmark scheduling algorithms such as market-based mechanisms (Section 6). Figures 11 and 12 show the real execution time of the algorithm in a simulation with 4 OHBs and 15 repair lines.

[2] The objective is to let the engines fly for as long as possible until the deadline. However, they cannot be left to run until the very last moment because, due to limited OHB capacities, staggered repair plans are required.

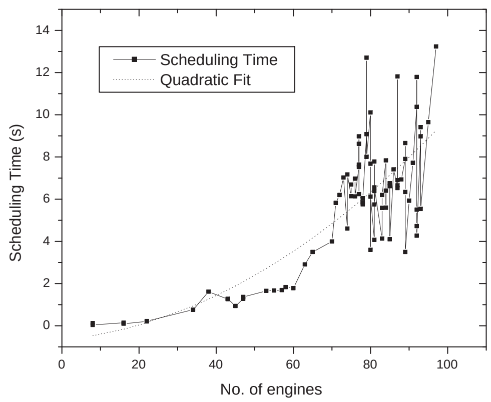

Figure 11: Scheduling algorithm execution time

In Figure 11, up to about 60 engines, the algorithm run time increases linearly with the number of engines. The time to schedule a larger number of engines increases quadratically. What is important, however, is that the algorithm run time always remains several orders of magnitude smaller than the typical interval between consecutive scheduling requests. One notable feature of Figure 11 is the "noisiness" of the plot at larger numbers of engines. With a large number of engines flying with different usage patterns and repair requirements, there occur instances when a lot of these are overhauled at OHBs simultaneously whereas on other occasions the scheduling load remains light. These happen alternately but not necessarily with a fixed periodicity due to the largely dissimilar usage patterns across fleets.

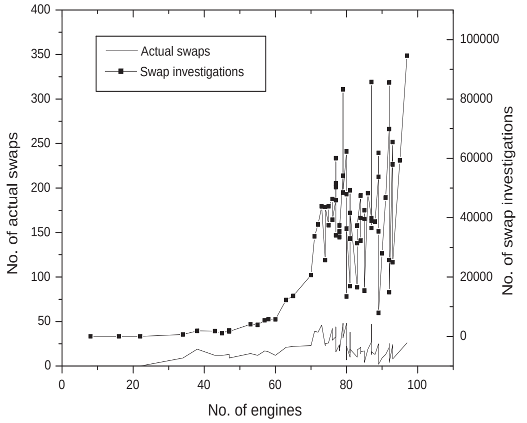

Figure 12: Comparing schedule swaps

The increase in scheduling time is caused by an increase in the number of investigations if a swap should be made (see description in Section 4.1). This is shown in Figure 12. With more engines requesting repair slots simultaneously, the number of appointments with non-optimal repair dates increases since OHBs have fixed and finite capacities. As mentioned in Section 4.1, swap investigations occur for such appointments. Hence, the number of swap investigations increases. However, because of the finite OHB capacity, there is very limited leeway for a profitable swap when the demand for repair dates is high. Thus, the actual number of swaps remains small, as shown in Figure 12.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented the Overhaul Prediction and Scheduling tool for simulation of complex AR&O operations. The tool's capability includes reliable estimation of whole-engine composite risk, scheduling of overhaul shop visits, adaptive re-scheduling to mitigate effects of unforeseen events, and ability to be used as a decision-making support tool by exploring different business policies via scenario modelling and "what-if" analysis.

In our future work, we aim to use market paradigms as a distributed solution to the scheduling problem. One of the many advantages of such solutions is their scalability with increasing problem size with no requirement of a single planner to have complete knowledge of all other entities in the system (e.g. fleet managers and OHB owners). Thus, individual preferences can be kept private as they compete against each other in a marketplace that intrinsically adapts to the changing demand and supply (of repair time slots).

Acknowledgement:.

This research was undertaken as part of the ARGUS II DARP. This is a collaborative project involving BAE SYSTEMS, QinetiQ, Rolls-Royce, University of Oxford and University of Southampton, and is funded by the industrial partners together with the EPSRC, MoD and DTI. We thank Peter Swann and Graham Hesketh for their valuable contributions in developing OPS.

7. REFERENCES

[1] Java Agent DEvelopment Framework (JADE). http://jade.tilab.com.

[2] Value Improvement through a Virtual Aeronautical Collaborative Enterprise (VIVACE). http://www.vivaceproject.com, 2004.

[3] Aerogility. http://www.aerogility.com.

[4] R. Agarwal and J. Gilmer. Back from the future: Using projected futures from recursive simulation to aid decision making in the present. In *Proceedings of the International Symposium on Military, Government and Aerospace Simulation*, 2004.

[5] D. Buxton, R. Farr, and B. McCarthy. The aero-engine value chain under future business environments: using agent-based simulation to understand dynamic behaviour. In *Proceedings of MITIP*, 2006.

[6] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan. MASON: A multiagent simulation environment. *SIMULATION*, 81:517–527, 2005.

[7] J. T. McDonald and M. L. Talbert. Agent-based architecture for modeling and simulation integration. In *Proceedings of the National Aerospace and Electronics Conference*, 2000.

[8] G. McLachlan and D. Peel. *Finite Mixture Models*. John Wiley and Sons, Inc., 2000.

[9] C. L. N. Minar, R. Burkhart and M. Askenazi. The swarm simulation system: A toolkit for building multi-agent simulations. Technical report, Sante Fe Institute, 1996.

[10] M. J. North, N. T. Collier, and J. R. Vos. Experiences creating three implementations of the Repast agent modeling toolkit. *ACM Transactions on Modeling and Computer Simulation*, 16(1):1–25, 2006.

[11] A. R. Pritchett, M. M. van Paassen, F. P. Wieland, and E. N. Johnson. *Applied system simulation: methodologies and applications*, chapter Aerospace vehicle and air traffic simulation, pages 365–389. Kluwer Academic, 2003.

[12] T.-T. Yu. A Comparison of Agent-based Modelling with Traditional Discrete-event Modelling. Technical Report ECSTR-LSL07-006, University of Southampton, 2007.