electronics

Article

A Hybrid Model for Prognostic and Health Management of Electronic Devices

Alessandro Murgia [1,*], Chaitra Harsha [1], Elena Tsiporkova [1], Chinmay Nawghane [2] and Bart Vandevelde [2]

[1] EluciDATA Lab of Sirris, Blvd Auguste Reyers 80, 1030 Brussels, Belgium; chaitra.harsha@sirris.be (C.H.); elena.tsiporkova@sirris.be (E.T.)
[2] Imec, 3001 Leuven, Belgium
* Correspondence: alessandro.murgia@sirris.be

Abstract: Techniques for prognostic and health management are becoming common in the electronic domain to reduce the cost of failures. Typically, the proposed techniques rely either on physics-based or data-driven models. Only a few studies explored hybrid models to combine the advantages of both models. For this reason, this work investigates the potential of hybrid modeling by presenting a new framework for the diagnostics and prognostics of an electronic system. The methodology is validated on simulation data describing the behavior of a QFN package subject to die delamination. The main results of this work are twofold. First, it is shown that the hybrid model can achieve better performance than the performance obtained by either the physics-based or the data-driven models alone. Second, a baseline is set for the best performance achievable by the hybrid model, allowing us to estimate the remaining useful life of the package.

Keywords: reliability; QFN; package delamination; PHM; RUL; health index; hybrid modeling

check for updates

Citation: Murgia, A.; Harsha, C.; Tsiporkova, E.; Nawghane, C.; Vandevelde, B. A Hybrid Model for Prognostic and Health Management of Electronic Devices. *Electronics* **2024**, *13*, 642. https://doi.org/10.3390/electronics13030642

Academic Editor: Fernando De la Prieta Pintado

Received: 29 November 2023
Revised: 30 January 2024
Accepted: 31 January 2024
Published: 3 February 2024

1. Introduction

Microelectronics technology has made significant advancements in the past decades, enabling the creation of integrated circuits (ICs), i.e., the building blocks of digital devices, leading to technological innovation and industrial revolution. The ICs are realized by combining multiple materials such as silicon, copper, solder, etc. These materials react differently to the applied loading such as temperature cycles and thermal shocks, leading to internal stresses which eventually can cause thermomechanical failures. This type of failure is responsible alone for more than 65% of the total electronics failures [1].

To address the costs related to these failures (e.g., reactive maintenance, warranties, downtime, etc.), it is becoming common in the electronic domain to rely on prognostic and health management (PHM) techniques [2,3]. PHM is a research field which develops methods to evaluate system reliability in order to provide fault detection (i.e., diagnostics), and remaining useful life forecasts (i.e., prognostics). Historically, in the electronic domain, PHM methods have been applied either relying on physics-based models or data-driven (DD) ones [4]. The former, also known as the physics-of-failure (PoF) model, relies on first-principles physics models (numerical models) to describe the system and its degradation mechanism [5]. The latter relies on empirical collected data for training a data-driven model (e.g., neural networks) and infers the expected behavior of the model. However, in complex electronic systems, the usage of these models alone presents shortcomings. Physics-based models are generally incomplete (e.g., cover limited mission profiles) and consequently difficult to scale, and thus, they fail to correctly describe the degradation process in real scenarios. Furthermore, a high-fidelity physics-based model is highly demanding in computational terms. Consequently, when deployed in real applications, certain parts are simplified to meet computational demands, leading to an impact on model accuracy [6]. Data-driven models can correctly describe the behavior of a component as long as they have “sufficient” and representative data for its characterization. Unfortunately, this is often not the case, since in electronics, the run-to-failure data, i.e., data collection until the component presents an observable failure, are hard to capture. Indeed, even with accelerated tests, not all faults become observable, and consequently, only a small set of failure types are generally available for training the data-driven models. Moreover, the black box nature of several data-driven models poses a challenge in terms of trustability, as the outcome cannot be interpreted by physics, and its results are opaque to users [7].

To address these shortcomings, a few researchers have started to explore hybrid models able to combine the physics-based and data-driven models. Hybrid models can indeed cope with some of the drawbacks of physics-based and data-driven models and still retain many of their advantages. For instance, physics-based models can support data-driven models by reducing the amount of data required for training. The latter can be realized by providing informative input features, by ensuring the physical consistency of data-driven models [8] or by generating synthetic data [9]. On the other hand, data-driven models can complement physics-based models by integrating or replacing some of its parts [10] (e.g., for modeling unseen failure modes) or empirically determining its model parameters [11]. Furthermore, physics-based and data-driven models combined proved to allow for a better RUL forecast [12]. In this context, it is opportunistic to investigate the potential of hybrid modeling for PHM in electronics. More specifically, this work presents a novel framework based on a hybrid model for diagnostics and prognostics of the health index of an electronic system using as a case study a QFN package subject to die delamination. According to our literature study, we are the first to validate hybrid modeling for PHM at the package level.

The main results of this work are twofold. First, at package level, the proposed hybrid model for diagnostics achieves better performance than the ones based exclusively on either the physics-based or the data-driven models. Second, the framework produces a baseline in terms of estimation accuracy of the remaining useful life of the package.

The structure of the manuscript is organized as follows: in Section 2, the related works are summarized; in Section 3, the background notions related to the physics of delamination in the QFN package are described; in Section 4, the methodology is reported in general; in Section 5, the test case, the dataset and the concrete instantiation of the methodology on the test case are reported; Section 6 is devoted to the presentation and discussion of the results; and conclusions are drawn and further directions are outlined in Section 7.

2. Related Work

In this section, and in the rest of the paper, to avoid confusion, we will refer to the physics-based model as PoF (physics-of-failure) models since the latter is frequently used in the electronic literature.

Prognostic and health management is a well-established research field which only recently is becoming applied to the electronic domain. This delay is mainly due to the lack of electronic systems opportunely sensorized for data collection, which is the root cause of the scarcity of suitable datasets that can be used for validation of PHM solutions [4]. Methods for PHM are generally presented and validated using either a PoF model or a DD one [4]. To our best knowledge, only few exceptions exist where hybrid models are (conceptually) presented to perform PHM at the component (e.g., capacitor) or system level (e.g., computer). Patil et al. presented conceptually a hybrid methodology for diagnostics of the IGBT transistor [13]. The work demonstrated that the threshold voltage, transconductance, and collector-emitter ON voltage are all relevant failure precursors. The authors suggest that those precursors can be exploited by a PoF model for anomaly detection and remaining useful life prediction. Cheng and Pecht presented conceptually a hybrid methodology for the prognostics of capacitors [14]. In this methodology, the PoF model is used to identify the potential failure mechanisms, failure models, and failure criteria, whereas the DD model is used for feature extraction and to define the normal behavior of the components for anomaly detection and remaining useful life prediction. Sachin and Kuman presented a hybrid approach for the fault diagnostics of a computer system [7]. In this approach, the PoF model is used to monitor the environmental loading conditions, and the DD model is used for anomaly detection and fault isolation.

Methods for PHM at the die and package level are still in an early research stage. Riegel et al. investigated the die delamination at the board level on a QFN component [15]. As a first step, an accelerated aging experiment based on thermal cycling was created to reach a rapid component failure. During the experiment, run-to-failure field data were captured using stress sensors. As a second step, a physical failure analysis for health assessment was performed. The results highlighted the correlation between the amount of delamination and solder joint fatigue, which is coupled with the number of thermal cycles. In this sense, this work paved the way for the usage of PHM for die delamination. On the same research line, Prisacaru et al. analyzed the delamination process in automotive electronic packages subject to delamination at the interface of the molding compound/copper pad and silicon die/die attachment [16]. In this case too, to collect sufficient run-to-failure data, an accelerated testing via thermal cycles was performed on a package monitored via stress sensors. On this data, a DD model was trained for the estimation and prediction of degradation, but no remaining useful life was extracted. Results confirmed that the DD model was suitable to characterize the package behavior during operation.

For a full overview of prognostic and health management methods for electronic devices, the interested reader can refer to the surveys [4,17,18]. To the best of the authors knowledge, no hybrid model for a remaining useful life forecast has been evaluated at the package level, and no hybrid model for PHM at the die level has even been presented.

3. Background

This section provides the background for understanding how the delamination process takes place and how it occurs in a QFN.

3.1. The Physics of Delamination

Delamination is an irreversible process which progresses over time. Delamination is the separation of bi-material interfaces, and it occurs in two phases, delamination initiation and delamination propagation. Chip fabrication and packaging processes can cause delamination in the package. Delamination can also develop because of high interfacial stresses caused by severe thermal loads, and in this case, it is known as stress-induced delamination [19]. The delamination usually initiates at the locations with geometrical discontinuities such as edges, corners where stress concentration is observed. The driving mechanism behind package delamination, such as die delamination, is the coefficient of thermal expansion (CTE) mismatch between various materials involved in multi-layer packages. Expansion and contraction of the package will happen during thermal cycling, inducing stresses and strains on the interfaces.

3.2. QFN Package

A QFN (Quad Flat No-leads) is a low cost, compact, lightweight package with improved thermal performance, and it is widely used for programmable modules and microcontrollers. QFNs are characterized by abundant geometrical discontinuities (such as edges, corners) and multilayer construction with several materials which makes them prone to delamination. QFNs are widely used in automotive applications where the delamination of QFN packages is a known issue which affects the package reliability. For these reasons, the QFN is used as a case study in this work. The QFN package consists of a silicon die connected to a copper pad using die attachment material, and the full system is encapsulated using an epoxy molding compound (EMC). The cross-sectional view of the QFN package is presented in Figure 1.

