Proceedings of the 33rd European Safety and Reliability Conference (ESREL 2023)
Edited by Mário P. Brito, Terje Aven, Piero Baraldi, Marko Čepin and Enrico Zio

doi: 10.3850/978-981-18-8071-1_P064-cd

Predictive maintenance planning using renewal reward processes and probabilistic RUL prognostics - analyzing the influence of accuracy and sharpness of prognostics

Mihaela Mitic

Faculty of Science, Utrecht University, The Netherlands.
E-mail: m.a.mitici@uu.nl

Ingeborg de Pater

Faculty of Aerospace Engineering, Delft University of Technology, The Netherlands.
E-mail: i.i.depater@tudelft.nl

Zhiguo Zeng

Laboratoire Genie Industriel, CentraleSupélec - Université Paris-Saclay, France.
E-mail: zhiguo.zeng@centralesupelec.fr

Anne Barros

Laboratoire Genie Industriel, CentraleSupélec - Université Paris-Saclay, France.
E-mail: anne.barros@centralesupelec.fr

We pose the maintenance planning for systems using probabilistic Remaining Useful Life (RUL) prognostics as a renewal reward process. Data-driven probabilistic RUL prognostics are obtained using a Convolutional Neural Network with Monte Carlo dropout. The maintenance planning model is illustrated for aircraft turbofan engines. The results show that in the initial monitoring phase, the accuracy and sharpness of the RUL prognostics is relatively small. The maintenance of the engines is therefore scheduled far in the future. As the usage of the engine increases, the accuracy of the prognostics improves, while the sharpness remains relatively small. As soon as the estimated probability of the RUL is skewed towards 0, the maintenance planning model consistently indicates it is optimal to replace the engines immediately, i.e., "now". This shows that probabilistic RUL prognostics support an effective maintenance planning of the engines, despite being imperfect with respect to accuracy and sharpness.

Keywords: predictive maintenance planning, probabilistic RUL prognostics, aircraft engines, renewal processes, convolutional neural network, Monte Carlo dropout.

1. Introduction

The increasing use of sensors to monitor the health of systems has generated large volumes of data. This has incentivized the development of data-driven Remaining Useful Life (RUL) prognostics for these systems in the last years. Ultimately, these prognostics are expected to support the maintenance planning Shi et al. (2020).

Since the degradation of technical systems is stochastic, it is of interest to quantify the uncertainty associated with the RUL prognostics Fink et al. (2020). This is particularly important for maintenance planning. In this line, several studies determine probabilistic RUL prognostics, and integrate these prognostics in the maintenance planning. In Nguyen and Medjaher (2019), the probability that an aircraft engine fails in predefined time-windows is estimated using a neural network. Using these probabilities, the optimal moment to order a new spare part and to replace the engine are determined. In Lee and Mitici (2023), the probability density function (PDF) of the RUL of aircraft engines is predicted using a Convolutional Neural Network (CNN) with Monte Carlo dropout. These probabilistic RUL prognostics are further integrated in a deep reinforcement learning framework for maintenance planning. In de Pater and Mitici (2021), the PDF of the RUL of aircraft cooling units is estimated using particle filtering. Using a linear program together with these prognostics, the maintenance of the units is planned. Last, in Consilvio et al. (2020), the probability of failure of a railway track is estimated using a physics-based model. The maintenance of the railway tracks is then planned using a linear program that minimizes the risk.

Despite the current methodological advancements for (probabilistic) RUL prognostics, prognostics are still imperfect: The estimated PDF of the RUL is not necessarily centered around the true RUL (low accuracy), the variance of the estimated PDF may be large (low sharpness), and the mass of the estimated PDF of the RUL may be concentrated such that the true RUL is underestimated/overestimated. Moreover, current machine learning algorithms that estimate the RUL often consider as loss function only metrics related to the accuracy, such as the Mean Square Error (MSE), without explicitly considering the sharpness of the prognostics, or a combination of the accuracy and the sharpness. However, we expect that both the accuracy and sharpness of the RUL prognostics influence the maintenance planning. In this paper, we thus investigate the influence of the accuracy and sharpness of the RUL prognostics on the maintenance planning. Specifically, we analyze the ability to plan maintenance and avoid failures with imperfect RUL prognostics.

