CIRP Journal of Manufacturing Science and Technology 37 (2022) xxx–xxx

Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

journal homepage: www.elsevier.com/locate/cirpj

Multistep prediction of dynamic uncertainty under limited data

Alex Grenyer[a,*], Oliver Schwabe[b], John A. Erkoyuncu[a], Yifan Zhao[a]

[a] Through-life Engineering Services Centre, Cranfield University, Cranfield MK43 0AL, UK
[b] Rolls-Royce Deutschland Ltd & Co KG, Hohemarkstraße 60-70, 61440 Oberursel, Germany

ARTICLE INFO

Keywords:
Forecast
Limited data
Long-short term memory (LSTM)
Multistep
Prediction
Spatial geometry
Uncertainty

ABSTRACT

Engineering systems are growing in complexity, requiring increasingly intelligent and flexible methods to account for and predict uncertainties in service. This paper presents a framework for dynamic uncertainty prediction under limited data (UPLD). Spatial geometry is incorporated with LSTM networks to enable real-time multistep prediction of quantitative and qualitative uncertainty over time. Validation is achieved through two case studies. Results demonstrate robust prediction of trends in limited and dynamic uncertainty data with parallel determination of geometric symmetry at each time unit. Future work is recommended to explore alternative network architectures suited to limited data scenarios.

© 2022

Introduction

The growing complexity in engineering systems manifests a range of uncertainty surrounding in-service maintenance. Such systems are comprised of various equipment units, many of which are maintained on a corrective or time-based basis. Unexpected failures outside planned maintenance periods require reactive maintenance to repair or replace units. Sampling rates of maintenance data in this context are often sporadic due to manual recording methods and disjointed signals from equipment units. The resulting quality and availability of data, as well as the influence of expert experience, assumptions, and environmental operating conditions, drive uncertainty that increases the likelihood of under or overestimating factors such as turnaround times, equipment availability and resulting costs [1–3]. This can lead to increased failure rates or, more often, unnecessary maintenance carried out. Uncertainty in this industrial context is defined as the difference between the degree of information required and information held to make a decision concerning a given entity. As well as deviations in quantitative, recorded data, this definition encompasses information sourced from qualitative, subjective opinions, assumptions and environmental factors. The resulting risk is the impact the uncertainty will have on the given entity [4–6]. Accommodating for uncertainty requires the determination of key contributors, their influence on interconnected units how this might change over time.

Limited available or poor-quality data directly hinders forecast accuracy and robustness. This paper focuses on time-series data. Once quantified, predictions of the uncertainty in such data and assumptions made surrounding it can enhance decision-making capabilities for the maintenance of increasingly complex systems and equipment units. This sets the motivation of the paper, which presents a framework to predict dynamic uncertainty under limited available data. The framework is designed to be embedded in a range of systems such as engines, radar, and heating systems as well as uncertainty in associated maintenance costs. The aim is not to mitigate or reduce the uncertainty, but to provide a holistic view as to which factors require mitigation or may become an issue in the future.

Supporting literature covering emerging techniques to forecast uncertainty for increasingly complex systems is given in Section “Supporting literature”. The framework structure is detailed in Section “Framework overview: Uncertainty prediction under limited data (UPLD)”, along with key mathematical formulae and assumptions made. Section “Framework implementation and results” applies the framework to two use cases: US SAR cost uncertainty data and simulated turbofan engine degradation. Results of each step are given to illustrate the multistep prediction for each long-short term memory (LSTM) network allocation. Section “Discussion” discusses the strengths and limitations of each step of the framework, while Section “Conclusions and future work” summarises the study along with future work in this area.

* Corresponding author.
E-mail address: a.h.grenyer@cranfield.ac.uk (A. Grenyer).

https://doi.org/10.1016/j.cirpj.2022.01.002
1755-5817/© 2022

Supporting literature

Uncertainty manifestation with growing complexity in engineering systems

Engineering systems are growing in complexity, consisting of numerous equipment units interacting in a linear or non-linear manner, with a collective behaviour distinct from and not often determined by that of individual units [7–10]. Applied in the engineering domain, these are termed complex engineering systems (CES). The level of uncertainty increases with growing complexity. Accurate depiction of turnaround times, equipment availability and resulting costs for optimum maintenance therefore become increasingly complicated and uncertain [9,11,12]. The level of uncertainty can change throughout the in-service life. This calls for adaptive and intelligent approaches to forecast uncertainty based on a combination of available data and expert opinion [10].

An overwhelming issue in the forecasting of equipment states and related maintenance is the quality and availability of data [12–16]. To make accurate and robust predictions, a degree of historic data is required. Where this does not exist, the solution is generally to model the physical system and obtain data through simulation. As systems grow in complexity, robust and dependable models are harder and more expensive to produce [12,17]. In the CES context, sampling rates are rarely consistent and feature a highly variable number of signals from different system units [15]. It then falls to subjective opinions and assumptions of experts—designers and manufacturers—to determine when maintenance will be required, equipment needed, time scales and resulting costs. This naturally places a large degree of uncertainty on the accuracy and robustness of such predictions, which must be quantified and considered when performing forecasts. A forecast is the determination of future outcomes based on historic and new data (Bayesian), while a prediction is an indication of a future event with or without prior information [18,19].

Multistep forecasting methodologies

Three prominent areas that have seen advancements in forecasting capability in recent years for uncertainty consideration are remaining useful life (RUL) prediction [12,17,20], cost estimation [2,21–24] and meteorology [25]. RUL prediction is a central task for maintenance practices of CES [26]. There are a myriad of RUL prediction approaches, notably reviewed by Lei et al. [12]. While the theory, general implementation and evaluation metrics of many approaches are open source, several industries have developed their own protected approaches for their specific requirements. Uncertainty is a significant point of vulnerability in long-term RUL predictions. Bayesian filtering algorithms are typically applied, the most prominent of which are Kalman filters; only suitable for linear systems and variations thereof for non-linear systems [12]. A more flexible algorithm is the particle filter, designed for use with non-linear systems, which has become a widely used method for performing real-time uncertainty assessment of RUL predictions [12,20,27,28].

Another key forecasting approach is deep learning, which makes use of neural networks (NNs) to learn from existing data. Applications are covered in detail for RUL prediction by Lei et al. [12] and for uncertainty forecasting by Grenyer et al. [10]. NNs are composed of multiple layers, allowing them to learn complex non-linear relationships. Bayesian deep learning (BDL) and variations thereof have been widely applied to forecast future events given existing data and update when presented with new data [29–37]. Deep learning models are only as accurate as the data they are trained on and, as such, typically require large datasets with defined trends over time [37]. They must therefore be flexible to consider all data properties necessary to achieve robust predictions. Flexible models can make better predictions, but all predictions involve assumptions that manifest uncertainty [10,29,37]. Naturally, the structure of NNs and training options applied have a significant impact on prediction accuracy and robustness for specific applications [12]. Determination of parameters that result in minimal prediction error can be achieved through hyperparameter tuning, often performed via a grid search technique [26,38]. Modification of deep learning models as more data becomes available can make implementation more complex and require extensive computation time [25,36,38].

This issue can be mitigated in part by dropout training, applied by Gal [36] in a method to approximate Bayesian inference in Gaussian processes (GPs) in deep neural networks and more generally by Cicuttin et al. [39] and Srivastava et al. [38]. GPs are highly flexible non-parametric models widely used for regression and classification, growing in complexity in line with the density of training data [29,37,40,41]. Defined as a layer within the network structure, dropout randomly sets input sequences below a defined probability to zero. This alters the underlying network structure for each iteration to prevent overfitting [36,39]. The uncertainty assessed by Gal [36] was in the deep learning process itself, not the resulting uncertainty interval It was highlighted that the use of distributions other than normal in Gaussian processes will result in different uncertainty estimates, the use of which may trade-off uncertainty quality with computational complexity [10,42]. These methods still require enough prior data of sufficient quantity and quality to fulfil the Central Limit Theorem, where the normalised sum of variables will tend towards a normal distribution [6,12,21,22,43].

Recurrent neural networks (RNNs) are a form of NN with a feedback loop to better capture non-linear relationships. Long short-term memory (LSTM) networks are a type of RNN increasingly used in sequential time-series forecasting and RUL prediction, the theory of which is widely covered in literature [17,26,44–46]. A key advantage of LSTMs over other types of NN is their ability to use gates to avoid vanishing or exploding gradients, increasing prediction accuracy [17,44,46]. Wu et al. [26] applied a vanilla LSTM model to predict RUL and identify physical degradation mechanisms, the parameters of which were defined through hyperparameter tuning. Shi and Chehade [17] proposed a Dual-LSTM framework to predict uncertain change points from which degradation accelerates and heath indexes that can be used to determine RUL in real-time. Both studies were compared with and found to outperform benchmark methods. A common trait among the examined publications is the use of a turbofan engine degradation dataset to test and demonstrate RUL prediction with proposed methodologies [17,26,45,47–49]. Further applications of this dataset are examined and ranked by Ramasso [50]. Different approaches to account for uncertainty in the datasets are also covered—the most popular being probability theory. The turbofan engine degradation dataset is also used in this study—detailed in Section “Case study 2: Turbofan engine degradation”.

Uncertainty in cost estimation is largely examined in the context of product-service systems (PSS) [2,10,13,21–24]. Multiple equipment uncertainties arise in this context including reliability, availability, and maintainability. Smart [19] applied Bayes’ Theorem to estimate costs from trends with minimal data points. Existing data and assumptions were combined with limited real-time data to produce accurate forecasts with a degree of confidence. Schwabe et al. [21] devised an approach focusing on the topology and symmetry of variance data given by its geometric shape at the time of estimation. This was initially driven by the idea that most statistical conclusions obtained arithmetically can also be achieved by geometry, which can additionally simplify otherwise complex conclusions [21,51]. Rather than interdependencies between individual

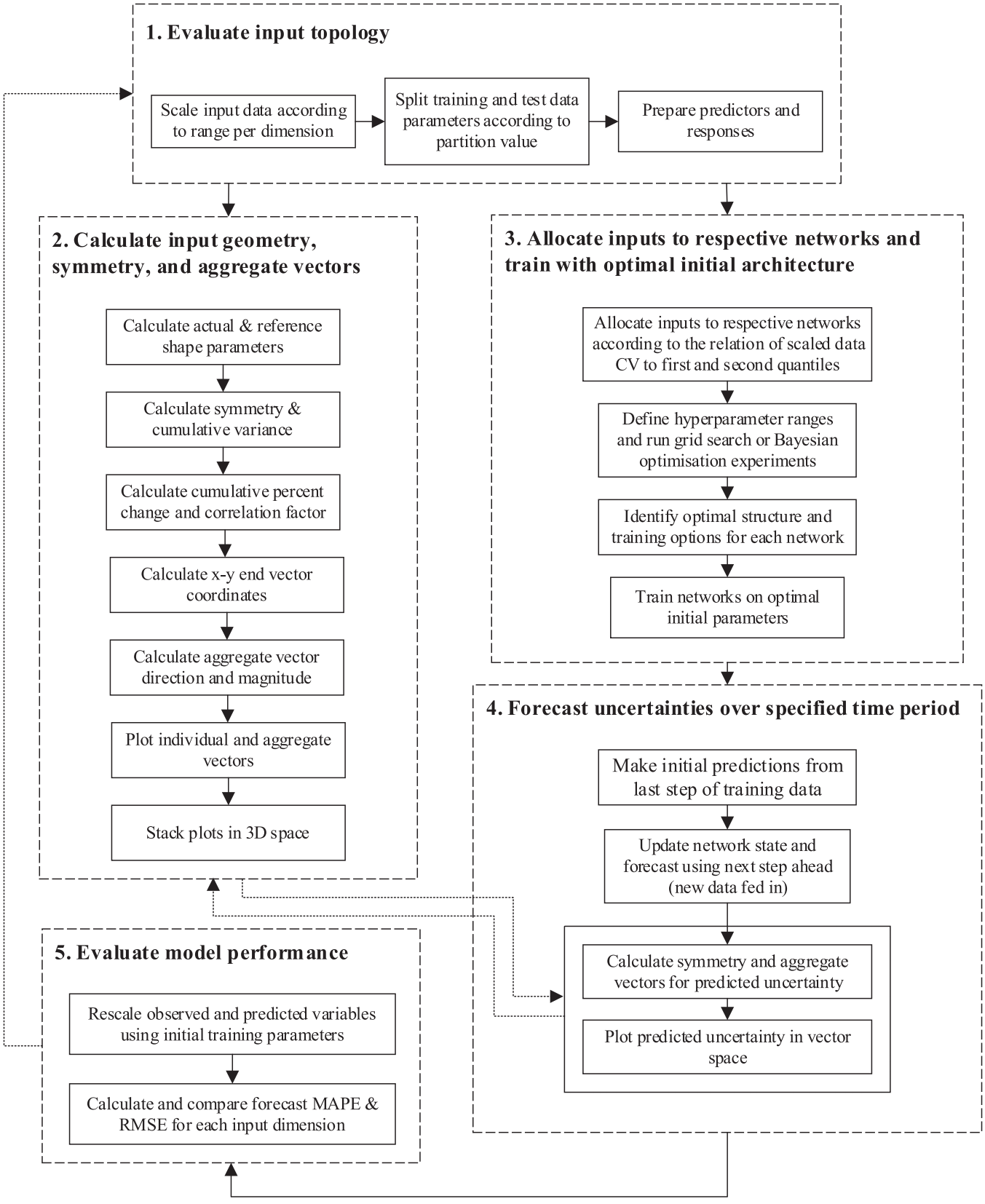

Fig. 1. UPLD framework overview.

data points, spatial geometry describes the behaviour of a space created by connecting outlying data points, represented in vector space in a point cloud around an origin, forming a regular cyclical polygon. A shape with greater symmetry requires less information to be described. A positive correlation was therefore hypothesised between symmetry and information entropy. The symmetry of the space created by each input dimension (cost variances) was able to predict future development without requiring significant volumes of data [21,22].

Research gaps

The quality and availability of data is the greatest driver of uncertainty in the forecasting of equipment states, RUL, and determination of when and how maintenance should be carried out. Growing complexity in engineering systems makes precise modelling of physical systems harder and more expensive to produce in order to obtain reliable simulated data [12,17]. Inconsistent data sampling from system units coupled with subjective assumptions made place a significant degree of uncertainty on predictions concerning the maintenance of such systems.