During its usage, the QFN is subject to different loading conditions, such as current and temperature. The temperature induces internal stresses due to the different CTE of the materials. Over time, the prolonged thermal cycle can eventually lead to a progressive reduction in the electrical contact between silicon and the mold, which eventually leads to the complete delamination of the IC. This process is irreversible and leads the QFN to reach its end of life. As demonstrated by Schindler-Saefkow [20], such mechanical stress, which can be captured by stress sensors, can be exploited to evaluate the health status of an electric component during the 'early fail' as well as during the 'wear out' phases. Besides the loading condition, the progress of the delamination depends on QFN material properties (e.g., E-modulus, coefficient of thermal expansion) and its geometry (e.g., solder thickness).

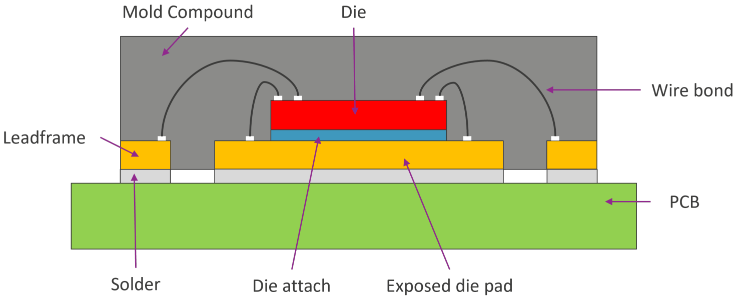


Figure 1. Cross-sectional view of the QFN package.

4. Methodology

This section presents the methodology implemented by the framework to perform PHM on a QFN.

The framework, presented in Figure 2, consists of diagnostic and prognostic modules. The former provides the current health index of the QFN, whereas the latter forecasts the evolution of the health index over time, i.e., the remaining useful life of the QFN. The health index of the QFN is described by a metric which we call the delamination index. This metric is used to characterize the monotonic evolution of die delamination from 0, no die delamination, to 1, end of the useful life of the die. As input, the framework relies on loading conditions, stress sensor values, delamination criteria, and QFN parameters. As output, the framework provides the delamination index and the remaining useful life of the QFN. The rest of this section presents in more detail the diagnostic and prognostic modules leveraging on the use case as a running example.

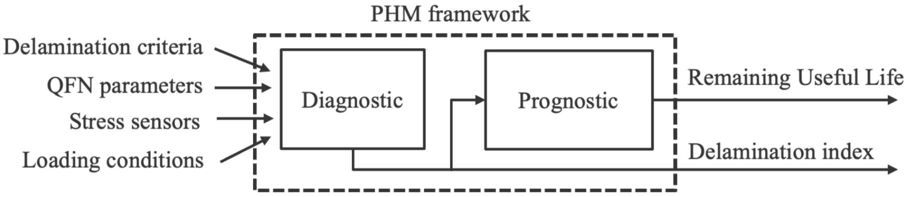


Figure 2. PHM framework.

4.1. Diagnostic Module

The diagnostic module provides as output the delamination index of the QFN. The delamination is determined by a hybrid model which combines the delamination indexes predicted by the PoF and DD models. Aizpurua et al. [21] have developed a decision framework which guides the designer in deciding the layout of the hybrid model without implementation details. From this framework, we selected a parallel layout to improve the estimation accuracy of the single models, and then, we further elaborated on the hybrid model by describing how the output of the PoF and DD models are combined in practice for diagnostic purposes. The hybrid model also presents limitations and assumptions which depend on the data-driven and PoF model used. Indeed, a qualitative dataset is required in order to train correctly the data-driven model. Furthermore, the specific nature of the PoF model limits the hybrid model's usability across different integrated circuits or conditions. Figure 3 illustrates the three components of the diagnostic module.

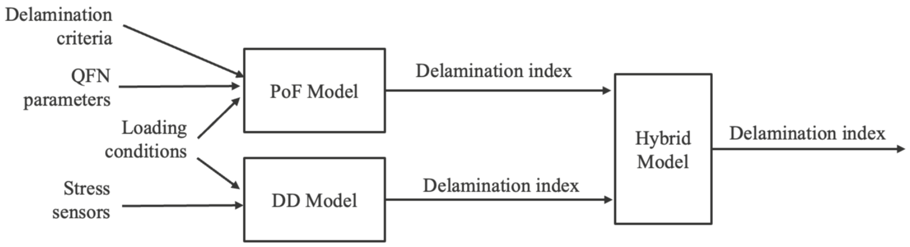


Figure 3. Overview of the components which compose the diagnostic module.

4.1.1. PoF Model

The PoF model characterizes the physical behavior of the QFN. The expected physical behavior described by the PoF model can differ from the one empirically observed on the QFN model in the field. This happens because there are also other factors which influence the delamination which the PoF model cannot take into account (e.g., the environment in which the IC is operating like humidity, the real delamination profile, etc.). The PoF model, leveraging on underlying engineering and failure principles, can isolate the failure mechanisms behind the die-delamination progress.

The methodology implemented by the model, depicted in Figure 4, is divided into the following three steps:

- Computation of the local damage. A 2.5D model is a finite element model used to mimic the QFN behavior. Based on the input parameters, this model provides the evolution of the local damage index for each one of its nodes. The model discretizes the structure of the material into smaller shapes called elements (lines, plates, bricks, tetrahedral), and these elements are bounded by nodes (e.g., line element is bounded by one node at each end) [22]. The local damage index is calculated by monitoring normal and tangential stresses at each node to evaluate delamination criteria. The local damage index is set to 1 as soon as the delamination criteria is fulfilled.
- Computation of the delamination percentage. Figure 5 shows a dashed line for the region of interest where the delamination takes place. In this region, the stress value at each node is continuously monitored to check if it exceeds the delamination criteria. Such delamination criteria characterizes the stress criterion which provides the allowable normal and tangential stresses at every interface in the package. When the stress value exceeds the delamination criteria, the interface at the location of the node detaches, and the local damage index of the nodes reaches one. At the package level, a delamination percentage is computed as the ratio of detached nodes with respect to the total number of nodes on the interface. As an illustrative example, Figure 6 shows the monotonic-like evolution of the delamination percentage over time.
- Computation of the delamination index. The delamination percentage is converted to the delamination index for the QFN package. The delamination index ranges from 0, no die delamination, to 1, end of remaining useful life of the die. This end of life is application-dependent, and it is reached when the delamination percentage goes above a certain threshold. This threshold depends on the type of electronic system where the delamination takes place. On the one hand, in applications with small power dissipation, just 10% delamination in the die attachment contact area can still assure good functionality in an application-specific integrated circuits. On the other hand, in an application with larger power dissipation, 20% delamination of the contact area can cause the end of life of the component [16]. To take into account this

threshold, the delamination index, which we call DI^{PoF}, is mapped to the delamination percentage as described in Equation (1) as follows:

$$DI^{PoF}(t) = \begin{cases} 1 & \text{if } DP(t) > \text{threshold} \\ \frac{DP(t)}{100} & \text{if } DP(t) \leq \text{threshold} \end{cases} \tag{1}$$

where $DP(t)$ represents the delamination percentage at time t; the threshold represents the level of delamination percentage after which the end of life is reached. The latter, being domain-dependent, can be defined based on the end user specifications.

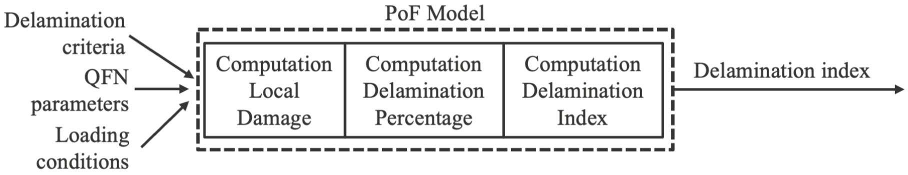


Figure 4. Workflow for PoF model.

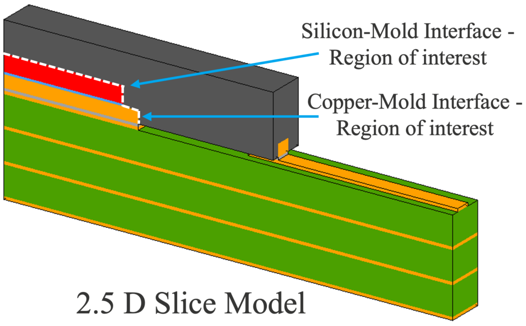

Silicon-Mold Interface - Region of interest
Copper-Mold Interface - Region of interest
2.5 D Slice Model

Figure 5. Region of interest shown by a dashed white line at silicon–mold interface and copper–mold interface.

