Neural ordinary differential equations and recurrent neural networks for predicting the state of health of batteries

Simona Pepe[a], Jiapeng Liu[a], Emanuele Quattrocchi[a], Francesco Ciucci[ab]*

[a] Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

[b] Department of Chemical and Biological Engineering, The Hong Kong University of Science and Technology, Hong Kong, China

* Corresponding author: francesco.ciucci@ust.hk

Phone: +852 2358 7187

Fax: +852 2358 1543

Abstract

Battery management systems require efficient battery prognostics so that failures can be prevented, and efficient operation guaranteed. In this work, we develop new models based on neural networks and ordinary differential equations (ODE) to forecast the state of health (SOH) of batteries and predict their end of life (EOL). Governing differential equations are discovered using measured capacities and voltage curves. In this context, discoveries and predictions made with neural ODEs, augmented neural ODEs, predictor-corrector recurrent ODEs are compared against established recurrent neural network models, including long short-term memory and gated recurrent units. The ODE models show good performance, achieving errors of 1% in SOH and 5% in EOL estimation when predicting 30% of the remaining battery's cycle life. Variable cycling conditions and a range of prediction horizons are analyzed to evaluate the models' characteristics. The results obtained are extremely promising for applications in SOH and EOL predictions.

Keywords

Lithium-ion batteries, neural ordinary differential equation, deep learning, state of health, end of life.

1 Introduction

To enable large-scale decarbonization of our planet, energy storage technologies such as lithium-ion batteries (LIBs) will need to play an increasingly central role. The growing success of portable electronics and electric vehicles has been propelled by the realization of LIBs, which are characterized by high energy density, efficiency, and long lifespans [1, 2]. It is projected that LIBs will also be used to buffer the intermittent electricity produced by renewable energy sources through large-scale battery power stations such as Hornsdale Power Reserve in Southern Australia and Gateway Energy Storage in California, USA [3]. In this regard, new policy scenarios predict strong commercial penetration of batteries with a utility-scale deployment of 220 GW by 2040 from 4 GW in 2020 [4].

The optimization of battery performance and lifespan is critical. The state of health (SOH) is a key piece of information that can be used to predict the battery's remaining useful life (RUL) and, therefore, help end-users avoid system failure and manage required maintenance. For instance, in electric vehicles, accurate prognostics of a battery SOH can prevent failure, thereby avoiding service interruptions. Battery degradation can be assessed using several methods. Direct techniques, such as via scanning electron microscopy, transmission electron microscopy, or Raman spectroscopy can be used to observe the microstructure and chemical state of batteries directly [5]. However, these direct methods are destructive and applicable to small-scale prototypes. Electrical methods, including incremental capacity, differential voltage, and equivalent circuit analysis, have also been used [6-8]. However, those approaches necessitate *ad hoc* testing conditions and data processing to achieve high reliability. To overcome these issues, Kalman and particle filter techniques [9-14] have been developed to tackle real-time estimation and handle uncertainty in the data. Thanks to the exponential growth of computational power, the large amounts of data available, and the ease of programming with scripted languages, data-driven analytics is becoming more widespread. Related data-driven models have been shown to score the highest accuracies among all the other techniques in SOH applications [15-17]. Among the various methods used, support [18-22] and relevance [23] vector machines, Box-Cox kernel techniques [24], Bayesian and Gaussian processes [25-28], random forest trees [29, 30], and deep neural networks [31, 32] have been applied to estimate the battery's SOH and state of charge. However, most of these techniques have not been developed specifically for time-dependent problems. Methods particularly valuable for time series [16, 33-37] include recurrent neural network (RNN), long-short-term memory (LSTM), and gated recurrent neural networks (GRU) [38-43]. In the field of lithium-ion battery lifetime prediction, we can find applications of LSTM algorithms [12, 44, 45]. Works from Chemali *et al*. and Zhang *et al.* demonstrated exceptional abilities of LSTM in state of charge and RUL prediction in batteries [40, 46]. Although previous models showed good performances in short-term predictions, improving the prediction accuracy while limiting computational resources needed for onboard prognostic is still challenging [47-49]. Furthermore, much of the literature analyzes *ad hoc* datasets, making the comparisons among prior works difficult and the results less general [50].