RUL prediction is a key determinate for maintenance scheduling and costing in CES [12,26]. Sensor-fed or manually recorded time series data is typically translated into a health index, the methods for which are dependent on the application [17,49,52,53]. Variants of RNN are widely used to predict RUL for CES, many using the C-MAPSS dataset. However, these methods still require sufficient data with which to train networks to make accurate and robust predictions. This also limits the robustness of probabilistic methods such as BDL [37]. Initial training parameters can be set by hyperparameter tuning to determine settings that result in minimal prediction error. This can require significant computation time when comparing multiple training options and network structures [46]. It is therefore not suitable for regular updates to the network when new data becomes available but can provide an effective starting point to make initial predictions. Dropout training can help improve the robustness of predictions by preventing overfitting for each input parameter, as well as updating the LSTM state at each prediction step as more data becomes available over time.

Uncertainty manifested under limited data and assumptions as discussed above should be forecast to allow decision-makers to plan with greater confidence. Doing so will prevent under or over estimation of factors such as equipment availability and cost, incorporating complex and non-complex factors. Uncertainties that may pose an undesirable risk at a given point in time can be mitigated to reduce the likelihood of unforeseen costs and delays [10]. Predictions need to be robust and as accurate as possible despite being produced under limited data where traditional probabilistic methods are not applicable.

The critical research gap is therefore summarised as a lack of approaches to predict uncertainty in engineering systems with complex and non-complex entities under limited data [10], and to do this without the need to produce complicated and expensive models of physical systems. LSTMs can make flexible forecasts based on prior data and update when new data becomes available. Spatial geometry offers a novel approach to predict uncertainty under limited data, though not yet applied outside of cost estimation. A combination of LSTMs with spatial geometry is seen as the best methodology to fill the research gap as it will enable robust uncertainty prediction under limited data, updating predictions as new data becomes available over time. Other approaches reviewed in this section are not suited to limited data scenarios as they rely on prior data sufficient to fulfil the Central Limit Theorem. The need to fill this gap is growing as data volumes and variability increase with system complexity.

Framework overview: uncertainty prediction under limited data (UPLD)

This paper contributes a framework for dynamic uncertainty prediction under limited data (UPLD). Spatial geometry is combined with LSTMs to enable covariant analysis of dynamic variables within state space, whereby a change in one variable will affect another. For each time step of the input sequence, the network learns to predict the value of the next time step. This work builds on a conceptual model presented in Grenyer et al. [42]. Here it is further developed and validated through two case studies. The steps were developed from emerging studies in literature utilising LSTM networks to forecast time-series data, extended to consider the geometric symmetry between input variances to improve prediction robustness under limited data. This addresses the research gap above and is achieved through a 5-step framework developed in MATLAB, described below and illustrated in Fig. 1.

Step 1. Evaluate input topology. To examine interactions, uncertainties and knock-on effects within the system, its topology must first be defined. Input uncertainty data is given as a time series of changing variance, formatted as row vectors where each column represents one time unit. The number of rows gives the number of input dimensions. The variance data is scaled according to the range of each input dimension i over each time slice j by Eq. (1), where n is the number of input dimensions. Under limited data a robust standard deviation cannot be applied, making traditional standardisation methods with mean and standard deviation undesirable [21].

$$dScaled_{i,j} = \frac{data_{i,j} - dMin_i}{dRange_i} \cdot \left(1 - \frac{1}{n}\right) + \frac{1}{n} \tag{1}$$

Where: $dScaled$ = scaled dataset; $data$ = initial dataset; $dMin$ = minimum value of each input over the time series; $dRange$ = range of each input over the time series.

The scaled variance data is split into training and test data according to a defined partition. The default partition is set to 60% to allow for a comparable proportion of observed and predicted values to determine prediction accuracy and robustness with varying input dimensions. The number of training steps is given by Eq. (2).

$$nStepsTrain = |P \cdot nStepsTotal| \tag{2}$$

Where: P = Partition; $nStepsTrain$ = Number of training steps; $nStepsTotal$ = Total number of time steps.

The input to be forecast is the scaled training data. The forecast data is then compared against the test data to determine prediction error. In parallel, the symmetry and aggregate vectors are calculated for the training, test and predicted data, detailed in the next step, and compared in the same manner. The LSTM networks then take the next time step, update the network state and corresponding prediction.

Step 2. Calculate input geometry, symmetry, and aggregate vectors. Spatial geometry determines an uncertainty range based on the geometric symmetry between input variances for each available time unit via polar force-field analysis in vector space. The procedures in this step to calculate symmetry and vector coordinates are based on previous work by Schwabe et al. [21]. Symmetry is defined as the relationship between the actual shape area of the evaluated time slice and the maximum possible area from the created geometry, illustrated in Fig. 2 by an example time slice with six input dimensions [21]. For each calculation in this step, the radial degree between each input vector and their input order (dimensional sequence) is kept constant [54].

Coordinate data points for vertices of the actual area shape are given by the scaled input variances for each time slice. The space between each vector dimension (D) out from the origin is a triangle (six in Fig. 2). The sum of each triangle's area gives the full actual

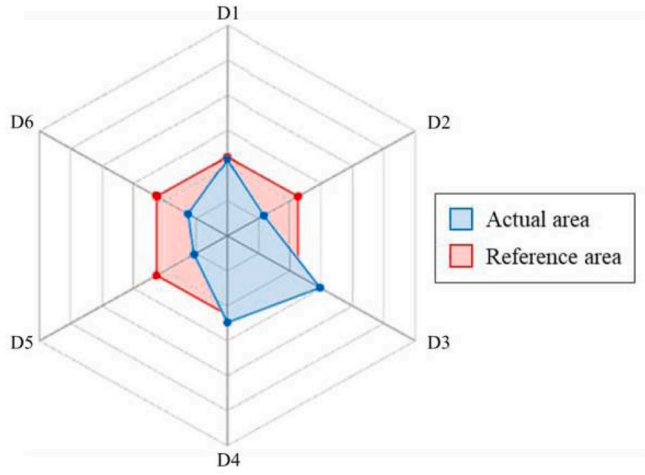

Fig. 2. Spatial geometry actual vs. reference shape area example.

shape area. This is calculated by Eq. (3), where a_i and b_i are the respective magnitudes of each vector that make up the triangle sides and *rad* is the radial degree. The sum of the outer face lengths then gives the shape perimeter, calculated by Eq. (4).

$$Area_{Act} = \sum_{i=1}^{n} \left(\frac{1}{2} \cdot a_i \cdot b_i \cdot \sin(rad) \right) \tag{3}$$

$$Perimeter_{Act} = \sum_{i=1}^{n} \sqrt{a_i^2 + b_i^2 - 2(a_i \cdot b_i) \cdot \cos(rad)} \tag{4}$$

The reference shape perimeter is calculated by the mean of the outer face lengths multiplied by the number of input dimensions. This creates a regular polygon, for which the apothem (line from centre to midpoint of each side) is given by Eq. (5). This is then used to calculate the reference shape vertex magnitude (Eq. (6)), which in turn is used to calculate the reference shape area (Eq. (7)). The symmetry between the actual and reference shape areas is then calculated by Eq. (8) [21]. Spatial geometry uses a ring topology, analysing the linear progression of symmetry. There is a positive correlation between the percent change in the cumulative increase of actual area and symmetry. The correlation factor for each time slice can be used to determine an uncertainty metric against the baseline estimate. This is the most likely or best guess value of the data point from which the input variance is obtained, explored further in Section "Framework implementation and results".

$$Apothem_{Ref} = \sqrt{FaceLength_{Ref}^2 - \left(\frac{FaceLength_{Ref}^2}{4} \right)} \tag{5}$$

$$Vertex_{Ref} = \sqrt{Apothem_{Ref}^2 + \left(\frac{FaceLength_{Ref}}{2} \right)^2} \tag{6}$$

$$Area_{Ref} = \frac{1}{2} \cdot n \cdot Vertex_{Ref}^2 \cdot sin\ (rad) \tag{7}$$

$$Symmetry = \frac{Area_{Act}}{Area_{Ref}} \tag{8}$$

To plot the change in shape geometry over time, X and Y end vector coordinates for each dimension, i over the time period, j are obtained by Eq. (9), iterated through each radial degree around the unit circle [42]. The sum of these points identifies the aggregate vector (Eq. (10)), whose magnitude is given by Eq. (11).

$$absEndX_{i,j} = cos(rad) \cdot dScaled_{i,j} absEndY_{i,j} = sin(rad) \cdot dScaled_{i,j} \tag{9}$$

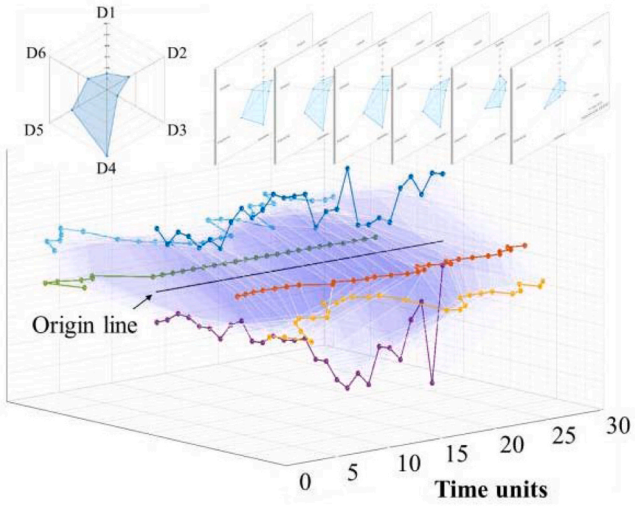

Fig. 3. Stacked plot example [42].

$$aggVectX_j = \sum_{i=1}^{n} absEndX_i aggVectY_j = \sum_{i=1}^{n} absEndY_i \tag{10}$$

$$aggVectMag_j = \sqrt{aggVectX_j^2 + aggVectY_j^2} \tag{11}$$

The aggregate vector magnitude and degree are assumed to represent the source of greatest uncertainty for each time slice. The resulting plots for each time unit are stacked to illustrate a dynamic change in the uncertainty of each input and aggregated vectors over time. An example illustration is given in Fig. 3 for six input dimensions, with aggregate vectors removed for illustrative purposes.

Step 3. Allocate inputs to respective networks and train with optimal initial architecture. While some inputs are relatively constant, others can vary significantly over the time series. The mix of dynamic and comparatively constant trends on a single network limits that network's ability to accurately and robustly forecast future time steps. To reduce under or over estimation, training is split across three networks with different architectures and initial training options. Different parameters are applied for different ranges of data according to the relation of the coefficient of variation (CV) of the scaled data to the first and second quantiles of each input dimension. CV is a dimensionless measure of relative variability, given by the ratio of the standard deviation to the mean [55,56]. This is illustrated in Fig. 4, with the networks hence referred to as "LSTM networks".

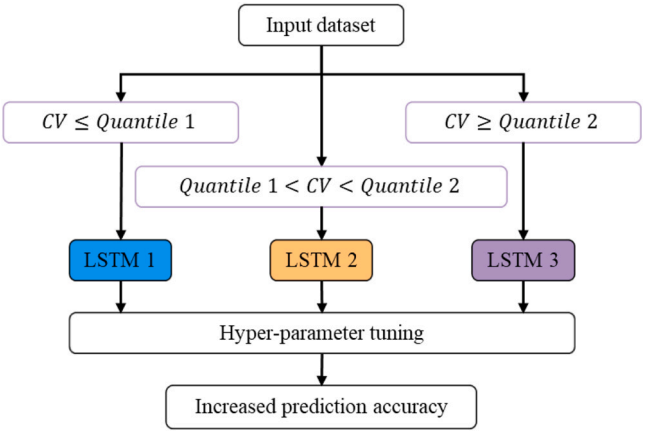

Fig. 4. LSTM network allocation according to input parameters.

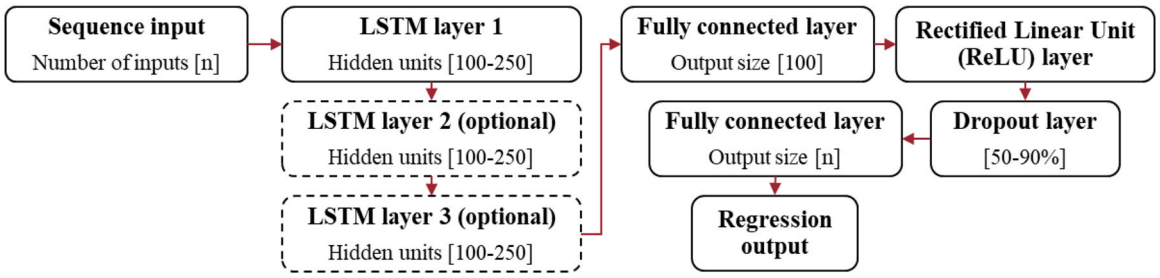

Fig. 5. Hyperparameter setup metrics: network structure.

Each network has a variable structure and range of training options to best suit the variability in the data applied to it. The best of these, i.e., the combination that gives the lowest prediction error, is determined through hyperparameter tuning according to the mean absolute percentage error (MAPE), discussed further in Step 5. As shown in Fig. 5, the variable structure for each network consists of 1–3 LSTM layers, each with 100–250 hidden units, a rectified linear unit (ReLU) layer, dropout layer (50–90%) and a regression output layer. Increasing the number of LSTM layers can make predictions more robust but also increases computation time [46]. The ReLU layer simply sets any value less than zero to zero, avoiding the vanishing gradient problem found in tanh and sigmoid functions [39,40]. The dropout layer then sets input sequences below a defined probability to zero to prevent overfitting [36,39]. The fully connected layers compile all neurons in the previous layer to a defined output size. The final regression output layer computes the half-mean-squared-error loss of the output.

The range of training options compared by hyperparameter tuning is denoted in Table 1. Three solvers are compared: Adaptive moment estimation (Adam), Stochastic gradient descent with momentum (SGDM) and root mean square propagation (RMS prop). Each of these are variations of gradient descent algorithms that update network parameters (weights and biases) to minimise prediction error by taking steps towards the negative gradient of the loss function [17,39,44]. The number of epochs is the number of full passes over the training data. The learning rate controls the changes made to the model for every epoch.

The optimal network structure and training options can be found by two methods: an exhaustive grid search, comparing every possible combination with set interval ranges for the parameters, or by Bayesian optimisation, where the software selectively alters a specified range of hyperparameters to minimise or maximise a selected evaluation metric. The three LSTM networks are then trained sequentially using the optimal hyperparameters.

Step 4. Forecast uncertainties over specified time period. The scaled variance data is forecast using the trained networks from the partition to the end of the initial dataset. Initial predictions are made using the last time step of the training response. When making predictions using standardised data (according to mean and deviation) the same training data parameters are used for the test (observed) data [17,42]. Therefore, this approach uses the same range and minimum parameters from the training data to compare observed data against predicted data. The corresponding symmetry and aggregate vectors are calculated in parallel by Step 2 and compared in the same manner. The network state is updated to use observed values at each step in place of the predicted values to increase robustness [42]. The observed and predicted data is then plotted in vector space, stacked for each time slice.

