A Self-Adaptive Physics-Informed Gated Recurrent Unit Neural Networks Model for Estimating the Lifetime of Li-ion Batteries

Mohammad AlShaikh Saleh¹, Alamera Nouran Alquennah¹, Ali Ghrayeb¹, Shady S. Refaat¹, Haitham Abu-Rub¹, and Sunil P. Khatri¹

¹Affiliation not available

June 07, 2024
A Self-Adaptive Physics-Informed Gated Recurrent Unit Neural Networks Model for Estimating the Lifetime of Li-ion Batteries

Mohammad AlShaikh Saleh, Member, IEEE, Alamera Nouran Alquennah, Member, IEEE, Ali Ghrayeb, Fellow, IEEE, Shady S. Refaat, Senior Member, IEEE, Haitham Abu-Rub, Fellow, IEEE, and Sunil P. Khatri, Senior Member, IEEE,

Abstract—Physics-Informed Neural Networks (PINNs) have recently emerged as a promising approach for applying deep neural networks to solve partial differential equations (PDEs). However, accurately addressing challenging regions in the solutions of stiff PDEs necessitates adaptive methods. Additionally, the inherent limitations of baseline PINN in handling sequential or time-series data significantly constrain their applicability. In light of this, this paper introduces a Self-Adaptive Physics-Informed Attention-based Gated Recurrent Unit (SA-PI-AGRU) model, which enhances the baseline PINN framework to address these critical issues. The proposed SA-PI-AGRU model advances PINNs by integrating an attention-based GRU layer, which is particularly effective at analyzing sequential data. This dual objective of minimizing losses while optimizing the weighting parameters ensures a robust training and testing process, which can be used for many applications, including language modelling and text generation, prognostics and health management, and other prediction/forecasting problems. The efficacy of the SA-PI-AGRU model is demonstrated through an essential case study, which is to predict the state of health (SoH) of lithium-ion batteries (LIBs), utilizing four different battery datasets from the National Aeronautics and Space Administration (NASA). The obtained results suggest significant improvements in predictive accuracy and network initialization capabilities compared to the baseline PINN and other benchmark models.

Index Terms—Attention mechanism, battery management systems, Gated recurrent unit, Li-ion batteries, physics-informed neural networks.

I. INTRODUCTION

A. Background and Motivation

The emerging area of physics-informed neural networks (PINNs), also known as scientific machine learning, combines data-driven and theory-driven methods to address science and engineering problems [1], [2]. PINN algorithms are informed by strong prior knowledge in the form of differential equations relating the variables in the problem. Unlike traditional numerical methods, they do not require elaborate mesh grids, produce differentiable solutions, and can assimilate data easily. PINNs are able to extrapolate to regions where there is little or no data, since they also rely on physical knowledge and unlike traditional scientific computation, any missing physics may be learned from sample data. However, the problem with the current PINN models is that they fall short when it comes to predicting long-term temporal sequential data applications. Therefore, it is worth investigating the possibility of incorporating recurrent-based NN (RNN) models such as long short-term memory (LSTM) or gated recurrent units (GRU) and their variants, into the PINNs.

The GRU model is devised to deal with the issue of vanishing gradients in conventional RNNs. An RNN is a type of neural network that is used to process sequential data, such as time series or natural language [3]. In RNNs, the information from previous time steps can be lost as it is passed through the network. This is known as the vanishing gradient problem, and it makes it difficult to train RNNs on long sequences of data. In light of this, the GRU was designed to handle sequential data effectively, capturing long-range dependencies in time-series data.

Furthermore, attention-based DL models are a type of neural network architecture that utilizes the Attention Mechanism (AM) to improve the performance of sequential data processing tasks such as natural language processing and speech recognition. AMs allow the model to prioritize various segments of the input sequence by assigning different weights, thereby enabling it to concentrate on the most pertinent parts of the input for making predictions [4]. Such a process is achieved by obtaining different attention scores for each time step of the input sequence, which are then used to allocate weights to the hidden states of the GRU layers. The attention scores are typically computed using a feed-forward neural network (FNN) that takes the hidden states of the GRU layers as input. In light of this, this paper introduces a Self-Adaptive Physics-Informed Attention-based Gated Recurrent Unit (SA-PI-AGRU) model, which enhances the baseline PINN framework by integrating an attention-based GRU layer, which is particularly effective at analyzing sequential data. The proposed model is projected to perform efficiently and accurately for many applications.
involving prediction and forecasting problems that possess sequential/long term time-series data, where the baseline PINNs would fall short for prediction. Among these applications are battery prognostics and health management (BPHM).