We develop probabilistic RUL prognostics using Convolutional Neural Networks (CNN) with Monte Carlo dropout for the aircraft engines in the C-MAPSS dataset Saxena and Goebel (2008). As expected, these RUL prognostics are imperfect. Next, we determine an optimal maintenance moment for these engines using renewal-reward processes. In the initial usage phase, the accuracy of the prognostics is relatively small. The prognostics become more accurate as the engine degrades over time, and more measurements are collected. Also, the sharpness of the prognostics remains relative small until the engines are close to failure. However, although the prognostics are neither very accurate nor very sharp in the beginning of the engines' lifetime, we replace all engines before their failure. For the considered dataset, the optimal replacement time is *now* for all engines when the actual RUL of the engines is ten cycles or less. At this moment the mass of the estimated PDF of the RULs is concentrated around 0. In general, the optimal replacement moment is *now* for all engines when the estimated probability $P(\mathrm{RUL} = 0) \geq 0.004$.

2. Data-driven predictive maintenance scheduling

2.1. *Maintenance scheduling using Renewal Reward Processes*

We consider a component (aircraft engine) whose health is continuously monitored. At a generic time step k during the life of the component, we are interested in determining the optimal time $k + t_k^*$ to replace this component. The component is in a brand-new state after replacement, i.e., perfect maintenance.

At time step k, using the measurements recorded up to time step k and a CNN with Monte Carlo dropout (see Section 2.2), we estimate the probability that the RUL of the component is i time steps, $i \geq 0$. Let $\phi_k(i)$ denote the probability that the component, after being used for k steps, has a RUL of exactly i time steps. To determine an optimal time to replace the component, we consider the expected cost per unit of time:

$$\frac{\text{[Expected cost over the current life cycle]}}{\text{[Expected current life cycle]}}$$

Formally, at time step k, we are interested in finding t_k^* such that:

$$t_k^* := argmin_{t_k > 0} \frac{\mathbb{E}[C(k, t_k)]}{\mathbb{E}[L(k, t_k)]}, \qquad (1)$$

with $C(k, t_k)$ the cost of replacing the component at time $k + t_k$, given that this component has already been used for k time steps, and $L(k, t_k)$ the lifetime of the component, given that this component is replaced either upon failure, or preventively after being used for $k + t_k$ steps.

If the component is scheduled for replacement at time $k + t_k$ from now, and this component does not fail from now (time step k) until $k + t_k$, then the component is replaced preventively at time $k + t_k$ at a cost c_r. If, however, the component fails at some time $j, k < j < k + t_k$, then a failure cost c_f is incurred (corrective replacement), and the component is immediately replaced.

With this, the expected cost over the current life cycle of the component is:

$$\mathbb{E}[C(k,t_k)] = c_f \sum_{i=1}^{t_k-1} \phi_k(i) + c_r(1 - \sum_{i=0}^{t_k-1} \phi_k(i)).$$

Also, the expected current life cycle is:

$$\mathbb{E}[L(k,t_k)] = k + \sum_{i=1}^{t_k-1} i\phi_k(i) + t_k(1 - \sum_{i=0}^{t_k-1} \phi_k(i)).$$

Eq. (1) is solved using a numerical grid search. In the next section we estimate $\phi_k(i)$ after every flight cycle k of the aircraft engines using sensor measurements and a CNN.

2.2. *Probabilistic RUL prognostics using a CNN with Monte Carlo dropout*

We obtain RUL prognostics for the aircraft engines in the C-MAPSS data set (see Saxena and Goebel (2008)). The C-MAPSS dataset consists of four subsets (FD001, FD002, FD003, FD004), each with a test and training set. Each subset has different failure and flight conditions. For each engine in each training set, one measurement of the engine is available per flight cycle per sensor until the failure of the engine. Using the recorded sensor measurements up to flight cycle k, we predict a probability density function (PDF) of the RUL of the engines. These prognostics are updated after each flight cycle k. For this, we train a CNN with Monte Carlo dropout for each subset, following Mitici et al. (2023).

