Hindawi Publishing Corporation
ISRN Aerospace Engineering
Volume 2014, Article ID 204546, 13 pages
http://dx.doi.org/10.1155/2014/204546

Research Article

Condition Based Maintenance Optimization of an Aircraft Assembly Process Considering Multiple Objectives

J. Li,[1] T. Sreenuch,[2] and A. Tsourdos[3]

[1] *Shanghai Aircraft Manufacturing Co., Ltd., Shanghai 200436, China*
[2] *Integrated Vehicle Health Management Centre, Cranfield University, Bedford MK43 0AL, UK*
[3] *Division of Engineering Sciences, Cranfield University, Bedford MK43 0AL, UK*

Correspondence should be addressed to T. Sreenuch; t.sreenuch@cranfield.ac.uk

Received 31 October 2013; Accepted 29 December 2013; Published 11 February 2014

Academic Editors: V. G. M. Annamdas, C. Bigelow, R. V. Rao, Y. Shi, and A. Yesildirek

The Commercial Aircraft Cooperation of China (COMAC) ARJ21 fuselage's final assembly process is used as a case study. The focus of this paper is on the condition based maintenance regime for the (semi-) automatic assembly machines and how they impact the throughput of the fuselage assembly process. The fuselage assembly process is modeled and analyzed by using agent based simulation in this paper. The agent approach allows complex process interactions of assembly, equipment, and maintenance to be captured and empirically studied. In this paper, the built network is modeled as the sequence of activities in each stage, which are parameterized by activity lead time and equipment used. A scatter search is used to find multiobjective optimal solutions for the CBM regime, where the maintenance related cost and production rate are the optimization objectives. In this paper, in order to ease computation intensity caused by running multiple simulations during the optimization and to simplify a multiobjective formulation, multiple Min-Max weightings are used to trace Pareto front. The empirical analysis reviews the trade-offs between the production rate and maintenance cost and how sensitive the design solution is to the uncertainties.

1. Introduction

Nowadays, aircraft manufacturers are operating in a global competitive environment. Increasing production rate and reducing costs are the key drivers in aircraft manufacturing. In order to meet the required production rate while meeting high quality requirements, (semi-) automatic assembly machines (e.g., Flexible Drilling Head [1], GRAWDE (Gear Rib Automated Wing Drilling Equipment), and HAWED (Horizontal Automated Wing Drilling Equipment) [2]) are increasingly being used in the aircraft assembly line. These machines can deliver significant productivity gains on the shop floor by reducing the manual multistep processes and overcoming the restricted worker access [3]. This in effect has shifted the production throughput to be now very much dependent on the operational availability of these (semi-) automatic machines [4]. Consequently, machine breakdowns and maintenance are therefore a major cause of bottlenecks in the assembly line. How to manage these machines in an efficient and cost-effective way to maximize the overall product rate is still a key challenge to the aircraft manufacturers [2].

Maintenance involves fixing when equipment becomes out of order (corrective maintenance) and also includes performing routine actions which will keep the equipment working in order or prevent failures from arising (i.e., preventive maintenance) [5, 6]. A maintenance strategy in general includes identification of parameters, inspection methods, plan execution, and repair [7, 8]. In the recent decade, Condition Based Maintenance (CBM) has increasingly been integrated as part of the manufacturing system [4, 9–12]. Its goal is to minimize unscheduled downtime and shift towards a more forward-looking approach by monitoring deterioration of equipment conditions. Examples of integrated CBM in manufacturing system are found in the areas of measurement equipment [13], plastic injection [14], plastic yoghurt pots [15], food and drink industry [16], PBL (performance-based logistics) contracts [17], and generic stochastically deteriorating systems [18]. In these examples, it has been shown that CBM can potentially improve the overall cost and production rate of the manufacturing systems by increasing the machine availability while reducing the maintenance cost [17, 18].

In many cases, manufacturing for an example, where many high-value assets (machines) are part of it, is impractical and economically not feasible to experiment different manufacturing processes based on the real objects [9]. Simulation could allow complex process interactions of assembly, equipment, and maintenance to be captured and empirically studied in a virtual environment without having to build a real manufacturing system. Agent Based Simulation (ABS) and Discrete Event Simulation (DES) have been used in the manufacturing domain. ABS is based on the dynamic interaction of entities involved in the process. Examples of the ABS are autonomic manufacturing execution system [19] and intelligent manufacturing (e.g., enterprise integration and collaboration, manufacturing process planning, and scheduling) [20]. DES is on the other hand based on a fixed sequence of operations or process being performed over entities [9]. DES is more widely adopted in manufacturing as fixed sequence of operations can be naturally captured [9]. However, in a highly complex process it is simpler to model using ABS; complex interactions between entities (e.g., machine, service, and process) can be naturally captured in ABS without having to reformulate a problem into the queue theory framework as required by DES [21, 22]. Both DES and ABS allow the important aspects like quality, cost, and time to be simulated and analyzed which provide the basis in Manufacturing System Development (MSD) and Product Realization Process (PRP) [23].

In simulation, a model comprises several input variables or model parameters such as scheduling properties, process leap time, and machine reliability. The aim of MSD or PRP is to find optimal controllable parameters that will result in the most desirable outputs of the process. In the case of (semi-) automatic assembly lines, examples of performance indicators are maximum production rate and minimum maintenance cost. To find an optimal solution, the simulation is iterated until the most optimal combination of variables is found; at each iteration the controllable variables are adjusted, the model is simulated, and the simulation output is then evaluated against the design objectives [24, 25]. Evolutionary techniques (e.g., scatter search and genetic algorithms) are often applied to solve difficult simulation optimization problems [26–28].

In this paper, CBM is exploited as part of a design solution for a (semi-) automatic aircraft assembly process that demands high production rate and until now there are no studies of CBM in an aircraft assembly process reported in the literature. This and the simulation optimization of a CBM integrated aircraft assembly process model will be the contribution of this paper. In this study, Commercial Aircraft Cooperation of China (COMAC) ARJ21 regional jet final assembly is used as a what-if representative example to illustrate the impact of CBM on the aircraft assembly process.

The outline of this paper is as follows: Section 2 describes an aircraft assembly process and identifies the key performance bottlenecks in the process. Sections 3 and 4 explain CBM and how a CBM enabled aircraft assembly system is modelled using agent concepts. A multiobjective simulation optimization approach is described in Section 5. Section 6 describes performance measures and then evaluates tradeoff between competing objectives and their relation to design parameters. Finally, the concluding remarks are made in Section 7.

2. Aircraft Assembly Process

2.1. ARJ21 Structure Assembly. ARJ21 is one of two ongoing COMAC's regional jet development programs [29]. It is a new type of turbofan short/medium range 78–90 seat regional aircraft. The ARJ21 program is in the ongoing certification process and is currently in transition from development stage to batch serial production. The ARJ21 has received a total of 309 orders as of 2013. COMAC has planned to increase its production rate to 30 aircrafts per year by 2015. However, at this state, the production of ARJ21 is heavily relying on manual processes and inevitably limited to 1-2 aircrafts per year. To meet the delivery target (i.e., 30 aircrafts per year) while maintaining quality and cost effectiveness, the manufacturing and assembly processes of the ARJ21 have to be less of manual work, but more automated by adopting the concept of (semi-) automatic assembly process.

Similar to other integrated aircraft manufacturing networks like B777, B787, A340, and A380, the main structure components of the ARJ21 are manufactured and assembled across China by three other ARJ21 consortium members (Tier-1) located in Xi'an, Chengdu, and Shenyang. The parts are then transported and finally assembled by COMAC itself in Shanghai. This also means any delay from the Tier-1 airframe component suppliers or in the final assembly will respectively cause holdup in the production rate or accumulation of components from the suppliers. Hence, in order to maximize the overall production rate, it is important that disruptions in each assembly line at different sites will have to be minimized.

In this paper, the subfinal assembly of ARJ21 fuselage joint is used as a case study to illustrate the impact of maintenance on the assembly process performance. This can be subsequently extended to cover the whole final assembly process or applied to the other Tier-1 component-level assemblies. At the ARJ21 final assembly line, each ARJ21 arrives in seven substructures: nose section, front fuselage, central fuselage, aft fuselage, rear fuselage (including tails), and both wings. The components are uploaded to transporters and taken to three specific assembly stations, where in parallel the forward fuselage is constructed of the nose section and front fuselage, the wings are joined to the central fuselage, and the aft and rear fuselages are joined which form the aft fuselage, see Figure 1. The three main fuselage substructures are then transported to the final assembly station where they are joined together into a complete airframe. In this paper, we will focus on the assembly processes (i.e., Stages 200A and 200B) carried out this station.