Studies indicate that retired lithium-ion batteries (LIB) from EVs will exceed a total mass of over 12 million tons by 2030, underscoring the significance of promoting BPHM systems for ensuring optimal safety and reliability while operating these batteries [5]. Thus, BPHM’s main goal is to oversee, forecast, and uphold the lifetime and functionality of LIBs over their operational lifespan by utilizing a variety of algorithms and tools [6]. The crucial internal states of batteries such as state of charge, state of health, and state of power have a substantial impact on the safety, effectiveness, and overall durability of battery storage systems within BPHM [7]. Consequently, there has been an increased focus on developing advanced algorithms and techniques for effective BPHM [8].

Lithium-ion batteries (LIB) are currently one of the most significant advancements due to their exceptional safety features, high energy density, long lifespan, and recyclability. The battery market was valued at 36.7 billion USD in 2019 and is predicted to culminate up to a whopping 128.3 billion USD by 2027, with a predicted annual growth rate of 18% from 2020 to 2027. This growth is mainly driven by the transition from traditional combustion engine vehicles to hybrid and electric vehicles (EV) [9], [10]. LIBs contributes heavily in facilitating the adoption of various technologies such as EVs, portable electronic devices, renewable energy deployment, and grid stability uphold [11]. However, the anticipated expansion of LIB usage highlights the crucial necessity for reliable assessment and estimation of battery performance metrics because complex aging mechanisms during charging cycles can reduce their longevity [12].

The proposed model incorporates a battery-based PDE-constrainer physics degradation model for the AGRU (data-driven component). Moreover, this paper incorporates Self-Adaptive PINNs (SA-PINNs), a novel technique to train PINNs adaptively, which was introduced by [2]. A defining characteristic of SA-PINNs is that even though the loss function is minimized with respect to network weights w through gradient descent, “as is customary”, it is maximized with respect to the self-adaptive weights λr and λb through gradient ascent [2]. The model’s performance demonstrated significant improvements in predictive accuracy relative to the baseline PINN and other benchmark models. The self-adaptive and AGRU layers contributed heavily to the adaptation of the battery degradation dynamics, where the proposed model achieved the smallest standard deviation in the error metrics across repeated training and testing cycles for four different battery datasets. The model’s feasibility and scalability arise from the fact that it accounts for the impact of different initial network weights and biases better than the baseline PINN.

B. Contributions

The main contributions are summarized as follows:

• A Self-Adaptive Physics-Informed Attention-based Gated Recurrent Unit (SA-PI-AGRU) model is proposed. This model represents a significant advancement in the estimation of the SoH of LIBs, focusing on achieving enhanced accuracy.

• The SA-PI-AGRU incorporates an attention layer with GRU (instead of baseline NNs) to allow effective handling and analysis of sequential battery usage data, improving the model’s ability to focus on relevant temporal patterns critical for accurate SoH estimation. The model’s performance is further validated against benchmark shallow and deep learning models, showcasing significant improvements in predictive accuracy, with enhancements ranging from 0.60% to 5.61% over the baseline PINN and other benchmark models.

• A self-adaptive layer is considered as it tunes the weights of each loss component for different epochs, adapting better to the dynamics of battery degradation. Therefore, achieving a small standard deviation in the error metrics across repeated training and testing cycles for the different battery datasets is essential to prove the viability of the proposed self-adaptive layer. Accordingly, the standard deviation values acquired were 0.0873 for the MAE, 0.0690 for R², and 0.0436 for the RMSE, highlighting the model’s feasibility and scalability as it accounts for the impact of different initial network weights and biases better than the baseline PINN.

• The model’s generalization capabilities is tested and ensured on various battery conditions/datasets along with a case study that assesses the proposed model’s ability to yield accurate results in spite of repeated network training initialization. The latter is essential as the initial network weights and biases at the start of the training can have an impact on the overall results, therefore acquiring a small standard deviation in the error metrics across the repeated training and testing network initialization cycles is ideal to ensure model feasibility and scalability.

C. Paper Outline

The rest of the paper is organized as follows: Sections 2 reviews the related works. Section 3 provides an overview of PINN, self-adaptive PINN, and AGRU. Section 4 introduces our application of the proposed SA-PI-AGRU training and testing scheme to LIBs. Section 5 demonstrates the experimental results and performance evaluation of the proposed model. Finally, Section 6 concludes the paper.

II. RELATED WORKS

A. Current Advancements of PINN

As of now, PINN methods primarily use deep neural networks and Gaussian processes conditioned by PDEs and initial/boundary conditions. The foundational technique involves constraining a deep neural network’s output through the loss function to penalize deviations from the PDE, approximating satisfaction post-training [2]. Moreover, the continuous PINN algorithm introduced in [1], also known as the “baseline” PINN, is proficient at predicting solutions that are adequately smooth given computationally manageable initial and boundary conditions. However, the baseline PINN was tested by different
researchers and have observed computational errors and issues pertaining to convergence and accuracy. This was especially the case when using stiff PDEs as mentioned by [2]. Such stiff PDEs come with sharp spatial transitions or rapid temporal evolution [13]. This issue arises, for instance, when solving the nonlinear Allen-Cahn PDE inside the PINN model [14]. Consequently, Self-Adaptive PINNs (SA-PINNs), a novel approach for adaptively training PINNs, was introduced by [2]. SA-PINNs employ trainable weights for each training point, which was inspired by the soft multiplicative attention masks utilized for computer vision tasks [15]. These SA weighting parameters are trained simultaneously with the NN weights \( w \), thereby automatically giving more attention to the residual component (including the initial and boundary conditions) in challenging locations of the output within the loss function, improving the approximation at those points.