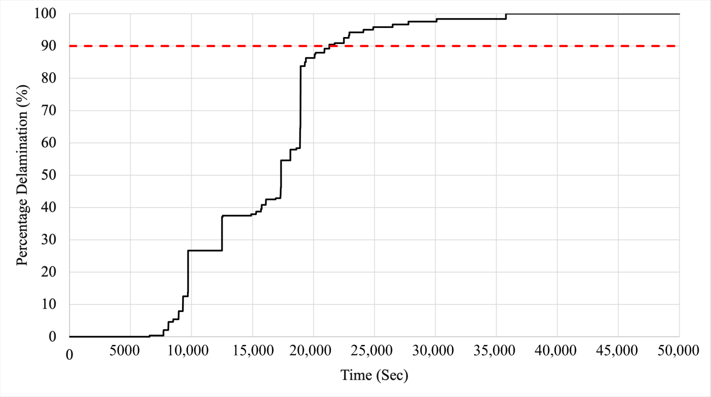


Figure 6. An example of the evolution of delamination percentage. The horizontal dashed line represents a possible threshold to define the end of life of the package.

4.1.2. DD Model

The DD model characterizes the field behavior of the QFN, i.e., it models the QFN's behavior observed empirically. This model, leveraging the data captured by the sensors (see Figure 7), can monitor the die-delamination progress.

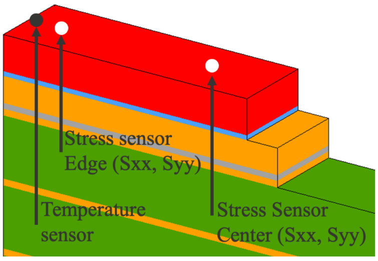


Figure 7. Cross-sectional view of the QFN package with the locations of the temperature sensor and stress sensors.

The methodology implemented by the model is depicted in Figure 8. As a first step, the raw stress sensor values are preprocessed to highlight the monotonic evolution of the delamination progress. As a second step, the delamination progress is converted into a delamination index. These steps will now be discussed in more detail.

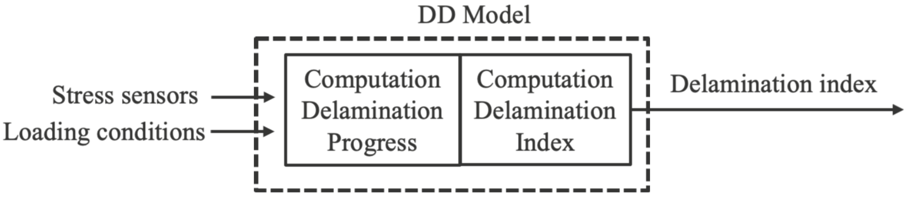


Figure 8. Overview of the data-driven module for diagnostics.

Computation of the Delamination Progress

The evolution data of each stress sensor value is initially preprocessed making it monotonic to highlight the progress of the delamination. This is achieved by factoring out from the evolution sensor values the influence of the loading condition. As a concrete example, we can imagine that the loading condition presents a periodic pattern which leads to a periodic evolution of the sensor values (due to the continuous expansion and contraction of the material). In this case, by just identifying empirically the periodicity of the loading condition, the evolution of the stress sensor value can be resampled accordingly. In Figure 9, the raw and monotonic evolution of the stress sensor values are described by the blue and red curves, respectively.

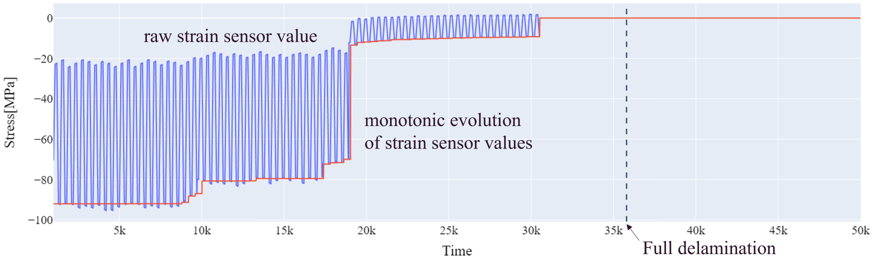


Figure 9. Evolution of the stress sensor values. The blue curve reports the raw stress sensor values, whereas the red curve reports their monotonic evolution. The dashed vertical line represents the point in time when the full delamination is reached.

Computation of the Delamination Index

The delamination progress is converted into a delamination index. This is carried out using a machine learning regression model trained offline using run-to-failure data. The choice of the regression model depends on multiple factors such as the amount of data available, model understandability, computational resources available, etc. In electronic reliability, considering the typical lack of run-to-failure data, machine learning algorithms which have low demands in terms of data training size and yet that can guarantee a good tradeoff between bias and variance are preferable. Well-known machine learning models which fall into this category are linear regression, random forest regression, support vector machine (SVM) regression, etc. In any case, the final choice of the algorithm requires a careful analysis of the learning curves. These curves, by describing the performance of a machine learning model as a function of the amount of training data, can be used to avoid possible cases of overfitting or underfitting.

4.1.3. Hybrid Model

The proposed hybrid model features a regression model which combines as input the delamination indexes generated by PoF and DD models and provides as output an adjusted delamination index. The choice of the regression model follows the same rules decided before for the DD model.

4.2. Prognostic Module

The prognostic module provides as output the remaining useful life of the QFN. The methodology can be divided into an offline and online stage. The offline stage is concerned with the construction of a library of delamination index (DI) curves. The online stage consists of three steps. First, the DI curves in the library are rescaled with respect to the target curve, i.e., the DI curve for which the remaining useful life needs to be forecasted. Second, the rescaled DI curves are retained based on their similarity with the target curve. Finally, the retained curves are combined to forecast the remaining useful life of the target curve. The offline and online stages are discussed in more detail below.

4.2.1. Creation Library Delamination Index Curves

A library Σ of DI curves is created. In this library, described as $\Sigma = \{C^{\{j\}}\}(j = 1, 2, \ldots, J)$, each curve has a different time length T_j and presents a monotonic-like evolution from 0 (no die delamination) to 1 (end of life), i.e., $C^{\{j\}} = [c_1^{\{j\}}, c_2^{\{j\}}, \ldots, c_{T_{\{j\}}}^{\{j\}}]$, $c_1^{\{j\}} = 0$ and $c_{T_j}^{\{j\}} = 1$, as shown in Figure 10.

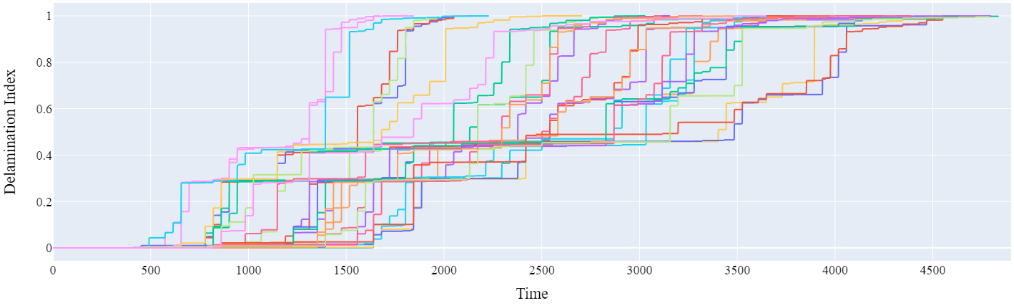


Figure 10. Library of DI curves. Each curve is represented with a different color.

Each DI curve, generated offline by the diagnostic model, describes the evolution of DI in multiple scenarios (e.g., loading conditions, QFN parameters, etc.). On the one hand, the more DI curves that are present in the library, the higher the chance to characterize the possible evolution of future delamination indexes as best as possible. On the other hand, in scenarios where there are limited resources to store or process these DI curves, it is not possible to use all possible DI curves for the library. In this case, to select a representative sample of DI curves, it is possible to cluster all available DI curves and then retain only one curve per cluster based on its capacity to represent the entire cluster well.

4.2.2. Rescaling of the Delamination Index Curves

The methodology is based on a similarity-based interpolation which assumes that the evolution of all DI curves follows a similar pattern [23–25]. As a consequence, the evolution of any target curve can be inferred based on the run-to-failure evolution of similar reference curves present in the library Σ [26]. In case of package delamination, the associated reference DI curves follow similar—yet not equal—patterns. Indeed, although they all have a monotonic evolution, the DI curves differ from the point in time where the step changes occur. In real case scenarios, this can be observed due to differences in manufacturing tolerances, environmental conditions, defects, etc. To cope with this issue, in this step, each reference curves is rescaled to increase its similarity with the target curve $C' = c'_1, c'_2, \ldots, c'_{T'}$ where T' is the length of the target curve, i.e., the point in time where the forecast is computed. Such rescaling is performed to fully exploit each single DI curve for computing the remaining useful life of the test curve. As a representative example, Figure 11 shows the target curve C' as a blue line and two other curves representative of a reference curve before ($C^{\{j\}}$) and after the rescaling ($R^{\{j\}}$) as the red and green curves, respectively. The rescaling of each reference curve proceeds as follows:

- A dynamic time warping [27] is applied between C' and the reference curve ($C^{\{j\}}$). As a result, each time stamp in the target curve is mapped with a one (or more) time stamp in the reference curve. From this mapping, the first time stamp in the reference curve is identified, which is mapped to T' in the target curve. This time stamp is called T^R.
- A rescaling is applied from the beginning to the end of the reference curve $C^{\{j\}}$ to create a rescaled reference curve $R^{\{j\}}$. This is completed by shrinking (or expanding) the original reference curve in order to match T^R with T'. The shrinking of the reference curve is carried out by undersampling the curve, whereas the expansion is carried out by oversampling the curve. The first consecutive T' points of the rescaled reference curve define the reference curve $R^{\{j\}}$. It is worth noting that each reference curve can be rescaled differently based exclusively on the value of T^R and T'.