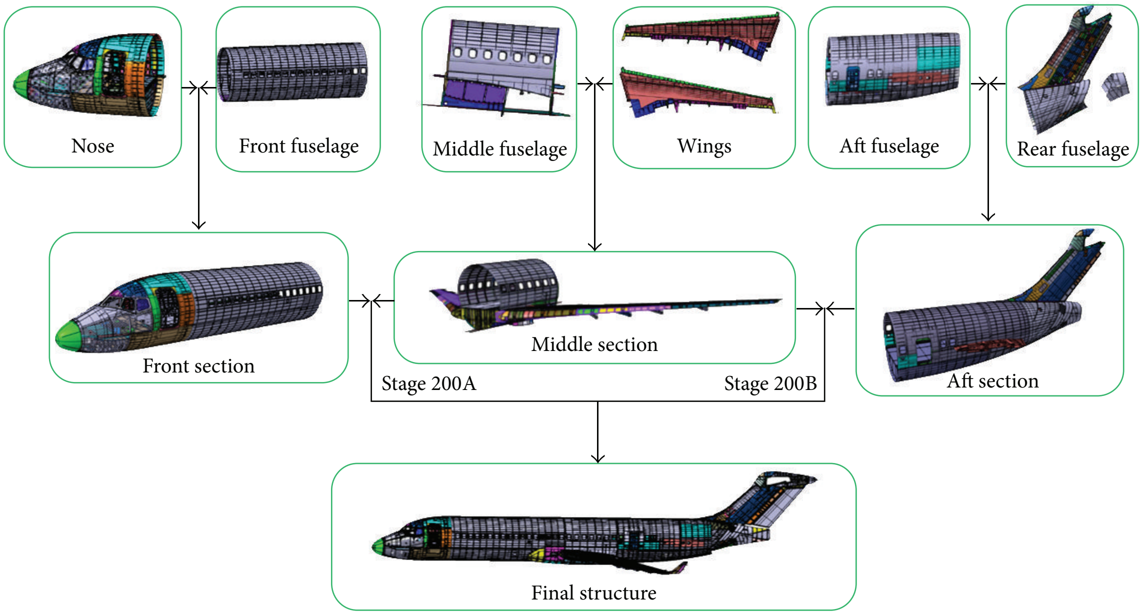


Figure 1: COMAC ARJ21 structure assembly.

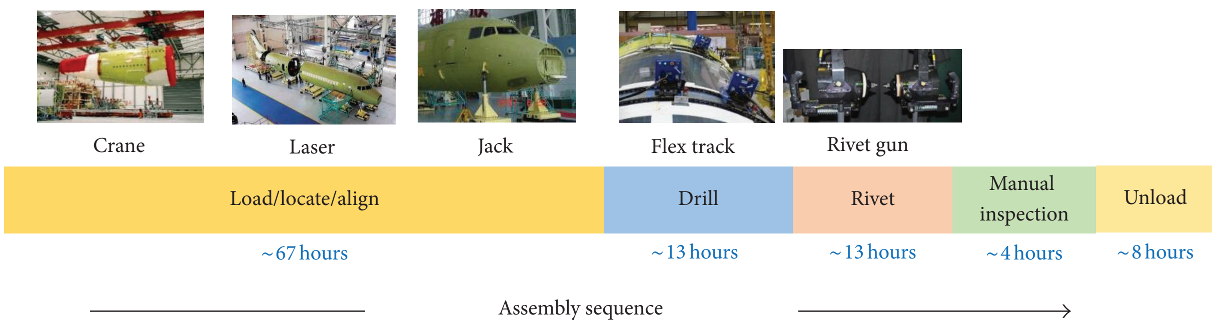


Figure 2: ARJ21's main sequence of final joints and related assembly machines.

2.2. Fuselage Joining Process. The main sequence of the final joints (Stages 200A and 200B) can be divided into 5 steps as shown Figure 2. The joint work starts by using an overhead crane to move and prealign the fuselage components into an assembly jig. These components are then aligned with high accuracy by adjusting the supported jacks guided by measurement data from the laser tracking devices. It takes approximately 67 hours to complete this sequence. The next assembly step is to drill the joint structures, that is, front and middle sections and middle and aft sections. In this process, the skins, frames, stringers, and other joint parts of the structure pair are drilled together and then deburred. In manual drilling, this process relies on precision measurements and drilling skills which is time consuming and often imperfect (e.g., oblique holes and excessive countersinks). In an (semi-) automated assembly line, a light weight portable computer numerical control (CNC) drilling machine (a.k.a Flex Track [30]) is used to reduce the lead time while maintaining the required drilling quality. This process is estimated to be 13 hours assuming that 2 Flex Tracks are used. In the third step, the joint fuselage sections are fastened using hammered solid rivets. In this process, "Handheld Electromagnetic Rivet Guns" are used in place of manual riveting to reduce lead time and noise hazard level. For 2 pairs of rivet guns, the lead time of this process is estimated to be 13 hours. The fourth step is to manually inspect the riveting quality such as position, depth, and angle. This process takes approximately 4 hours. Finally, the jig and jacks are removed from the completed fuselage. The fuselage is then towed away from the assembly area. This final unloading process takes about 8 hours.

Table 1 details subprocesses of the main assembly sequence and their estimated lead time. These values were obtained from interviews of COMAC engineer. Note that these estimates do not take into account any of process disruptions which could be caused by part delays, machine breakdowns and maintenance, or other factors.

2.3. Bottlenecks. During the design and development phases, manual work is relevant and sufficient as the assembly process

Table 1: Estimated work process lead time.

Work contents	Assembly machines	Lead time (Hrs)
Load/locate/align		
Load middle section	Crane	9
Prelocate and -align Jig and middle section	Jacks and lasers	19
Load front section	Crane	9
Load aft section	Crane	9
Locate and align Jig and fuselage sections	Jacks and lasers	21
Drill (in parallel)		
Front and middle sections		
Load flex tracks	Flex tracks	2
Drill	Flex tracks	10
Unload flex tracks	Flex tracks	1
Front and middle sections		
Load flex tracks	Flex tracks	2
Drill	Flex tracks	10
Unload flex tracks	Flex tracks	1
Rivet (in parallel)		
Front and middle sections		
Load rivet guns	Rivet guns	2
Rivet	Rivet guns	10
Unload rivet guns	Rivet guns	1
Middle and aft sections		
Load rivet guns	Rivet guns	2
Rivet	Rivet guns	10
Unload rivet guns	Rivet guns	1
Manual inspection		4
Unload		8

is not finalized and the required production rate is limited to 1 or 2 aircrafts per year. To meet the delivery target of 30 aircrafts per year, the assembly process of the ARJ21 has to be of less manual work but to be more automated. However, disruption caused by machine breakdowns and maintenance is one of the key performance bottlenecks in a (semi-) automatic assembly line [2]. From interviews of COMAC engineers, an overhead crane, Flex Tracks, and rivet guns are less reliable in relation to other types of assembly equipment and will likely be the common causes of machine breakdown.

From Table 1, it can be seen that three of the main processes of the ARJ21 fuselage joining process are heavily dependent on the overhead crane, Flex Tracks, and rivet guns. Hence, the downtime of these machines will essentially affect the throughput of the assembly process. Table 2 summarizes estimated maintenance parameters of these assembly machines. In this paper, the crane, Flex Tracks, and rivet guns are the focus of an application of CBM.

Table 2: Estimated maintenance parameters.

Assembly machines	MTBM[1] (hrs)	Maintenance time (hrs)	Loss time[2] (hrs)
Overhead crane	720	1	169
Flex track	900	2	506
Rivet gun	1440	1	336

[1]MTBM: mean time before maintenance; [2]Loss time: minimum downtime caused by unexpected breakdown.

3. Condition Based Maintenance

3.1. Degradation Process. Machine failures can be divided into two categories, random failures and those as a consequence of degradation. In this paper, we only consider the degradation failures in which preventive maintenance strategies can be applied. A simplified degradation process is illustrated in Figure 3. R_{PM}, R_F, and $T_M^{(i)}$ are the preventive maintenance threshold, failure threshold, and required duration to perform the ith maintenance (or repair), respectively. The degradation process can be represented by a stochastic process of increasing wear, and hence decreasing in system reliability, finally leading to machine failure. The degradation stages can be modelled using either discrete steps or continuous process in time. The failure occurs when the machine degradation stage reaches a certain reliability level. In Figure 3, maintenances are used to intervene with the degradation process and bring about an improvement to a certain reliability level before failures occur. However,

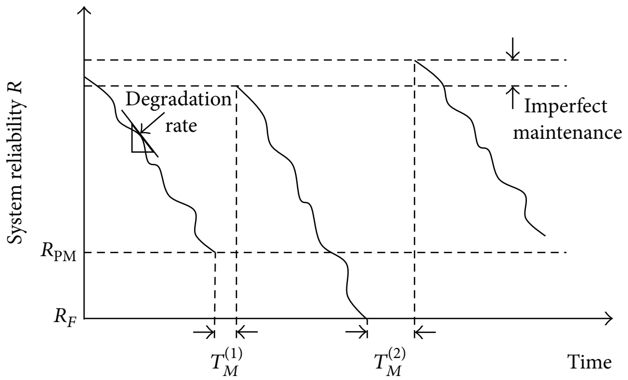


Figure 3: Degradation process.