The issue of rapid temporal evolution is particularly critical in forward problems, where all data are restricted to the initial and boundary components, and no data is made available within the PDE’s domain. For instance, in time-evolution problems, advancing the behaviour of the physical problem at hand, from the given initial point to the next points during neural network training can be challenging, as several authors have observed [2], [14], [16], [17]. An approach proposed in [14], have postulated the possibility of propagating the time-dependent features forward in time by decreasing the time steps and sequentially training the PINN model accordingly, starting with the time step closest to the initial point. However, this method drastically increases the computational time and will require additional training data, which is usually unavailable when it comes to real industrial settings. Consequently, to solve this time-dependency issue without requiring additional data and decreasing the time step, modifying the baseline PINN architecture by leveraging a recurrent-based neural network method is proposed in this paper, instead of using the conventional NN as the data-driven component.

B. Current Advancements of AI-based Lifetime Estimation Models for LIB

Li et al. [18] devised a new approach that combines LSTM with the Electrochemical Model (ECM) to identify faults in lithium-ion batteries (LIBs) in electric vehicles (EVs). The ECM delivers a comprehensive understanding of the battery’s internal dynamics, improving the accuracy of battery diagnostics. Yet, this method was firstly employed for simulations or laboratory settings, with limited real-world validation in EV applications. Wang et al. [17] introduced a Conditional Temporal Convolutional Encoder-Decoder (CTCED) to estimate the current capacity of LIBs given varying conditions, incorporating different battery types and environmental changes. The CTCED model, being non-recursive, delivers quicker performance on advanced hardware tailored for vectorized operations, though it requires higher GPU RAM power when training, particularly with longer data sequences. Furthermore, they proposed a CNN-LSTM-DNN model for Remaining Useful Life (RUL) prediction that employs an adaptive and nonlinear technique [19].

Experimental validation with datasets from NASA and the Center for Advanced Life Cycle Engineering shows that this method outperforms individual machine learning techniques in terms of predictive accuracy, providing an enhanced accuracy and reasonable execution times. Yet, this technique’s dependence on real data may overlook unexpected variations or anomalies in the battery’s degradation profile, such as heat and aging. Additionally, a calendar-based health prognostic model that incorporates knowledge-data-driven attention has been proposed in [5], [6]. This model incorporates empirical battery knowledge, leading to a significant enhancement in prognostic performance especially under unforeseen conditions. Nevertheless, potential biases in semi-empirical models within this framework may require further refinement. Wen et al. [20] introduced a PINN framework that combines uncovered dynamics information with a surrogate neural network and an enhanced Verhulst’s battery degradation PDE model. Additionally, an uncertainty-based adaptive weighting method is utilized to solve this multi-objective problem during PINN training.

The problem so far is to predict accurately the fast-changing dynamics of battery degradation using a more robust PINN model. Since, aging time is an essential parameter that is normally not taken into account using the surrogate NN (integrated with PINN), it is worth exploring the possibility of integrating RNN-based variants into the PINN framework to capture accurately the temporal correlations during battery degradation. To this end, this paper proposes a self-adaptive physics-informed gated recurrent unit model using a soft attention mechanism to enhance the battery lifetime predictive accuracy.

III. PROBLEM SETTING

In this section, we provide a brief overview of PINNs, self-adaptive weights, and the AGRU.

A. Physics-Informed Neural Networks

During forward modelling, the initial and boundary conditions are assumed to be known, bringing us to the common PDE solving problem, where PINN is leveraged as a conventional numerical solver approach. With that, let the unknown \( u(x, t) \) be determined by an \( \text{ansatz} \) \( u(x, t; w) \), comprising a NN with input features \( x = (x_1, ..., x_d) \), time \( t \) to study the temporal correlations, and network weights \( w \). Merely one output is approximated (regression problem) as \( u \) is a scalar function. However, given a PDE system, multiple outputs can be acquired, one for each variable. To this end, the objective is to obtain network weights \( w \) such that [2]

\[
F(u(x, t; w), x, t, Du(x, t; w), D^2u(x, t; w), ..., D^k u(x, t; w); \lambda) = f(x, t). \quad x, t \in \Omega
\]

