*Aerospace Technology Congress, 8-9 October 2019, Stockholm, Sweden*
*Swedish Society of Aeronautics and Astronautics (FTF)*

# Phase Out Maintenance Optimization

Olle Wijk[a], Patric Andersson[b], Jan Block[c] and Thord Righard[a]

[a] Systecon, [b] P2J Solutions AB, [c] Saab Support and Services
E-mail: olle.wijk@systecon.se, patric.andersson@p2j.se, jan.block@saabgroup.com, thord.righard@systecon.se

## Abstract

This research studies the performance of an optimization technique for phase out optimization of technical systems. The optimization search is done by a genetic algorithm where good candidate samples can be drawn and evaluated in a fast manner using a novel concept denoted matrix simulation. The method is illustrated by applying it to a phase-out scenario of an aircraft fleet, where the optimal stop-maintenance strategy is determined for a set of repairable items (rotables). The main contribution of this paper is a parametric study of how the optimization search is expected to perform when varying the sampleset and problem size. In the parametric study the genetic algorithm result is compared against the real optimum which was found in a separate time-consuming brute force search, going through all possible solutions.

**Keywords**: phase-out, maintenance optimization, genetic algorithm, matrix simulations

## 1 Introduction

For capital-intensive systems, such as aircraft, the retirement process will often stretch over a decade or more, during which the operational fleet gradually contracts. This requires managing the resources for the remaining fleet at acceptable levels of cost, availability and risk. Examples of resources include maintenance personnel, tools, workshops and spares. The improvement of key profits on logistics and maintenance performance can be achieved by inventory management of costly components, which is important for equipment-intensive industries [1]. In a phase-out scenario, the number of aircraft in operation will decrease with time, and the demand for spares will also normally diminish. This combination may increase the relative quantity of spares in stock during the phase-out, which can represent an undesired capital investment.

Several studies have been done to manage spares during the operation phase of a product. In terms of maintenance policy, many researchers studied the joint optimization of maintenance and stock provisioning policy for spare part logistics, see [2], [3], [4] and [5]. In addition, Ferreira and Wang in [6] proposed a hybrid of simulation and analytical models for spare parts optimization, taking into account the residual life of equipment.

During a phase-out period, retired aircraft are usually dismantled to support the remaining operational fleet; called "parting-out" process, see the right-hand side of fig. 1. The "parting-out" refers to the collection of spares from retired aircrafts to be used in remaining operational fleet, i.e. the retired fleet is a source for increasing the spares stock.

In [7] it is suggested from a top-down overview perspective, how a Part-out Based Spare Provisioning (PBSP) management programme can be structured for managing a phase-out scenario in an effective manner. Block in [7] introduces the parting-out approach for spares provisioning management of a fleet and describes the prerequisites of such a method, furthermore, associated key decision criteria are discussed and a framework is presented for the phase-out management process.

This paper focuses on analyzing a method for optimizing the maintenance strategy decisions during a phase-out scenario [8], which is one out of several factors necessary to establish a cost-effective PSBP management programme. More specifically, the paper focuses on finding cost efficient points in time to; stop corrective maintenance (CM), stop preventive maintenance (PM), stop the parting-out process from disposed aircraft, see fig. 1.

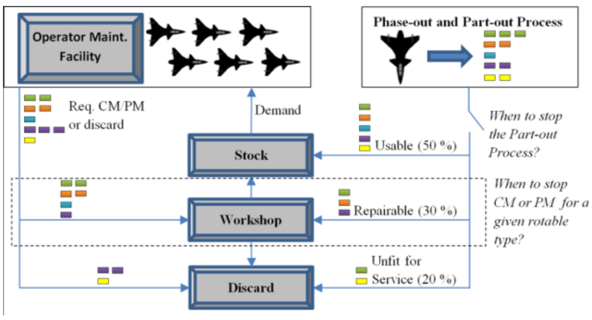


*Figure 1: Schematic of the rotable flow in a PBSP management program [7]. The rotables are here represented by rectangles and the rotable type (see tab. 1) is represented by the rectangle color.*

## 2 Scenario

The optimization technique applied in this paper is illustrated using field data gathered during the operation life of a military aircraft system. The data cover $N = 10$ rotables and have been slightly modified from actual values to allow publication.

In total the data set covers 90 aircraft that are phased out during a $T = 10$ year period; each aircraft in operation accumulates 20 flight hours/month. The aircraft phase-out scheme is shown in fig. 2, at the end of the phase-out period there are still 30 aircraft remaining in operation.