Table 1
Hyperparameter setup metrics: training options.

Training options	Value range
Solver	Adam, SGDM, RMS prop
Max. Epochs	150–250
Initial learn rate	0.005–0.01
Learn rate drop factor	0.1–0.5

Step 5. Evaluate model performance. Prediction error between the observed O_i and predicted P_i uncertainty can be attributed to the model parameters, unexpected changes in the inputs (causing no clear trend) and the amount of data available. Prior to performance evaluation, all variables are rescaled up to their original values by rearranging Eq. (1), using the initial range and min. parameters, given by Eq. (12). Common evaluation metrics are root-mean-square error (RMSE) (Eq. (13)), mean absolute percentage error (MAPE) (Eq. (14)) and custom score functions. RMSE is widely used in RUL prediction and regression problems. MAPE is a widely applied evaluation metric to determine forecast accuracy and robustness, providing a distinct percentage evaluation.

$$data_{i,j} = \frac{dScaled_{i,j} - \frac{1}{n}}{\left(1 - \frac{1}{n}\right)} \cdot dRange_i + dMin_i \tag{12}$$

$$RSME = \sqrt{\frac{\sum_{i=1}^{n}(P_i - O_i)^2}{n}} \tag{13}$$

$$MAPE = \frac{1}{n} \cdot \sum_{i=1}^{n} \left| \frac{O_i - P_i}{O_i} \right| \tag{14}$$

As new data is recorded, the framework loops back to Step 1 to reassess the input topology and feeds through to performance evaluation.

Framework implementation and results

Case study 1: US SAR data

The initial spatial geometry approach utilised US Department of Defense Air Force Selected Acquisition Report (SAR) summary tables to test and validate the method [21,54]. This is made up of a mixture of summary cost data over various phases of product life cycles in aerospace, land and sea sectors. Cost variances used were considered significant enough to require monitoring by stakeholders [21]. The same dataset is applied here to provide comparable consistency in the application and demonstrate the wide applicability of the framework.

Step 1. Evaluate input topology. Annual cost variances in US $ Mil are given over the life cycle of a range of US Air Force military platforms for a 28-year period from 1986 to 2013. Further detail is given by Schwabe et al. [21,54]. The data is categorised into 6 cost variance factors and formatted as absolute integers as [42]:

Quantity: Change in the number of units of an end item of equipment.

Schedule: Change in procurement or delivery schedule, completion date, development, or production milestone.

Engineering: Alterations to physical or functional characteristics of a system.

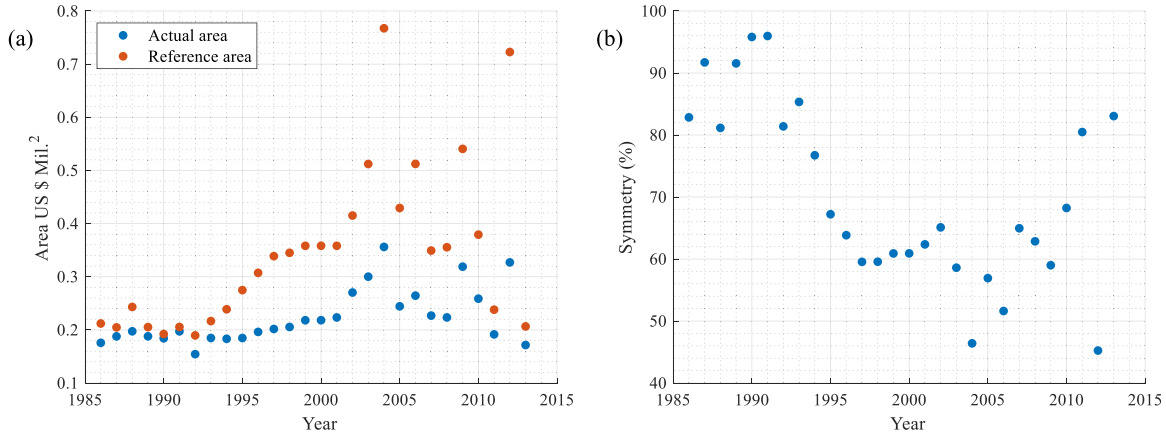

Fig. 6. SAR data: (a) Change in actual and reference shape area over time and (b) change in symmetry.

Estimating: Correction of previous estimating errors or refinements of current estimates.

Other: Unforeseeable events not covered in any other category (e.g., natural disaster or strike).

Support: Cost changes for support equipment of major hardware items not included in other costs.

Step 2. Calculate input geometry, symmetry, and vector coordinates. Following Eqs. (2)–(8), the resulting change in the actual area, reference area and symmetry over time is illustrated in Fig. 6. Initial observations can be made here to highlight the reduction in symmetry through to 2005, indicating an increase in the amount of information required to describe the shape. From here, the symmetry fluctuates up to the end of the observed period.

The cumulative increase of actual area and symmetry gave a linear trend for the observed period. The percent change for this increase between each unit is plotted in Fig. 7a. This displays a negative exponential trend with a correlation coefficient of 0.95 (Fig. 7b).

The gradient and intercept values from the actual area and symmetry trend line equations were plugged in for 100 time units. Their correlation factor, given by the *actualArea*/*symmetry* is illustrated in Fig. 8 with an R^2 value of 1. The shaded area shows the region of available data given by the 28 time units.

An interesting phenomenon occurs when taking a linear trend line from Fig. 7(a) and calculating the correlation factor in the same manner (Fig. 9) where a lognormal relationship is displayed. The asymptote where Y = 1 appears to meet the x-axis just prior to

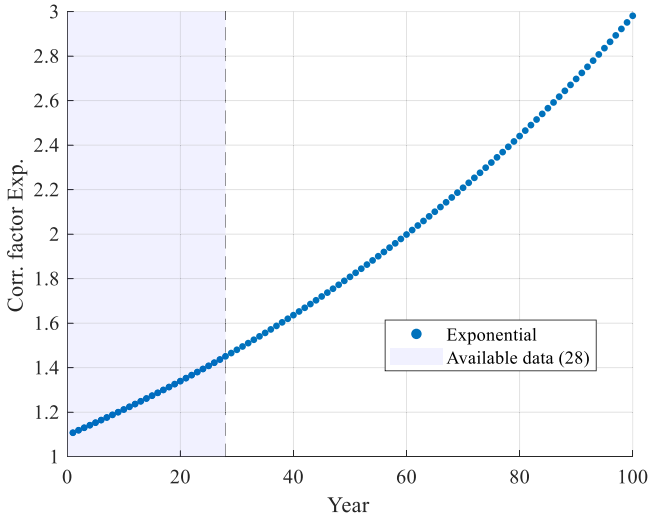

Fig. 8. SAR data: Actual vs. symmetry correlation factor for exponential trend.

where the available data ends. The reasons for this warrant further investigation but are out of scope for this study.

Next, the X and Y end vector coordinates are calculated for each input dimension over the 28-year time period. The resulting endpoints and aggregated vectors for each input dimension are stacked and plotted in Fig. 10. The dynamic shape area is shown in Fig. 10a by the white lines and blue fill between each vector coordinate. The

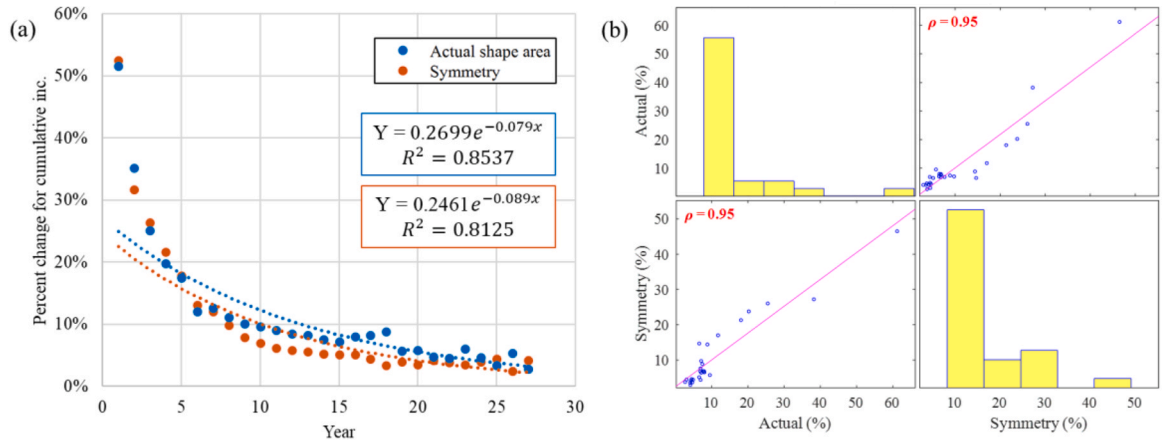

Fig. 7. SAR data: (a) Percent change for cumulative increase and (b) correlation matrix.

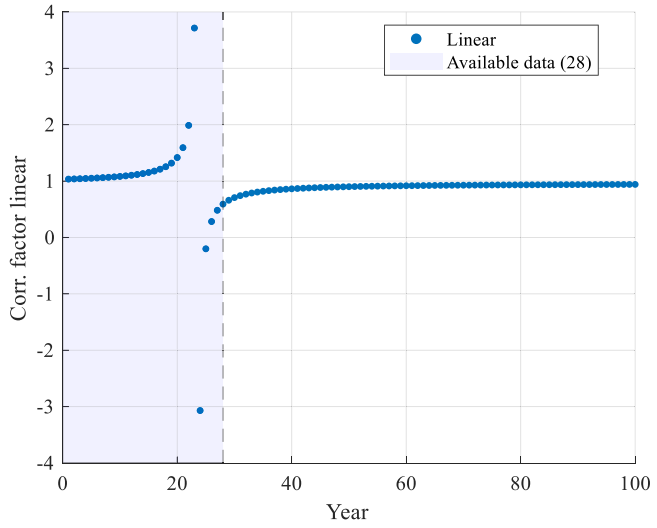

Fig. 9. SAR data: Actual vs. symmetry correlation factor if assuming linear trend in cumulative % increase.

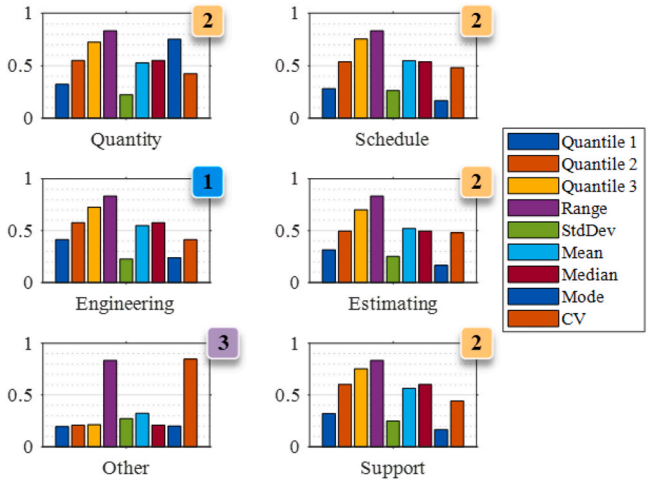

Fig. 11. SAR data: Summary statistics for each input and corresponding LSTM network allocation.

aggregate vector magnitude in Fig. 10b is visualised by the end marker, scaled up 40x for illustration. It can be seen here that the estimating factor prompts the greatest variance over the analysed period. It should also be noted that the radial degree between each input dimension is kept constant – in this case 1.0472 radians (60°). The apparent difference between e.g., Quantity-Schedule and Schedule-Engineering is due to the scaling of the figure produced in MATLAB.

Step 3. Allocate inputs to respective networks and train with optimal initial architecture. The range and deviation of each input used to train the network varies significantly. Using the CV as the deterministic parameter allows inputs with higher variation to be trained separately from those with lower deviation. Summary statistics are illustrated in Fig. 11 and categorised into the relevant LSTM networks according to Fig. 4.

For the training data, this fits the Engineering factor into the first network, Quantity, Schedule, Estimating and Support into the second and Other into the third. Hyperparameters were filtered through Bayesian optimisation to minimise the resulting average MAPE between inputs for the respective networks, using defined ranges (e.g., 100–250 hidden units). A maximum sweep time of 2 h was set, which gave approximately 200 runs. The resulting initial network structure is illustrated in Fig. 12. The optimal initial network training options are given in Table 2. For all three networks, the learning rate schedule was set to 'Piecewise' and gradient threshold to 1.

Step 4. Forecast uncertainties over specified time period. Initial predictions made using the trained networks are shown in Fig. 13a, where the solid lines are the training data for each input dimension (partitioned at 60% of the full dataset), the dashed lines are the predictions, and the thin dotted lines are the actual (observed) data for the test period. After resetting the network state, predictions were updated for each time step accounting for the previous step in Fig. 13b. Following the updated prediction, it was observed that values for the Estimating factor now lie within the observed range. More detail in the prediction error can be seen in Fig. 15.

The corresponding symmetry, aggregate vectors and stacked vector plot built in Step 2 were calculated and updated in Table 3 and Fig. 14. The percentage difference between the observed and predicted symmetry and aggregate vector magnitudes over the test period is directly influenced by the prediction error for each input dimension. Significant percentage difference in the aggregate vector magnitude for years 25 and 26 is due to the change in shape area, illustrated in the stacked 3D plot. Initial training data was removed for the illustration. Due to the significant difference in variance

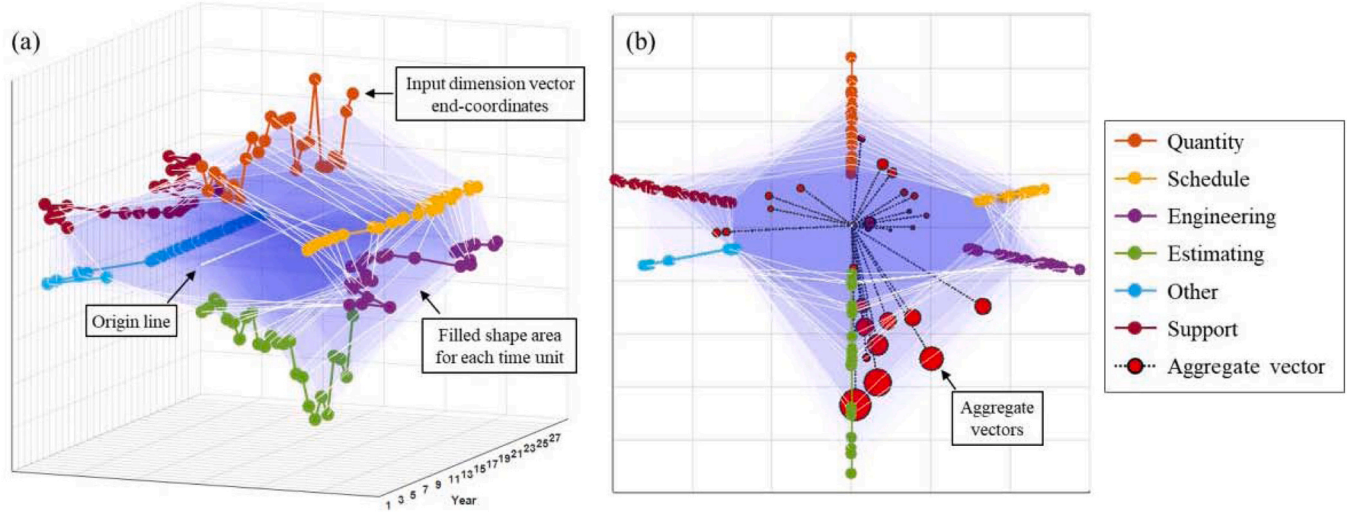

Fig. 10. SAR data: (a) Stacked vector 3D plot and (b) face-on with aggregated vectors over 28-year period.