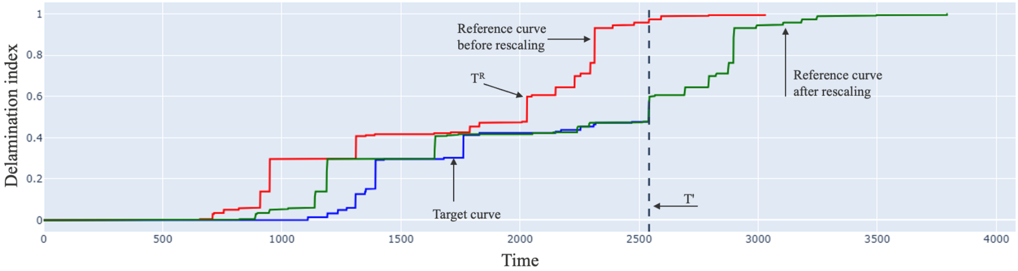


Figure 11. Rescaling of a DI reference curve based on the target curve. The vertical line T' represents the point in time where the forecast is computed. T^R represents the time stamp mapped to T' in the reference curve. The red and green curves represent the reference curve before and after the rescaling, respectively; the blue curve represents the target curve.

4.2.3. Retention of the Reference Curves

The rescaled reference curves are ranked based on their similarity to the target curve, and the most similar ones are retained. This is carried out in two steps. First, the Euclidean distance is computed between the target curve C' and each rescaled reference curve $R^{\{j\}}$ using Equation (2). The metric Euclidean distance is used since it provides a measure of goodness-of-fit between two curves, and the smaller the distance, the higher the similarity between the curves.

$$d(C',R^{\{j\}}) = \sqrt{\sum_{i=1}^{T'} (C'_i - R_i^{\{j\}})^2} \tag{2}$$

Then, the rescaled reference curves are sorted based on their similarity, and the top N are retained. We call this sorted list S. The number N of retained curves is domain-dependent and needs to be manually identified offline looking at the evolution of the DI curves in the reference sample. On the one hand, the number should be large enough to allow each target curve to be compared with at least a few other reference curves when the forecast is made. This avoids the situation where eventually no reference curve is identified for the forecast. On the other hand, the number should be small enough to avoid too many reference curves being retained. This avoids the forecasts being just the expected ones of the entire (or large part) of the reference sample. A possible trade-off between these two needs is to set N as 5.

4.2.4. Remaining Useful Life Forecast

In this step, the remaining useful life (RUL) on the QFN is computed. This is carried out using Equation (3) as follows:

$$RUL(t) = Q2(S) - t \tag{3}$$

where $Q2(S)$ represents the second quartile, i.e., the median, of the EoLs associated with the retained S curves; t the point in time where the RUL is computed.

The usage of $Q2(S)$ for EoL estimation provides robustness against outliers which may be present in S and aims to minimize the forecast error. However, as early forecasts are better than late ones, it can be convenient in several cases to be more conservative on the estimated RUL. In these cases, Equation (4) can be used as follows:

$$RUL(t) = Q1(S) - t \tag{4}$$

where $Q1(S)$ represents the first quartile of the EoLs associated with the retained S curves.

It is worth noting that the forecast of the RUL can be theoretically computed at any point in time. However, the RUL forecast should be triggered only when the delamination index changes. Indeed, if the delamination index remains constant, then the remaining useful life of the QFN does not change. This approach would also make the computation of the RUL possible in the case of limited computational resources available in the chip.

5. Experimental Setup

This section describes how the QFN is monitored and how the collected data are used for validation.

5.1. QFN Parameters

The QFN parameters used in the case study are presented in Tables 1 and 2. Table 1 reports the QFN geometry, whereas Table 2 reports the material properties of the QFN package.

Table 1. QFN package standard dimensions.

Parameter	Dimensions
Die (X, Y, Z)	6 mm × 6 mm × 0.27 mm
Die attachment thickness	35 µm
Exposed die pad (X, Y, Z)	7 mm × 7 mm × 0.2 mm
QFN package (X, Y, Z)	9 mm × 9 mm × 1.05 mm

Table 2. Material properties of QFN package.

Materials	E Modulus (GPa)	Poisson's Ratio	In-Plane CTE (ppm/°C)
Silicon die	169	0.25	2.6
Copper die pad, Leadframe	120	0.35	17.6
EMC (epoxy molding compound)	30	0.25	7

The QFN is modeled as a 2.5D model, a slice of a full QFN package model, using a commercial finite element software. The package is modeled with silicon–mold and copper–mold interfaces. A structural symmetry boundary condition is also applied on the model to define the 2.5D slice model as shown in Figure 5. Under thermal loading conditions, the QFN package is subject to EMC-Die and EMC-Leadframe delamination. This model uses the delamination criteria shown in Equation (5).

$$\left(\frac{\sigma_n}{S_n}\right)^m + \left(\frac{\sigma_t}{S_t}\right)^n > 1 \tag{5}$$

Here, σ_n is the normal stress, σ_t is the tangential stress, S_n is the maximum allowable normal stress, S_t is the maximum allowable tangential stress, and m and n are the material-specific exponents. Material properties of the QFN are modeled as elastic–plastic isotropic materials.

5.2. Dataset

In real life, full package-delamination is a long and complex process which is hard to capture from the field. This is a main limitation to validate the PHM framework since it requires multiple run-to-failure samples, collected in a controlled environment, to characterize over the time the typical delamination behavior. To cope with this shortcoming, we decided to rely on simulated data, and we created the experimental setup reported in Figure 12.

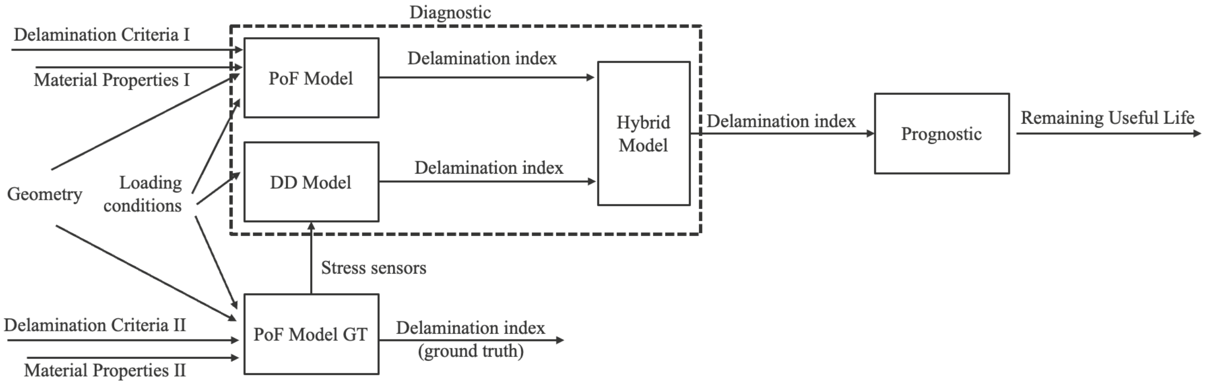


Figure 12. Setup describing the PHM framework for validation.

In this setup, besides the diagnostic and prognostic modules, a new PoF model GT (ground truth) is used to generate the stress sensor values and the *ground truth* delamination index. The PoF model for diagnostics and the PoF model GT are different. The difference between the models lies in the fact that they are created using different material properties and geometry. As a consequence, they generate different delamination indexes. This difference is fundamental since this allows us to represent a real scenario where the PoF model simulation provides a delamination index which is different from the one actually observed in the QFN.

The setup in Figure 12 is used to generate the validation dataset. This dataset is created by executing 29 run-to-failure simulations. The input of each simulation is the loading conditions, the QFN parameters, and the delamination criteria. These inputs can change within and across the simulations as follows:

- Within the same simulation, the loading condition, geometry, and material properties are the same, whereas the delamination criteria differ. The delimitation criteria are varied to introduce a slight difference between the behavior described by the PoF model GT and the PoF model. This choice is made because it is assumed that the interface strength at every interface will vary because of surface roughness, porosity in materials, and adhesion strength. Figure 13 shows an example of the variation in the delamination criteria which will result in a different delamination index for two samples with the same geometry.

- Across the simulations, the geometry is changed. During the development of the hybrid model, manufacturing tolerances are introduced in the simulations by changing the crucial geometry parameters such as solder thickness and EMC thickness. For instance, total QFN thickness (ideal thickness = 1.05 mm) varies between 1.005 mm and 1.095 mm by assuming solder and EMC thickness tolerance of +5%. Variations in the geometry combined together with variations in delamination criteria allow us to create a unique delamination behavior in every sample while imitating real-life uncertainties such as harsh environmental conditions and manufacturing defects.
- Across simulations, the loading condition, defined via a temperature profile depicted in Figure 14, is kept constant. The temperature profile starts at 185 °C to include the effect of the EMC curing process. Temperature variations are introduced at dwell time to mimic the real-life loading scenario. The thermal boundary condition is shown in Figure 14 where the loading profile starts at 185 °C to simulate the effect of EMC curing, and it is followed by thermal cycling. A thermal cycling profile of −40 °C to 125 °C is chosen since it is typically used to test the thermal reliability of electronic packages such as QFN and electronic assemblies. The stress sensors monitor the expansion and contraction of the material caused by the temperature profile. During the simulation, the stress values range between +25 MPa and –100 MPa.