The partial derivatives \( D^p u(x; w) \) of the NN output, for \( x \) and \( w \) at time \( t \) in function \( F \), is calculated using automatic differentiation methods. Therefore, it is worth noting the loss
functions as \[2\]

\[
\mathcal{L}_u(w) = \int_\Omega |\hat{u}(x, t) - u(x, t)|^2 \, dx,
\]

\[
\mathcal{L}_r(w, \lambda) = \int_\Omega |F(u(x, t; w), x, t, Du(x, t; w), D^2u(x, t; w), \cdots, D^k u(x, t; w); \lambda) - f(x, t)|^2 \, dx,
\]

\(\mathcal{L}_u\) is the mean-squared error (MSE\(_u\)) that the NN incorporates to forecast the initial and boundary conditions, as well as utilizing training data for calibration represented via \(\{x_i^r, t_i^r, u_i \}_{i=1}^{N_r}\) \(\lambda\) with \(\hat{u}\) defined as the predicted label based on the input \(x_i^u\) at time \(t_i^u\) and compared with the true label \(u^i\). \(\mathcal{L}_r\) is the MSE\(_r\), which consists of evaluations of the residual/physics function \(f\) over \(\{x_i^r, t_i^r\}_{i=1}^{N_r}\) with input \(x_i^r\) at time \(t_i^r\) \([21, 22]\).

Conventionally, the integrals in Equation (2) is numerically approximated by constructing Monte-Carlo estimates through uniformly distributed random samples \(\{x_i^r, t_i^r\}_{i=1}^{N_r} \subset \Omega\) and \(\{x_i^u, t_i^u\}_{i=1}^{N_u} \subset \Omega\) as \(2\)

\[
\mathcal{L}_u(w) = \frac{1}{N_u} \sum_{i=1}^{N_u} |\hat{u}(x_i^u, t_i^u) - u(x_i^u, t_i^u)|,
\]

\[
\mathcal{L}_r(w, \lambda) = \frac{1}{N_r} \sum_{i=1}^{N_r} |F(u(x_i^r, t_i^r, w), x_i^r, t_i^r, Du(x_i^r, t_i^r; w), D^2u(x_i^r, t_i^r; w), \cdots, D^k u(x_i^r, t_i^r; w); \lambda) - f(x_i^r, t_i^r)|^2.
\]

The weighting parameter \(\lambda\), which controls the trade-off between data loss, physics loss, and the initial loss (included in the physics component) along with the hyperparameters of the PINN model, is optimized by obtaining a minimum of the subsequent weighted loss function \([23]\)

\[
\mathcal{L} = \lambda_u \cdot \mathcal{L}_u + \lambda_r \cdot \mathcal{L}_r,
\]

through different optimization methods like stochastic gradient descent, Adam, adaptive gradient descent, RMS Prop, etc. \([23]\).

The weights \(\lambda_u\) (data weight) and \(\lambda_r\) (residual weight) are carefully selected to keep training balanced between the two losses. These weights can be fixed or changed during training, using a variety of methods, which brings us to self-adaptive PINNs.

B. Self-Adaptive Physics-Informed Neural Networks

When training PINN using the loss function detailed in Equation 4, the network is not only tasked with regression on a given dataset but also constrained by the losses imposed by a specific set of PDEs. Studies show that networks are extremely sensitive to the relative weights given to the various objective functions associated with the intended training problems \([20]\).

Manually adjusting these weighting coefficients, denoted as \(\lambda\), can be prohibitively costly. Thus, there is a strong preference for methods that adaptively tune these weights during training. The self-adaptive component of our proposed model addresses this issue by training the adaptation weights alongside the network weights \([2]\). This method ensures that points at initial, boundary, or residual locations in challenging areas of the solution space receive higher weighting in the loss function, compelling the network to enhance approximations at these critical points. Self-adaptive PINNs (SA-PINNs) weight each individual point in the loss criteria as \([2]\)

\[
\mathcal{L}_u(w, \lambda_u) = \frac{1}{N_u} \sum_{i=1}^{N_u} g(\lambda_u^i) |\hat{u}(x_i^u, t_i^u) - u(x_i^u, t_i^u)|,
\]

\[
\mathcal{L}_r(w, \lambda_r) = \frac{1}{N_r} \sum_{i=1}^{N_r} g(\lambda_r^i) |F(u(x_i^r, t_i^r, w), x_i^r, t_i^r), \cdots, D^m u(x_i^r, t_i^r; w); \lambda) - f(x_i^r, t_i^r)|^2,
\]

where \(\lambda_u = (\lambda_u^1, \cdots, \lambda_u^{N_u})\) and \(\lambda_r = (\lambda_r^1, \cdots, \lambda_r^{N_r})\) are the vectors of the SA weights with a self-adaptation mask function \(g\). Accordingly, the modified SA-PINN loss \([2]\)

\[
\mathcal{L}(w, \lambda_u, \lambda_r) = \mathcal{L}_u(w, \lambda_u) + \mathcal{L}_r(w, \lambda_r),
\]