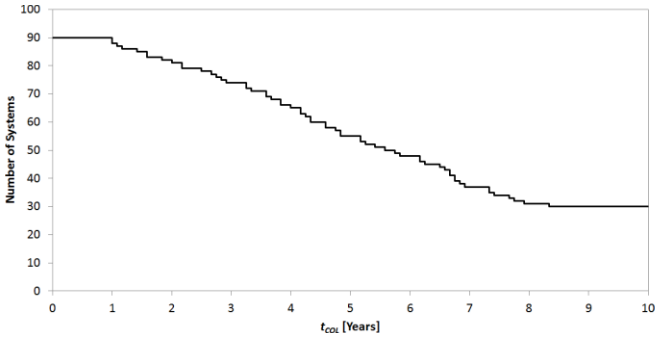


*Figure 2: The phase-out scheme for the 90 aircraft system. After 10 years only 30 aircraft remain in operation.*

The recovered rotables from a retired aircraft are classified in three categories:

- Usable (serviceable, sent directly into storage).
- Repairable (unserviceable, but can be reused after a repair action, sent to the workshop).
- Unfit for service (neither reusable nor worth repairing and should be scrapped).

On average 50 % of the recovered items are classified as usable, 30 % as repairable and 20 % as unfit for service.

Data for all maintenance significant events on the 10 rotables is listed in tab. 1. The reason for studying the particular rotables given in tab. 1 is that they comprise a relatively large maintenance volume and that the maintenance cost for these rotables is relatively high. In tab. 1 it is seen that there is a cost associated with recovering an item from a phased-out aircraft, which means that it can be beneficial to stop recovering items at some point in time. Furthermore, there is a cost for performing CM/PM on an item.

*Table 1: Empirical data used for modelling.*

| Item ID | Quantity per aircraft | Initial stock | Fail rate [$1/10^3$ fh] | CM TAT [Days] | CM cost [SEK] | PM Interval | PM TAT [Days] | PM cost [SEK] | Recover cost [SEK] |
|---|---|---|---|---|---|---|---|---|---|
| Warning Beacon | 1 | 11 | 0.13 | 20.1 | 1500 | 1600 fh | 23.6 | 2000 | 560 |
| Relief Valve | 1 | 11 | 0.07 | 40.3 | 10000 | 1010 fh | 91.2 | 14000 | 3750 |
| Transducer | 1 | 15 | 0.20 | 23.1 | 12000 | 510 fh | 35.3 | 10000 | 4500 |
| Cut-Off Valve | 4 | 23 | 0.20 | 30.2 | 6000 | 2010 fh | 90.3 | 18000 | 2250 |
| A/C - Start Generator | 2 | 18 | 0.16 | 89.6 | 20000 | 800 fh | 69.6 | 20000 | 7500 |
| Hydraulic Generator | 1 | 10 | 0.13 | 30.2 | 6000 | 3600 days | 71.5 | 8000 | 3000 |
| Electric Motor | 2 | 25 | 0.72 | 41.3 | 6000 | 2600 fh | 42.6 | 6000 | 2250 |
| Electric Jack | 1 | 12 | 0.88 | 19.1 | 12000 | 1300 fh | 41.9 | 20000 | 4500 |
| Oxygen Hose | 1 | 18 | 0.34 | 1.0 | 1000 | 510 fh | 21.0 | 400 | 500 |
| Cooling Turbine | 1 | 20 | 0.27 | 115.3 | 30000 | 1010 fh | 132.5 | 43000 | 15000 |

As long as maintenance activities are performed there is also an associated maintenance capability cost

$$c_K = 500\,000 \text{ SEK/year}$$

i.e. costs for keeping necessary maintenance resources, facilities etc. available. The higher the maintenance capability $c_K$ cost is, the more cost efficient it becomes to stop maintenance on all items as soon as possible and preferably at the same time, i.e. due to the $c_K$ parameter it is not possible to find an optimal solution with an item-by-item approach.