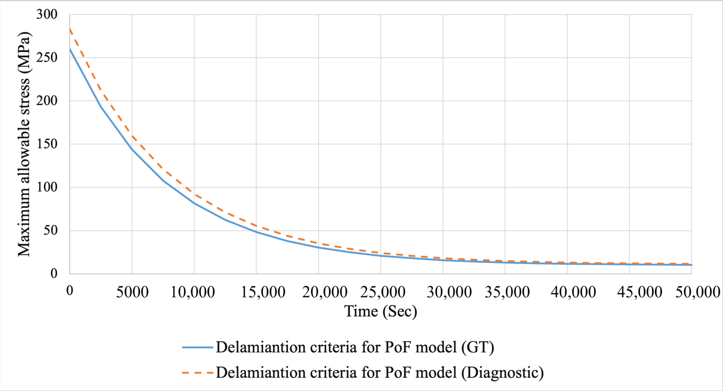


Figure 13. Variation in delamination criteria.

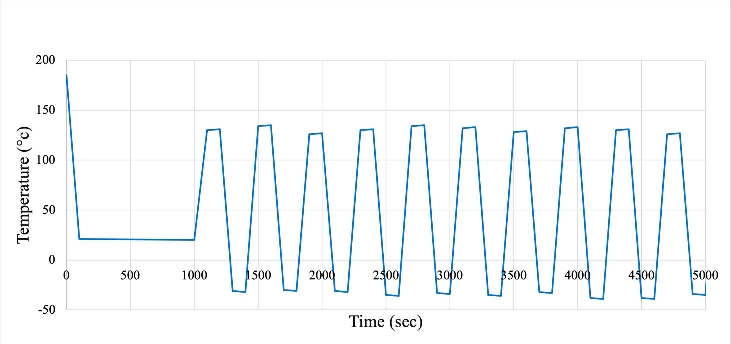


Figure 14. Thermal loading profile.

The evolution of the delamination percentages differs across simulations; although, in each case the delamination percentage is at least 90%. As a consequence, for the experiment, it was decided that a delamination index of 90% corresponds to the end of life (EoL) of the die (see Equation (1)). The validation dataset consists of the data generated from the 29 simulations. More specifically, from each simulation, the following data are retained:

- Temperature profile.
- Evolution delamination index predicted by the PoF model (which we call DI^{PoF}).
- Evolution delamination index generated by the PoF GT model (which we call $DI^{PoF\ GT}$).

- Evolution stress sensor values. These are four curves describing the evolution of the four stress sensor values generated by the PoF GT model.

5.3. Validation

The validation of the diagnostic module is performed in two steps.

In the first step, the most suitable machine learning regression model for the DD model is identified by performing a two-fold cross-validation ten times. Figure 15 illustrates how the DD model can be trained for one-fold validation. The candidate models tested are linear regression, random forest, and gradient boosting, and SVM regression, i.e., the commonly used machine learning algorithms for regression problems, which are less demanding in terms of data training size. At the end of the validation, to compare the performance of the regression models, the MSE is computed. Equation (6) shows the MSE associated with the *i*-th simulation as follows:

$$MSE(i) = \frac{1}{n}\sum_{i=1}^{n}(y_j - \hat{y}_j)^2 \tag{6}$$

where y_j and $\hat{y}_j$ represent the ground true and predicted delamination index for *i*-th simulation; n is the total number of data points in the *i*-th ground truth simulation. At the end of this step, the best performing algorithm is selected as a regression model of the DD model.

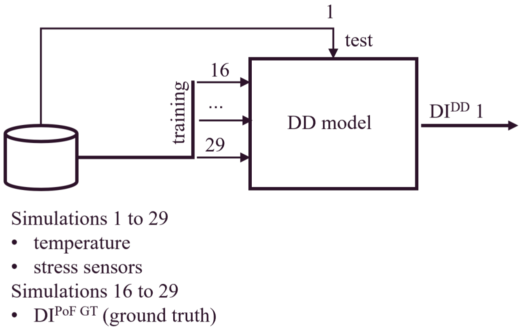


Figure 15. Example of training and testing of the DD model via two-fold cross validation. The DD model is trained using ground truth, temperature, and stress sensors from half of the simulations. The other half of simulations, with the exception of the ground truth, are used for testing. The output DI^{DD} 1 represents the DI curve computed by the model when it receives the test simulation 1 as input.

In the second step, the most suitable machine learning regression model for the hybrid model is identified by performing a *revisited* ten-fold cross-validation ten times. For each fold, five steps are followed as follows:

1. From the validation dataset, 1/10 of the simulations (i.e., 3 simulations) are used as tests. The remaining 26 simulations are split equally into training A and training B.
2. The 13 simulations of training A are used to train a single DD model. This model is then used to transform each test simulation data (i.e., the stress sensors and temperature) in a test delamination index which we call test DI^{DD}. Figure 16 shows how the first test simulation can be transformed by the DD model in test DI^{DD} 1.
3. The 13 simulations in training B are used to train 13 DD models using a leave-one-out approach. More specifically, 12 simulations are used to train a DD model, and the 13-th simulation is used to generate its associated DI. Figure 16 illustrates how simulation 14 can be transformed by the DD model 14 in the training delamination index which we call DI^{DD} 14.
4. The hybrid model is trained using the 13 DIs generated by the DD and PoF models. In this case, the same models are evaluated: linear, random forest, gradient boosting, and SVM regression models.
5. The model is used to combine test DIs generated by the DD and PoF models. The output of the model is the adjusted delamination index. The latter can consist of a

weighted sum of the DD and PoF's DIs if the model is implemented as a linear regression. For other models, the DD and PoF's DIs can be combined in more complex manners. Figure 16 shows for the first simulation, how the test DIs associated with the DD and PoF models generate the adjusted delamination index which we call DI^{Hybrid} 1. It is worth noting that by keeping the simulations in training A and B separate, the test and training DIs of the hybrid model pass through DD models which are not trained using common simulations. The latter prevents the training simulations from influencing the shape of the testing and training DIs, which may potentially lead to overly optimistic performance metrics due to the common type of reshaping introduced.

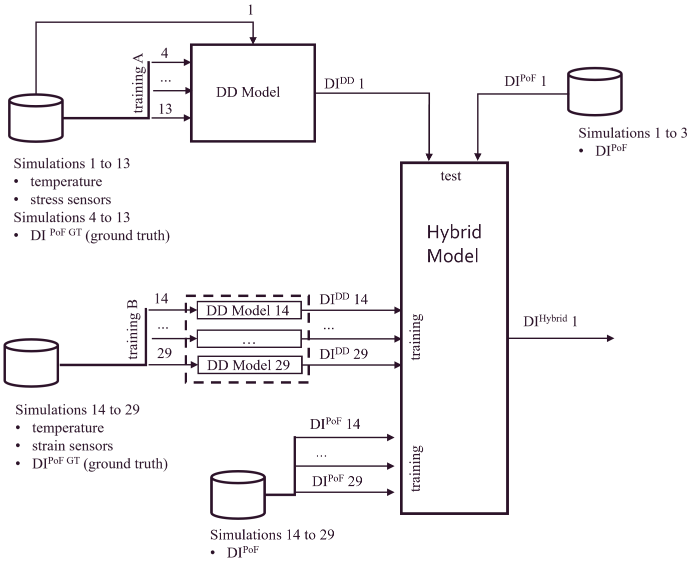


Figure 16. Example of training and testing of the hybrid model. The hybrid model is trained using DI curves generated from simulations in training B, whereas it is tested using DI curves generated from simulations in training A. The output DI^{Hybrid} 1 represents the DI curve computed by the hybrid model when it receives as input the testing DI curves. The latter two testing curves are generated by the DD and PoF models when the test simulation is 1.

At the end of the validation, the performance of the PoF, DD, and hybrid models are compared by computing the MSE.

To validate the prognostic module, a leave-one-out cross-validation is performed following these steps:

1. From the validation dataset, the data associated with the i-th simulation are retained and used to generate the curve test DI. Figure 17 illustrates how the first test simulation can be transformed by the hybrid model in test DI^{Hybrid} 1. The remaining simulations are used to compute via hybrid models the N-1 DI curves which will form the library of DI curves (see Section 4.2.1). These hybrid models are trained using a leave-two-out approach. More specifically, the hybrid model j (which generates the library DI curve j) is trained using all other DI curves except for the j-th and i-th ones. A similar approach is used for training the DD model within each hybrid model.
2. The RUL is computed for the curve test DI any time that the DI has a step increase of 0.1. The 0.1 threshold for computing the RUL is decided taking into account two aspects. On the one hand, if the threshold is too high, then the selectable step changes are too few, and this would hinder the possibility of highlighting the presence of a trend in the forecast performance. On the other hand, if the threshold is too low, then the selectable step changes may become too many. As a result, it would become more challenging to abstract a trend across the test curves. At the end of the validation, to evaluate the accuracy of the forecasted RUL, the relative error (RE) and absolute

relative error (|RE|) are used. These metrics are chosen since they are commonly used for performance measures of the RUL. Equation (7) reports how RE is computed:

$$RE = \frac{E\hat{o}L - EoL}{EoL} \tag{7}$$

where EoL and $E\hat{o}L$ represent the true and predicted EoL.