The C-MAPSS dataset contains a total of 21 sensors, of which seven sensors produce constant measurements. We therefore use the remaining $H = 14$ sensors as input to the CNN. We normalize the sensor measurements with min-max normalization with respect to the operating condition (see Mitici et al. (2023)). Specifically, after each flight cycle k of each engine v, we consider the data sample X_k^v as input to the CNN. This data sample contains the sensor measurements of engine v of the past $N = 30$ flights:

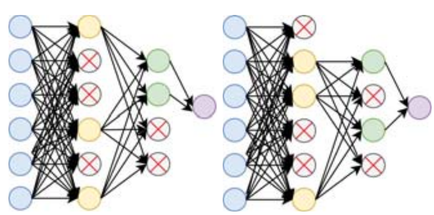
$$X_k^v = [x_{k-N}^v, x_{k-N+1}^v, \ldots, x_k^v], \tag{2}$$

$$x_i^v = [\hat{m}_{i1}^v, \hat{m}_{i2}^v, \ldots, \hat{m}_{iH}^v], \tag{3}$$

with $\hat{m}_{ij}^v$ the normalized sensor measurement of flight cycle i of engine v from sensor j.

Following hyper-parameter tuning using grid search Mitici et al. (2023), we consider a CNN where the first five convolutional layers each contain a ten one-dimensional filters of size 1×10. Then, all ten feature maps are combined in a single feature map by a sixth convolutional layer with only a single filter, of size 1×3. Last, we predict the RUL of the engine by inputting this final feature map in two fully connected layers. The first fully connected layers contains 100 nodes, and the last fully connected layer contains one node and outputs the RUL prediction. Throughout the neural network, we apply the tanh activation function, except for the last layer, where we apply the ReLU activation function Mitici et al. (2023).

To obtain a PDF of the RUL, we apply Monte Carlo Dropout Gal and Ghahramani (2016) as follows. During the training phase, we apply a dropout rate of 0.5 to each layer of the CNN, except for the first layer. This is often done to prevent overfitting. However, we apply Monte Carlo dropout with a rate of 0.5 also in the testing phase, when we predict the RUL belonging to a new data sample. In general, the dropout rate can be further optimized using hyper-parameter tuning. For each new test sample, we perform $M = 1000$ forward passes of this sample through the neural network. Due to the dropout, different, randomly selected neurons (50 percent per layer) are dropped during each forward pass. Thus, a different RUL prediction is obtained with each forward pass. The PDF of the RUL is constructed by giving each individual RUL prediction a probability of $\frac{1}{M}$. This procedure is illustrated in Figure 1.



(a) First pass (b) Second pass

Fig. 1.: Schematic example - Monte Carlo dropout for a neural network with 3 fully connected layers, during 2 passes of a sample through the network.

3. Results - Probabilistic RUL prognostics for aircraft engines

For the test subsets of C-MAPSS, the measurements stop at some moment before failure, i.e., they are not run-to-failure instances. For our maintenance planning methodology, however, we need to have the sensor measurements until the failure time of the engine. Otherwise, we cannot evaluate how the optimal maintenance moment changes when an engines degrades over time. We thus use the engines in the C-MAPSS training set for maintenance planning. We randomly select 80% of the engines of each training set Nguyen and Medjaher (2019) to train the CNNs. Then, we use the trained CNN with Monte Carlo dropout to predict a PDF of the RUL after each flight cycle for the remaining 20% of the engines. These predictions are subsequently used in the maintenance planning.

We train the neural network using the Adam optimizer Kingma and Ba (2014), with a training validation split of 80%-20% and 250 epochs. The initial learning rate is 0.001, and is divided by two when there is no improvement in the validation loss for ten epochs in a row Mitici et al. (2023).

Table 1.: RMSE, MAE, mean standard deviation (STD) and CRPS of the PDFs of the RUL.