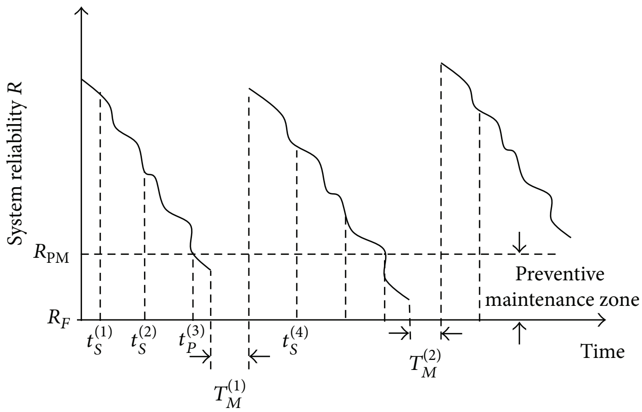


Figure 4: Condition based maintenance (CBM) framework.

when there is no ambiguity, the term "maintenance" will also include "repair" operations in this paper. The randomness (being stochastic) is from uncertainties in the degradation rate and maintenance. The latter results from imperfect maintenance.

3.2. Maintenance Model. The purpose of maintenance is to increase the mean time to failure. It is assumed that maintenance will bring about an improvement to the conditions in the previous stage of degradation [31], see Figures 3 and 4. In CBM framework, a maintenance policy relies on continuous condition monitoring which can be carried out by embedded sensors or periodic inspection (i.e., $t_S^{(i)}$ in Figure 4) [11, 14]. This paper assumes that embedded sensors are used to support online continuous condition monitoring (CM). In CBM, instead of traditional fix schedule, maintenance interventions are performed only when the system reliability degrades below a certain preventive maintenance threshold. In this way, unnecessary maintenance actions and unexpected breakdowns can be reduced; a real-time CM system provides an estimate for the reliability level due to degradation. In CBM, when to take maintenance actions, that is, defining the R_{PM} level, is essentially the main design maintenance parameter. This parameter will be based on both the system reliability level at inspection time and the potential evolution of the system's degradation process. R_{PM} must be sufficiently high to allow maintenance actions to be performed before the machine degrades to the failure level R_F.

4. Agent Based Model

DES is widely used in modelling and simulation of manufacturing systems where fixed sequence of operations can be naturally captured. However, in a highly complex process, it is simpler to model using ABS; complex active interactions between entities (e.g., machine, maintenance, and process) can be naturally captured in ABS without having to reformulate a problem into a series of discrete events as required by DES [21, 22]. In our case, CM system, whose self-aware (or active) properties are required in order to trigger maintenance activities, is an example of entities that cannot straightforwardly be modeled using DES; the behaviour of a CM system has to be determined by the system in a DES model (unintuitively being passive). For this reason, ABS is preferred to DES in this paper.

In this paper, for simplicity in the analysis, a CBM enabled ARJ21 assembled system is composed of minimally assembly machines, maintenance and the assembly process itself, and the interdependencies among these components. In ABS, these entities are called agents. Here, we use statechart to model behaviours of an agent and a messaging concept to model interactions between agents [32]. The statechart allows active stochastic behaviours of an agent to be modelled [33]. In ABS, agents independently evolve in parallel using the same universal time tick generated by the simulation environment.

Figure 5 depicts a generic agent based model of the assembly machines described in Section 2, that is, crane, flex tracks and rivet guns. Δt, RUL, ΔR, $\mathcal{X}(\cdot,\cdot)$, μ_{RUL}, σ_{RUL}, $\mu_{\Delta R}$, and $\sigma_{\Delta R}$ are the simulation time step, remaining useful life, degradation rate, random variable, mean remaining useful life after maintenance, uncertainty caused by imperfect maintenance, mean degradation rate threshold, and degradation uncertainty, respectively. A machine can be in either "In Operation" or "Out of Order" states. When in operation, the machine is "Idle" if it is not needed by an assembly process. The changes in state to "Busy" and back to "Idle" are triggered by the messages "In Use" and "Work Completed" sent from the process, respectively. When the machine is in use, the system reliability ~RUL is decreasing at the rate of ΔR. Note that RUL is a function of operating hours not time.

When the RUL falls below the preventive maintenance threshold R_{PM}, if have not yet sent one, a maintenance order, that is, the "Need Maintenance" message, is sent to the related maintenance service. The machine goes into "Out of Order" triggered by either the breakdown (i.e., RUL $\leq$ 0) or "In Maintenance" messages from the related service and is changed back to be operable (either "Idle" or "Busy" depending on which one is the last state) after being maintained or repaired triggered by the message "Maintained". After maintenance, the system reliability is brought back to a certain level randomly defined by the imperfect maintenance

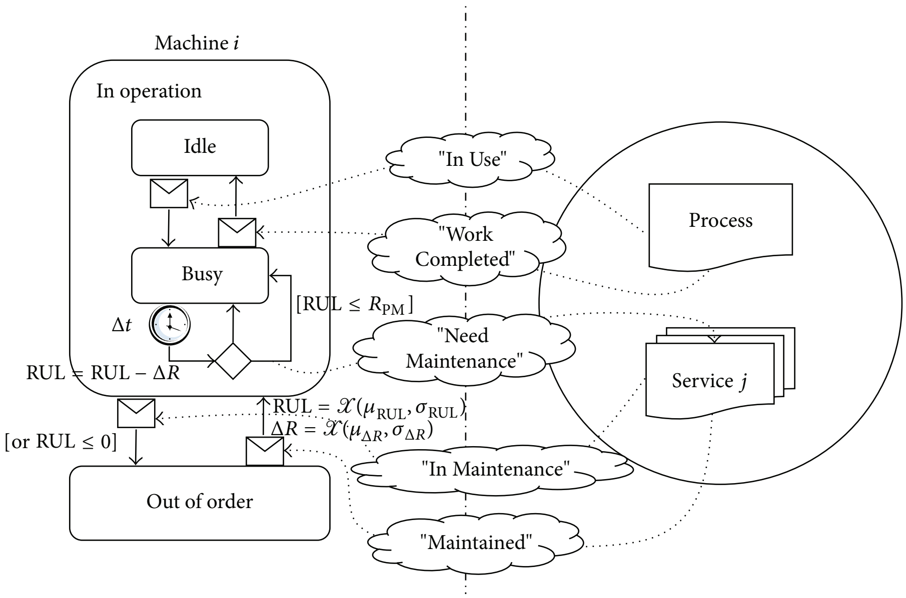


Figure 5: Agent based model of machines.

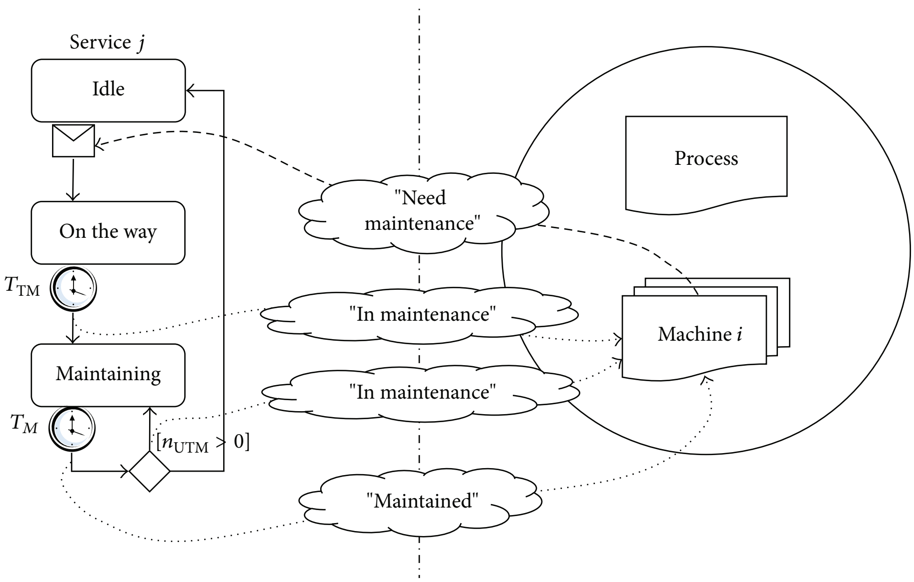


Figure 6: Agent based model of services.

parameters μ_{RUL} and σ_{RUL}. Depending on the density function, $\sigma_{...}$ can be either a bound, standard deviation, or no value depending on the probability distribution of the random variable $\mathscr{X}$ for normal. In this model, the stochasticity in the degradation process is simulated by randomly resetting ΔR according to the degradation parameters $\mu_{\Delta R}$ and $\sigma_{\Delta R}$.