The objective of this work is to forecast accurately the battery state of health (SOH) and end of life (EOL). To do that, we interpreted the battery SOH evolution to be a dynamical system. In the models developed herein, the battery state $\boldsymbol{y}$, discretizing capacities and charge voltage profiles, evolves following an ordinary differential equation, *i.e.*,

$$\dot{\boldsymbol{y}} = \boldsymbol{F}(\boldsymbol{y}, t) \tag{1}$$

where $\boldsymbol{F}(\boldsymbol{y}, t)$ is a vector-valued function, which we shall assume to be independent of time, *i.e.*, $F(\boldsymbol{y}, t) = F(\boldsymbol{y})$. After learning $F(\boldsymbol{y})$, $\mathrm{SOH}(\boldsymbol{y})$ is predicted. First, the neural ordinary differential equation (neural-ODE) approach [51], which parametrizes the function $F(\boldsymbol{y})$ through an infinitely deep neural network, is analyzed. Second, the data is studied using augmented neural ODE (ANODE) [52] framework, as this method has been shown to be more robust and lightweight and better capable of lowering training losses than the neural-ODE [52]. The (1) was then discretized as an RNN [53], called predictor-corrector RNN (PC-RNN), in which an explicit correction follows a forward Euler step.

The three ODE-based methods are then compared to the established LSTM and GRU RNNs [54, 55] on two different battery datasets [56, 57]. The computational results show that the ODE-based models outperform LSTM and GRU. In particular, ANODE and PC-RNN can accurately forecast the RUL based on fewer data. Finally, a multi-battery approach is used to leverage full-cycle data and multiple cells. The results show significant improvements in performance for early-stage predictions.

2 Dataset

Datasets from Oxford University [57, 58] and NASA [56] were used as they are heterogeneous and characterized by established discharge patterns and constant-current charge. B1-B8 and A1-A4 correspond to the Oxford and NASA datasets, respectively, consistent with [25]. It should be noted that the Oxford and NASA cells have different degradation behavior, as shown from the capacity *versus* time plots, see Figures 1(a) and (c). In particular, irregular patterns are observed on the capacity *versus* cycle curves in the NASA data (Figure 1(c)) [25]. Moreover, even within the same dataset, different cells show a different EOL, suggesting distinct degradation patterns, thereby making predictions challenging.

The features of the dataset included the capacity, Q_k, at cycle k and the charge voltage profiles. From Q_k we derived the state-of-health at cycle k, SOH_k, which is defined as [1, 59]

$$\mathrm{SOH}_k = \frac{Q_k}{Q_0} \tag{2}$$

where Q_k is capacity at cycle k ($k = 0$ corresponds to the fresh battery). As various authors have shown that charge voltage profiles correlate strongly to aging, they were also included in the state vector $\boldsymbol{y}_k$ [58, 60]. To convert the voltage charge curves into usable features, we selected N_V equispaced voltage points, from 3V for Oxford and 3.6 V for NASA datasets to 4.2V for both, and used as features the corresponding normalized charge times (Figure 2(a)). In turn, this leads to the time/voltage features, $\boldsymbol{V}_k$. This procedure generates a feature vector $\boldsymbol{y}_k = \begin{pmatrix} \mathrm{SOH}_k \\ \boldsymbol{V}_k \end{pmatrix}$ of dimension $N_\mathrm{V} + 1$ ($N_V = 21$ for the Oxford and $N_\mathrm{V} = 19$ for the NASA datasets). Therefore, the state evolution $\{\mathbf{y}_k |\, k = 1, \dots, \mathrm{EOL}\}$ can be interpreted as a multivariate time series or a dynamical system.