Subset	RMSE	MAE	CRPS	Mean STD
FD001	13.1	10.1	7.1	12.0
FD002	15.2	11.7	8.4	11.5
FD003	13.6	10.2	7.3	12.4
FD004	15.9	10.7	7.1	12.2

Table 1 shows the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE), calculated with the mean RUL prediction of each engine and after each flight cycle. To also evaluate the quality of the PDF of the RUL, we also show the mean standard deviation of the PDF of the RUL and the mean Continuously Ranked Probability Score (CRPS, de Pater and Mitici (2022)). Let y_{ij} denote the actual RUL for an engine i belonging to the j^{th} flight cycle, and let $F_{\widehat{y_{ij}}}(x)$ denote the estimated, empirical CDF of the RUL of a engine i and flight cycle j. The mean CRPS is defined as follows:

$$\text{CRPS} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{F_i}\sum_{j=1}^{F_i}\text{CRPS}_{ij}, \tag{4}$$

$$\text{CRPS}_{ij} = \int_{-\infty}^{\infty}(F_{\widehat{y_{ij}}}(x) - \mathcal{I}\{y_{ij} \leq x\})^2 dx,$$

with

$$\mathcal{I}\{y_{ij} \leq x\} = \begin{cases} 1, & y_{ij} \leq x \\ 0, & y_{ij} > x. \end{cases}$$

with N the number of engines selected for maintenance planning, and F_i the number of flight cycles for which we have a RUL prediction for an engine i. Intuitively, the CRPS is a probabilistic generalization of the absolute error, and analyzes both the accuracy of the PDF (i.e., whether the estimated RUL distribution is centered around the actual RUL), and the sharpness of the PDF (i.e., if the variance is low). The CRPS is small when the corresponding PDF of the RUL is accurate and sharp. For example, when all M individual RUL predictions are close to the actual RUL, the accuracy and sharpness of the corresponding PDF of the RUL is high, and the CRPS is thus low. The lowest possible value of the CRPS equals zero, which we only obtain for a perfect RUL prediction without any uncertainty.

4. Results - Maintenance scheduling using probabilistic RUL prognostics

We considered a preventive replacement cost of $c_p = 10$, and a corrective maintenance cost of $c_f = 100$ for a failed engine. Table 2 shows the planning results for four engines selected for maintenance, one from each subset of C-MAPSS. Here, y_k and $\hat{y_k}$ denote the actual and predicted RUL, respectively, after the engine has been used for k cycles. The optimal maintenance moment is close to the lower bound of the 99% confidence interval of the RUL for all engines in Table 2. This is because the extra lifetime gained by postponing the moment of maintenance does not outweigh the extra expected failure cost gained from this postponement.