An agent based model for maintenance services is shown in Figure 6. T_{TM}, T_M, and n_{UTM} are the time to maintenance caused by delay or travel, maintenance time, and number of machines that require maintenance, respectively. A transition from "Idle" to "On the Way" is triggered by the message "Need Maintenance" sent from one of the related machines. On average, the service needs T_{TM} to respond to the maintenance order. Before the service goes into the "Maintaining" state, a message "In Maintenance" is sent to the machine at the first in the requested list for maintenance. It takes on average T_{TM} for a maintenance action. When the timer expires, the message "Maintenance Complete" is sent to the maintained machine which causes a state transition from "Out of Order" to "In Operation". It is assumed that every triggered maintenance action is completed successfully. The service's new state now depends on the number of machines that required maintenance. If another machine is to be maintained (i.e., $n_{UTM} > 0$), the service chooses the first machine in the requested list and as before maintains the machine. The service's state goes into "Idle" if there are no more machines in the requested list. Whenever the "Need Maintenance" message is received, the requested machine will be added to

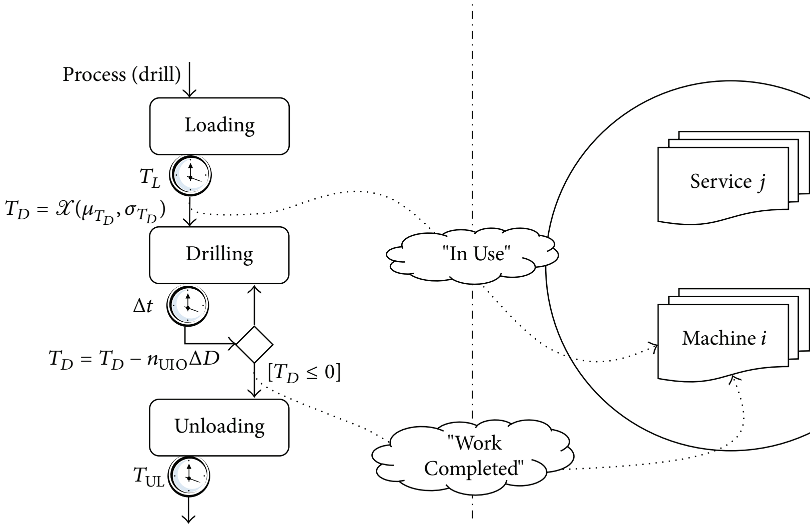


Figure 7: Partial agent based model of assembly process.

the requested list for maintenance. This allows the service to respond to the maintenance request even though it is not in the "Idle" state.

Figure 7 depicts an agent based model of the assembly process. For the sake of explanation, it only shows the drill subassembly process. T_L, T_D, μ_{T_D}, σ_{T_D}, ΔD, T_{UL}, and n_{UIO} are the loading time, required drill time, mean required drill time, drilling process uncertainty, drill rate, and number of machines currently in operation, respectively. After the load/locate/align process is completed, the state is transitioned to the "Drill" composite state in which "Loading" is the entry state. On average, it needs T_L to position the two Flex Tracks at the font/middle and middle/aft section joints. When the timer expires, the message "In Use" is sent to the Flex Track machines which causes a state transition from "Idle" to "Busy", and the required drill time is set to a random value determined by the process uncertainty parameters μ_{T_D} and σ_{T_D}. In this paper, it is assumed that maintenance actions do not interrupt the "Loading" and "Unloading" processes. When the machine is in use, the required (or remaining) drill time T_D is decreasing at the rate of $n_{\mathrm{UIO}}\Delta D$. How fast the task can be completed will depend on how many machines are available (i.e., in operation) to perform the tasks. When the drill process is completed (i.e., $T_D \leq 0$), the message "Work Complete" is sent to the related machines, and the process's new state is now "Unloading". It takes on average T_{UL} to remove the machines (i.e., Flex Track in the drill case) from the fuselage. When the timer expires (also means the drill subassembly process completed), the state is then transitioned to the "Rivet" composite state.

5. Optimization Considering Multiple Objectives

5.1. Simulation Optimization. The agent based simulation model described in Section 4 is used to empirically study the impact of CBM on the (semi-) automatic fuselage assembly process. Through simulation, the aim is to find the best CBM configuration that will maximize the production rate and at the same time minimize the maintenance cost of the assembly process. A simulation optimization scheme is shown in Figure 8. $\overrightarrow{x}$, Δ, $f(\cdot)$, $g(\cdot)$, and $E[\cdot]$ are the design variable vector, uncertainties, objective function, constraint function, and mean (or expected) value, respectively. In our case, the simulation model (i.e., the agent based simulation model of the fuselage assembly system) takes the controllable maintenance design variables as the input and outputs the simulated time series data which are then used to evaluate the performance objectives of the assembly process. In contrast to the deterministic simulation, two successive simulations of the same input variables return two different simulation results due to the uncertainties in the machine degradation, maintenance, and process lead time. In the simulation optimization, the optimization objective is therefore the mean of the objective evaluations $E[f(\overrightarrow{x}, \Delta)]$ from multiple simulation runs (called Monte-Carlo simulation) of the model [35–37]. Here, "Optimal" means on average that this is the best design solution.

5.2. Scatter Search. In this paper, AnyLogic Multimethod Simulation Software is used to implement the agent based fuselage assembly model described in Section 4. OptQuest scatter search package is the only built-in optimization engine/method in AnyLogic [38]. Scatter search is a population-based metaheuristics for optimization and has been successfully applied to many hard optimization problems [28, 39]. In contrast to Evolutionary Algorithms, scatter search does not emphasize randomization. It is instead designed to incorporate strategic responses, both deterministic and probabilistic, that remember which solutions worked well (i.e., both high quality solutions and diverse solutions) and recombined them into new, better solutions [40, 41].

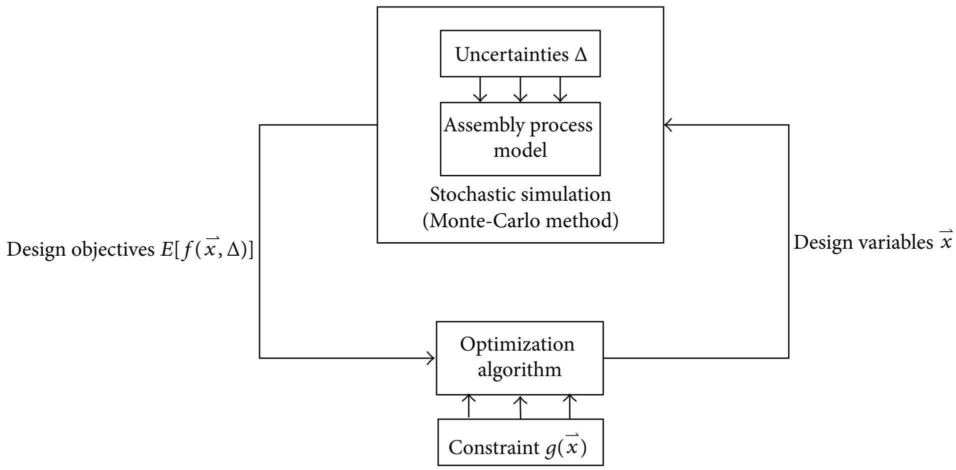


Figure 8: Simulation optimization.