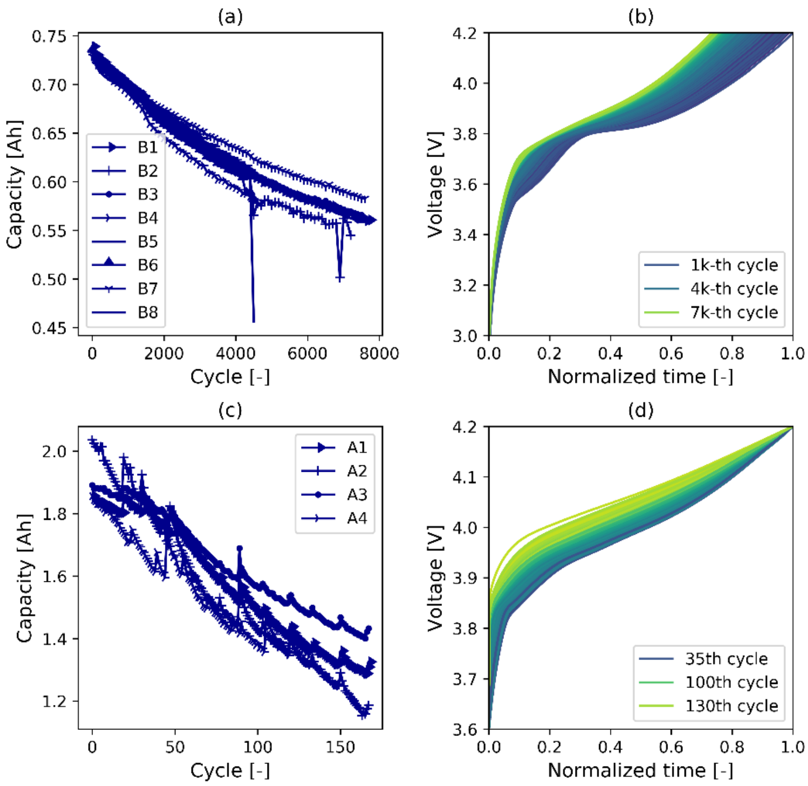


Figure 1 (a) Capacity curve of batteries from the Oxford dataset (b) Voltage curves at different time life of the B1 battery (c) Capacity curve from the NASA dataset (d) Voltage curves at different time life of the A1 battery.

3 Methods

3.1 Overview

Training was done on the dataset $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \dots \boldsymbol{y}_{N_{\mathrm{TP}}}\}$ with $N_{\mathrm{TP}} < \mathrm{EOL}$. To study different prediction horizons, we trained different portions of the complete degradation data (i.e. $N_{\mathrm{TP}}/\mathrm{EOL} = 50, 60, \dots, 90\%$), three additional cases (i.e. 20, 30, 40 %) were included in the early prediction analysis (see Multi-battery approach section). The forecasting goal was to predict EOL and $[\mathrm{SOH}_{N_{\mathrm{TP}}+1}, \mathrm{SOH}_{N_{\mathrm{TP}}+2}, \dots, \mathrm{SOH}_{\mathrm{EOL}}]$. First, we considered a single-battery approach, see Figure 2(b), where, for each battery, we trained the first N_{TP} points and used later datapoints for validation or testing. Validation was performed on a selection of batteries to tune the models' hyperparameters (*i.e.* number of neurons, layers, and number of iterations). Testing was performed on the remaining batteries.