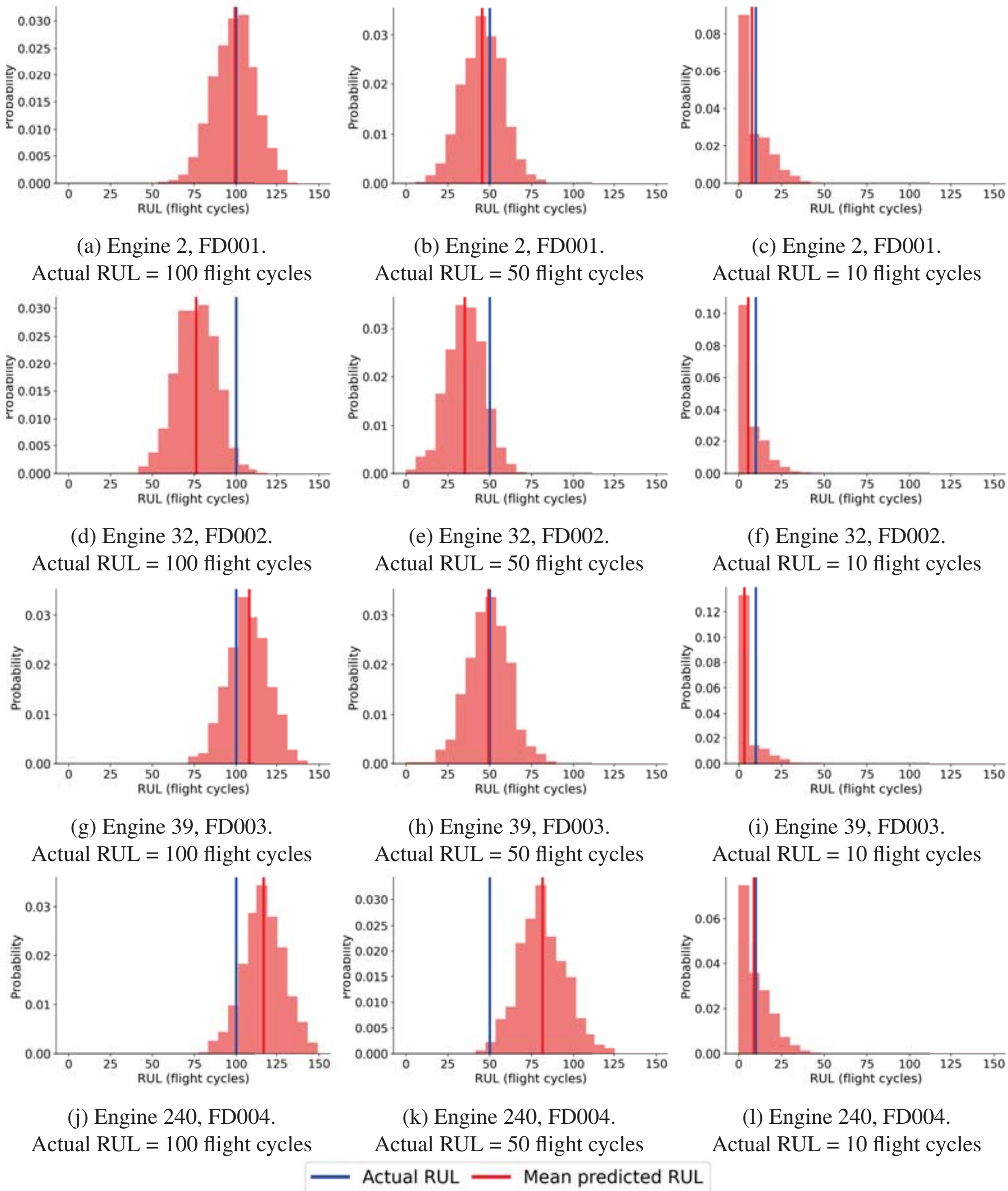
Table 2 shows that for engine 2 (FD001) and engine 39 (FD003) the mean and median RUL prediction are close to the actual RUL throughout

Table 2.: Performance of probabilistic RUL prognostics and maintenance planning for engines 2 (FD001), 32 (FD002), 39 (FD003) and 240 (FD004). When $y_k - t_k^*$ is negative, it is optimal (given the predicted PDF of the RUL) to maintain the engine after the actual failure time.