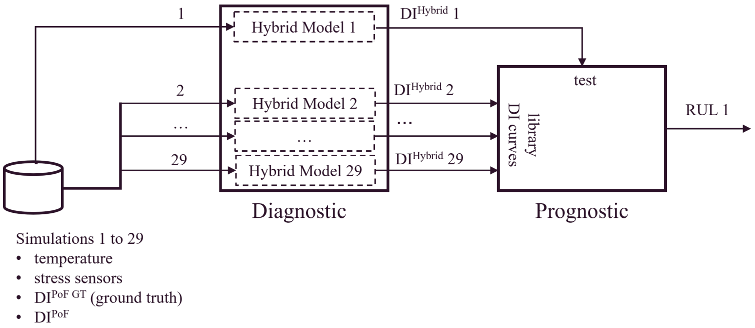


Figure 17. Example of training and testing of the prognostic module. Each hybrid model j is trained using all other DI curves except for the DI curves *j*-th and 1. The output RUL 1 is provided by the prognostic module when it receives as input the testing DI curve DI^{Hybrid} 1. The latter is generated by the diagnostic model when it receives as input the test simulation 1.

6. Results and Discussion

This section evaluates to what extent the instantiated methodology is suitable for the diagnostics and prognostics of the health index of a QFN package.

6.1. Diagnostic

This section investigates the performance of the PoF, DD, and hybrid models in monitoring the level of the package delamination. As a first step, the number of available run-to-failure simulations is evaluated to verify whether it is sufficient for training the DD model.

Figure 18 reports the learning curve of the gradient boosting regression model used for the DD model. In this figure, once the training size goes above 10, it is possible to notice that both learning curves become flat. This means that (i) adding more samples would have only a little impact on the performance of the model, and (ii) the model is not overfitting. The learning curves of the other models (reported in the Appendix A) exhibit similar behaviors confirming that even with a limited number of run-to-failure simulations, there exist multiple machine learning algorithms suitable for training a DD model robust against overfitting.

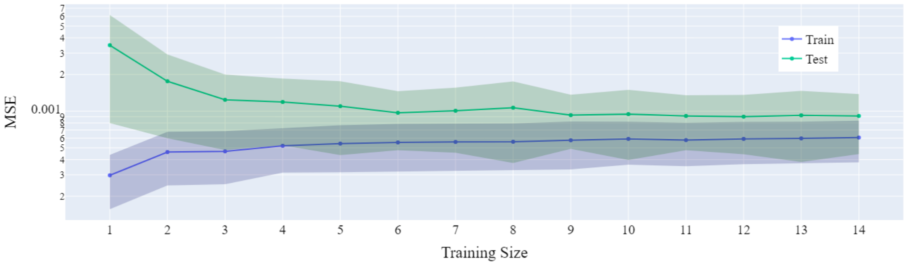


Figure 18. Learning curve of the gradient boosting regression model used for implementing the DD model. The green and blue curves represent the mean MSE for the test and train set, respectively. The transparent area corresponds to one standard deviation above and below the mean value.

Figure 19 reports the performance of the DD model when implemented using different machine learning regression models. Among the models, the gradient boosting is the best performing one, and for this reason, it is selected to implement the DD model. As a second step, we replicate the previous analysis to select the machine learning regression model for the implementation of the hybrid model. Figure 20 reports the performance of the hybrid model when implemented using different machine learning regression models. In this case, the gradient boosting and linear models are the best performing ones. By comparing their learning curves in Figures 21 and 22, it is possible to observe that the linear regression model is already robust against overfitting once trained with three or more samples. Furthermore, in Figure 20, if we use the median values as reference points, it is possible to observe that multiple regression models (e.g., linear regressor, gradient boosting regressor) have lower MSE than the PoF and DD models. This result suggests that the hybrid model for monitoring the health index of the package is able to achieve better performance than the ones obtained by the PoF and DD models alone.

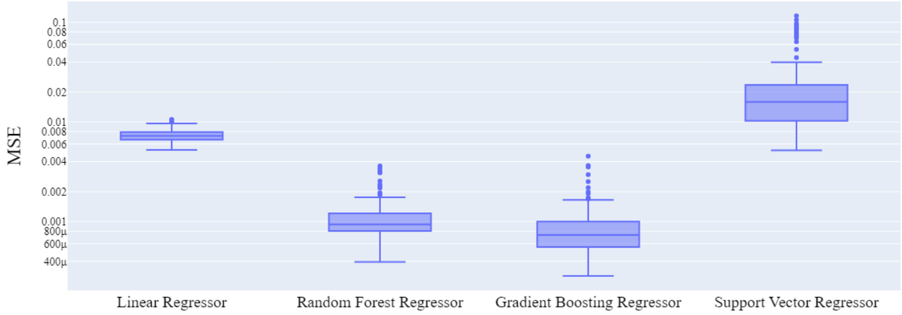


Figure 19. The diagnostic performance achieved by the different regression models (linear, random forest, gradient boosting and SVM) used for implementing the DD model.

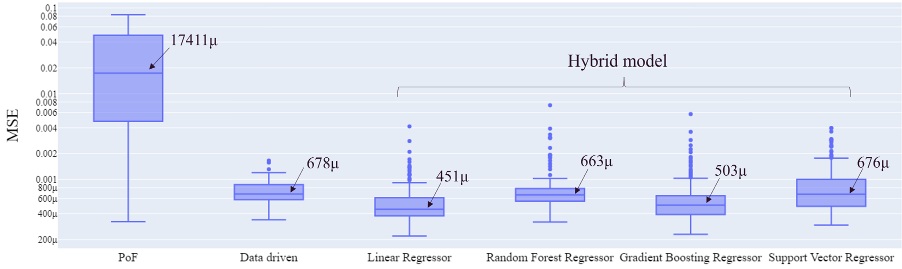


Figure 20. The diagnostic performance achieved by the PoF, DD, and hybrid model. Linear, random forest, gradient boosting, and SVM regression models are considered for the hybrid model.

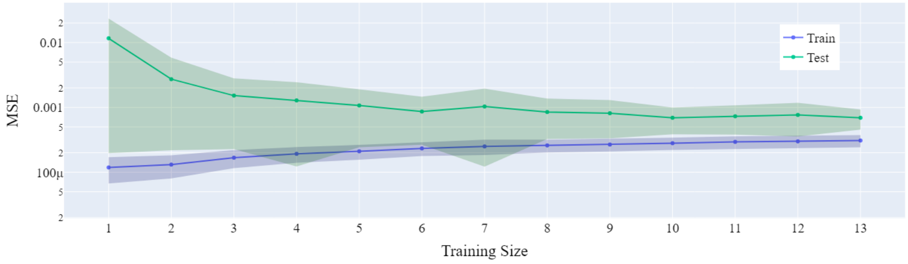


Figure 21. Learning curve of the gradient boosting regressor used for implementing the hybrid model.

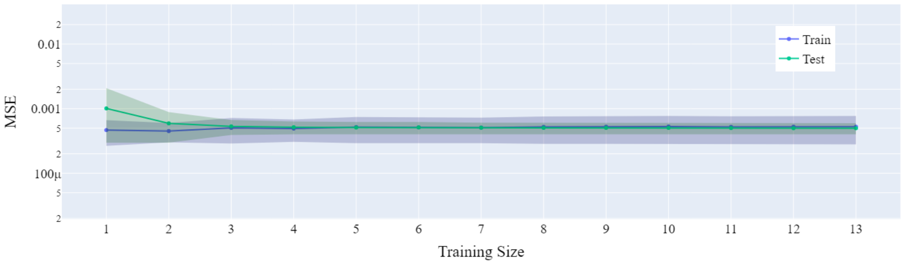


Figure 22. Learning curve of the linear regression model used for implementing the hybrid model.

6.2. Prognostic

This section investigates the performance of the prognostic model in forecasting the remaining useful life of the package. For the analysis, only the linear regression model is used for the hybrid model.

Figure 23 presents the boxplots of the |RE| for the first four step changes of each test curve. In this figure, by focusing on the median values, it is possible to observe a downward trend which highlights that the forecasts become better over time. The reason behind these results depends on the selection process of the top five reference curves. In the first step change, where the highest median value of 14% is reached, the test curve has a limited time span. Consequently, the test curve does not hold yet a recognizable pattern that can be exploited for the retention of the most suitable reference curves, namely the ones which eventually evolve into the test curve. This issue tends to reduce once the test curve has more step changes, i.e., recognizable patterns, which increases the chances to retain look-alike reference curves.