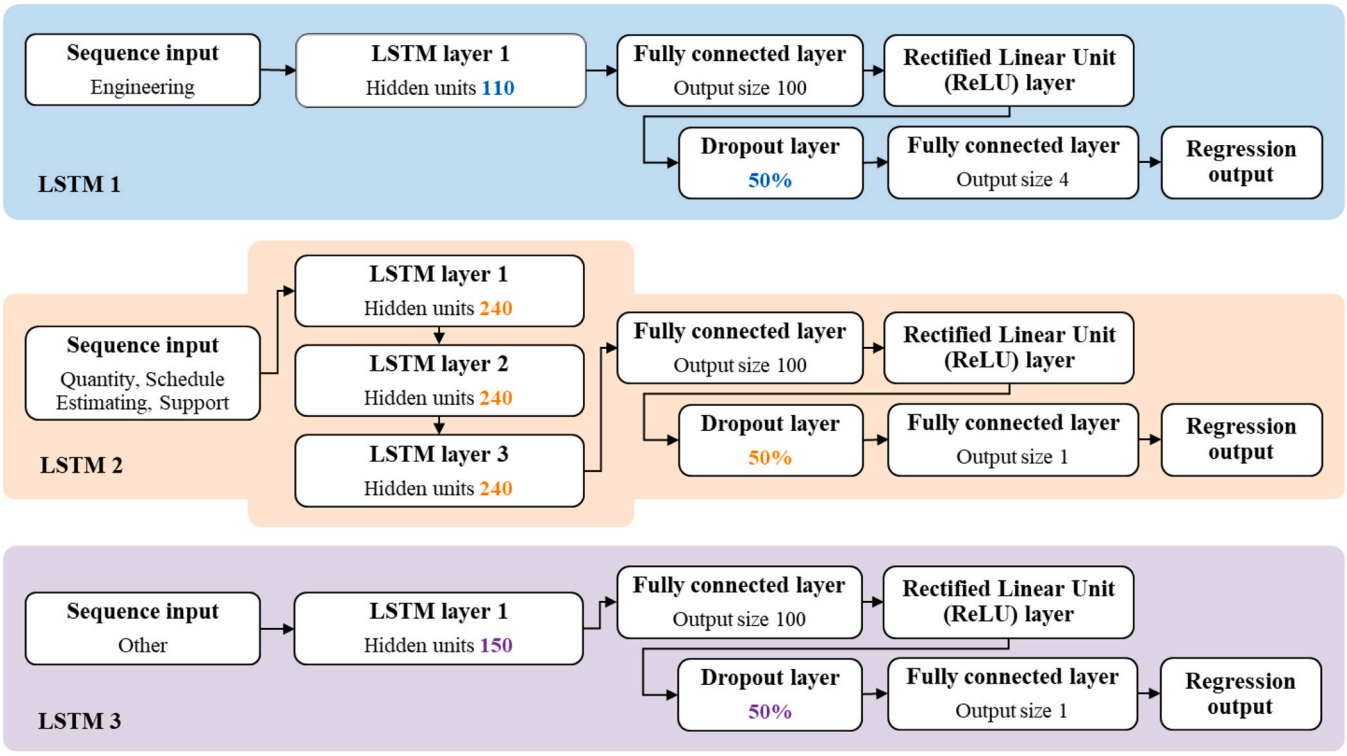

Fig. 12. SAR data: LSTM network input allocation and structure following hyperparameter tuning.

Table 2
SAR data: defined training options following hyperparameter tuning.

Training options	LSTM 1	LSTM 2	LSTM 3
Solver	SGDM	Adam	Adam
Max. Epochs	235	200	130
Initial Learn Rate	0.003	0.002	0.002
Learn rate drop factor	0.14	0.11	0.10

magnitude of the Estimating factor to all other factors, the aggregate vector direction is relatively unchanged other than in years 25 and 26. Further evaluation is made in the final step.

Step 5. Evaluate model performance. The difference in the observed and predicted data is illustrated in Fig. 15, scaled back up to the original variances, with corresponding line and stem plots for each input dimension. The stem plots show the difference in the observed to predicted data. Prediction error is noticeably variable over the time period for each of the six input dimensions. This is due to the quantity of data on which the networks were trained and the unpredictable peaks and troughs in the observed data. For example, the Schedule factor (Fig. 15b) is underestimated but the overall downward trend is picked up in the prediction. The observed data of the Engineering and Estimating factors (Fig. 15c,d) is scaled to 1.0×10^4 US $Mil. The relatively constant variance from year 20 to year 23 is accurately predicted, but the sudden increase was not predictable in the training data. As the model updated the multistep prediction, the increasing trend was identified up to year 28. Similarly, the Estimating factor (Fig. 15d) was able to predict the overall downward trend in the test data period but not the sudden changes in variance.

The MAPE and RMSE are calculated in Table 4. The lowest MAPE was observed in the Engineering factor due to low prediction error in year 20 to year 23 and the following positive trend. While the estimating factor appears to hold the trend of the observed data, the scale of the variance means it has the highest MAPE and RMSE. The

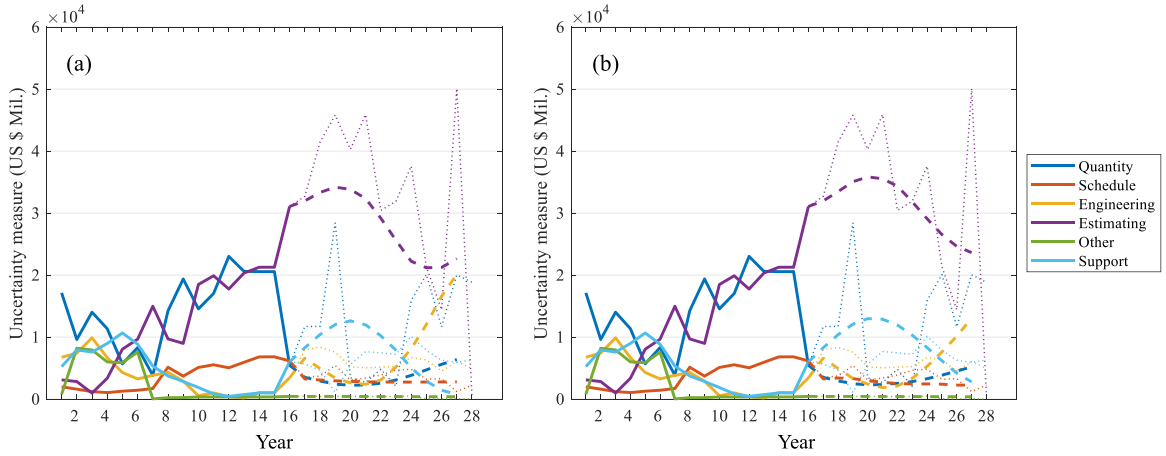

Fig. 13. SAR data: Observed vs. predicted uncertainty for (a) initial forecast and (b) updated forecast.

Table 3
SAR data: observed vs. predicted symmetry and aggregate vectors.

	Symmetry				Aggregate vector magnitude			
Time	Observed	Predicted	Diff.	% Diff.	Observed	Predicted	Diff.	% Diff.
18	81.44	87.71	6.27	7%	0.51	0.65	0.14	24%
19	76.35	71.30	5.05	7%	0.34	0.65	0.31	62%
20	64.47	60.25	4.23	7%	0.59	0.66	0.07	11%
21	53.23	60.66	7.43	13%	0.67	0.65	0.03	4%
22	62.47	65.30	2.82	4%	0.38	0.61	0.23	47%
23	55.55	74.49	18.94	29%	0.45	0.55	0.10	21%
24	71.55	88.60	17.05	21%	0.30	0.49	0.20	50%
25	76.11	97.51	21.40	25%	0.03	0.48	0.45	177%
26	80.37	80.68	0.31	0%	0.02	0.55	0.53	185%
27	54.18	42.64	11.54	24%	0.49	0.66	0.17	29%
28	72.53	27.33	45.20	91%	0.34	0.77	0.43	78%

Other factor holds constant up to year 23 before an unexpected dive, which the network was not able to account for in the prediction. These sudden changes and the scale in the observed variance data directly impact the mean prediction error, causing the high variation in MAPE and RMSE over the test period. While the predictions cannot be considered accurate, the ability to reflect the observed trends despite outliers in the observed data allows predictions to be deemed robust [57].

Case study 2: turbofan engine degradation

As discussed in Section "Supporting literature", a number of studies have applied a turbofan engine degradation dataset to forecast RUL using LSTMs, as well as other areas of prognostics and health management (PHM) [17,26,45,48,50,58]. Simulated using the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) tool, this publicly available dataset consists of four degradation scenarios. The FD001 training set was selected for this study because it consists of a range of quantitative data measured by sensors and qualitative factors given as noise. Uncertainties in the data and assumptions made were calculated by dividing the data into sub-arrays. The resulting uncertainty data over 16 time cycles was applied to the framework to further demonstrate the capability to predict uncertainty under limited data.

Step 1. Evaluate input topology. The initial dataset consisted of 21 sensors measuring temperature, pressure and speed for 100 engine units, each with a random start time and normal operating level, running to failure. Previous studies using this dataset discounted any uniform sensor data as these will not change in any forecasts made or contribute to the uncertainty. The same approach is applied here, as well as discounting parameters whose individual uncertainty has a minimal impact on the aggregated uncertainty or overall forecast. A description of the resulting 10 input dimensions forecast is given in Table 5. As for the SAR data case study, a partition of 60% was applied to split the training and test data, which was then scaled according to the range of the training data. The C-MAPSS dataset does consist of defined training and test sensor data for RUL prediction. The focus of this study is to forecast the uncertainty of that data, where there is limited previous data on which to base predictions. Using the training set from the database was therefore deemed sufficient.

Step 2. Calculate input geometry, symmetry, and vector coordinates. The actual and reference shape areas and resulting symmetry over time are illustrated in Fig. 16, given by Eqs. (2)–(8). For the 16 cycles observed, a trend cannot be identified for the shape areas or symmetry. As for the SAR data, the percent change for the cumulative increase in actual area and symmetry shows a negative exponential trend (Fig. 17a) with a highly significant correlation coefficient of 0.97 (Fig. 17b).

Following the line equations given by the actual area and symmetry, the correlation factor over 100 units given by the *actualArea*/*symmetry* is illustrated in Fig. 18 with an R^2 value of 1. The same asymptote trait occurred as for the SAR data when calculating the correlation factor assuming a linear trend line, occurring where

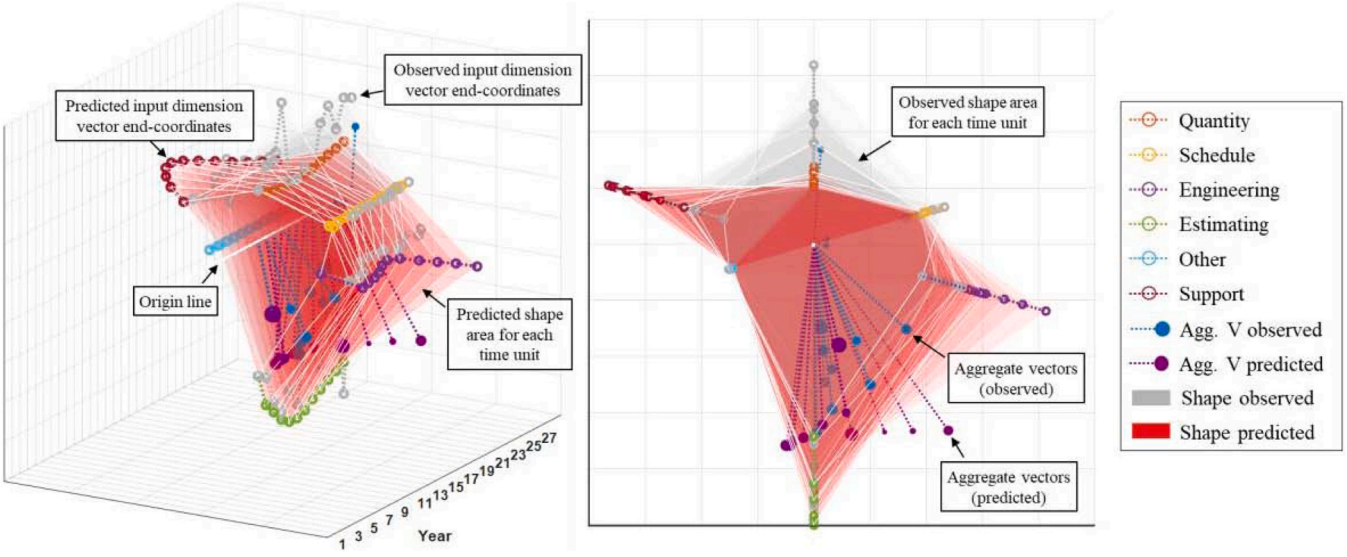

Fig. 14. SAR data: Stacked 3D vector plot including observed and predicted data.