## 3 Optimization problem

The phase-out maintenance optimization problem studied in [8] can be formulated as

$$\begin{aligned} & min\, C(\boldsymbol{t}_{CP}, t_{COL}) \\ & \text{s.t. } B_i(t) = 0 \text{ for } i = 1,2,\dots,N. \end{aligned} \qquad (P)$$

where $C(\boldsymbol{t}_{CP}, t_{COL})$ is the objective cost function that can be chosen freely to map reality, $t_{COL}$ represents the time when we stop recovering items from phased out aircraft and $\boldsymbol{t}_{CP}$ is a time vector

$$\boldsymbol{t}_{CP} = (t_{CM_1},\ t_{PM_1}, t_{CM_2},\ t_{PM_2}, \dots, t_{CM_N},\ t_{PM_N})$$

where the two first time elements $t_{CM_1}$ and $t_{PM_1}$ is the time to stop CM/PM respectively for the first rotable item, the third and fourth element $t_{CM_2}$ and $t_{PM_2}$ is the time to stop CM/PM respectively for the second rotable item and so on.

For the specific scenario studied in this article, a backorder constraint of zero is used for each rotable, i.e. no backorders allowed. A more detailed description of the optimization problem and the terminology used is found in [8], where it is also outlined how one can handle non-zero backorder constraints.

## 4 Solving the optimization problem by brute force

An optimization problem can always be solved by going through all possible solution, one obstacle though is that this might be just too time consuming. In [8] a novel concept denoted matrix simulation is developed in order to narrow down the solution space of the optimization problem ($P$). For the scenario described in section 2 the matrix simulations narrows down the solution space to 43 866 471 255 357 candidates, which are tested one by one in a time-consuming 4000 hour brute force search, see [8]. The minimum cost found is

$$C_{min} = 24\,046\,241 \text{ SEK}$$

which in the subsequent sections is used for comparison when evaluating the ability of a genetic algorithm to solve the optimization problem ($P$).

From [8], the optimal time to stop collecting items from phased out aircraft is

$$t_{COL}^{opt} = 7.92 \text{ years}$$

and the optimal time to stop CM/PM on the respective item is listed in tab. 2.

*Table 2: Optimal time instances to stop CM and PM.*

| Item ID | $t_{CM}^{opt}$ | $t_{PM}^{opt}$ |
|---|---|---|
| Warning Beacon | 7.65 years | 3.20 years |
| Relief Valve | 6.62 years | 4.11 years |
| Transducer | 1.37 years | 7.53 years |
| Cut-Off Valve | 7.53 years | 3.88 years |
| A/C - Start Generator | 7.65 years | 5.82 years |
| Hydraulic Generator | 5.71 years | 1.26 years |
| Electric Motor | 4.79 years | 6.74 years |
| Electric Jack | 7.65 years | 4.68 years |
| Oxygen Hose | 2.74 years | 7.88 years |
| Cooling Turbine | 7.53 years | 4.00 years |

## 5 Attacking the optimization problem with a genetic algorithm

The optimization problem ($P$) in section 3 can be solved using a genetic algorithm as described in fig. 3

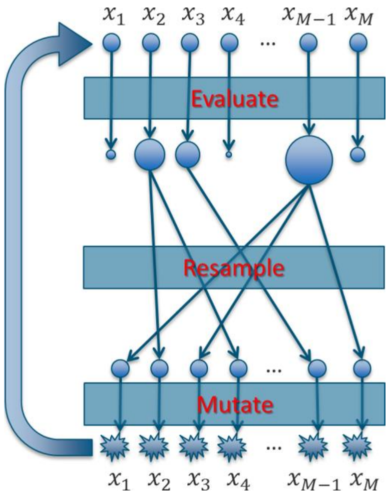


*Figure 3: Genetic algorithm: 1) Evaluate the samples; 2) Survival of the fittest samples by resampling; 3) Mutate the samples.*

The samples used in the genetic algorithm can be described as

$$x_m = (\boldsymbol{t}_{CP}, t_{COL}), \quad m = 1,2, \dots, M$$

When ***initializing*** the genetic algorithm, the $M$ samples are drawn randomly. In [8] it is described how good sample candidates can be randomly drawn from a narrowed solution space satisfying the optimization problem backorder constraints. This is achieved by a novel concept denoted matrix simulation, furthermore as a byproduct of the matrix simulations, fast ***evaluation*** of samples becomes possible. For the specific scenario described in this article, samples were evaluated close to a speed of 3 000 000 samples/sec on a laptop with a 2.7 GHz processor. In the evaluation process samples with low cost $C(\boldsymbol{t}_{CP}, t_{COL})$ are given more support at the expense of samples with higher cost, see [8] for details.

In the ***resampling*** step of the genetic algorithm the next sample population is drawn from the support distribution obtained in the evaluation step. The new sample population then undergoes a ***mutation*** process where the samples are randomly modified slightly within the narrowed solution space obtained by the matrix simulations, see [8] for details.