is minimized with respect to the network weights \(w\) by gradient descent, as usual, but is maximized with respect to the self-adaptive weights \(\lambda_u\) and \(\lambda_r\) by gradient ascent. A fundamental concept in the proposed model is to allow weights to increase in response to corresponding losses. This is accomplished by training the network to concurrently minimize the losses and maximize the weights, effectively seeking a saddle point on the cost surface. Therefore, training the proposed model by minimizing Equation (7) instead of Equation (4) is anticipated to balance the data and residual loss, referred to as AdpBal

\[
\mathcal{L} = \exp (-\lambda_u^r) \cdot \mathcal{L}_u + \exp (-\lambda_r^r) \cdot \mathcal{L}_r + \lambda_u^r + \lambda_r^r.
\]

where \(\lambda_u^r = -\log(\lambda_u^r)\) and \(\lambda_r^r = -\log(\lambda_r^r)\) are trained for numerical stability, which is derived from the log-likelihood function \([20, 24]\). As training progresses, the \(\lambda_u\) and \(\lambda_r\) change, adjusting the impact of each loss component on the total training loss. In early stages, the model focuses more on fitting the data (lower \(\lambda_u\)) to quickly learn basic patterns before refining its parameters to adhere more strictly to physical laws. Later in training, after achieving a reasonable fit to the data, the model prioritizes the physics-informed loss (higher \(\lambda_r\)) in order to refine outputs for physical plausibility, particularly in regions with sparse data. This dynamic adaptation is crucial for training PDE-constrained models, especially when balancing data fitness and physical accuracy is challenging due to noisy or complex dynamics.

The weighting parameters are trainable, meaning they are modified during the training cycles using backpropagation, based on how well the model is performing on both the data fitting and the adherence to physical laws. This adaptiveness allows the model to dynamically prioritize between fitting the empirical data and satisfying the physical constraints, depending on which aspect is deemed more critical at different stages of the training.

C. Attention-based Gated Recurrent Unit (AGRU)

The GRU has been proposed for learning sequential patterns in natural language \([3]\). It has the ability to learn the long-term dependencies between time series windows while being simpler to compute and implement compared to the LSTM. A gate
controller, \( z \), manages both the input and forget gates: when \( z \) equals 1, the forget gate closes and the input gate opens; when \( z \) equals 0, the forget gate opens and the input gate closes. At each step, the previous \((t-1)\) memory is preserved while clearing out the input for that time step. The operation of a GRU cell follows specific equations [25]

\[
\begin{align*}
    r_t &= \sigma(W_r x_t + U_r x_{t-1}), \\
    z_t &= \sigma(W_z x_t + U_z x_{t-1}), \\
    c_t &= \tanh(W_c (h_{t-1} \odot r) + U_c x_t), \\
    h_t &= (z_t \odot c_t) + ((1 - z_t) \odot h_{t-1}),
\end{align*}
\]

where \( r_t \) is the reset gate, \( z_t \) is the update gate, \( c_t \) is the candidate activation, and \( h_t \) is the hidden state at time step \( t \). \( W, U, \) and \( b \) are learnable parameters. A longer input sequence within the GRU network leads to reduced accuracy in predicting the output sequence as it evenly emphasizes all input variables, despite their differing correlations with the prediction variable. To address this issue, an attention mechanism was employed to prioritize more relevant input features.

Soft attention allows for the visualization of the learned attention weights during each prediction cycle, enhancing interpretability while maintaining predictive power in GRU networks [4]. The attention mechanism first computes a set of attention weights \( \alpha \), interpreted as probabilities. Therefore, the attention weights during each prediction cycle, enhancing interpretability while maintaining predictive power in GRU networks.

The resulting weighted sum is used as the output of the attention mechanism. This output can then be passed on to subsequent layers of the proposed network.

IV. APPLICATION OF THE PROPOSED SA-PI-AGRU SCHEME TO LITHIUM ION BATTERIES

A. Improved Verhulst’s Battery Degradation Model

The time at which a Li-ion battery’s usable capacity drops below a specified threshold is widely known as its failure time [27]. The State of Health (SoH) of a battery is closely associated with its current capacity and is generally expressed as the ratio of the current capacity to the nominal capacity [28]. This paper considers the Percent Capacity Loss \( u \), which indicates the relative difference between the usable capacity and nominal capacity described as follows [20]

\[
    u_k = 1 - \frac{Q_k}{Q_{Nom}} \times 100\%.
\]

where \( Q_{Nom} \) is the nominal capacity and \( Q_k \) is the usable capacity at the \( k \)th cycle. Therefore, given an \( S \)-dimensional health indicator \( x = [x_1, x_2, ..., x_S]^T \in \mathbb{R}^S \) that describes the health condition as the battery deteriorates, and both collected data for monitoring and selected features that serve as indicators of battery health, the improved Verhulst’s battery degradation model can be constructed [29], [30]. It is possible to measure