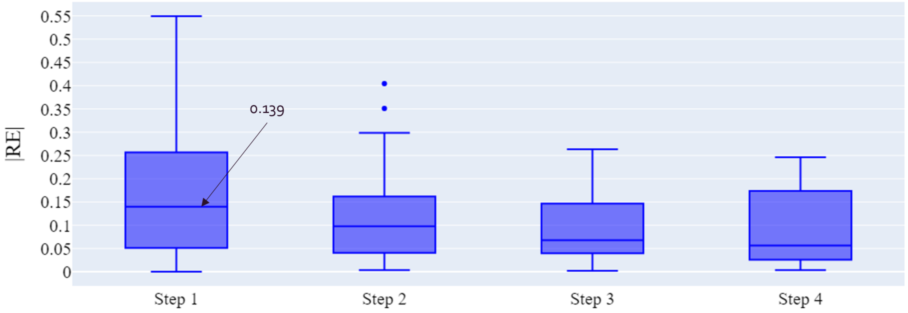
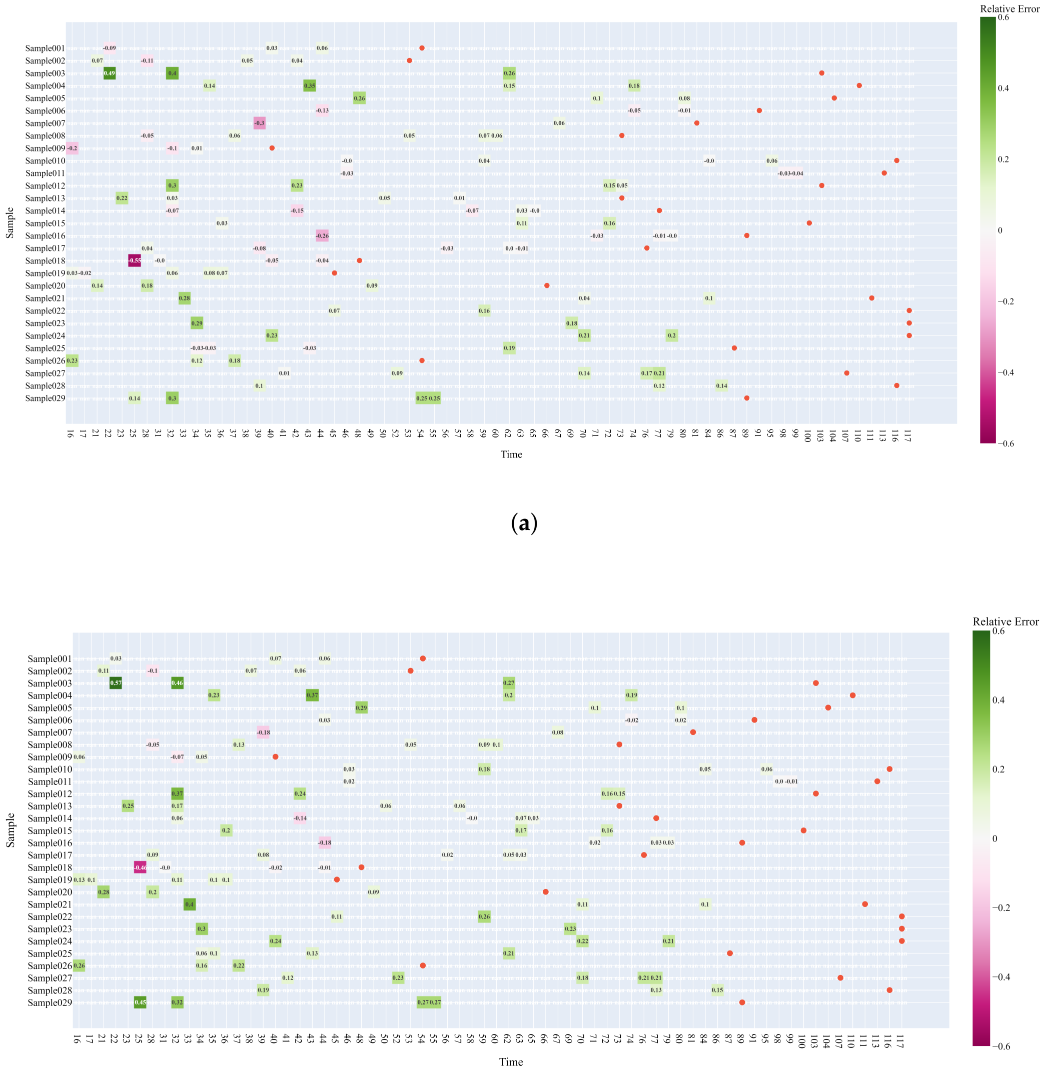


Figure 23. Evolution of the prognostic performance for the first four step changes of each test curve using Equation (3).

To better understand the improvement in performance forecast over time, Figure 24a provides, for each single test curve and for each step change, the associated RE based on Equation (3). In this figure, it is possible to notice that for 33 out of 103 forecasts, i.e., 32% of the cases, the RE is above 0, namely the forecast goes beyond the real end of life. Since late forecasts may represent major problems (e.g., for safety-critical applications), Figure 24b shows the RUL forecast using a more conservative estimation based on Equation (4). In this case, only 13% of the forecasts are delayed, and yet, the highest median value of RE, for the first four step changes of each test curve, reaches just 18%. To the best of the authors knowledge, this is the first study to evaluate a hybrid method for an RUL forecast at the package level. While further analysis with field data is needed, our results provide a reference point of the accuracy level in the RUL forecast for the package. This is a fundamental first step to verify to what extent hybrid-based methods for the RUL forecast can provide actionable information to implement a predictive maintenance strategy.



(**a**)

(**b**)

Figure 24. Evolution of the prognostic performance for each step change of each test curve. The y-axis reports the test curves, whereas the x-axis reports the time stamps uniformly binned (in time windows of 41 time stamps). The color and number of each square highlight to what extent the RUL forecast is delayed or early. The red circle reports the true end of life of the package. Figure (**a**) reports the prognostic performance using Equation (3); figure (**b**) reports the prognostic performance using Equation (4).

7. Conclusions and Future Work

In the electronic domain, techniques for prognostic and health management typically rely on either physics-based or data-driven models. Despite the advantages of combining both models, only a limited number of studies explored hybrid models, and none of them have applied these models for RUL forecasting at the package level. For this reason, this study delves into the potential of hybrid modeling for PHM of electronic systems. The key contribution of this research is the introduction of a novel framework for diagnostics and prognostics of the health index of an electronic system using as a case study a QFN package subject to delamination.

This study proves that, even with a limited number of run-to-failure simulations, hybrid modeling can be used to improve the original diagnostic capabilities of data-driven and physics-of-failure models alone. For prognostics, this study addresses the existing research gap by establishing a performance benchmark for the estimation of the remaining useful life of the package. While the preliminary results are promising, the next step of this study is to validate the methodology on real field data to ensure its robustness and performance accuracy. More specifically, we aim to collect more extensive datasets which include normal behavior data, data under various failure conditions, and data coming from different electronic components.

Author Contributions: Conceptualization, A.M., C.H., E.T., C.N. and B.V.; data curation, A.M., C.H., C.N. and B.V.; formal analysis, A.M., C.H. and C.N.; investigation, A.M., C.H. and C.N.; methodology, A.M. and C.H.; project administration, A.M., E.T. and B.V.; software, C.H. and C.N.; supervision, A.M., validation, A.M. and C.H.; writing—original draft, A.M.; writing—review and editing, A.M., C.H., C.N., B.V. and E.T. All authors have read and agreed to the published version of the manuscript.

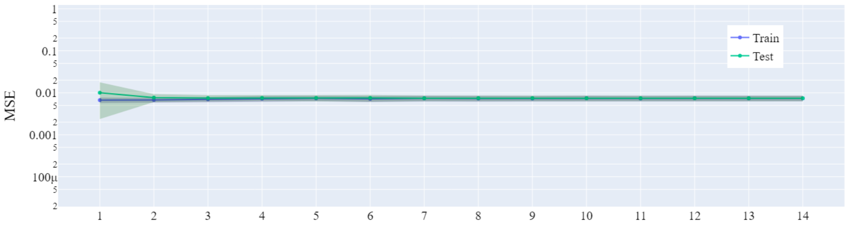
Funding: This research was funded by the Flemish Government through the AI Research Program; by the Region of Bruxelles-Capitale—Innoviris; and by VLAIO (Flanders Innovation & Entrepreneurship) through the project iRel40. iRel40 is a European co-funded innovation project, granted by the ECSEL Joint Undertaking (JU) under grant agreement No 876659. The funding of the project comes from the Horizon 2020 research program and participating countries. National funding is provided by Germany, including the Free States of Saxony and Thuringia, Austria, Belgium, Finland, France, Italy, the Netherlands, Slovakia, Spain, Sweden, and Turkey.

Data Availability Statement: The dataset containing the simulations used in this work can be downloaded at the following address: https://figshare.com/s/89aad43135daca13a581, accessed on 30 January 2024.

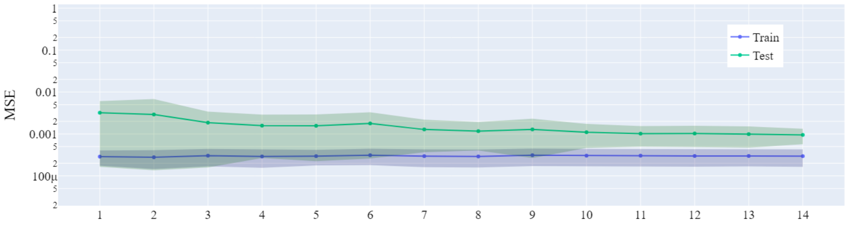
Conflicts of Interest: Author Chinmay Nawghane and Bart Vandevelde were employed by the company Imec. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A. Learning Curves

This section reports the learning curves for all machine learning regression models tested for the DD and hybrid models.