Two training losses were used. A first loss, $\mathcal{L}_F$, defined as

$$\mathcal{L}_F = \sum_{k=1}^{N_{\mathrm{TP}}} |\hat{\boldsymbol{y}}_k - \boldsymbol{y}_k|^2 \tag{3}$$

where we attributed equal weight to all features, the hat indicates model values. A second loss $\mathcal{L}_{F-\mathrm{norm}}$, attributing equal weight to SOH_k and the combination of time/voltage features, was defined as

$$\mathcal{L}_{F-\mathrm{norm}} = \sum_{k=1}^{N_{\mathrm{TP}}} [\left(\widehat{\mathrm{SOH}}_k - \mathrm{SOH}_k\right)^2 + \frac{1}{N_{\mathrm{V}}}\left|\hat{\boldsymbol{V}}_k - \boldsymbol{V}_k\right|^2] \tag{4}$$

where $\widehat{(\cdot)}_k$ and $(\cdot)_k$ indicate model prediction and experimental values.

A similar workflow to that reported in Figure 2(b) was used to study the multi-battery learning where multiple cells were included in the training, and longer prediction windows were examined. Only ANODE, GRU, and $\mathcal{L}_F$ were considered.

Model performance was benchmarked on the SOH RMSE defined as

$$\mathrm{RMSE}_{\mathrm{SOH}} = \sqrt{\sum_{k=N_{\mathrm{TP}}+1}^{\mathrm{EOL}} \frac{\left(\widehat{\mathrm{SOH}}_k - \mathrm{SOH}_k\right)^2}{N_{\mathrm{test}}}} \tag{5}$$

where $N_{\mathrm{test}} = \mathrm{EOL} - \mathrm{N_{TP}} - 1$ is the number of testing points. In addition, the error on EOL prediction given by [48]:

$$\mathrm{DEOL} = \frac{\widehat{\mathrm{EOL}} - \mathrm{EOL}}{\mathrm{EOL}} \tag{6}$$

was also tracked. A positive/negative DEOL denotes optimistic/pessimistic predictions.