![Fig. 1: Processed data representing the SoH degradation (at a temperature = 24°C).](image1.png)

![Fig. 2: Proposed SA-PI-AGRU model architecture for SoH estimation of LIB.](image2.png)
cell-to-cell differences using distinct combinations of values within the feature space. The degradation rate given \( x \) and monitoring time \( t \) is therefore given by [20]

\[
\frac{\partial u(x, t)}{\partial t} = r[u(x, t) - C] \left[ 1 - \frac{u(x, t) - C}{K - C} \right],
\]

(16)

where \( r > 0, 0 < u(t) < K, 0 < C \leq u_0 \), \( r \) is the degradation constant, \( u_0 \) is the initial loss, which is set to 10% [31], \( K \) is a constant showing the upper bound of capacity loss, and \( C \) denotes the initial capacity loss [20]. Since, a battery is usually classified as failed given that \( u \) exceeds 20%–30%, then \( K \) has a range of 20% to 100% [20].

B. Optimal Hyperparameters Selected for the Benchmark and Proposed Predictors

GridSearchCV was conducted on the proposed and benchmarked LIB lifetime predictors to maximize their performance [33]. During this process, hyperparameters were individually selected in a sequence. A predefined search space \( \mathcal{P} \) was used to select an optimal value for each hyperparameter. This included options such as number of layers \( \mathcal{L} = \{1, 2, 3, 4\} \), number of neurons \( \mathcal{U} = \{8, 16, 32, 64, 128\} \), dropout rate \( \mathcal{D} = \{0, 0.2, 0.4, 0.5\} \), optimizer \( \mathcal{O} = \{SGD, Adam, Adamax, and Rmsprop\} \), activation function \( \mathcal{A} = \{\text{Relu}, \text{Sigmoid}, \text{Swish}, \text{Gelu}\} \), and the learning rate \( \alpha = \{0.1, 0.01, 0.001, 0.005\} \). It is worth noting that both the baseline PINN and SA-PI-AGRU models had the same hyperparameters to ensure a consistent comparison.

C. Model Training and Testing

The architecture of the proposed model is depicted in Fig. 2 along with the proposed methodology outlined in Fig. 3. The input data (B0005) consists of 8 features, namely the cycle number, measured voltage, measured current, measured temperature, ambient temperature, load current, load voltage, and capacity. The data was cleaned and the columns were renamed to a more user-friendly format. After that, the data was split into 70% training, 15% testing, and 15% validation datasets and they were normalized to finally devise the proposed SA-PI-AGRU architecture. Furthermore, the model keeps training by calculating the loss in a batch for a pre-defined number of epochs/iterations via back-propagation. Finally, the model training is completed and evaluated by the validation dataset to ensure that overfitting is avoided and the testing datasets (B0006, B0007, and B0018) were evaluated to generate the SoH predictions.

The model consists of multiple hidden layers with varying number of neurons, an output layer representing the Verhulst’s degradation model parameters, and an activation function. The input layer comprises 8 features as mentioned above, 3 hidden layers with 32 neurons each, and one output (SoH) as outlined in Table II. The learning rate and training epochs were established at 0.01 and 10,000 respectively. The dense hidden layers possess GELU (Gaussian Error Linear Unit) as the activation function. The output layer has a single neuron with a Softmax activation function. The residual network is

<table>
<thead>
<tr>
<th>Settings ( \vartheta )</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Layers ( \mathcal{L} )</td>
<td>3</td>
</tr>
<tr>
<td>Number of Neurons ( \mathcal{U} )</td>
<td>32</td>
</tr>
<tr>
<td>Dropout Rate ( \mathcal{D} )</td>
<td>0.2</td>
</tr>
<tr>
<td>Optimizer ( \mathcal{O} )</td>
<td>Adam</td>
</tr>
<tr>
<td>Activation Function ( \mathcal{A} )</td>
<td>GELU</td>
</tr>
<tr>
<td>Learning Rate ( \alpha )</td>
<td>0.01</td>
</tr>
<tr>
<td>Training Epochs</td>
<td>10,000</td>
</tr>
</tbody>
</table>
TABLE III: SoH Estimation Results for Different Machine Learning Models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>B0005</th>
<th>B0006</th>
<th>B0007</th>
<th>B0018</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMSE</td>
<td>MAE</td>
<td>$R^2$</td>
<td>RMSE</td>
<td>MAE</td>
</tr>
<tr>
<td>LR [32]</td>
<td>1.45</td>
<td>1.21</td>
<td>97.84</td>
<td>2.27</td>
<td>1.75</td>
</tr>
<tr>
<td>LSTM [32]</td>
<td>0.85</td>
<td>0.61</td>
<td>99.26</td>
<td>1.46</td>
<td>0.97</td>
</tr>
<tr>
<td>GRU [32]</td>
<td>1.08</td>
<td>0.77</td>
<td>98.79</td>
<td>2.04</td>
<td>1.41</td>
</tr>
<tr>
<td>TCN [32]</td>
<td>2.44</td>
<td>2.14</td>
<td>93.71</td>
<td>4.26</td>
<td>3.69</td>
</tr>
<tr>
<td>Trans. [32]</td>
<td>1.68</td>
<td>1.46</td>
<td>97.13</td>
<td>2.38</td>
<td>2.03</td>
</tr>
<tr>
<td>ResNet-LSTM [32]</td>
<td>0.69</td>
<td>0.49</td>
<td>99.51</td>
<td>1.49</td>
<td>1.09</td>
</tr>
<tr>
<td>PINN (baseline)</td>
<td>1.10</td>
<td>0.86</td>
<td>98.73</td>
<td>1.80</td>
<td>1.32</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.61</td>
<td>0.39</td>
<td>99.61</td>
<td>1.32</td>
<td>0.92</td>
</tr>
</tbody>
</table>