Usage		RUL prediction metrics					Optimal maintenance time		
Actual RUL y_k	Usage time k	Mean predicted RUL $\hat{y_k}$	Mean prediction error $y_k - \hat{y_k}$	99% CI of the predicted RUL $\hat{y_k}$	STD predicted RUL $\hat{y_k}$	CRPS	t_k^*	$k + t_k^*$	$y_k - t_k^*$
Engine 2 - subset FD001 (True lifetime = 287 cycles)									
100	187	99.3	0.7	[66, 128]	12.6	3.0	61	248	39
50	237	45.5	4.5	[14,75]	11.9	3.4	13	250	37
40	247	43.2	-3.2	[13,71]	11.7	3.1	9	256	31
30	257	24.9	5.1	[0,54]	11.8	3.6	0	257	30
20	267	25.1	-5.1	[0,55]	12.2	3.7	0	267	20
10	277	7.7	2.3	[0,36]	8.9	3.3	0	277	10
Engine 32 of subset FD002 (True lifetime = 281 flight cycles)									
100	181	76.2	23.8	[45,107]	11.8	17.3	42	223	58
50	231	35.1	14.9	[6,62]	11.1	9.4	0	231	50
40	241	29.0	11.0	[2,56]	10.5	6.6	0	241	40
30	251	23.8	6.2	[0,52]	10.8	3.9	0	251	30
20	261	11.9	8.1	[0,42]	9.7	5.4	0	261	20
10	271	5.5	4.5	[0,30]	7.3	4.0	0	271	10
Engine 39 of subset FD003 9True lifetime = 288 flight cycles)									
100	188	107.8	-7.8	[77,137]	12.1	4.7	74	262	26
50	238	49.1	0.9	[18,80]	12.0	2.8	18	256	32
40	248	36.9	3.1	[8,65]	11.7	3.0	7	255	33
30	258	20.1	-0.7	[0,62]	12.3	2.9	0	258	30
20	268	20.2	-0.2	[0,49]	11.7	2.8	0	268	20
10	278	3.3	6.7	[0,28]	6.5	6.1	0	278	10
Engine 240 of subset FD004 (True lifetime = 149 flight cycles)									
100	49	117	-17.3	[87,151]	12.4	11.3	87	136	13
50	99	81.5	-31.5	[49,120]	14.0	23.7	46	145	4
40	109	85.2	-45.2	[51,117]	13.4	37.6	47	156	-7
30	119	56.3	-26.3	[25,88]	12.6	19.3	23	142	7
20	129	26.3	-6.3	[0,58]	12.1	4.0	0	129	20
10	139	8.9	1.1	[0,38]	8.9	2.7	0	139	10

their lifetime, i.e., the RUL prognostics are accurate. The high accuracy is also reflected by the low value of the CRPS. For both engines, however, the PDF's are wide (not sharp) with a standard deviation (STD) of approximately 12 flight cycles (see also the PDF's in Figure 2). Because the RUL prognostics of these 2 engines are accurate, but not sharp, the optimal maintenance moment is quite early relative to the failure time. From 30 flights before failure onwards, the estimated probability that the RUL is 0 is always at least 0.002. From this moment on, it is consistently optimal to immediately replace these 2 engines (*now*, $t_k^* = 0$) (see also Figure 3). To limit the wasted life for these engines, it would be desirable that the RUL prognostics are not only accurate, but also sharp.

For engine 32 (FD002), the RUL is underestimated: the mean RUL prediction is only 76.2 cycles while the actual RUL is 100.0 cycles. Because the RUL is underestimated in the beginning of the engine's lifetime, we replace this engine much earlier than its failure time. When the actual RUL = 50 cycles, it is already optimal to immediately ($t_k^* = 0$) replace engine 32 (Figure 3). Here, underestimating the RUL early on resulted in a significant waste of the engine' lifetime.



(a) Engine 2, FD001. Actual RUL = 100 flight cycles

(b) Engine 2, FD001. Actual RUL = 50 flight cycles

(c) Engine 2, FD001. Actual RUL = 10 flight cycles

(d) Engine 32, FD002. Actual RUL = 100 flight cycles

(e) Engine 32, FD002. Actual RUL = 50 flight cycles

(f) Engine 32, FD002. Actual RUL = 10 flight cycles

(g) Engine 39, FD003. Actual RUL = 100 flight cycles

(h) Engine 39, FD003. Actual RUL = 50 flight cycles

(i) Engine 39, FD003. Actual RUL = 10 flight cycles

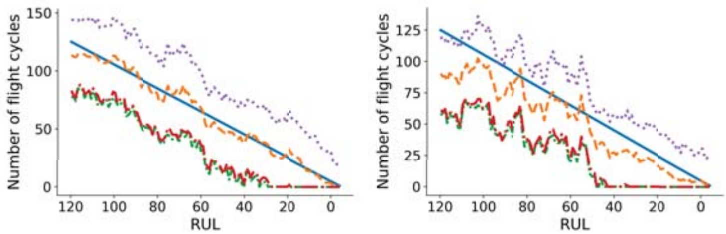
(j) Engine 240, FD004. Actual RUL = 100 flight cycles