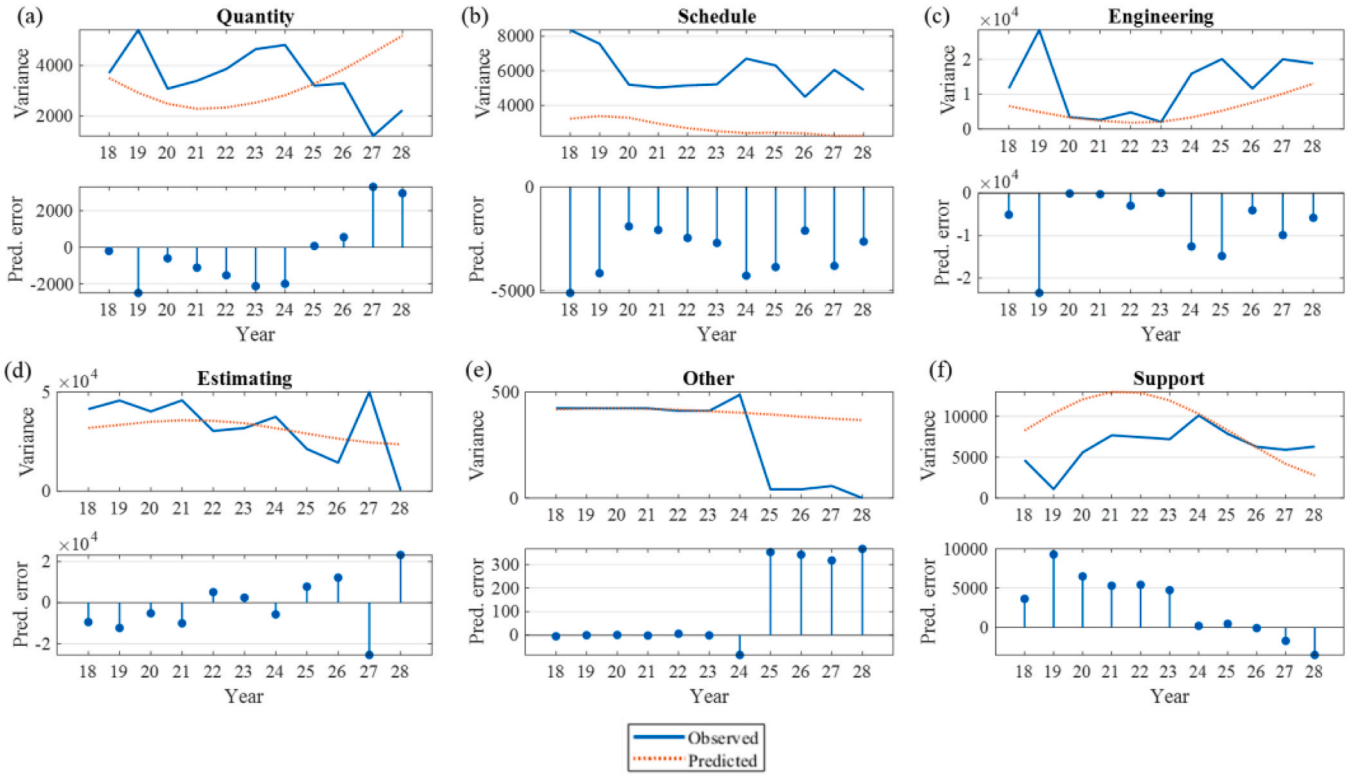

Fig. 15. SAR data: Observed vs. predicted variance over the test period for each input dimension (a–f).

Table 4
SAR data: MAPE and RMSE of observed vs. predicted values over the test period.

Input	MAPE	RMSE
Quantity	140%	2003
Schedule	92%	3834
Engineering	64%	8363
Estimating	867%	17,110
Other	381%	208
Support	248%	4200

Table 5
C-MAPSS data: description of input dimensions to be forecast [58].

Sensor number	Notation	Description
3	T30	Total temperature at HPC inlet
4	T50	Total temperature at LPT inlet
7	P30	Total pressure at HPC outlet
9	Nc	Physical core speed
11	Ps30	Static pressure at HPC outlet
12	Phi	Ratio of fuel flow to Ps30
14	NRc	Corrected core speed
15	BPR	Bypass ratio
20	W31	HPT coolant bleed
–	–	Process noise

the available data ends. The 16 time units signify uncertainty in the data up to engine failure. This plot is purely illustrative to expand the decreasing correlation factor. A key difference here to the SAR data is the opposite (negative) trend. As the variation and corresponding uncertainty in the sensor data increases up to failure, the relation of the geometric shape area to its symmetry reduces.

The coordinate endpoints and aggregated vectors over the 16 time cycles for each input dimension are stacked and plotted in Fig. 19. The shape area is starkly different here compared to the SAR data. This is due not only to the four additional input dimensions but also the contrast in variability between the dimensions about the radial degree (0.63 radians, 36°). The aggregate vector magnitude in Fig. 10b is scaled up 10x for illustration and tend towards the low-pressure turbine inlet temperature (S4).

Step 3. Allocate inputs to respective networks and train with optimal initial architecture. The variation in the uncertainty data is illustrated in the summary statistics in Fig. 20. Categorised by Fig. 4 according to the respective CV, the majority of dimensions fell into LSTM network 3. The turbine core speed (S9) was placed in LSTM 1 and the high-pressure turbine coolant bleed (S20) in LSTM 2. The hyperparameter ranges applied to identify the optimal initial network structure are the same limits as for the SAR data, trained using Bayesian optimisation for the same maximum of 2 h. The resulting structure is summarised by Fig. 21 and training options in Table 6. For all three networks, the learning rate schedule was set to Piecewise and gradient threshold to 1.

Step 4. Forecast uncertainties over specified time period. Initial predictions made using the trained networks are shown in Fig. 22a. As for the SAR data, the solid lines are the training data, dashed lines are the predictions and thin dotted lines are the observed data for the test period. Fig. 22b shows the predictions after updating for each time step to account for the previous step. Changes in the predicted values are examined in Step 5. While not immediately noticeable in the plots, a reduction is noted in S3 (blue) from time cycle 12, a reduction in the negative gradient in S20 (orange) and a constant period in S7 between cycles 13 and 14.

The corresponding symmetry and aggregate vectors from Step 2 are compared in Table 7 for each time step and through the stacked vector plot in Fig. 23. The large difference in symmetry for time cycles 13 and 14 is due to the exploding gradients within the networks, which a higher dropout percentage could avoid. However, when tested at 80% rather than 50% the networks were found to give a constant line as the prediction. Further testing of combinations and degrees of dropout layers may alleviate the errors within the LSTM.

The stacked vector plot demonstrates this further (initial training data was removed for the illustration). Notable differences in the observed and predicted data can be seen where the grey shape area

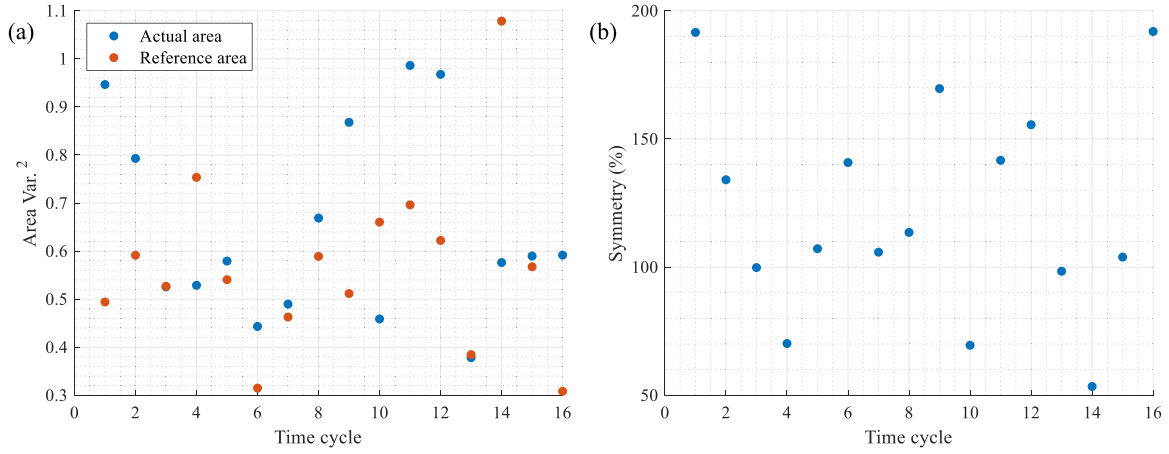

Fig. 16. C-MAPSS data: (a) Change in actual and reference shape area over time and (b) change in symmetry.

(observed) is not covered by the red area (predicted). Caused mainly by S4, S9 and S14, these errors alter the resulting symmetry and corresponding aggregate vector magnitude and direction towards different factors.

Step 5. Evaluate model performance. The difference in the re-scaled observed and predicted data is illustrated in Fig. 24 in the line and stem plots for each input dimension. The significant prediction error for a number of factors is most likely due to the very limited number of steps on which it was trained and the lack of defined trends. Development of the network allocation methodology or inclusion of additional networks to train further variabilities in the data may improve robustness in the prediction. As for the previous case study, the multistep prediction was not able to pick up sudden changes in the variance data. Variances considered here are magnitudes smaller than those used in the previous case study and propagated over a smaller time period.

The tracing of overall positive or negative trends in the test data where they are apparent, such as for S7 (Fig. 24c), S14 (g) and S15 (h), and predicting within the range boundaries of the observed data is therefore considered a satisfactory result. This case study illustrates the pitfalls of making predictions based on very limited data. The MAPE and RMSE over the observed period for each input dimension are given in Table 8. The range and scale of variances are comparatively small against the SAR data in Section "Case study 1: US SAR data", which is where the MAPE demonstrates the ability to better compare the prediction errors. Large variation in prediction error is due to unpredictable changes in the data. This naturally drives up the prediction error, as seen in time cycle 14 in S4 (Fig. 24b). Even if the remainder of the test period has a very low prediction error, that increased error will increase the overall MAPE. As for Section "Case study 1: US SAR data", predictions are robust as they reflect observed trends despite outliers and limited data on which to train [57].

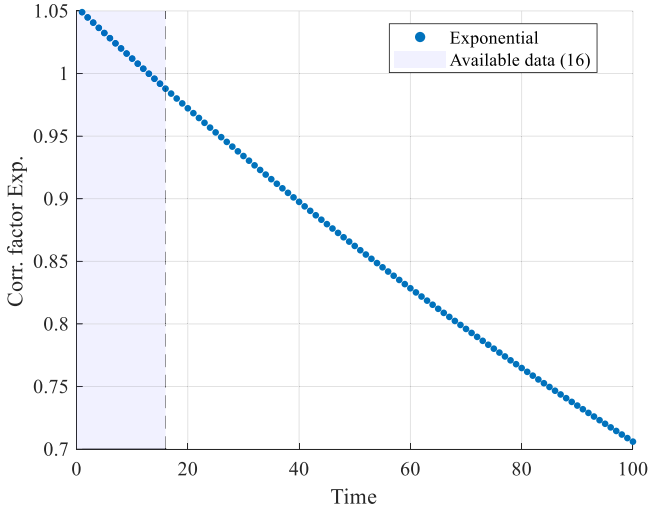

Fig. 18. C-MAPSS data: Actual vs. symmetry correlation factor for exponential trend.

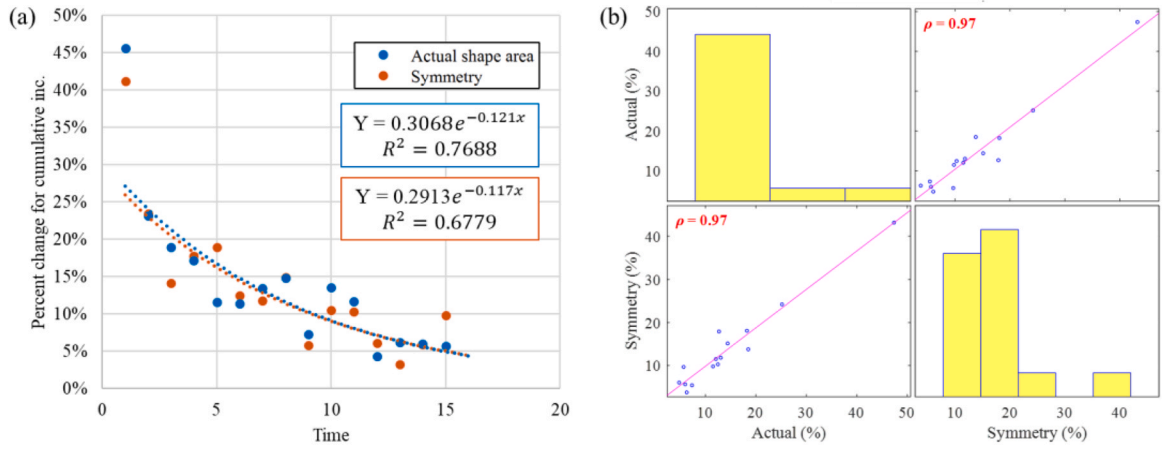

Fig. 17. C-MAPSS data: (a) Percent change for cumulative increase and (b) correlation matrix.

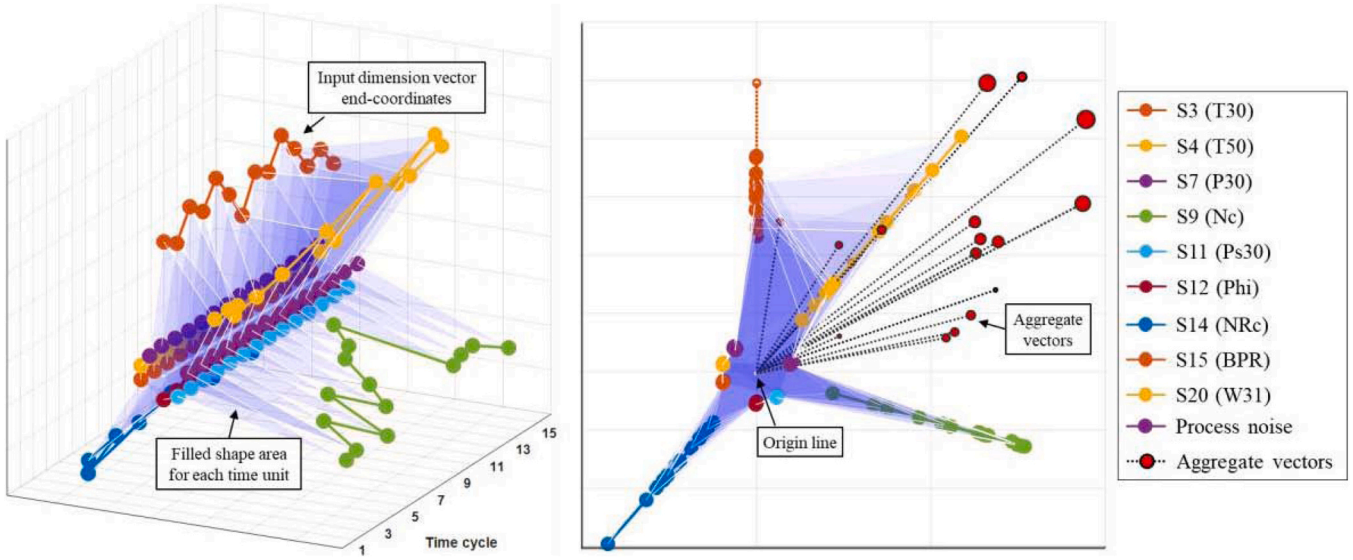

Fig. 19. C-MAPSS data: Stacked vector 3D plot and face-on with aggregated vectors over 16 time units.