The evaluation, resampling and mutation steps of the genetic algorithm are repeated for a number of iterations until the best solution found stabilizes. In each iteration the best sample, i.e. the sample with lowest cost $C(\boldsymbol{t}_{CP}, t_{COL})$, is treated separately and is always guaranteed to survive to the next iteration (without mutation) as long as not a new better sample is found taking its place.

In [8], the proposed genetic algorithm is run $I = 100$ iterations using two different sampleset sizes $M = 100$ and $M = 2000$. When comparing with the brute force optimum (see section 4) it is concluded that with $M = 100$ the genetic algorithm comes very close to the optimum (only 0.05 % higher cost) while with $M = 2000$ the optimum is found.

To estimate the probability of the genetic algorithm producing a solution with a cost within a given distance from the minimum cost (found by brute force), it is necessary to run the genetic algorithm multiple times to get sufficient statistical confidence. The next section, which is the main contribution of this article, shows the results from such parametric study when varying the samples set size $M$.

## 6 Parametric study of the genetic algorithm when varying the sampleset size *M*

In [8] it is shown that it is possible to find the optimal phase-out strategy for a field case with 10 items by running a genetic algorithm $I = 200$ iterations with a sampleset size $M = 2\,000$. However, the fact is that the result of the genetic algorithm depends on random numbers drawn during the optimization search (resample and mutate steps). Therefore, changing the random initializer seed of the computer and re-running the genetic algorithm will generate a different search trajectory that may or may not end up in the optimum. Therefore, an interesting question that this article tries to answer is:

*Q: What is the probability of finding the minimum cost of problem (P), or being a certain distance from it, after running the genetic algorithm a certain number of iterations I using a sampleset size M?*

To answer this question one can run the genetic algorithm many times, each time using a different initializer seed to the random generator in order to get a different optimization path. This type of experiment is done 1000 times for the sampleset sizes

$$M = 10, 20, 50, 100, 200, 500, 1\,000, 2\,000, 5\,000, 10\,000$$

and the results are shown in fig. 4 which represents the probabilistic state of the best sample for the genetic algorithm after $I = 200$ iterations.

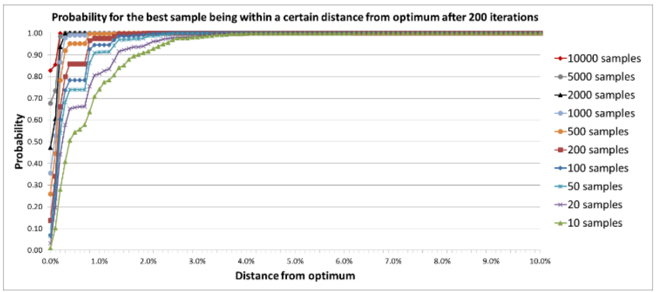


*Figure 4: Probability of being a certain distance from the optimum after running the optimization $I = 200$ iterations.*

Although running the genetic algorithm once is fast, 2 seconds when $M = 100$ and 1 minute when $M = 2000$, doing it repeatedly 1 000 times for the different sampleset sizes $M$ takes a long time. Therefore, in order to speed up the the parameter study the genetic algorithm searches were spread out on a number of computers.

In the graph of fig. 4 the distance to the minimum cost (found by brute force search) is given in percent on the x-axis. The statistically estimated probability of being a certain percentage distance from the minimum cost after 200 iterations (based upon 1 000 trials) is shown on the y-axis. As expected, the probability of finding the optimum increases as the sampleset size M increases. If using $M = 10\,000$ the probability of finding the optimum after $I = 200$ iterations is greater than 0.8. Furthermore, it is seen that all 1 000 trials for a sampleset size of $M = 10\,000$ manages to get within a distance of 0.2 % of the optimum.

## 7 Parametric study repeated with a modified mutation step

In [8], the mutation step of the genetic algorithm is achieved by modifying the stop time for CM/PM for *one* of the rotables (randomly selected) to a neighbor candidate in the solution space produced by the matrix simulations. There is however a risk of getting stuck in local minima when mutating the solution in small steps, and therefore an alternative mutation process is tested where the stop time for CM/PM (for the randomly selected rotable) was randomly selected among *all* possible solutions provided by the matrix simulations (equal chance of being drawn). When re-running the parametric study of sampleset sizes $M$ with this slight modification of the genetic algorithm the results of fig. 5 are obtained.