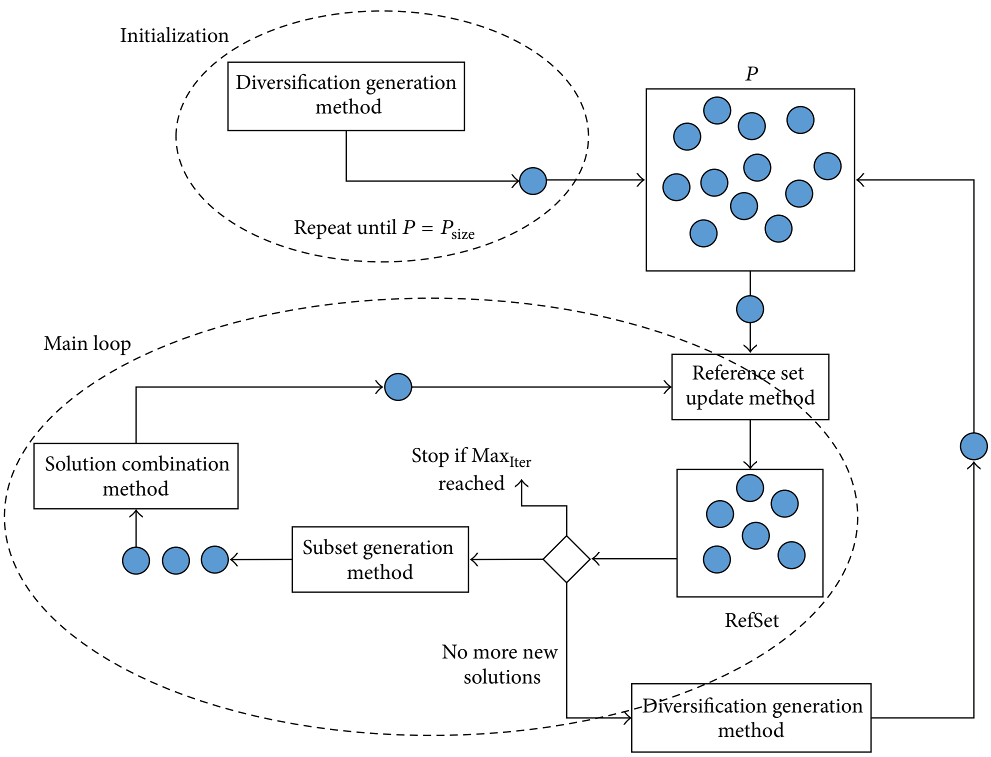


Figure 9: Scatter search optimization method [34].

Figure 9 outlines a simplified scatter search optimization method. The search consists of four methods: (1) diversification generation, (2) reference set update, (3) subset generation and (4) solution combination. The search starts with the diversification generation method which is used to generate a large set P of diverse solutions (i.e., scattered across the parameter space) that are the basis for initializing the search. The process repeats until $|P| = P_{\text{Size}}$. The initial RefSet is built according to the reference set update method, which can take the b best solutions (as regards their quality or diversity in the problem solving) from P to compose the RefSet. The search is then initiated by applying the subset generation method which produces subsets of reference solutions as the input to the combination method. The solution combination method uses these subsets to create new combined solution vectors. The reference set update method is applied once more to build the new RefSet and the main loop repeats again. The iteration stops when the maximum iteration is reached. However, at the end of each iteration, if no more better solutions are found, the new

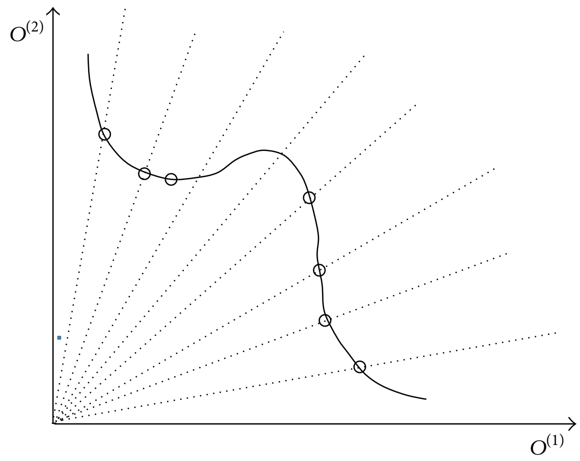


Figure 10: Pareto optimal solutions corresponding to a set of target vectors.

RefSet is partially rebuilt by refilling the RefSet with a new diverse solution generated using the diversification method [26, 34, 38, 42].

5.3. Multiple Single Objective Pareto Sampling. In addition to the production rate, the cost of maintenance is another important objective in the integrated maintenance (semi-) automatic aircraft assembly process. In deterministic multi-objective optimization, nondominated ranking methods are generally used in finding the Pareto optimal solutions [43–45]. These methods rely on a large number of populations to find all Pareto solutions and consequently are practically not feasible for simulation optimization primarily due to the add-on computational burden required by the Monte-Carlo simulation. Moreover, the optimization experiment in AnyLogic in any case is limited to single objective optimization and possibly because of the computational reason. This limitation in fact applies to most of the commercial of-the-shelf (COTS) DES or ABS software.

In this paper, a multiple single objective Pareto sampling method described in [46, 47] is used to find the Pareto optimal solutions. It is based on the classical weighed metric method. The method locates some specific solutions on the Pareto front corresponding to a given set of target vectors $V = \{\overrightarrow{v}_1, \ldots, \overrightarrow{v}_T\}$, where $\overrightarrow{v}_i = (1/w_i^{(1)}, \ldots, 1/w_i^{(\mathrm{NOBJ})})$ is the direction corresponding to the given ith weight vector, and $w_i^{(k)}$ is the weight of the kth objective, see Figure 10. For each rerun $\overrightarrow{v}_i$, the corresponding optimal solution is found by minimizing the objective function [48–50]

$$L_i\left(\overrightarrow{x}\right) = E\left[f_i\left(\overrightarrow{x}, \Delta\right)\right], \tag{1}$$

where

$$f_i\left(\overrightarrow{x}, \Delta\right) = \max\left\{w_i^{(1)}O_i^{(1)}, \ldots, w_i^{(\mathrm{NOBJ})}O_i^{(\mathrm{NOBJ})}\right\}. \tag{2}$$

The uncertainties Δ represent a Monte-Carlo simulation replication. In this way, the multiobjective optimization can be carried out using the AnyLogic built-in optimization engine and as well computationally is more plausible.

The primary advantage of this method is that the target vectors can be arbitrarily generated focusing on the regions of interest. Note also that if the optimization process converges to a solution that exactly matches the weight vector, then $w^{(1)}O^{(1)} = \cdots = w^{(\mathrm{NOBJ})}O^{(\mathrm{NOBJ})}$. Thus, the angle between the vectors $\overrightarrow{v}$ and $\overrightarrow{O}$ indicates whether the solution lies in where it was expected or not. If the vector $\overrightarrow{v}$ lies within a discontinuity of the Pareto set or is outside of the entire objective space, then the angle between the two vectors will be significant. By observing the distribution of the final angular errors across the total weight set, the limits of the objective space and discontinuities within the Pareto set can be identified.

6. Results

6.1. Measures of Performance. The agent based simulation model described in Section 4 is by design developed to empirically study the impact of CBM on the (semi-) automatic fuselage assembly process and find a best CBM system configuration for maximizing production throughput while keeping related costs minimum. From Table 1, it can be seen that most parts of the fuselage joint sequence are dependent on the operational availability of the assembly machines, that is, crane, Flex Tracks, and rivet guns. The downtime of these machines will essentially incur loss in the production. For consistency, "Optimize" means "Minimize" in this paper.

For the first performance measure, in order to maximize the production rate, we seek to minimize the production loss objective defined by

$$O^{(1)} = \frac{N_A - n_A}{\epsilon N_A}. \tag{3}$$

n_A, N_A, and ϵ are the number of aircrafts produced per year, maximum number of aircrafts per year without part delays, machine breakdowns and maintenance or other factors, and in percentage maximum allowable production loss from breakdown and maintenance, respectively. The product ϵN_A is the normalization factor of the incurred production loss.

Besides the production loss, at what cost in keeping the machines operable is another performance measure in the integrated CBM aircraft assembly system. In this paper, the maintenance cost consists of the fixed maintenance cost and service level related maintenance cost, and its corresponding performance objective is defined by

$$O^{(2)} = \frac{C_M + C_{\mathrm{MRT}}\left(T_R\right)}{C_M + C_{\mathrm{MRT}}\left(T_{\mathrm{NR}}\right)}. \tag{4}$$

C_M, T_R, and T_{NR} and $C_{\mathrm{MRT}}(\cdot)$ are the fixed maintenance cost, maintenance response time, nominal maintenance response time, and cost function related to required maintenance response time defined by $\rho T_R/T_{\mathrm{NR}}$, respectively. The response time parameters T_{NR} and T_R can be considered as the level of service required for machine maintenance. In simulation, these parameters define T_{TM} the time to maintenance caused by delay or travel shown in Figure 6. For this objective, the nominal maintenance cost $C_M + C_{\mathrm{MRT}}(T_{\mathrm{NR}})$ is the normalization factor.

In this paper, CBM is applied to multiple assembly machines, which will individually be contributing to the overall maintenance cost. The fixed and service level maintenance costs can simply be defined as linearly formulated as

$$C_M = \sum_{i=1}^{N} C_M^{(i)},$$
$$C_{\mathrm{MRT}} = \sum_{i=1}^{N} C_{\mathrm{MRT}}^{(i)} \left(T_R^{(i)}\right). \quad (5)$$

N and ith are the maximum number of assembly machines (i.e., 5 = 1 crane + 2 Flex Tracks + 2 pairs of rivet guns) and the machines' index, respectively. ith indexes the machine specific related costs and required response times.