utilized merely in the training stage to determine the solution to the PDE involved, where 5,000 collocation test points were considered. This represents the error between the Verhulst’s model and the SA-PI-AGRU’s prediction. It ensures that the model learns the underlying degradation behavior of the battery. During the inference stage, only the AGRU is utilized for solving the Verhulst’s degradation model.

D. Evaluation Metrics used to Assess the Performance of the Benchmark and Proposed Predictors

We evaluate the performance of the proposed and benchmark models by calculating the mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination ($R^2$) as follows [34]

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|,$$  \hspace{1cm} (17)

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2},$$  \hspace{1cm} (18)

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{N} (\bar{y} - y_i)^2}, \hspace{1cm} \bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i,$$  \hspace{1cm} (19)

where $y_i$ is the true SoH value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean value, and $N$ is the total number of data points.

V. RESULTS

A. Experimental Data Preprocessing

The testing data for LIBs was obtained from the Prognostics Center of Excellence’s data repository at NASA’s Ames Research Center [35]. The experiment involved collecting LIB data at the Idaho National Laboratory using various commercial lithium-ion 18650 LIBs tested on a specialized prognostics testbed [5], [35]. The battery charging process followed the constant-current constant-voltage method, and the collected time series data comprised 34,866 measurements including cycle number, measured voltage, measured current, measured temperature, ambient temperature, load current, load voltage, capacity, and SoH values. The testbed included diagnostic tools such as a power supply, programmable DC electronic load, voltmeter, thermocouple sensor, environmental chamber, electrochemical impedance spectroscopy (EIS), and a PXI-based data acquisition system [36].

Operational tests conducted at room temperature of 24 °C involved charge/discharge cycles and impedance assessments through EIS over a frequency range spanning from 0.1 kHz to 5 kHz [36]. These cycles expended wear until reaching an end-of-life state when batteries exhibited a capacity reduction of around 30%. Experiments were halted upon reaching this point, which signifies rated capacity fading from 2 A-h to approximately 1.4 A-h for four different types of batteries with distinct SoH degradation behavior (B0005, B0006, B0007, and B0018) as illustrated in Fig. 1 and Table I. The battery aging conditions outlined in Table I, includes the charge/discharge cut-off voltage (C/DCV), minimal charge current (MCC), and the constant discharge current (CDC) [5].

B. Performance Evaluation and Comparison with Benchmark Models

In this study, the performance of several machine learning models are evaluated for the estimation of the SoH for LIBs across four different datasets (B0005, B0006, B0007, and B0018) as shown in Table III, where battery B0005 was used for training, and batteries B0006, B0007, and B0018 were used for testing. The models assessed included Linear Regression (LR), LSTM, GRU, Temporal Convolutional Networks (TCN), Transformers (Trans.), residual neural network with LSTM (ResNet-LSTM), baseline PINN, and the newly proposed SA-PI-AGRU model as shown in Fig. 4. The efficacy of these models was gauged based on three key metrics: RMSE, MAE, and $R^2$.

The LR model exhibited consistent performance with ($R^2$) values above 94% for all datasets, although it tended to have higher RMSE and MAE values, suggesting lower precision relative to more sophisticated models. The LSTM model, noted for its ability to capture temporal dependencies, showed superior performance, particularly on datasets B0005 and B0006, with ($R^2$) values approaching 99%. This suggests a strong capacity for LSTM models to handle sequences and time-dependent data which is critical in battery SoH prediction. Moreover, GRU models also performed robustly, with slightly lower accuracy compared to LSTM. This could be attributed to the similar architecture of GRU to LSTM, albeit with fewer parameters, which might affect capturing complex patterns in the data. TCN and Transformer models, however, showed a decrease in performance, particularly on datasets B0006 and B0018, where the more complex patterns possibly present in the data could not be effectively modeled by these architectures.
The proposed SA-PI-AGRU model demonstrated the most promising results, outperforming all other models on three of the four datasets, particularly on dataset B0005 with an \((R^2)\) value of 99.61%, which is expected as B0005 was used for training the proposed model (5-fold cross-validation was applied). It also achieved the lowest RMSE and MAE scores on most datasets, indicating its superior predictive accuracy and robustness. The architecture and learning algorithm enhancements of the SA-PI-AGRU evidently provide significant improvements over traditional and more recent approaches, highlighting its potential for practical applications in BPHM.

