Online Supplement for

"A Bayesian Deep Learning Framework for Interval Estimation of Remaining Useful Life in Complex Systems by Incorporating General Degradation Characteristics" by Minhee Kim and Kaibo Liu

A. Sensitivity to the number of training systems

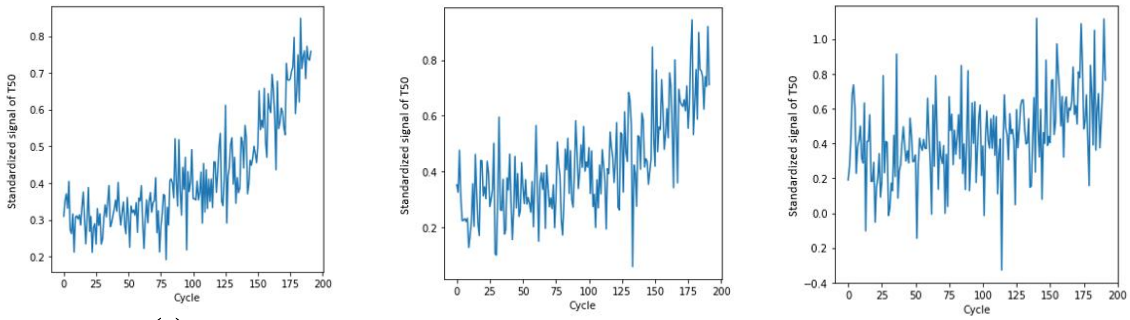
Table A shows the stability of the prognostic performance of the proposed method on sub-dataset FD001 according to the different number of training systems. The evaluations are repeated 10 times for each value of the number of training systems, and the detailed training settings follow Section 4.4.1 of the manuscript. We can see that as we have more training data, the prognostic performance of the proposed method gets more accurate and precise.

Table A. The mean and standard deviation (in parentheses) of prognostic results of the proposed method on FD001.

Number of training systems	40	60	80	100
Score	722.77 (45.82)	549.36 (32.46)	372.58 (23.14)	267.21 (14.78)
RMSE	23.32 (8.21)	17.43 (3.74)	15.49 (2.44)	12.42 (0.21)

B. Sensitivity to the sensor signal noise level

Here, we investigate how the noise and disturbance of the available data affect the stability of the proposed method. To do that, we introduce additional Gaussian random noise $\varepsilon_{\mathcal{N}} \sim N(0, \sigma_{\mathcal{N}}^2)$ to the standardized sensor signals of sub-dataset FD001 and apply the proposed method. Figure A shows examples of how $\varepsilon_{\mathcal{N}}$ affects standardized sensor signals. The evaluations are repeated 10 times for each value of the noise level $\sigma_{\mathcal{N}}^2$ and the detailed training settings follow Section 4.4.1 of the manuscript. Table B shows the prognostic results of the proposed method according to different noise levels. We can see



(a) (b) (c)

Figure A. Signals of T50 of Training Unit 1 in FD001 when the noise level $\sigma_{\mathcal{N}}^2$ is (a) 0, (b) 0.1, and (c) 0.2.

Table B. The mean and standard deviation (in parentheses) prognostic results of the proposed method on FD001 according to different $\sigma_{\mathcal{N}}^2$ values.

$\sigma_{\mathcal{N}}^2$	0	0.01	0.05	0.1	0.2
Score	267.21 (14.78)	287.61 (19.24)	294.60 (20.28)	306.94 (23.36)	1117.51 (61.93)
RMSE	12.19 (0.22)	13.31 (0.76)	13.78 (1.24)	14.45 (2.59)	18.71 (2.76)

that the smaller the noise level, the better the prognostic performance. Given that the C-MAPSS dataset itself (i.e., $\sigma_{\mathcal{N}}^2 = 0$) has a certain noise level already, the proposed method shows fairly good robustness against noise level.

C. Comparison of computational costs

Table C shows the average computation time of the proposed method and other benchmark methods over 10 trials on FD001 sub-dataset. The proposed method, Benchmarks (2) and (5) are tested using Intel Core i5-6300U CPU 2.40-GHz and 16-GP RAM. The Benchmark (4) is tested with Intel Core i7-3770 3.40-GHz CPU and 16-GB RAM. The training time of Benchmark (3) is excluded as the authors of the paper did not provide detailed settings to reproduce the results and calculate the computational costs. The table shows that the proposed method yields much lower training time than the existing deep learning approach (MODBNE) and comparable training time to Deep LSTM (Benchmark (2)). Although the proposed method takes longer training time than the existing parametric approach (HI), please note that this training procedure is carried out offline. In online, we can obtain the interval estimations of RUL which take around 0.1s seconds using two Intel(R) Xeon(R) CPU E5-4620 0 2.20GHz processors and 192 GB RAM and around 0.5s using Intel Core i5-6300U CPU 2.40-GHz and 16-GP RAM. To obtain the interval estimation, we repeat multiple stochastic forward passes (for R repetitions) to obtain R empirical

(Monte Carlo) samples. Owing to these samples being independent, computation time can be further reduced by using parallel computing if needed in practice.