In CBM, a maintenance order is triggered when the system reliability falls below a predefined preventive maintenance threshold R_{PM}, see Section 3. Moreover, together with R_{PM}, when the maintenance is actually performed after being triggered, T_R will essentially determine the operation availability of the machine, which indirectly determines the production rate of the aircraft assembly process. The cost of maintenance is determined by how frequent maintenance is performed and maintenance service level. These two cost factors are as well determined by the parameters R_{PM} and T_R. In this study, there are 3 types of assembly machines. Hence, R_{PM}^{C}, T_R^{C}, $R_{\mathrm{PM}}^{\mathrm{FT}}$, T_R^{FT}, $R_{\mathrm{PM}}^{\mathrm{RG}}$, and T_R^{RG} will be in total the 6 design variables for the optimization of the CBM enabled ARJ21 assembly process. R_{PM}^{C} and T_R^{C} are for the crane. $R_{\mathrm{PM}}^{\mathrm{FT}}$ and T_R^{FT} are for the two Flex Tracks. $R_{\mathrm{PM}}^{\mathrm{RG}}$ and T_R^{RG} are for the two pairs of rivet guns. In our ABS, these design variables are corresponding to the machine i's R_{PM} and service j's T_{TM}.

6.2. Trade-Off Analysis. In the paper, AnyLogic and OptQuest scatter search are employed for the simulation optimization experiments of the agent based CBM enabled ARJ21 assembly system described in Section 4. The target vectors were distributedly formed to outline the Pareto front using the weights $w^{(1)} \in [0.5, 0.99]$ and $w^{(2)} \in [0.01, 0.5]$. In this paper, the uncertainties in lead time, maintenance, and machine degradation are modelled using the triangular distribution. The lower and upper uncertainty bounds are assumed at ±25% of the mean values. The design variables are in the intervals $R_{\mathrm{PM}}^{(\cdot)}, T_R^{(\cdot)} \in [0, \mathrm{MTBM}^{(\cdot)}]$, where the subscript $(\cdot)$ is the machine specific type indices C, FT, or RG. Table 3 summarizes the other related simulation parameters required for the evaluation of the performance objectives $O^{(1)}$ and $O^{(2)}$ defined in (3) and (4). It is not possible to disclose related costs in term of money figures due to commercial reasons. In this paper, the ratios and normalization values are alternatively used to quantify the parameters related to the maintenance cost.

Figure 11 shows the Pareto optimal solutions obtained by multiple simulation optimization runs for different weightings (or target vectors). The simulation optimization experiments were based on 20 and 100 Monte-Carlo replications, see also Sections 5.1 and 5.3. Note that the optimization process can be rerun to ensure the consistencies of the Pareto solution (for both 20 and 100 Monte-Carlo replications). The uncertainties in the simulation can be observed from the fluctuations on the Pareto surfaces which is more apparent in the case of 20 Monte-Carlo replications. The resulting Pareto optimal solutions are more likely to represent a true Pareto surface if the optimization objectives are evaluated from a large number of Monte-Carlo simulation runs, and this can be seen in the case of 100 replications shown in Figure 11.

The Pareto surface shows the trade-off between the production loss and maintenance cost conflicting objectives. A reduction in product loss will generally increase in maintenance cost. This is in particular when the production loss $O^{(1)}$ is approximately less than 0.38 (i.e., 1.585 per year); a small decrease in production loss will result in a very significant increment in the maintenance cost. As a mean to increase the machines' operational availability with less number of maintenance performed, the response time T_R will have to be relatively small so that the machines are maintained before

Table 3: Parameters for evaluation of performance objectives.

Simulation parameters	Values
Maximum number of aircraft per year N_A	83.429
Allowable production loss ϵ	0.05
Crane	
Fixed maintenance cost C_M^{C}	1
Ratio for service level maintenance cost ρ^{C}	1
Nominal response time T_{NR}^{C}	168 hrs
Flex track	
Fixed maintenance cost C_M^{FT}	2
Ratio for service level maintenance cost ρ^{FT}	1.5
Nominal response time $T_{\mathrm{NR}}^{\mathrm{FT}}$	504 hrs
Rivet gun	
Fixed maintenance cost C_M^{RG}	0.4
Ratio for service level maintenance cost ρ^{RG}	0.6
Nominal response time $T_{\mathrm{NR}}^{\mathrm{RG}}$	336 hrs

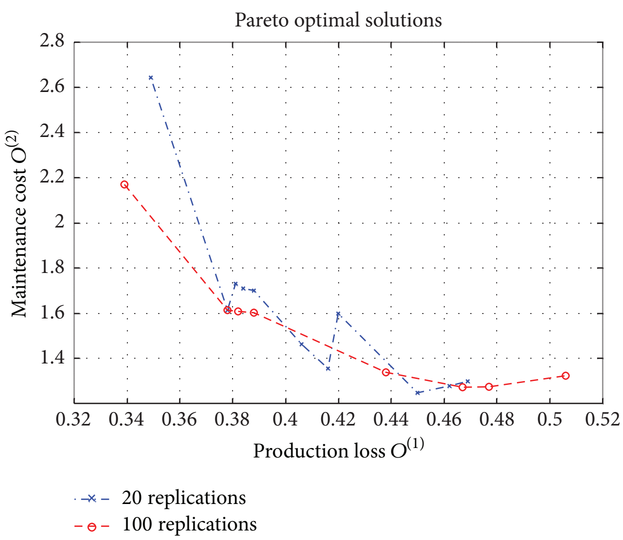


Figure 11: Trade-off surface of optimal CBM configurations.

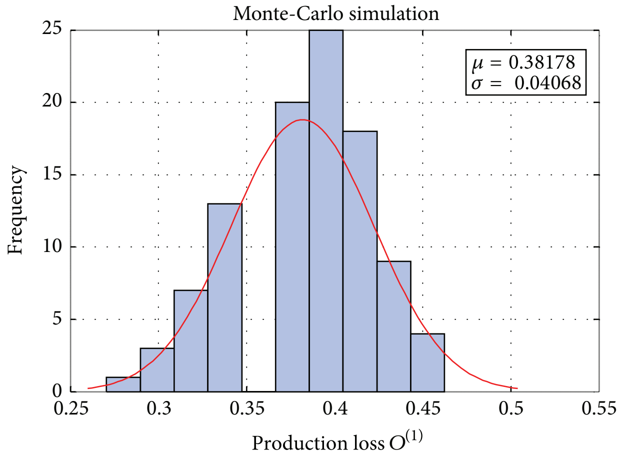


Figure 12: Monte-Carlo simulation results for production loss $O^{(1)}$.

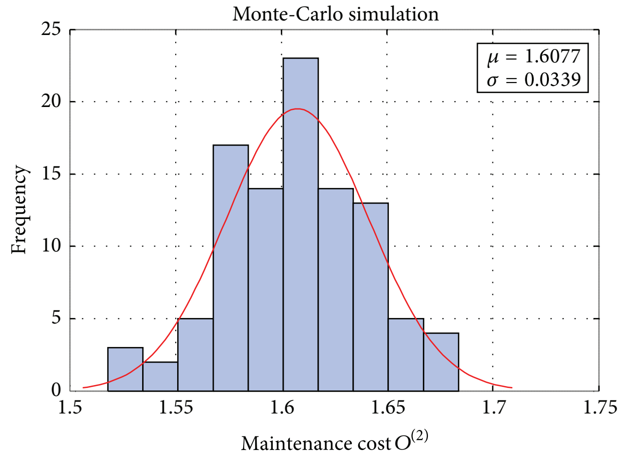


Figure 13: Monte-Carlo simulation results for maintenance cost $O^{(2)}$.

breakdowns happen. This will essentially increase the service level related maintenance cost C_{MRT}. This is an important insight as for $O^{(1)} < 0.38$ a decision maker can avoid a situation where a significant investment on the maintenance is made for a very small gain in the production rate and possibly in overall is not worth the investment.

The trade-off surface also outlines the optimal boundary for the maintenance cost. It can be seen that the maintenance cost $O^{(2)}$ approximately cannot be lower than 1.3 even when we deliberately set multiple target vectors to probe in the >0.5 production loss objective space. This result is from the fact that the assembly machines are essentially degrading no matter whether CBM is applied or not. From the ABS model, maintenance will always be performed on the machines regardless before or after the breakdown. Consequently, this will incur the minimum fixed maintenance cost. In addition, there is also a minimum cost for the required maintenance service level. These explain why the maintenance cost cannot be lower than a certain level, and effectively determine the boundary of the objective space.