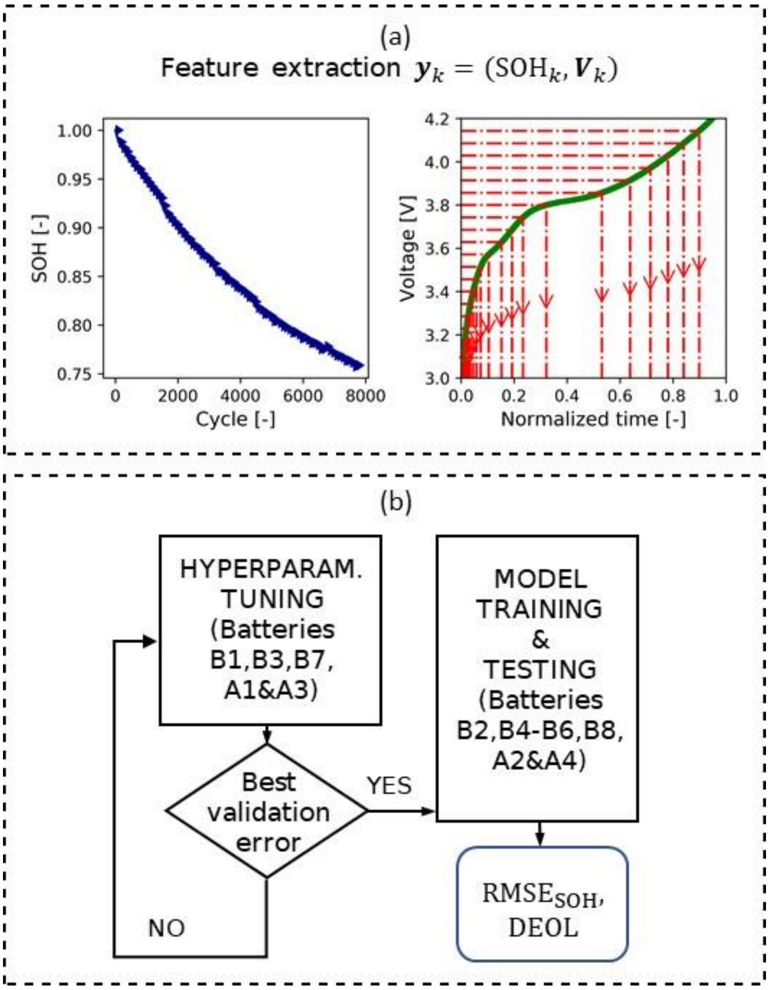


Figure 2 (a) Example of extracted features for the Oxford B1 battery: SOH *versus* cycle number (left panel) and charging times *versus* voltage points at $k = 10000 - th$ cycle (right panel). (b) Workflow for the SOH and EOL estimation from batteries data. First, the features input of our models are extracted, second, the model's hyperparameters are tuned on the validation set. Last, considering the optimal hyperparameters, the models are trained and the prediction is evaluated on the test set.

3.2 Models

3.2.1 Neural-ODE

Neural-ODEs are a family of deep neural networks introduced by Chen *et al.* [51, 61], which not only extend continuously residual networks and recurrent neural networks, but are also closely linked to normalizing flows [62]. Within an infinitesimal time step, the neural-ODE dynamics can be reduced to a continuous ODE problem as in (1), where an infinitely deep neural network parametrizes the sequence of system states. Therefore, the measured systems state, $\boldsymbol{y}_k$, can be computed using an ODE solver starting from an initial value $\boldsymbol{y}_{k=0} = \boldsymbol{y}_0$ by learning the function $F(\boldsymbol{y}, \boldsymbol{\theta}(t))$, where $\boldsymbol{\theta}(t)$ are the (time-dependent) neural network parameters.

3.2.2 Augmented neural-ODE (ANODE)

ANODEs have been introduced recently to resolve some of the limitations of neural-ODEs. In fact, as demonstrated in [52, 63], there are functions that neural-ODEs cannot represent (*e.g.* discrete jumps and flows with crossing trajectories). ANODE overcome this problem by introducing an augmented variable, $\boldsymbol{a}$, such that $d/dt \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{a} \end{bmatrix} = F(\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{a} \end{bmatrix}, \boldsymbol{\theta}(t))$ and $\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{a} \end{bmatrix}(0) = \begin{bmatrix} \boldsymbol{y}_0 \\ 0 \end{bmatrix}$. As shown elsewhere [52, 63, 64], augmenting the solution space allows more complex functions to be represented and may lower computational cost.

3.2.3 Predictor-corrector RNN (PC-RNN)

RNNs are attractive given their ability to process sequential data using recursive structures [53, 65]. Here we use a specific version of RNNs, which we use and call PC-RNN. $F(\boldsymbol{y}_k, \boldsymbol{\theta})$ is parametrized as a neural network where $\boldsymbol{\theta}$ are the network parameters and Euler forward differencing is used to obtain a "prediction" $\widehat{\boldsymbol{y}}_{k+1}$ from $\boldsymbol{y}_k$, namely,

$$\widehat{\boldsymbol{y}}_{k+1} = \boldsymbol{y}_k + \Delta t\, F(\boldsymbol{y}_k, \boldsymbol{\theta}) \tag{7}$$

Then, the "predicted" $\widehat{\boldsymbol{y}}_{k+1}$ are used to compute $F(\widehat{\boldsymbol{y}}_{k+1}, \boldsymbol{\theta})$. In turn the "corrected" $\boldsymbol{y}_{k+1}$ is obtained using

$$\boldsymbol{y}_{k+1} = \boldsymbol{y}_k + \frac{1}{2}\, \Delta t (F(\boldsymbol{y}_k, \boldsymbol{\theta}) + F(\widehat{\boldsymbol{y}}_{k+1}, \boldsymbol{\theta})) \tag{8}$$

3.2.4 LSTM and GRU

LSTM was introduced by Hochreiter and Schmidhuber [55, 66] to solve RNNs' vanishing gradient issue in long-term dependencies. Like RNNs, in LSTM the past states and the new information are recursively combined to return outputs. However, in RNN the distant information cannot persist as time elapses, due to feedback error decay. The more complex architecture of LSTM provides a solution to this problem by adding to the conventional hidden state a cell state, $\boldsymbol{C}_k$. The former state is responsible for maintaining short-term memory, whereas the latter state preserves long-term memory. Four gates process the information accordingly. Each LSTM unit is followed by a fully connected layer to give the output at the next time step $k + 1$.