Table C. The average model training time of the proposed method and other benchmark approaches on FD001.

	The Proposed Method	Deep LSTM (Benchmark (2))	MODBNE (Benchmark (4))	HI (Benchmark (5))
Average Model Training Time (s)	1014.75	961.47	1153760.67	0.85

D. Numerical study – Li-ion battery

In this section, we further apply the proposed method to predict the RULs of Li-ion batteries. In Section D.I, we provide an overview of the system and dataset. Section D.II demonstrates how we preprocess the raw degradation data and presents the prognostic results of the proposed method.

D.I. Overview of the system and dataset

In this dataset (Bole *et al.*, 2014), a set of four 18650 Li-ion batteries (RW1, RW2, RW7 and RW8) were continuously operated under a randomly generated sequence of charging and discharging profiles (also referred to as random walk discharging). Each system (battery) starts from a fully charged cell in a stationary condition. At each cycle time, three sensor signals are collected from a battery: voltage, current and temperature. Here, we consider the last signal measurement time of each battery as its failure time. More detailed experimental settings can be found in Bole et al., (2014).

D.II. Data preprocessing and results

We first conduct similar data preprocessing procedures to Section 4.2. Specifically, min-max normalization is applied to all three sensor signals, such that each degradation data is within the range of [0, 1]. Then, the sliding time window procedure is applied to the training systems to augment the training dataset. The hyper-parameters are optimized via 10-fold cross validation: $n_{TW} = 10$, the number of hidden layers of BDNN is set to 2, the number of hidden neurons per hidden layer of BDNN is set to 20.

Out of four batteries, the first three batteries (RW1, RW2 and RW7) are used to train the model while the fourth battery (RW8) is used to test the model. The RUL prediction results of RW8 are illustrated in Figure B. We can see that as the battery approaches the failure, the proposed method provides more accurate and tighter interval estimations of RULs.

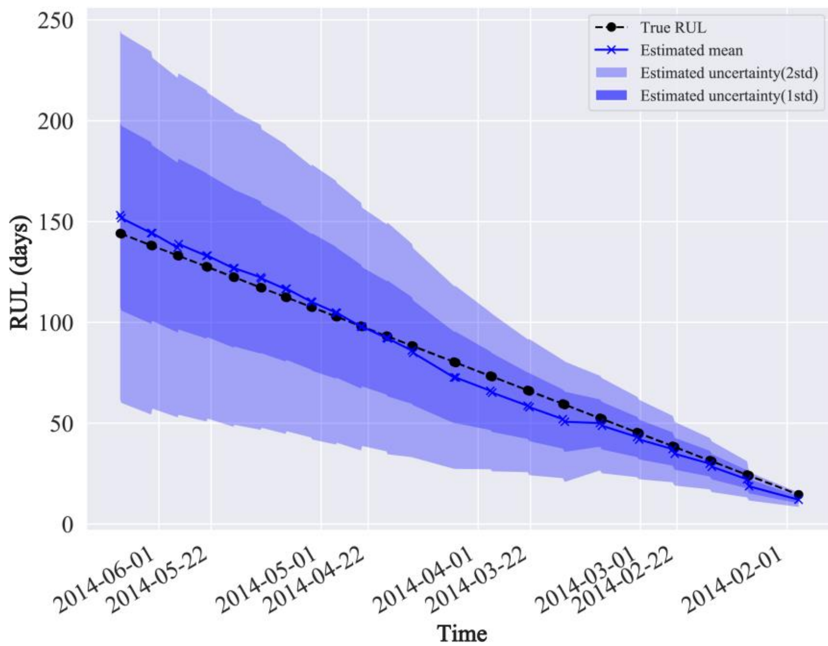


Figure B. Estimated RULs of RW8. The black dashed line represents the actual RULs of RW8. The X marker line shows the mean of estimated RULs using the proposed method. The shaded areas show the one and two standard deviations of the estimated RULs.

Bole, B., Kulkarni, C.S., and Daigle, M. (2014) Adaptation of an electrochemistry-based Li-ion battery model to account for deterioration observed under randomized use. In *PHM 2014 - Proceedings of the Annual Conference of the Prognostics and Health Management Society 2014*.