6.3. Monte-Carlo Experiment. Suppose the optimal solution (0.382, 1.608) in Figure 11 corresponding to the weights $w^{(1)} = 0.85$ and $w^{(2)} = 0.15$ is preferred. The resulting design variables for this optimal solution are $R_{\mathrm{PM}}^{C} = 390.54$, $T_{R}^{C} = 139.8$, $R_{\mathrm{PM}}^{\mathrm{FT}} = 452.7$, $T_{R}^{\mathrm{FT}} = 254.76$, $R_{\mathrm{PM}}^{\mathrm{RG}} = 458.46$, and $T_{R}^{\mathrm{RG}} = 337.02$. In paper, a Monte-Carlo simulation experiment is used to evaluate whether the obtained optimal solution is the performance expected from the system and how sensitive the design solution is to the uncertainties.

Figures 12 and 13 show the results of 100 runs of Monte-Carlo simulation. The mean production loss and maintenance cost are 0.3817 and 1.6077, respectively, and these follow what initially expected from the simulation optimization result. From the histograms, 100 Monte-Carlo simulation runs (or samples) is sufficient to represent the true probability distribution of the performance objective values. With ±25% of the mean values for the lower and upper uncertainties, the standard deviations of the production loss and maintenance cost are 0.04068 and 0.0339, respectively, 11% for the production loss and 2% for the maintenance cost in relation to the mean values. This indicates very low sensitivities for this design solution. In terms of number of aircrafts produced, this means in ~70% of the cases, the production loss will be between ±0.17 aircrafts per year from the expected value.

7. Concluding Remarks

In a current competitive environment, increasing production rate and reducing costs are the key drivers in aircraft manufacturing. More (semi-) automatic assembly machines have increasingly been used in the aircraft assembly lines as a mean to deliver high production rate while meeting high quality requirements. However, the production throughput is effectively dependent on the operational availability of these machines. Integration of CBM into the assembly system has potential benefits as a way to minimize the production loss and maintenance related cost. Maintenance are performed as needed, hence avoiding unnecessary downtime and maintenance cost.

In a CBM enabled aircraft assembly system, there are self-active interactions between the subsystems, for example, CM system self-triggers a maintenance order when the system reliability falls below a certain level. This example of active self-aware behavior cannot straightforwardly be modelled using DESs. In this case, where, besides the assembly process, independent entities are in addition parts of the system, ABS is proved effective as it allows complex active interactions between entities to be naturally captured.

Production rate and maintenance cost are the competing objectives in an integrated CBM aircraft assembly system. Finding trade-offs between the production rate and maintenance cost is equivalent to finding a Pareto optimal surface. The conventional nondominated ranking methods will not be practically feasible due to the computational burden required by the Monte-Carlo simulation. This limitation can be addressed by independently sampling the Pareto surface using the weighted Min-Max method. The approach allows less number of populations to be used in the optimization as it does not need to probe the whole Pareto front, and hence effectively a reduction in the computation intensity is required.

In our ARJ21 case study, the preventive maintenance threshold and required service level are the key design parameters that determine the overall performance of the assembly system. Because of uncertainties, increase production rate will require a high required service level (i.e., fast response time) to avoid breakdowns before the maintenance is performed, and consequently this will increase the maintenance cost and sometimes can be significant. However, compromising on the production rate does not always mean a further decrease in maintenance cost. The minimum cost is from the actual cost in maintaining the machines and the minimum service level. Pareto surface is an important piece of information to the system designer. Together with Monte-Carlo simulation, it can be used to support decision making in terms of cost-benefit of different design solutions and also what could be achieved.

In this example, even though in small scale, it can be seen that CBM has potential to be applicable in (semi-) automatic aircraft assembly lines. However, its claim has to be further researched in comparison with other maintenance regimes and with a high fidelity large-scale aircraft assembly example. Moreover, in terms of optimization, other different optimization methods like genetic algorithms (GAs), simulated annealing, and teaching-learning-based optimization (TLBO) should also be used in the optimization to ensure that the true Pareto font is found and consequently their performance in terms of computation and solution can be compared and analyzed.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is part of a research program at Cranfield University sponsored by Commercial Aircraft Cooperation of China (COMAC) and China Scholarship Council (CSC).

References

[1] MTorres, *Total Solution for Aircraft Automatic Assembly Jigs*, MTorres, Santa Ana, Calif, USA, 2013.

[2] P. Lute, *An Investigation of Airbus A380 Stage 01 Wing Box Assembly Using Discrete Event Simulation*, Cranfield University, Bedfordshire, UK, 2007.

[3] Electroimpact Inc., *A380 Stage 1 GRAWDE Machine*, Electroimpact Inc., Mukilteo, Wash, USA, 2003.

[4] IBM Corporation, *Predictive Maintenance for Manufacturing*, 2011.

[5] Wikipedia, “Preventive maintenance,” Wikipedia, http://en.wikipedia.org/wiki/Preventive_maintenance.

[6] ResolveFM, “Preventive/corrective maintenance,” ResolveFM, http://www.resolve.com.au/.

[7] A. Kelly, *Maintenance Strategy*, Butterworth-Heinemann, Oxford, UK, 1997.

[8] NACE, “Maintenance strategies,” 2013, http://www.nace.org/, http://events.nace.org/library/corrosion/Inspection/Strategies.asp.

[9] I. Grigoryev, *AnyLogic 6 in Three Days: A Quick Course in Simulation Modeling*, Anylogic North America, 2012.

[10] J. B. Leger, E. Neunreuthe, B. Iung, and G. Morel, “Integration of the predictive maintenance in manufacturing system,” in *Advanced in Manufacturing*, pp. 133–144, Springer, London, UK, 1999.

[11] Z. Tian, D. Lin, and B. Wu, “Condition based maintenance optimization considering multiple objectives,” *Journal of Intelligent Manufacturing*, vol. 23, no. 2, pp. 333–340, 2009.

[12] J. Yulan, J. Zuhua, and H. Wenrui, “Multi-objective integrated optimization research on preventive maintenance planning and production scheduling for a single machine,” *International Journal of Advanced Manufacturing Technology*, vol. 39, no. 9-10, pp. 954–964, 2008.

[13] D. Achermann, *Modelling, Simulation and Optimization of Maintenance Strategies Under Consideration of Logistic Processes*, Südwestdeutscher, 2008.

[14] Š. Valčuha, A. Goti, J. Úradníček, and I. Navarro, “Multi-equipment condition based maintenance optimization by multi-objective genetic algorithm,” *Journal of Achievements in Materials and Manufacturing Engineering*, vol. 45, no. 2, pp. 188–193, 2011.

[15] L. Tautou and H. Pierreval, “Using evolutionary algorithms and simulation for the optimization of manufacturing systems,” *IIE Transactions*, vol. 29, no. 3, pp. 181–189, 1997.

[16] D. Baglee, “Maintenance strategy development in the UK food and drink industry,” *International Journal of Strategic Engineering Asset Management*, vol. 1, no. 3, pp. 289–300, 2013.

[17] J. Reimann, G. Kacprzynski, D. Cabral, and R. Marini, “Using condition based maintenance to improve the profitability of performance based logistic contracts,” in *Proceedings of the Annual Conference of the Prognostics and Health Management Society*, 2009.

[18] A. Grall, C. Bérenguer, and L. Dieulle, “A condition-based maintenance policy for stochastically deteriorating systems,” *Reliability Engineering and System Safety*, vol. 76, no. 2, pp. 167–180, 2002.

[19] M. Rolón and E. Martínez, “Agent-based modeling and simulation of an autonomic manufacturing execution system,” *Computers in Industry*, vol. 63, no. 1, pp. 53–78, 2012.

[20] W. Shen, Q. Hao, H. J. Yoon, and D. H. Norrie, “Applications of agent-based systems in intelligent manufacturing: an updated review,” *Advanced Engineering Informatics*, vol. 20, no. 4, pp. 415–431, 2006.

[21] M. A. Majid, U. Aickelin, and P. O. Siebers, “Comparing simulation output accuracy of discrete event and agent based models: a quantitave approach,” in *Proceedings of the Summer Computer Simulation Conference (SCSC ’09)*, Vista, Calif, USA, 2009.

[22] P. O. Siebers, C. M. MacAl, J. Garnett, D. Buxton, and M. Pidd, “Discrete-event simulation is dead, long live agent-based simulation!,” *Journal of Simulation*, vol. 4, no. 3, pp. 204–210, 2010.

[23] L. Holst, *Integrating Discrete-Event Simulation into the Manufacturing System Development Process*, Division of Robotics, Lund, Sweden, 2001.

[24] J. A. B. Montevechi, R. d. C. Miranda, and J. D. Friend, "Sensitivity analysis in discrete-event simulation using design of experiments," in *Discrete Event Simulations-Development and Applications*, InTech, Rijeka, Croatia, 2012.