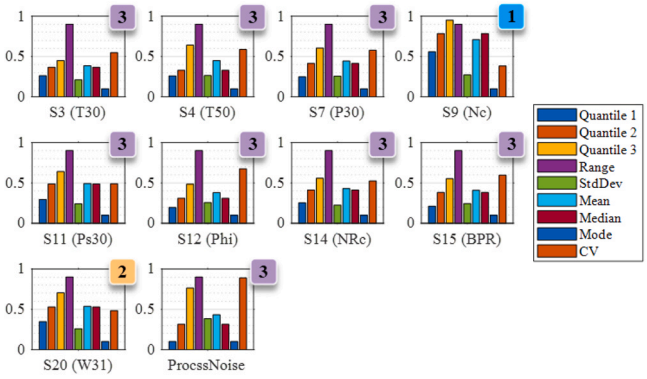

Fig. 20. C-MAPSS data: Summary statistics for each input and corresponding LSTM network allocation.

Discussion

The following paragraphs critique the framework steps through the results of the two case studies, concluding with an examination of industrial applications. Input uncertainty data for both case studies were given as a time series of variances, formatted as row vectors. The use of case studies in distinct domains demonstrated the framework's flexibility to be embedded in different systems. The data was scaled according to the range of each input dimension. As stated in Section "Framework overview: Uncertainty prediction under limited data (UPLD)", the number of time steps available under limited data is unlikely to provide a robust deviation measure required for traditional standardisation methods. The scaling equation (Eq. (1)) can theoretically be applied to any format of data such as standard deviation or raw sensor data. The useability and results of using such formats in the framework have not been explored and may warrant further investigation.

The scaled data was split into training and test data according to a defined partition, set to 60% to provide a comparable proportion of observed and predicted values to determine robustness of predictions. A lower partition would reduce the amount of data on which to train the networks, leading to reduced robustness, while an increased partition would reduce the data on which to test and update the networks and make predictions beyond the available time period. Comparisons with a varying partition would be beneficial for cases with a larger forecastable period, though up to a point the available data may no longer be considered "limited" and more traditional statistical approaches can come into play. Even when working with "big data", parametrics and statistics must be treated with caution prior to validation when significant correlations are unknown.

Spatial geometry was used as the uncertainty descriptor because of its ability to propagate interdependent cost uncertainties under limited data. Connecting outlying data points in vector space formed geometric shapes for each time slice, the area of which was used to determine the symmetrical relationship between inputs. This enabled a simplification of what may otherwise be complex conclusions [21,51]. The greater the symmetry, the greater the information entropy and therefore representative uncertainty for a given time slice. Symmetry and respective vector coordinates were calculated in Step 2 of the framework and run in parallel with Step 4 as uncertainties were predicted through the LSTM. The aggregated vectors for each time slice illustrated the greatest source of uncertainty and gave an indication of shape change for the next time interval, stacked in a point cloud in 3D space.

The visualisation provided an immersive view of shape change over time as well as the source of greatest uncertainty via the aggregated vector. These uncertainties require the most attention; be it mitigation, exploitation or simply increased awareness [42]. Employment of the 'shape' of data through spatial geometry for forecasting against the correlation of individual data points is a significant novelty in the developing field of big data analysis. Live and continuous forecasts are beneficial to industry in several areas including maintenance planning and digital twins in the face of mounting increases in technological complexity.

To calculate the symmetry and aggregate vectors and build the 3D visualisation, defined parameters had to be fixed while others were allowed to change over time. Summarised in Table 9, key fixed parameters were the radial degree and dimensional sequence of inputs, while changeable parameters included the vector coordinates of each dimension over time. Altering the sequence of input dimensions would change the magnitude and direction of the aggregate vector but should maintain the shape area. Adding a new input dimension part-way into the space will alter the radial degree and require rescaling of the full dataset. Repercussions and allowances for altering fixed parameters warrant further research for spatial geometry.

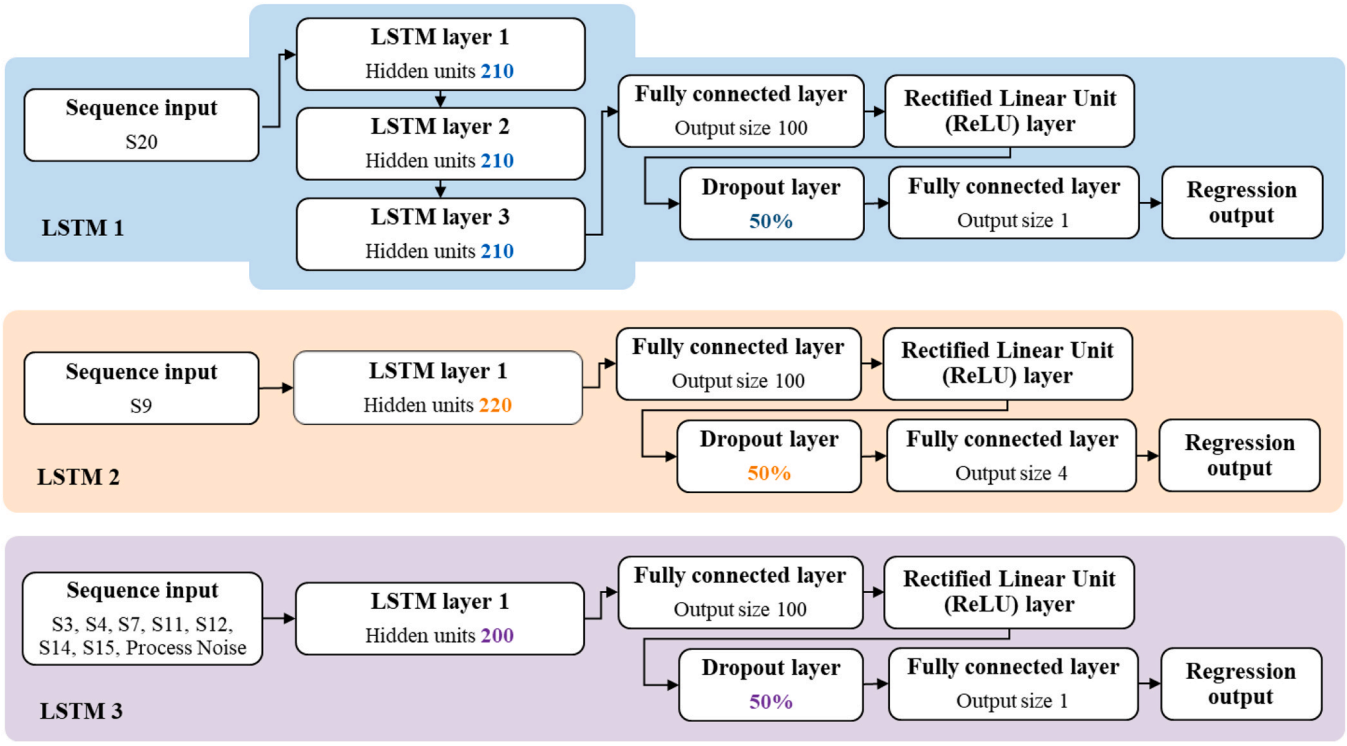

Fig. 21. C-MAPSS data: LSTM network input allocation and structure following hyperparameter tuning.

Table 6
C-MAPSS data: defined training options following hyperparameter tuning.

Training options	LSTM 1	LSTM 2	LSTM 3
Solver	Adam	Adam	Adam
Max. Epochs	220	180	120
Initial Learn Rate	0.009	0.009	0.021
Learn rate drop factor	0.23	0.061	0.138

The third step allocated the scaled inputs to one of three networks according to their coefficient of variation (CV) over the time period, then used hyperparameter tuning to define the optimal initial network structure and training options to yield robust predictions. The CV was used as the deterministic measure for network allocation because it provides a dimensionless measure of relative variability.

Alternative measures such as the mean are affected by outliers, while the mode and standard deviation are not suitable for small sets of data. Other methods to define bins in which to allocate input dimensions should be explored, such as interquartile range, and variable allocation methods based on the amount of available data and respective variability. The allocation of input dimensions to the three networks for UPLD has a significant effect on the robustness of resulting predictions, making this one of the most important steps of the framework.

The allocation approach is similar to the semi-double-loop learning concept proposed by Putnik et al. [59]. This was used to select the best learning models for predictive maintenance scenarios. This method could prove effective in further development of the UPLD framework, where the application of double-loop learning principles in reinforcement learning would be used to help allocate input parameters and define initial network architecture.

Initial parameters for the adaptable network architecture were defined through hyperparameter tuning. For both case studies, Bayesian optimisation was used to minimise the MAPE by comparing

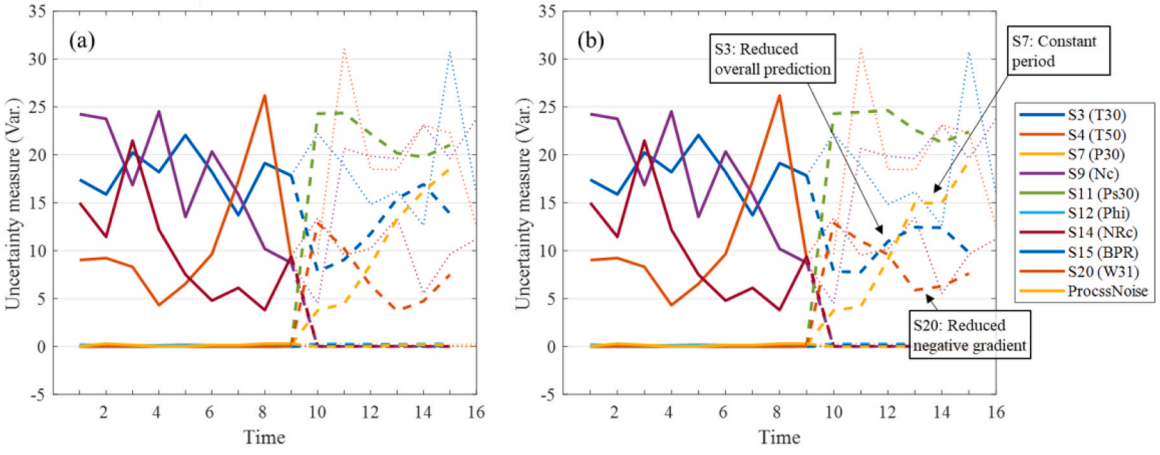

Fig. 22. C-MAPSS data: Observed vs. predicted uncertainty for (a) initial forecast and (b) updated forecast.

Table 7
C-MAPSS data: observed vs. predicted symmetry and aggregate vector magnitude.

	Symmetry				Aggregate vector magnitude			
Time	Observed	Predicted	Diff.	% Diff.	Observed	Predicted	Diff.	% Diff.
11	141.67	149.23	7.56	5%	1.28	0.33	0.95	118%
12	155.55	196.87	41.32	23%	0.83	0.40	0.42	69%
13	98.38	165.01	66.63	51%	0.75	0.56	0.19	29%
14	53.42	124.97	71.56	80%	1.10	0.78	0.32	34%
15	103.93	101.93	2.01	2%	1.27	0.87	0.40	37%
16	191.93	149.66	42.27	25%	0.74	0.86	0.12	15%

a defined range of parameters. Experiments were run for a maximum of 2 h for each network. This computation time is not viable for regular updates when new data becomes available so was only used to gain an optimal initial setup. This does not necessarily give the best possible initial setup as not every combination can be tested with the time frame. An exhaustive grid search comparing each parameter iteration would not be viable without extensive computing power.

The defined ranges for training options and network structure are detailed in Section "Framework overview: Uncertainty prediction under limited data (UPLD)". Different combinations will generate different results. Additional LSTM layers will typically improve prediction accuracy but take longer to train. The hidden units of each LSTM layer are equal. Comparisons of different sizes in each layer and additional training options such as mini-batch size may improve results with reduced computation time. The final output layer of each network was a regression layer. Regression typically relies on statistical data sufficient to fulfil the Central Limit Theorem. Under limited data scenarios, this is not the case without artificial propagation through Monte Carlo simulation [42,43,60]. This may therefore lead to reduced robustness in predictions based on the available training data. Alternative, custom output layers should be explored to provide more robust predictions.

Uncertainty was predicted in Step 4 using the trained networks. The symmetry and individual and aggregate vectors were calculated for the predicted data via Step 2 running in parallel with the LSTMs. It is important to note the distinction of the forecast direction where the LSTMs forecast through the time axis, while the symmetry and aggregate vectors are calculated for each predicted time unit. Predictions were rescaled to the original input variances and plotted to illustrate the difference in the observed and forecast uncertainty. The dropout layer prevented overfitting for each input parameter, but when set too high resulted in near-constant predicted values. The multistep model updated predictions as new data was fed in. The initial, observed and predicted data were plotted in the stacked 3D vector space. These plots provided an immersive view of the shape area through time as well the aggregate vector magnitude and direction. However, the visualisation can become chaotic when too many parameters are visualised at once. Further developments will therefore allow selected parameters to be visualised and removed, as well as value labels and altering the shape area fill transparency.

The fifth and final step of the framework evaluated model performance via the MAPE and RMSE. Other evaluation metrics such as custom scoring functions should be developed to gauge the quality of uncertainty prediction and develop a methodology to identify areas where more data is required to allow comprehensive decisions to be made concerning equipment availability, turnaround time and unforeseen costs through the system life cycle.

The pertinence of the framework was discussed with key personnel from a leading defence company in 4 h of semi-structured interviews. A large degree of uncertainty is portioned to numerous data repositories, maintenance formats and failure modes for different platforms. Sampling rates of maintenance data from different systems can have unpredictable gaps and varying sampling rates. The quality of signal reconstruction and determination of operational defects is used to determine when maintenance will be required. Rates of degradation or identification of other failure modes it is not always achievable. It was agreed that continuous forecasting of uncertainties resulting from these traits is vital to facilitate dependable maintenance costing and ensure equipment availability.