(k) Engine 240, FD004. Actual RUL = 50 flight cycles

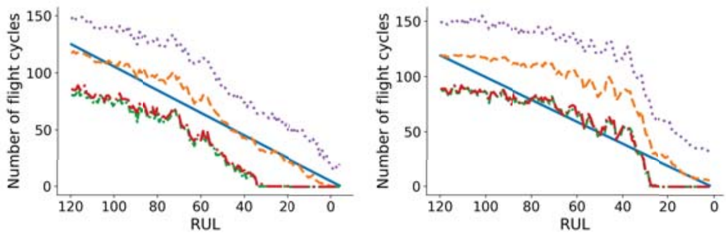
(l) Engine 240, FD004. Actual RUL = 10 flight cycles

Fig. 2.: Predicted PDF of the RUL of engine 2, 32, 38 and 240 of subset FD001, FD002, FD003 and FD004 respectively, when the actual RUL is 100, 50 and 10 flight cycles.

For engine 240 (FD004), the RUL is overestimated in the early phase. The mean predicted RUL is 85.2 cycles when the actual RUL is 40 cycles. At this moment, the optimal maintenance time is 47 cycles in the future, which is after the failure time of engine 240. As the usage of the engine increases, the estimated PDF of the RUL becomes more accurate and skewed towards 0. As a result, the planning model consistently recommends to immediately (*now*, $t_k^* = 0$) replace the engine. In fact, from 20 cycles before failure onwards, the estimated probability that the RUL is 0 is at

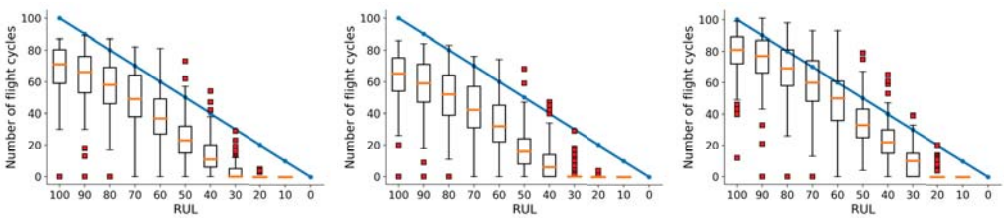


(a) Engine 2, FD001 (b) Engine 32, FD002



(c) Engine 39, FD003 (d) Engine 240, FD004

Fig. 3.: Optimal maintenance planning: t_k^* optimal number of cycles until replacement, given current time k - engines 2 (FD001), 32 (FD002), 38 (FD003), 240 (FD004).



(a) $c_p = 10$, $c_f = 100$ (b) $c_p = 1$, $c_f = 100$ (c) $c_p = 50$, $c_f = 100$

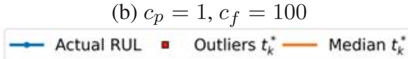


Fig. 4.: Boxplot of t_k^* for several values of the actual RUL, considering a sensitivity analysis of ratio of the preventive and corrective replacement costs c_p and c_f - all 141 engines selected for maintenance planning from the four subsets of C-MAPSS.

least 0.015. As a result, it is consistently optimal to immediately replace the engine.

As soon as the actual RUL is 10 cycles or less, the planning model consistently indicates for all engines immediate replacement as optimal. In these cases, the estimated probability that the RUL=0 is at least 0.052. In general, as soon as the estimated probability that the RUL=0 is at least 0.004, the optimal action is to immediately replace the engines. The accuracy of the prognostics improves as the engines approach their failure time. As a result, all engines are replaced before failure. Thus, to prevent failures, it is especially important to have accurate RUL prognostics in the final phase of the engines' lifetime.