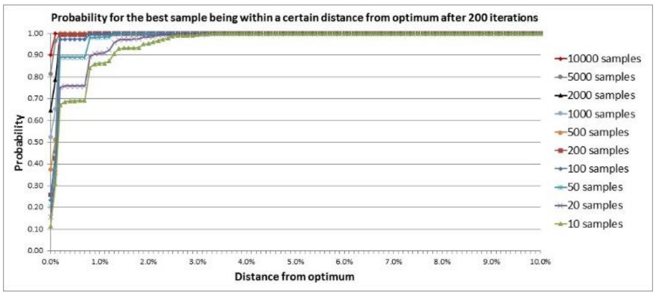


*Figure 5: Probability of being a certain distance from the optimum after running the optimization $I = 200$ iterations after modifying the mutation process.*

Comparing the results of fig. 4 with fig. 5 the convergence to the optimum is faster since every sampleset size $M$ in fig. 5 has a higher probability of finding the optimum compared to fig. 4. In particular, the probability of finding the minimum cost with $M = 10$ samples after $I = 200$ iterations has increased from 0 % to 10 %, and the probability of finding the optimum with $M = 10\,000$ samples after $I = 200$ iterations has increased from the 80 % range to the 90 % range. Furthermore, it is seen in fig. 5 that all 1000 trials for a sampleset size of $M = 10\,000$ manages to get within a distance of 0.1 % of the minimum cost (compared to 0.2 % in fig. 4)

## 8 Parametric study on a large scale problem

In a real-world application there will be more than $N = 10$ items involved in the phase-out optimization problem described in this article. Therefore, the parametric study of section 7 was repeated for a problem involving $N = 100$ items. In this case it was not possible to find the true optimum by a brute force search (too many solution candidates), but it was still possible to compare the genetic algorithm performance against the minimum cost value found in *all* genetic algorithm searches

$$MINVALUE = 174\,550\,398 \text{ SEK}.$$

The result of this study is found in fig. 6 which shows the probabilistic state the genetic algorithm after $I = 4000$ iterations compared to $MINVALUE$ for varying sampleset sizes $M = 10, 20, 50, 100$ and 200. For $I = 4000$ and $M = 200$ it is seen that there is a 40 % probability to find $MINVALUE$. An increased amount of iterations or samples is expected to increase the probability to find the $MINVALUE$ but unfortunately there was no time to finish such additional parametric variations within the given project time frame.

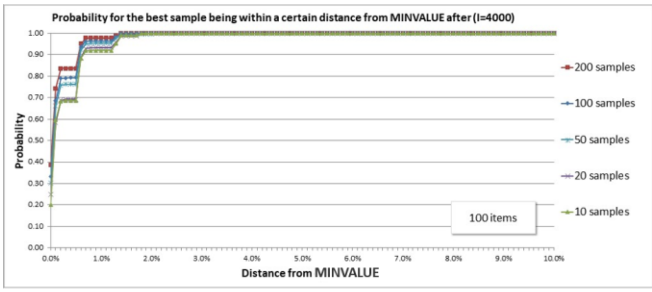


*Figure 6: Probability of being a certain distance from the $MINVALUE$ after running the optimization $I = 4000$ iterations.*

## 9 Conclusions

This article analyzes the capability of a genetic algorithm to solve a phase-out optimization problem for a fleet of aircraft, but the ideas are general and can be applied to other types of technical systems.

For a problem size of $N = 10$ items the genetic algorithm, after running $I = 200$ iterations, has a high probability (0.7) of getting within a 1 % range of the optimum already with a small sampleset size of $M = 10$, with higher sampleset sizes, $M = 10\,000$, there is a high probability (0.9) of finding the optimum (fig. 5).

When increasing the problem size to $N = 100$ items, more iterations are required to get close to the optimum. After running the genetic algorithm $I = 4000$ iterations with a sample set size of $M = 200$, there is a high probability (above 0.9) to be within a 1 % range of what could be an optimum (fig. 6).