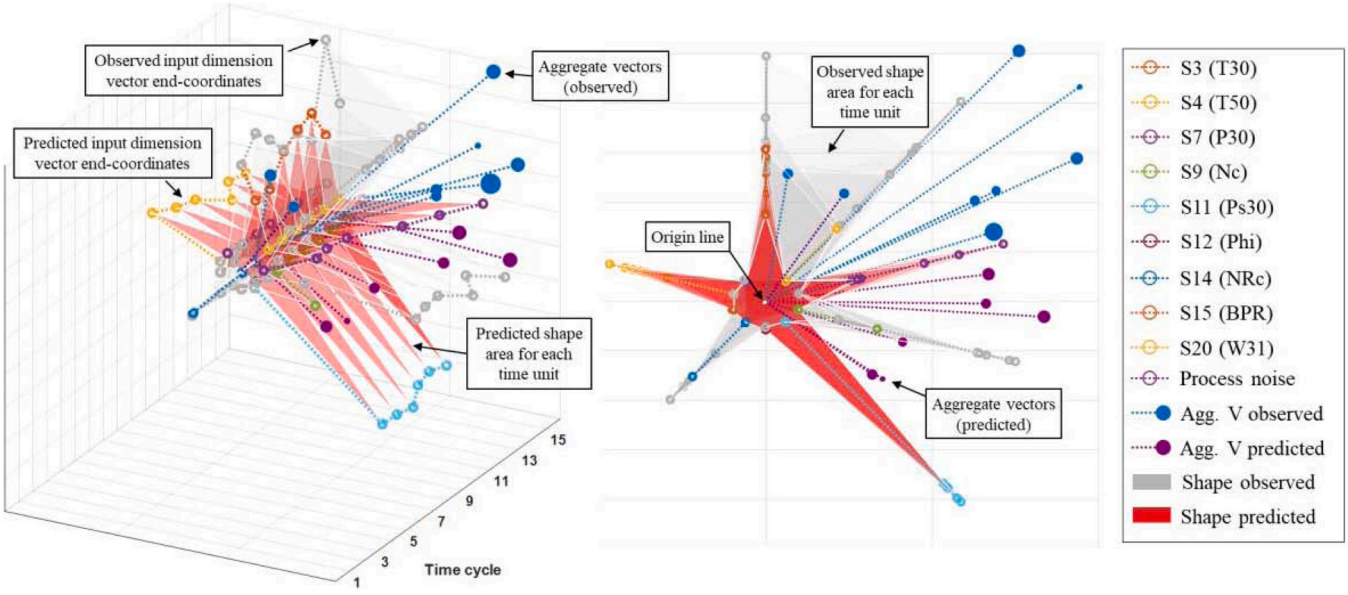

Fig. 23. C-MAPSS data: Stacked 3D vector plot for observed and predicted data.

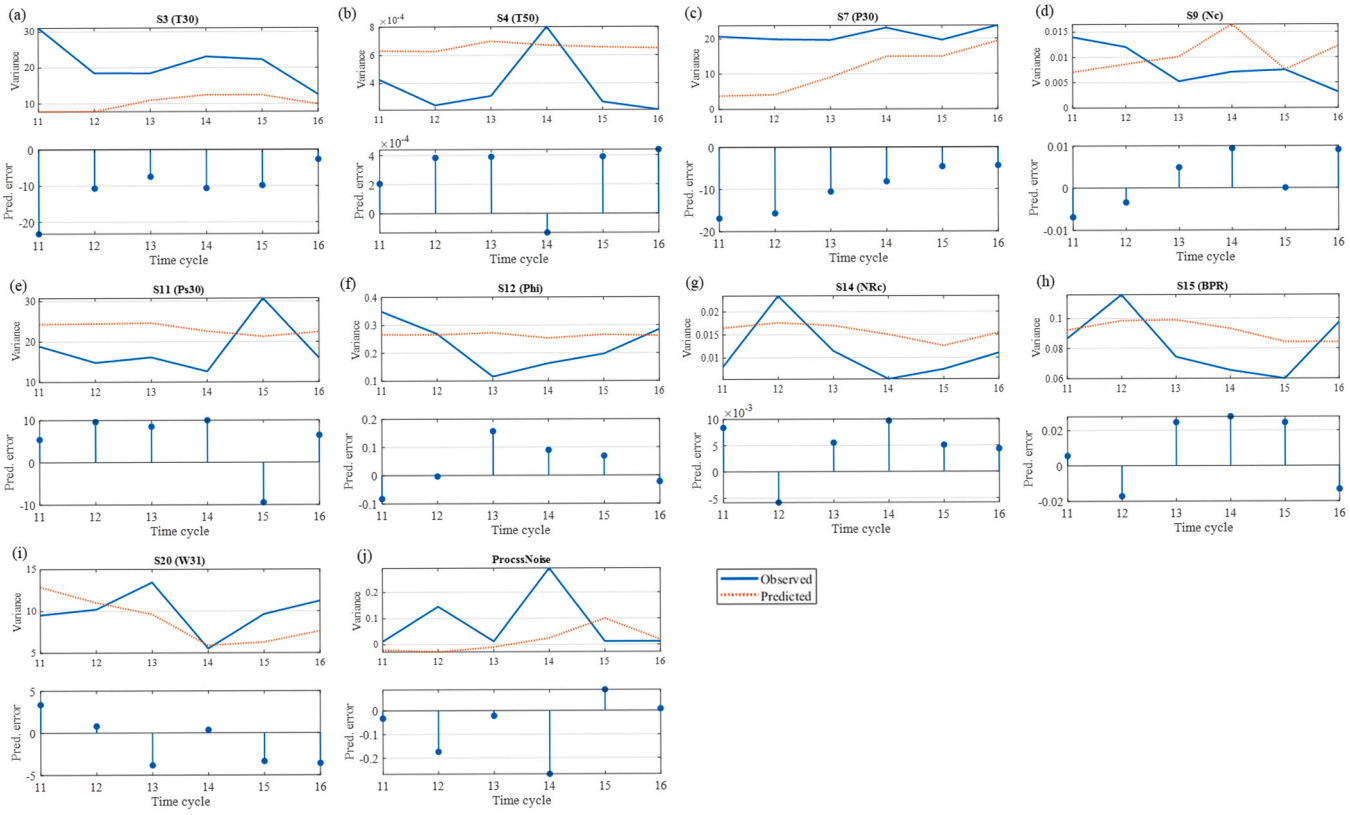

Fig. 24. C-MAPSS data: Observed vs. predicted variance over test period for each input dimension (a–j).

Table 8
C-MAPSS data: MAPE and RMSE of observed vs. predicted values over test period.

Input	MAPE	RMSE
S3 (T30)	23%	12.52
S4 (T50)	70%	0.00
S7 (P30)	30%	2.43
S9 (Nc)	46%	0.01
S11 (Ps30)	26%	7.25
S12 (Phi)	20%	0.09
S14 (NRc)	40%	0.01
S15 (BPR)	16%	0.02
S20 (W31)	20%	3.30
ProcssNoise	314%	0.12

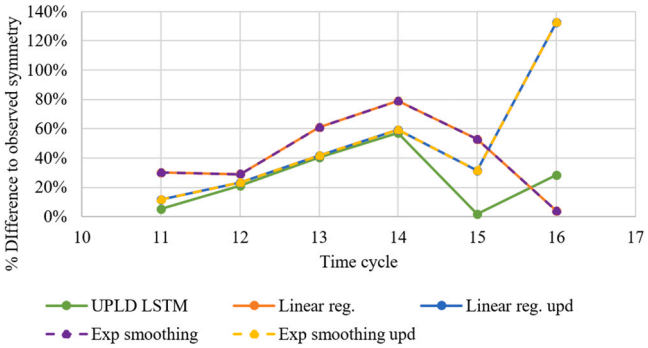

Fig. 25. Forecast method comparison – percentage difference of observed and predicted symmetry.

Further work towards implementation is discussed at the end of the next section.

A direct comparison of the UPLD framework with traditional, probabilistic forecasting methods such as regression is not suitable because they are designed for large volumes of data that fulfil the Central Limit Theorem (as stated above and in Section "Research gaps"). Such models aim to forecast statistical data, not the uncertainty in that data and surrounding qualitative factors. They are therefore not appropriate under limited data scenarios.

A comparison of the percentage difference in symmetry given by predicted variables over the test period of case study 2 against that observed is therefore plotted in Fig. 25. The results from Tables 7 and 8 using UPLD are plotted against predictions made by linear regression and exponential smoothing, using the training data from case study 2. Predictions were also updated (upd) as in the UPLD framework by including the data of the previous time step for each iteration. The symmetry for the resulting predictions was calculated according to Step 2 of the UPLD framework to provide comparative data. The percentage difference given by the UPLD LSTM gives the lowest

Table 9
Spatial geometry taxonomy for fixed and changeable parameters.

Fixed parameters	Translation space	Changeable parameters (over time)
• Scaling equation for all input dimensions • Shape area calculation • Symmetry calculation • Radial degree between inputs • Dimensional sequence of inputs • Origin location	• Layout/plotting functions for visualisation • Computational complexity	• Input dimension vector coordinates • Shape area • Symmetry • Aggregate vector direction and magnitude • Forecast most likely variance

percentage difference to the observed data, thus outperforming the other methods.

Conclusions and future work

This paper presented a framework to predict dynamic uncertainty exhibited under limited data (UPLD) for the maintenance of increasingly complex engineering systems. These uncertainties arise as a result of data quality and availability, operating conditions and assumptions made surrounding maintenance. Coded in MATLAB, the framework was designed to be embedded in a variety of systems, building on supporting literature to develop a flexible forecasting model capable of making predictions under limited data from complex and non-complex factors without the need to develop precise models of physical systems. LSTMs were applied in parallel with spatial geometry to predict uncertainty in time-series data through the geometric symmetry between input dimensions. Additional benefits include the ability to update uncertainty predictions as new data becomes available by comparing initial predictions against the observed data and projecting forecasts through the visualisation of polar force fields in 3D vector space. This allows factors that may require future mitigation to be identified, which can, in turn, reduce under or over estimation of turnaround times, equipment availability and resulting costs.

The framework was applied to two case studies in different contexts: Annual cost variances for a range of US Air Force military platforms (SAR data) [21,54] and precalculated uncertainties from a turbofan engine degradation simulation (C-MAPSS data) [50,58]. The SAR data consisted of six input dimensions with a widespread of variances over a 28-year period.

The C-MAPSS data consisted of 10 dimensions made up of sensor data and process noise, with variances precalculated for 16 time cycles, determined initially from raw sensor data for RUL prediction [58]. Section “Supporting literature” highlighted the wide use of LSTMs for RUL prediction, for which many studies use the C-MAPSS dataset. While the study does not use the dataset directly, findings on uncertainty in sensor and noise data will impact the determination of the acceptable range on which decisions are made when planning maintenance for related systems.

Key findings of this research were:

Employment of the ‘shape’ of data to describe uncertainty by the geometric symmetry between inputs for each point in time provided discernible information to determine and predict equipment health under limited data.

Allocation of inputs to one of three networks according to their variation enabled improved definition of initial network architecture and more robust predictions.

As technological complexity grows, live and continuous forecasts of uncertainty manifested by data quality and availability are of great benefit to industry.

Core contributions of the UPLD framework are:

1. Robust prediction of uncertainty under limited data
2. Adaptable allocation of inputs to networks with variable structure and training options
3. Initial technique for immersive visualisation of dynamic uncertainties and shape change with an indication of magnitude and direction of the greatest contributing factor

The authors propose future work to simulate and interpolate input data to fill gaps in signal data. Implications of altering fixed parameters within spatial geometry merit further research. The impact of changing the dimensional sequence of inputs (input order around the origin) for each time unit to a variable rather than a constant parameter is being investigated. Prediction robustness from the LSTM networks may be improved by exploring alternatives to the regression output layer. alternative evaluation metrics such as custom scoring functions should be developed to gauge prediction quality and identify where additional data is required.

In terms of implementation, the framework enables forecasting under limited data, though prediction robustness is highly dependent on the LSTM network architecture each respective input dimension is assigned to. Further development of the approach should explore implementation in real-time applications to receive live equipment data to update predictions to ultimately provide uncertainty predictions on factors such as RUL. The interoperability of such implementation will depend on data sampling rates, computational processing times and varying environmental and human factors [10,61,62]. Development of the visualisation for applications in AR will enhance useability and allow the user to access additional state information for a given point in time. In addition, suitable approaches to mitigate, tolerate or exploit uncertainty through deep learning according to the magnitude should be explored.

Funding

This research is funded by the Engineering and Physical Sciences Research Council (EPSRC) UK, project reference 1944319, and Doctoral Training Partnership (DTP) UK.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The project is based on the collaboration between the Through-life Engineering Services Centre (TES) at Cranfield University (UK) and BAE Systems. The authors would like to thank the Engineering and Physical Sciences Research Council (EPSRC), project reference 1944319, and Doctoral Training Partnership (DTP) for funding this research. For access to the data underlying this paper, please see the Cranfield University repository, CORD, at DOI: 10.17862/cranfield.rd.14381987.

References

[1] Lind, H., Muyingo, H., 2012, Building Maintenance Strategies: Planning Under Uncertainty. Property Management, 30/1: 14–28. https://doi.org/10.1108/02637471211198152.
[2] Erkoyuncu, J.A., Khan, S., Eiroa, A.L., Butler, N., Rushton, K., Brocklebank, S., 2017, Perspectives on Trading Cost and Availability for Corrective Maintenance at the Equipment Type Level. Reliability Engineering and System Safety, 168:53–69. https://doi.org/10.1016/j.ress.2017.05.041.
[3] Greenough, R.M., Grubic, T., 2011, Modelling Condition-based Maintenance to Deliver a Service to Machine Tool Users. International Journal of Advanced Manufacturing Technology, 52/9–12: 1117–1132. https://doi.org/10.1007/s00170-010-2760-x.
[4] Andretta, M., 2014, Some Considerations on the Definition of Risk Based on Concepts of Systems Theory and Probability. Risk Analysis, 34/7: 1184–1195. https://doi.org/10.1111/risa.12092.
[5] Grote, G., 2009, Management of Uncertainty: Theory and Application in the Design of Systems and Organisations, Decision Engineering. Springer, Zurich, London.
[6] NASA, 2010, Measurement Uncertainty Analysis Principles and Methods, NASA. ⟨https://standards.nasa.gov/standard/osma/nasa-hdbk-873919–3⟩.
[7] Newman, M.E.J., 2011, Complex Systems: A Survey. American Journal of Physics, 79/8: 800–810. https://doi.org/10.1119/1.3590372.
[8] Mayfield, M., Punzo, G., Beasley, R., Clarke, G., Holt, N., Jobbins, S., 2018, Challenges of Complexity and Resilience in Complex Engineering Systems. ENCORE Network++ White Paper⟨https://civil-struct.dept.shef.ac.uk/encore/public/storage/rnJInf6iHsqXZcXyAw0IsXb9tdAQnnJS2BTts4QA.pdf⟩.
[9] ElMaraghy, W., ElMaraghy, H., Tomiyama, T., Monostori, L., 2012, Complexity in Engineering Design and Manufacturing. CIRP Annals Manufacturing Technology, 61/2: 793–814. https://doi.org/10.1016/j.cirp.2012.05.001.
[10] Grenyer, A., Erkoyuncu, J.A., Zhao, Y., Roy, R., 2021, A Systematic Review of Multivariate Uncertainty Quantification for Engineering Systems. CIRP Journal of Manufacturing Science and Technology, 33:188–208. https://doi.org/10.1016/j.cirpj.2021.03.004.