[25] Y. Carson and A. Maria, "Simulation optimization: methods and applications," in *Proceedings of the Winter Simulation Conference*, pp. 118–126, Atlanta, Ga, USA, December 1997.

[26] R. Martí and M. Laguna, "Scatter search: basic design and advanced strategies," *Revista Iberoamericana de Inteligencia Artificial*, vol. 7, no. 19, pp. 123–130, 2003.

[27] A. Ghosh and S. Dehuri, "Evolution algorithms for multi-criterion optimization: a survey," *International Journey of Computing and Information Sciences*, vol. 2, no. 1, 2004.

[28] M. Laguna, R. Martí, M. Gallego, and A. Duarte, "The scatter search methodology," in *Wiley Encyclopedia of Operations Research and Management Science*, Wiley-Blackwell, Hoboken, NJ, USA, 2011.

[29] COMAC, "ARJ21 regional jet program," 2013, http://english.comac.cc/products/rj/pi2/index.shtml.

[30] Electroimpact, *Flex Track*, Electroimpact, Mukilteo, Wash, USA, 2013.

[31] IEEE/PES Task Force on Impact of Maintenance Strategy on Reliability of the Reliability, Risk and Probability Applications Subcommittee, S. Aboresheid, R. N. Allan et al., "The present status of maintenance strategies and the impact of maintenance on reliability," *IEEE Transactions on Power Systems*, vol. 16, no. 4, pp. 638–646, 2001.

[32] M. Kaegi, R. Mock, and W. Kröger, "Analyzing maintenance strategies by agent-based simulations: a feasibility study," *Reliability Engineering and System Safety*, vol. 94, no. 9, pp. 1416–1421, 2009.

[33] A. Borshchev, *Designing State-Based Behavior: Statecharts*, Anylogic, 2013.

[34] Process Engineering Group, *Introduction to SSm*, Instituto de Investigaciones Marinas (C.S.I.C.), Vigo, Spain, 2009.

[35] J. April, F. Glover, J. P. Kelly, and M. Laguna, "Practical introduction to simulation optimization," in *Proceedings of the Winter Simulation Conference*, pp. 71–78, Boulder, Colo, USA, December 2003.

[36] W. Abo-Hamad and A. Arisha, "Simulation optimisation methods in supply chain applications: a review," *Irish Journal of Management*, vol. 30, no. 2, pp. 95–124, 2011.

[37] C.-H. Chen and L. H. Lee, "Introduction to stochastic simulation optimization," in *Stochastic Simulation Optimization: An Optimal Computing Budget Allocation*, System Engineering and Operations Research, World Scientific, Hackensack, NJ, USA, 2010.

[38] M. Laguna, *OptQuest: Optimization of Complex Systems*, OptTek Systems, 2011.

[39] R. Martí, M. Laguna, and F. Glover, "Principles of scatter search," *European Journal of Operational Research*, vol. 169, no. 2, pp. 359–372, 2006.

[40] OptTek, "How the OptQuest engine works," OptTek, http://www.opttek.com.

[41] F. Glover and A. Reinholz, "Metaheuristics in science and industry: new developments," in *Proceedings of the Metaheuristics International Conference*, Montreal, Canada, June 2007.

[42] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Information Sciences*, vol. 237, pp. 82–117, 2013.

[43] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, Wiley-Blackwell, Chichester, UK, 2001.

[44] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York, NY, USA, 2000.

[45] A. Ghosh and S. Dehui, "Evolutionary algorithms for multi-criterion optimization: a survey," *International Journal of Computing and Information Sciences*, vol. 2, no. 1, pp. 38–57, 2004.

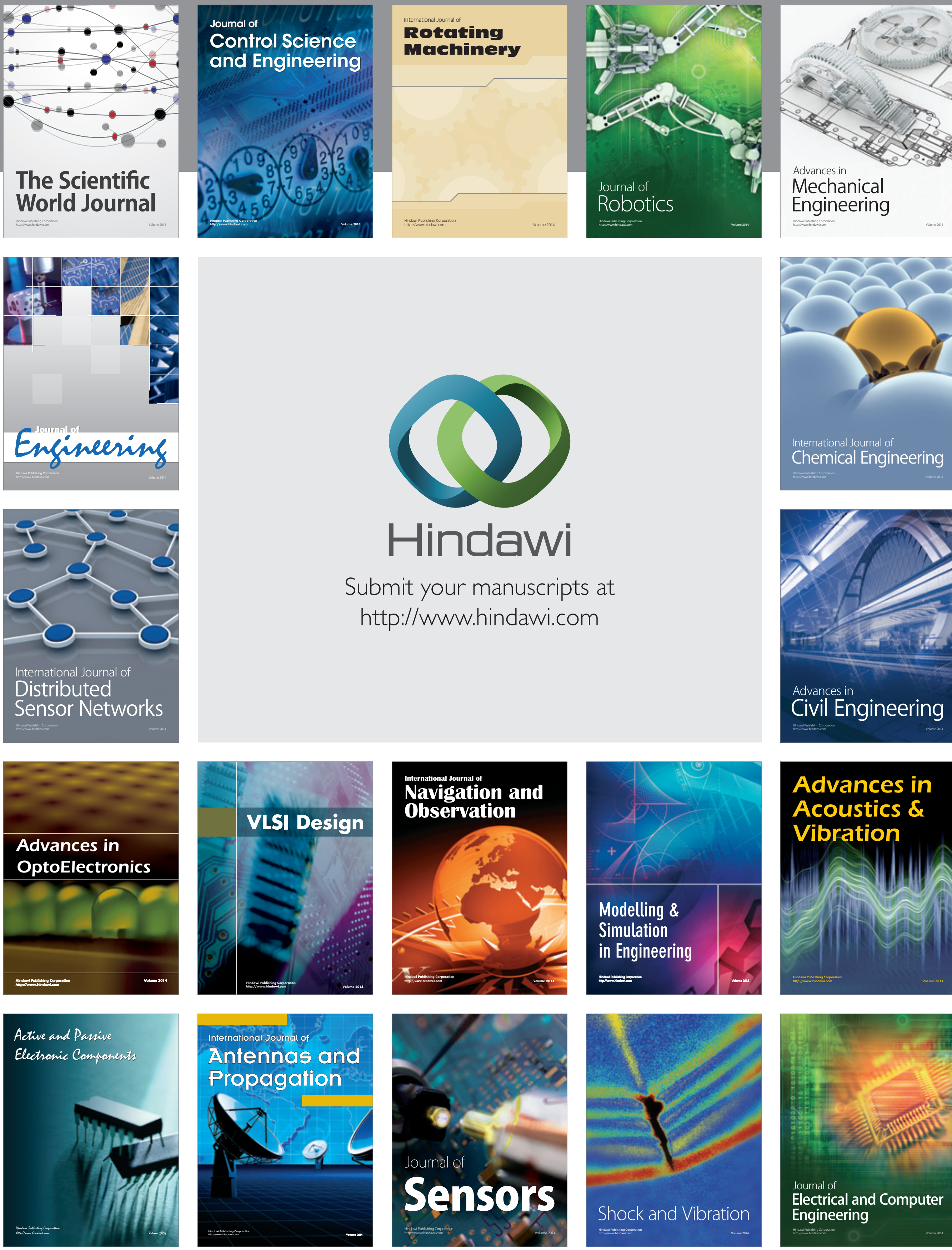
[46] E. J. Hughes, "Multiple single objective pareto sampling," *Evolutionary Computation*, vol. 4, pp. 2678–2684, 2003.

[47] T. Screenuch, A. Tsourdos, E. J. Hughes, and B. A. White, "Fuzzy gain-scheduled missile autopilot design using evolutionary algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4, pp. 1323–1339, 2006.

[48] Y. Jin and J. Branke, "Evolutionary optimization in uncertain environments—a survey," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 3, pp. 303–317, 2005.

[49] J. E. Fieldsend and R. M. Everson, "Multi-objective optimisation in the presence of uncertainty," in *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 1, pp. 243–250, Edinburgh, UK, September 2005.

[50] OptTek, "Multi-objective optimization," OptTek, http://www.opttek.com.


The Scientific World Journal
Journal of Control Science and Engineering
International Journal of Rotating Machinery
Journal of Robotics
Advances in Mechanical Engineering
Journal of Engineering
Hindawi
Submit your manuscripts at
http://www.hindawi.com
International Journal of Chemical Engineering
International Journal of Distributed Sensor Networks
Advances in Civil Engineering
Advances in OptoElectronics
VLSI Design
International Journal of Navigation and Observation
Modelling & Simulation in Engineering
Advances in Acoustics & Vibration
Active and Passive Electronic Components
International Journal of Antennas and Propagation
Journal of Sensors
Shock and Vibration
Journal of Electrical and Computer Engineering