4.1. *Sensitivity analysis - Costs*

In this section we analyze the impact of various ratios of preventive and corrective replacement costs. Figure 4 shows the optimal replacement time t_k^* for all engines selected for maintenance planning, for three different cost ratios: the original costs $c_p = 10, c_f = 100$ (Figure 4a), the costs $c_p = 1, c_f = 100$ (Figure 4b) and the costs $c_p = 50, c_f = 100$ (Figure 4c). As expected, we replace engines later when the preventive replacement costs c_p are larger (relative to the corrective replacement costs c_f). However, even when the ratio $c_f/c_p = 100/50 = 2$, i.e, when the cost of corrective replacement is relatively small compared with the cost of preventive replacement, all engines are still replaced immediately (i.e., *now*, $t_k^* = 0$) as soon as there are 10 flights or less before failure, i.e., in the final phase of the lifetime of the engines, the proposed model consistently indicated an immediate replacement.

5. Conclusion

We propose a stochastic renewal-reward process for predictive maintenance planning. This maintenance planning integrates data-driven probabilistic RUL prognostics. Our framework is illustrated for the aircraft engines from C-MAPSS. The probabilistic RUL prognostics are obtained using a Convolutional Neural Network with Monte Carlo dropout. As expected, the RUL prognostics are imperfect, i.e., the prognostics have a relatively small sharpness and accuracy, especially in the initial phase of the engines' usage. As the engines approach their failure time, the accuracy improves, while the sharpness improves only slightly in the last cycles of the life of the engines.

Despite these imperfect probabilistic RUL prognostics, the maintenance planning model schedules effectively engines for replacement. Initially, the engines are scheduled for replacement far in the future. As soon as the mass of the estimated PDF of the RUL is concentrated around zero, the planning model consistently indicates immediate engine replacement as an optimal action. As a result, all considered engines are replaced before failure.

As future work, we aim to further tune the dropout rate of the CNN used for prognostics and its impact of the maintenance planning. We also plan to further analyze the impact of the maintenance cost ratio on the planning.

Acknowledgement

The contribution of Anne Barros and Zhiguo Zeng is partially supported by the Chair of Risk and Resilience of Complex Systems (Chair EDF, SNCF and Orange).

References

Consilvio, A., A. Di Febbraro, and N. Sacco (2020). A rolling-horizon approach for predictive maintenance planning to reduce the risk of rail service disruptions. *IEEE Transactions on Reliability 70*(3), 875–886.

de Pater, I. and M. Mitici (2021). Predictive maintenance for multi-component systems of repairables with remaining-useful-life prognostics and a limited stock of spare components. *Reliability Engineering & System Safety 214*, 107761.

de Pater, I. and M. Mitici (2022). Novel metrics to evaluate probabilistic remaining useful life prognostics with applications to turbofan engines. In *PHM Society European Conference*, Volume 7, pp. 96–109.

Fink, O., Q. Wang, M. Svensen, P. Dersin, W.-J. Lee, and M. Ducoffe (2020). Potential, challenges and future directions for deep learning in prognostics and health management applications. *Engineering Applications of Artificial Intelligence 92*, 103678.

Gal, Y. and Z. Ghahramani (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International conference on machine learning*, pp. 1050–1059. PMLR.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Lee, J. and M. Mitici (2023). Deep reinforcement learning for predictive aircraft maintenance using probabilistic remaining-useful-life prognostics. *Reliability Engineering & System Safety 230*, 108908.

Mitici, M., I. de Pater, A. Barros, and Z. Zeng (2023). Dynamic predictive maintenance for multiple components using data-driven probabilistic rul prognostics: the case of turbofan engines . *Reliability Engineering & System Safety*, 109199.

Nguyen, K. T. and K. Medjaher (2019). A new dynamic predictive maintenance framework using deep learning for failure prognostics. *Reliability Engineering & System Safety 188*, 251–262.

Saxena, A. and K. Goebel (2008). *Turbofan engine degradation simulation data set.* NASA Ames Prognostics Data Repository, NASA Ames Research Center Moffett Field, CA, USA.

Shi, Y., W. Zhu, Y. Xiang, and Q. Feng (2020). Condition-based maintenance optimization for multi-component systems subject to a system reliability requirement. *Reliability Engineering & System Safety 202*, 107042.