[11] McManus, H., Hastings, D., 2005, A Framework for Understanding Uncertainty and Its Mitigation and Exploitation in Complex Systems. INCOSE International Symposium, 15/1: 484–503. https://doi.org/10.1002/j.2334-5837.2005.tb00685.x.

[12] Lei, Y., Li, N., Guo, L., Li, N., Yan, T., Lin, J., 2018, Machinery Health Prognostics: A Systematic Review from Data Acquisition to Rul Prediction. Mechanical Systems and Signal Processing, 104:799–834. https://doi.org/10.1016/j.ymssp.2017.11.016.

[13] Erkoyuncu, J.A., Durugbo, C., Roy, R., 2013, Identifying Uncertainties for Industrial Service Delivery: A Systems Approach. International Journal of Production Research, 51/21: 6295–6315. https://doi.org/10.1080/00207543.2013.794316.

[14] Xu, Y., Elgh, F., Erkoyuncu, J.A., Bankole, O., Goh, Y., Cheung, W.M., et al., 2012, Cost Engineering for Manufacturing: Current and Future Research. International Journal of Computer Integrated Manufacturing, 25/4–5: 300–314. https://doi.org/10.1080/0951192X.2010.542183.

[15] Grenyer, A., Dinmohammadi, F., Erkoyuncu, J.A., Zhao, Y., Roy, R., 2019, Current Practice and Challenges Towards Handling Uncertainty for Effective Outcomes in Maintenance. Procedia CIRP, 86:282–287. https://doi.org/10.1016/j.procir.2020.01.024.

[16] Laplante, P.A., 2004, The Certainty of Uncertainty in Real-time Systems. IEEE Instrumentation & Measurement Magazine, 7/4: 44–50. https://doi.org/10.1109/MIM.2004.1383464.

[17] Shi, Z., Chehade, A., 2021, A Dual-lstm Framework Combining Change Point Detection and Remaining Useful Life Prediction. Reliability Engineering and System Safety, 205/October 2020:107257https://doi.org/10.1016/j.ress.2020.107257.

[18] Everitt, B.S., Skrondal, A., 2010, The Cambridge Dictionary of Statistics. fourth edition Cambridge University Press, Cambridge.

[19] Smart, C., 2014, Bayesian Parametrics: How to Develop a Cer with Limited Data and Even Without Data. International Cost Estimating and Analysis Association, 1–23⟨http://www.iceaaonline.com/ready/wp-content/uploads/2014/07/PA-3-Paper-Bayesian-Parametrics-Developing-a-CER-with-Limited-Data-and-Even-Without-Data.pdf⟩.

[20] Jouin, M., Gouriveau, R., Hissel, D., Péra, M.-C., Zerhouni, N., 2016, Particle Filter-based Prognostics: Review, Discussion and Perspectives. Mechanical Systems and Signal Processing, 72–73:2–31. https://doi.org/10.1016/j.ymssp.2015.11.008.

[21] Schwabe, O., Shehab, E., Erkoyuncu, J.A., 2016, A Framework for Geometric Quantification and Forecasting of Cost Uncertainty for Aerospace Innovations. Progress in Aerospace Sciences, 84:29–47. https://doi.org/10.1016/j.paerosci.2016.05.001.

[22] Schwabe, O., Shehab, E., Erkoyuncu, J., 2015, Uncertainty Quantification Metrics for Whole Product Life Cycle Cost Estimates in Aerospace Innovation. Progress in Aerospace Sciences, 77:1–24. https://doi.org/10.1016/j.paerosci.2015.06.002.

[23] Goh, Y.M., Newnes, L.B., Mileham, A.R., McMahon, C.A., Saravi, M.E., 2010, Uncertainty in Through-life Costing-review and Perspectives. IEEE Transactions on Engineering Management, 57/4: 689–701. https://doi.org/10.1109/TEM.2010.2040745.

[24] Bertoni, A., Bertoni, M., 2020, PSS Cost Engineering: A Model-based Approach for Concept Design. CIRP Journal of Manufacturing Science and Technology, 29:176–190. https://doi.org/10.1016/j.cirpj.2018.08.001.

[25] Wang, B., Lu, J., Yan, Z., Luo, H., Li, T., Zheng, Y., et al., 2019, Deep Uncertainty Quantification: A Machine Learning Approach for Weather Forecasting. Proceedings of 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, USA: 2087–2095. https://doi.org/10.1145/3292500.3330704.

[26] Wu, Y., Yuan, M., Dong, S., Lin, L., Liu, Y., 2018, Remaining Useful Life Estimation of Engineered Systems Using Vanilla Lstm Neural Networks. Neurocomputing, 275:167–179. https://doi.org/10.1016/j.neucom.2017.05.063.

[27] Chen, X., Yu, J., Tang, D., Wang, Y., 2012, A Novel PF-LSSVR-based Framework for Failure Prognosis of Nonlinear Systems with Time-varying Parameters. Chinese Journal of Aeronautics, 25/5: 715–724. https://doi.org/10.1016/S1000-9361(11)60438-X.

[28] Cerdeira, P.B.O., Galvão, R.K.H., Malère, J.P.P., 2013, Particle Filter Prognostic Applied in Landing Gear Retraction. PHM2013 – Proceedings of Annual Conference of the Prognostics and Health Management Society. : 616–623. https://doi.org/10.36001/phmconf.2013.v5i1.2180.

[29] Maier, M., Zwicker, R., Akbari, M., Rupenyan, A., Wegener, K., 2019, Bayesian Optimization for Autonomous Process Set-up in Turning. CIRP Journal of Manufacturing Science and Technology, 26:81–87. https://doi.org/10.1016/j.cirpj.2019.04.005.

[30] Fujishima, M., Narimatsu, K., Irino, N., Mori, M., Ibaraki, S., 2019, Adaptive Thermal Displacement Compensation Method Based on Deep Learning. CIRP Journal of Manufacturing Science and Technology, 25:22–25. https://doi.org/10.1016/j.cirpj.2019.04.002.

[31] Ahmed, T., Srivastava, A., 2016, Predicting Human Interest: An Application of Artificial Intelligence and Uncertainty Quantification. Journal of Uncertainty Analysis and Applications, 4/1: 9. https://doi.org/10.1186/s40467-016-0051-2.

[32] Beven, K., Freer, J., 2001, Equifinality, Data Assimilation, and Uncertainty Estimation in Mechanistic Modelling of Complex Environmental Systems Using the Glue Methodology. Journal of Hydrology, 249/1–4: 11–29. https://doi.org/10.1016/S0022-1694(01)00421-8.

[33] Yang, L., Guo, Y., Kong, Z., 2019, On the Performance Evaluation of a Hierarchical-structure Prototype Product Using Inconsistent Prior Information and Limited Test Data. Information Science, 485:362–375. https://doi.org/10.1016/j.ins.2019.02.018.

[34] Hariri-Ardebili, M.A., Barak, S., 2020, A Series of Forecasting Models for Seismic Evaluation of Dams Based on Ground Motion Meta-features. Engineering Structures, 203:109657https://doi.org/10.1016/j.engstruct.2019.109657.

[35] Kwon, Y., Won, J.H., Kim, B.J., Paik, M.C., 2020, Uncertainty Quantification Using Bayesian Neural Networks in Classification: Application to Biomedical Image Segmentation. Computational Statistics & Data Analysis, 142:106816https://doi.org/10.1016/j.csda.2019.106816.

[36] Gal, Y., Ghahramani, Z., 2016, Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. Proceedings of the 33rd International Conference on Machine Learning. New York, USA: 1050–1059⟨http://proceedings.mlr.press/v48/gal16.pdf⟩.

[37] Ghahramani, Z., 2015, Probabilistic Machine Learning and Artificial Intelligence. Nature, 521/7553: 452–459. https://doi.org/10.1038/nature14541.

[38] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014, Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, 15:1929–1958.

[39] Cicuttin, A., Crespo, M.L., Mannatunga, K.S., Garcia, V.V., Baldazzi, G., Rignanese, L.P., et al., 2016, A Programmable System-on-chip Based Digital Pulse Processing for High Resolution X-ray Spectroscopy. 2016 International Conference on Advances in Electrical, Electronic and Systems Engineering. IEEE: 520–525. https://doi.org/10.1109/ICAEES.2016.7888100.

[40] Gal, Y., 2016, Uncertainty in Deep Learning (Ph.D. thesis). University of Cambridge, Cambridge⟨http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf⟩.

[41] Radaideh, M.I., Kozlowski, T., 2020, Surrogate Modeling of Advanced Computer Simulations Using Deep Gaussian Processes. Reliability Engineering and System Safety, 195:106731https://doi.org/10.1016/j.ress.2019.106731.

[42] Grenyer, A., Schwabe, O., Erkoyuncu, J.A., Zhao, Y., 2021, Dynamic Multistep Uncertainty Prediction in Spatial Geometry. Procedia CIRP, 96:74–79. https://doi.org/10.1016/j.procir.2021.01.055.

[43] Schwabe, O., Shehab, E., Erkoyuncu, J., 2015, Geometric Quantification of Cost Uncertainty Propagation: A Case Study. Procedia CIRP, 37:158–163. https://doi.org/10.1016/j.procir.2015.08.078.

[44] Mujeeb, S., Javaid, N., Ilahi, M., Wadud, Z., Ishmanov, F., Afzal, M., 2019, Long Short-term Memory: A New Price and Load Forecasting Scheme for Big Data in Smart Cities. Sustainability, 11/4: 987. https://doi.org/10.3390/su11040987.

[45] Chen, Z., Wu, M., Zhao, R., Guretno, F., Yan, R., Li, X., 2021, Machine Remaining Useful Life Prediction Via an Attention-based Deep Learning Approach. IEEE Transactions on Industrial Electronics, 68/3: 2521–2531. https://doi.org/10.1109/TIE.2020.2972443.

[46] R. Jozefowicz, W. Zaremba, I. Sutskever, An empirical exploration of recurrent network architectures, 32nd Int. Conf. Mach. Learn. ICML 2015. 3 (2015) 2332–2340.

[47] Wang, S., Zhang, X., Gao, D., Chen, B., Cheng, Y., Yang, Y., et al., 2018, A Remaining Useful Life Prediction Model Based on Hybrid Long-short Sequences Forengines. 2018 21st International Conference on Intelligent Transportation Systems. IEEE: 1757–1762. https://doi.org/10.1109/ITSC.2018.8569668.

[48] Behera, S., Misra, R., Sillitti, A., 2021, Multiscale Deep Bidirectional Gated Recurrent Neural Networks Based Prognostic Method for Complex Non-linear Degradation Systems. Information Sciences, 554:120–144. https://doi.org/10.1016/j.ins.2020.12.032.

[49] Sun, J., Zuo, H., Wang, W., Pecht, M.G., 2012, Application of a State Space Modeling Technique to System Prognostics Based on a Health Index for Condition-based Maintenance. Mechanical Systems and Signal Processing, 28:585–596. https://doi.org/10.1016/j.ymssp.2011.09.029.

[50] Ramasso, E., Saxena, A., 2020, Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets. International Journal of Prognostics and Health Management, 5/2: 1–15. https://doi.org/10.36001/ijphm.2014.v5i2.2236.

[51] Porter, T., 2004, Karl Pearson: The Scientific Life in a Statistical Age. Princeton University Press, Oxford⟨http://www.jstor.org/stable/j.ctt4cgc9j⟩.

[52] Yu, W., Kim, I.Y., Mechefske, C., 2021, Analysis of Different RNN Autoencoder Variants for Time Series Classification and Machine Prognostics. Mechanical Systems and Signal Processing, 149:107322https://doi.org/10.1016/j.ymssp.2020.107322.

[53] You, Y., Sun, J., wang Chen, Y., Niu, C., Jiang, J., 2021, Ensemble Belief Rule-based Model for Complex System Classification and Prediction. Expert Systems with Applications, 164/August 2020:113952https://doi.org/10.1016/j.eswa.2020.113952.

[54] Schwabe, O., 2018, Geometrical Framework for Forecasting Cost Uncertainty in Innovative High Value Manufacturing (Ph.D. thesis). Cranfield University, Cranfield⟨https://dspace.lib.cranfield.ac.uk/handle/1826/13616⟩.

[55] Muller, S., Lesage, P., Ciroth, A., Mutel, C., Weidema, B.P., Samson, R., 2016, The Application of the Pedigree Approach to the Distributions Foreseen in Ecoinvent v3. International Journal of Life Cycle Assessment, 21/9: 1327–1337. https://doi.org/10.1007/s11367-014-0759-5.

[56] Grenyer, A., Erkoyuncu, J.A., Addepalli, S., Zhao, Y., 2020, An Uncertainty Quantification and Aggregation Framework for System Performance Assessment in Industrial Maintenance. SSRN Electronic Journal, (3-4 November): 1–8. https://doi.org/10.2139/ssrn.3718001https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3718001.

[57] Ben-Haim, Y., Hemez, F.M., 2012, Robustness, Fidelity and Prediction-looseness of Models. in: Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 468/2137: 227–244. https://doi.org/10.1098/rspa.2011.0050.

[58] Saxena, A., Goebel, K., Simon, D., Eklund, N., 2008, Damage Propagation Modeling for Aircraft Engine Run-to-failure Simulation. 2008 International Conference on Prognostics and Health Management. IEEE: 1–9. https://doi.org/10.1109/PHM.2008.4711414.

[59] Putnik, G.D., Manupati, V.K., Pabba, S.K., Varela, L., Ferreira, F., 2021, Semi-double-loop Machine Learning Based CPS Approach for Predictive Maintenance in Manufacturing System Based on Machine Status Indications. CIRP Annals, 70/1: 365–368. https://doi.org/10.1016/j.cirp.2021.04.046.

[60] Schwabe, O., Shehab, E., Erkoyuncu, J.A., 2015, Long Tail Uncertainty Distributions in Novel Risk Probability Classification. Procedia CIRP, 28:191–196. https://doi.org/10.1016/j.procir.2015.04.033.

[61] Bate, I., Griffin, D., Lesage, B., 2020, Establishing Confidence and Understanding Uncertainty in Real-time Systems. Proceedings of 28th International Conference on Real-Time Networks System. ACM, New York, NY, USA: 67–77. https://doi.org/10.1145/3394810.3394816.

[62] Shah, S.A.B., Rashid, M., Arif, M., 2020, Estimating wcet Using Prediction Models to Compute Fitness Function of a Genetic Algorithm. Real-Time System, 56/1: 28–63. https://doi.org/10.1007/s11241-020-09343-2.