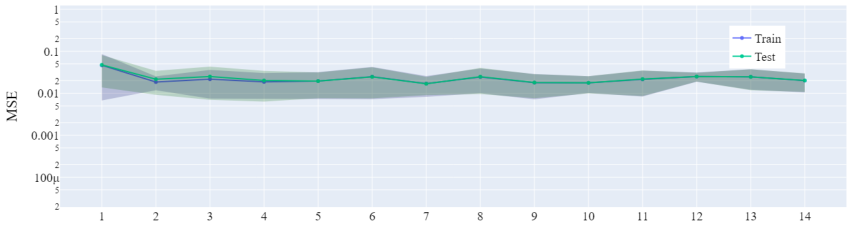


(**a**)



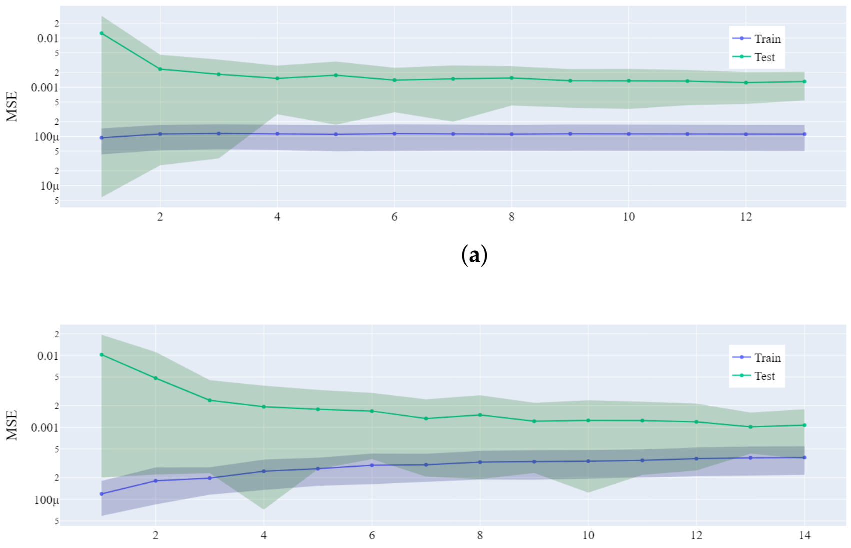
(**b**)

Figure A1. *Cont.*



(**c**)

Figure A1. Learning curves for the regression models tested for implementing the DD model. The green and blue curves represent the mean MSE for the test and train set, respectively. The transparent area corresponds to one standard deviation above and below the mean value. Figure (**a**) represents the Linear regression model; figure (**b**) represents the Random Forest regression model; figure (**c**) represents the Support Vector regressor.



(**a**)

(**b**)

Figure A2. Learning curves for the regressors tested for implementing the hybrid model. The green and blue curves represent the mean MSE for the test and train set, respectively. The transparent area corresponds to one standard deviation above and below the mean value. Figure (**a**) represents the Random Forest regression model; figure (**b**) represents the Support Vector regression model.

References

1. Zhang, G.Q.; Van Driel, W.D.; Fan, X.J. *Mechanics of Microelectronics*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2006; Volume 141.
2. Pecht, M. Prognostics and health management of electronics. In *Encyclopedia of Structural Health Monitoring*; Wiley Online Library: Hoboken, NJ, USA, 2009.
3. Vichare, N.M.; Pecht, M.G. Prognostics and health management of electronics. *IEEE Trans. Components Packag. Technol.* **2006**, *29*, 222–229. [CrossRef]
4. Prisacaru, A.; Gromala, P.J.; Jeronimo, M.B.; Han, B.; Zhang, G.Q. Prognostics and health monitoring of electronic system: A review. In Proceedings of the 2017 18th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE), Dresden, Germany, 3–5 April 2017; pp. 1–11. [CrossRef]
5. Kulkarni, C.S.; Celaya, J. Electronics Prognostics. In *Fault Diagnosis of Dynamic Systems: Quantitative and Qualitative Approaches*; Springer International Publishing: Cham, Switzerland, 2019; pp. 433–458.
6. Zhang, Y.; Yan, Z.; Zhu, H.; Tang, P. Physics-based model and data dual-driven approaches for predictive evacuation. *Dev. Built Environ.* **2023**, *16*, 100269. [CrossRef]
7. Kumar, S.; Torres, M.; Chan, Y.C.; Pecht, M. A hybrid prognostics methodology for electronic products. In Proceedings of the 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–8 June 2008; pp. 3479–3485. [CrossRef]
8. Karpatne, A.; Watkins, W.; Read, J.; Kumar, V. Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling. *arXiv* **2017**, arXiv:1710.11431.

9. Frank, S.; Heaney, M.; Jin, X.; Robertson, J.; Cheung, H.; Elmore, R.; Henze, G. *Hybrid Model-Based and Data-Driven Fault Detection and Diagnostics for Commercial Buildings*; Technical report; National Renewable Energy Lab. (NREL): Golden, CO, USA, 2016.
10. Forssell, U.; Lindskog, P. Combining Semi-Physical and Neural Network Modeling: An Example ofIts Usefulness. *IFAC Proc. Vol.* **1997**, *30*, 767–770. [CrossRef]
11. Hanachi, H.; Yu, W.; Kim, I.Y.; Mechefske, C. Hybrid Physics-Based and Data-Driven PHM. In Proceedings of the Canadian Machinery Vibration Association (CMVA) Annual Conference 2017, Edmonton, AB, Canada, 25–27 October 2017.
12. Sun, J.; Zuo, H.; Wang, W.; Pecht, M.G. Prognostics uncertainty reduction by fusing on-line monitoring data based on a state-space-based degradation model. *Mech. Syst. Signal Process.* **2014**, *45*, 396–407. [CrossRef]
13. Patil, N.; Das, D.; Goebel, K.; Pecht, M. Identification of Failure Precursor Parameters for Insulated Gate Bipolar Transistors (IGBTs). In Proceedings of the 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, 6–9 October 2008; pp. 1–5. [CrossRef]
14. Cheng, S.; Pecht, M. A fusion prognostics method for remaining useful life prediction of electronic products. In Proceedings of the 2009 IEEE International Conference on Automation Science and Engineering, Bangalore, India, 22–25 August 2009; pp. 102–107. [CrossRef]
15. Riegel, D.; Gromala, P.J.; Han, B.; Rzepka, S. Data-Driven Remaining Useful Life Prediction of QFN Packages on Board Level with On-Chip Stress Sensors. In Proceedings of the 2021 IEEE 71st Electronic Components and Technology Conference (ECTC), San Diego, CA, USA, 1 June–4 July 2021; pp. 904–909. [CrossRef]
16. Prisacaru, A.; Gromala, P.J.; Han, B.; Zhang, G. Degradation Estimation and Prediction of Electronic Packages Using Data-Driven Approach. *IEEE Trans. Ind. Electron.* **2022**, *69*, 2996–3006. [CrossRef]
17. Bhargava, C.; Sharma, P.K.; Senthilkumar, M.; Padmanaban, S.; Ramachandaramurthy, V.K.; Leonowicz, Z.; Blaabjerg, F.; Mitolo, M. Review of Health Prognostics and Condition Monitoring of Electronic Components. *IEEE Access* **2020**, *8*, 75163–75183. [CrossRef]
18. Gan, C.L. Prognostics and Health Management of Electronics: Fundamentals, Machine Learning, and the Internet of Things: John Wiley & Sons Ltd. (2018). pp. 731, ISBN: 9781119515326 (Print), 9781119515326 (Online). *Life Cycle Reliab. Saf. Eng.* **2020**, *9*, 225–226.
19. Lei, W.S.; Kumar, A. Delamination and reliability issues in packaged devices. In *Adhesion in Microelectronics*; Wiley Online Library: Hoboken, NJ, USA, 2014; pp. 267–312.
20. Schindler-Saefkow, F.; Rost, F.; Otto, A.; Faust, W.; Wunderle, B.; Michel, B.; Rzepka, S. Stress chip measurements of the internal package stress for process characterization and health monitoring. In Proceedings of the 2012 13th International Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems, Cascais, Portugal, 16–18 April 2012; pp. 1/10–10/10. [CrossRef]
21. Aizpurua, J.I.; Catterson, V.M. Towards a methodology for design of prognostic systems. In Proceedings of the Annual Conference of the PHM Society, Coronado, CA, USA, 18–24 October 2015; Volume 7.
22. Hutton, D.V. *Fundamentals of Finite Element Analysis*; The McGraw Hill Companies: New York, NY, USA, 2004.
23. Yu, W.; Kim, I.Y.; Mechefske, C. Remaining useful life estimation using a bidirectional recurrent neural network based autoencoder scheme. *Mech. Syst. Signal Process.* **2019**, *129*, 764–780. [CrossRef]
24. Wang, T.; Yu, J.; Siegel, D.; Lee, J. A similarity-based prognostics approach for remaining useful life estimation of engineered systems. In Proceedings of the 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, 6–9 October 2008; pp. 1–6.
25. Khelif, R.; Malinowski, S.; Chebel-Morello, B.; Zerhouni, N. RUL prediction based on a new similarity-instance based approach. In Proceedings of the 2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE), Istanbul, Turkey, 1–4 June 2014; pp. 2463–2468.
26. Yu, W.; Kim, I.Y.; Mechefske, C. An improved similarity-based prognostic algorithm for RUL estimation using an RNN autoencoder scheme. *Reliab. Eng. Syst. Saf.* **2020**, *199*, 106926. [CrossRef]
27. Bellman, R.; Kalaba, R. On adaptive control processes. *IRE Trans. Autom. Control* **1959**, *4*, 1–9. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.