Looking closer at the results from fig. 5 it is noted that with

$$N = 10, I = 200 \text{ and } M = 200$$

there is a probability of 0.25 to find the optimum. An interesting question is if $N$ and $I$ are increased with a factor 10, i.e.

$$N = 100, I = 2000 \text{ and } M = 200$$

is there then still a probability of 0.25 to find the optimum? More generally, are the results of the analysis done in fig. 5 ($N = 10$) and fig. 6 ($N = 100$) similar after $I$ and $10I$ iterations, respectively? The answer to this question is shown in fig. 7 (the 200 samples probability curve from fig. 5) and fig. 8 (the 200 samples probability curve from fig. 6, but after $\boldsymbol{I = 2000}$ iterations instead of $I = 4000$ iterations). Although the curves of fig. 7 and fig. 8 have some resemblances, it is clear that the probability curve of $N = 100$ items (fig. 8) is mostly below the probability curve of $N = 10$ items (fig. 7). This is an indication that if the problem size $N$ increases with a factor $k$ then the number of iterations $I$ needs to be increased more than a factor $k$ to achieve similar results, further research is required to establish a more detailed relationship between $I$ and $N$.

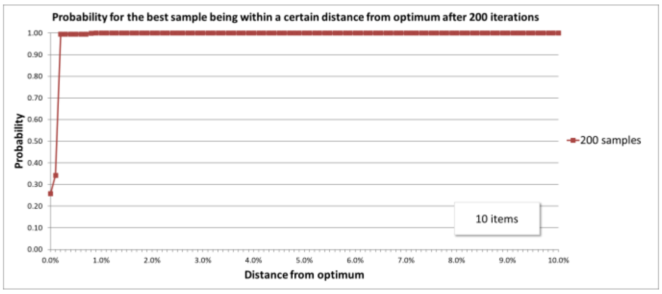


*Figure 7: Probability of being a certain distance from the optimum when N=10, I=200 and M= 200.*

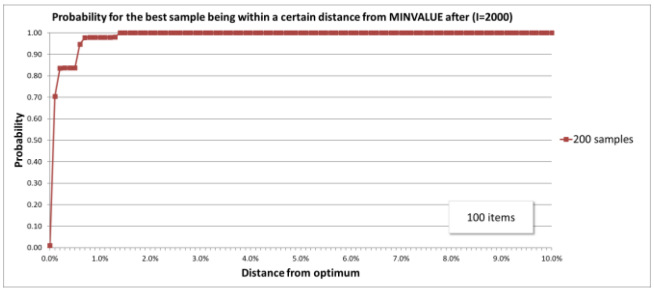


*Figure 8: Probability of being a certain distance from $MINVALUE$ when $N = 100$, $I = 2000$ and $M = 200$.*

Producing a genetic algorithm parameter study, varying the number of items $N$, the number of iterations $I$ and the sampleset size $M$ is a time consuming process and was enabled by a fast sample evaluation, through matrix simulations described in [8].

## Acknowledgments

We would like to acknowledge gratefully the financial support granted through the NFFP6 project "Efficient Performance Based Air Vehicle Maintenance", Supported by VINNOVA, Luleå University of Technology, Saab Support and Services, SAAB Aeronautics and Systecon

## References

[1] Braglia, M. & Frosolini, M. (2013), "*Virtual pooled inventories for equipment-intensive industries. An implementation in a paper district*", Reliability Engineering & System Safety, 112, 26-37

[2] Chen, M. C., Hsu, C. M. & Chen, S. W. (2006), "*Optimizing joint maintenance and stock provisioning policy for a multi-echelon spare part logistics network*", Journal of the Chinese Institute of Industrial Engineers, 23, 289-302

[3] Geiger, C. D., Martinez, O. E. & Lodree Jr, E. J. (2007), "*A multiobjective modeling approach for joint maintenance and spare parts inventory policy optimization*", IIE Annual Conference. Proceedings. pp. 1740-1745

[4] Scarf, P. A. & Cavalante, C. A. V. (2012), "*Spare parts provision for a maintained system with a heterogeneous*

*lifetime*", Advances in Safety, Reliability and Risk Management pp. 983-988

[5] Ilgin, M. A. & Tunalis S. (2007), "*Joint optimization of spare parts inventory and maintenance policies using genetic algorithms*", International Journal of Advanced Manufacturing Technology, 34, pp. 594-604

[6] Ferreria, R. J. P. & Wang, W. (2012), "*Spare parts optimisation subject to condition monitoring*", 11th International Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference 2012, PSAM11 ESREL, Vol 1, pp. 163-170

[7] Block, J., Ahmadi, A., Tyrberg, T., Peter Söderholm (2014), "*Part-out-based spares provisioning management: A military aviation maintenance case study*", Journal of Quality in Maintenance Engineering, Vol. 20 Iss: 1, pp.76 – 95

[8] Wijk, O., Andersson P., Block J. & Righard T. (2017), "*Phase-out maintenance optimization for an aircraft fleet*", International Journal of Production Economics, Vol. 188, pp 105-115