The weight magnitude plot observed in Fig. 6, provides further insights into the model’s adaptive behavior. The adaptive nature of the weight magnitudes can be attributed to the self-adaptive mechanism employed in the model. By allowing the weights to adjust dynamically with respect to the learning epochs, the model can effectively balance the contributions of the data-driven and physics-based components. In the early stages of training, the model focuses more on fitting the training data, as substantiated by the higher magnitude of the data loss weight. As the model learns to capture the underlying patterns in the data, the importance of the data loss component decreases, allowing the physics-based constraints to play a more significant role in guiding the model’s predictions. It is worth noting that the proposed model has a memory size of 101.78 KB (lightweight), making it suitable to be downloaded for practical applications.

### Table IV: LIB lifetime prediction results for the B0005 dataset.

<table>
<thead>
<tr>
<th>PINN Method</th>
<th>MAE</th>
<th>RMSE</th>
<th>(R^2)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline PINN</td>
<td>0.855 ± 0.173</td>
<td>1.095 ± 0.192</td>
<td>98.73 ± 0.446</td>
<td>5m 28.76s</td>
</tr>
<tr>
<td>SA-PI-AGRU (Proposed)</td>
<td>0.395 ± 0.0811</td>
<td>0.614 ± 0.0445</td>
<td>99.61 ± 0.0697</td>
<td>12m 19.77s</td>
</tr>
</tbody>
</table>
The results obtained from the bar graphs shown in Fig. 7a and Fig. 7b indicate a clear differentiation in the performance of the model based on the number of neurons and layers used. The MAE and RMSE values demonstrate how the model’s accuracy and error rates vary across different configurations. For a single layer, the MAE values for 16, 32, and 64 neurons are notably high, approximately 8.41 to 8.5, suggesting a higher prediction error. In contrast, configurations with two layers show a significant reduction in error for 32 and 64 neurons, with MAE values dropping to 0.53. The three-layer configuration shows the lowest MAE values for 32 neurons, around 0.4, indicating improved model performance with this setup. The four-layer configuration results in varied MAE values, with 16 neurons achieving the lowest error at 0.93.

Finally, Table IV compares the performance of the baseline PINN with the proposed SA-PI-AGRU model. The baseline PINN achieved an MAE of $0.855 \pm 0.173$, RMSE of $1.095 \pm 0.192$, and $R^2$ of $98.73 \pm 0.446$, with an average training time of 5 minutes and 28.76 seconds. In comparison, the SA-PI-AGRU method significantly outperformed the Baseline PINN, with a lower MAE of $0.395 \pm 0.0811$, lower RMSE of $0.614 \pm 0.0445$, and higher $R^2$ of $99.61 \pm 0.0697$. However, this improved performance comes with a longer average training time of 12 minutes and 19.77 seconds. The smaller standard deviations in the SA-PI-AGRU method’s error metrics indicate more consistent and reliable performance across multiple runs, highlighting the proposed model’s generalization capabilities despite the network initialization at the different runs conducted. These features help in stabilizing the training process by ensuring more effective gradient flow and breaking symmetry in a more controlled manner, which reduces variability in the performance metrics and leads to a more robust model. Effective initialization breaks symmetry in the network weights, allowing different neurons in the GRU cells to learn the present diverse battery inputs.

VI. CONCLUSIONS

A Self-Adaptive Physics-Informed Attention-based Gated Recurrent Unit (SA-PI-AGRU) model was proposed for handling sequential data. Accordingly, the proposed model was tested for prognostics and health management of lithium-ion batteries, surpassing traditional machine learning and deep learning models in accuracy and viability. This study verified...
the model’s effectiveness across four battery datasets, where it consistently achieved superior performance metrics such as lower RMSE and MAE, alongside higher \( R^2 \) values, confirming its robust predictive capabilities. Particularly notable, were its results on the training and testing datasets, where an \( R^2 \) value of approximately 99\% was achieved. The integration of attention mechanisms and an adaptive weighting method for the loss function within the SA-PI-AGRU model, effectively harnesses sequential data, enhancing the model’s ability to adapt to varying conditions and improving the overall accuracy of SoH estimation. The testing and validation procedure was conducted on different NASA Battery datasets and have ensured that the proposed model holds practical applicability across diverse real-world scenarios. Hence, the proposed model holds strong potential to solve the time-dependency issue present in the baseline PINN without requiring additional data and decreasing the time step.

**REFERENCES**