GRU was introduced by Cho *et al*. in 2014 [54, 56]. It is considered a variation of LSTM and has similar architecture. The main differences with LSTM are that (i) in GRU two gates (*vs*. the four of LSTM) are used; and (ii) the cell state and hidden state are merged. Few examples in the literature demonstrate results comparable to and sometimes better than LSTM [42, 46, 67].

3.3 Implementation

In neural-ODE, two linear layers with tanh activation were used to model $F(\boldsymbol{y}, \boldsymbol{\theta}(t))$. Dopri5, the default solver, and the adjoint method [68] were used to compute trajectories and gradients, respectively. The batch size was set to 1. The reader can refer to [51, 69] for more details. In ANODE, the network setup was as described in the literature [52]. As for LSTM and GRU, dropout (p=0.2) was applied to reduce overfitting. A schematic of the model architectures is provided in Figure 3. All codes were implemented using Pytorch.

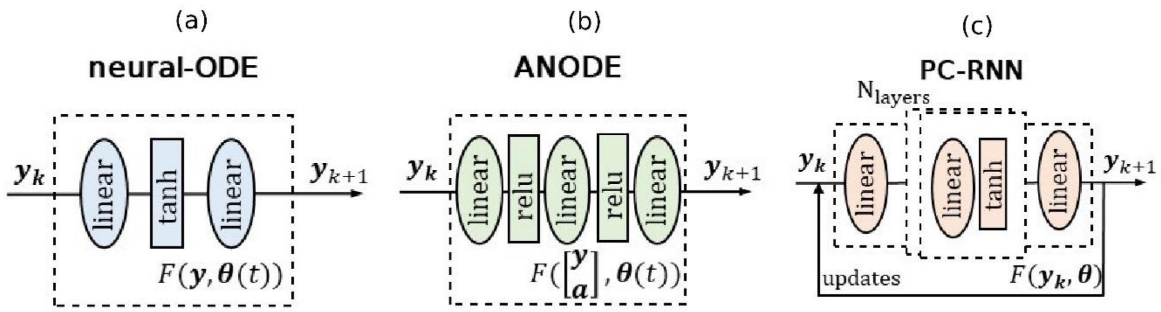


Figure 3 Simplified schematics of neural-ODE, ANODE, and PC-RNN architectures.

3.4 Hyperparameter optimization

For neural-ODEs, the width of the linear layer was selected among 50, 100, 200, and 500. In ANODE, the optimal neurons were chosen among 50, 100, 500, and 1000 and the augmented space dimensions ($n_{\boldsymbol{a}}$) among 1, 5, and N_v+1. The depth and width of the PC-RNN underlying neural network were chosen among 1, 2, 4, 6, and 20, 40, 50, 100, 125, 256, respectively. In LSTM, 250, 500, 1000 neurons in a 1-layer configuration and 50, 100, 250 in a 2-, 4-, and 6-layers network were explored as in [41, 46]. In GRU, 1, 2, 3, 4, and 6 layers of width 50, 100, 150, 200, and 300 were studied according to [67]. The number of iterations was also considered as a network hyperparameter and tuned against the validation data. Batteries B1, B3, B7, and

A1, A3, were selected for the hyperparameter tuning; for each one, training and validation were performed on the first 70% and last 30% of the data, respectively. The optimal (lowest validation error) hyperparameters were obtained by weighing each battery equally.

		neural-ODE	ANODE	PC-RNN	LSTM	GRU
				$\mathcal{L}_F$		
Oxford	neurons	100	1000	20	250	300
	layers	-	22 (n_a)	1	1	1
NASA	neurons	50	1000	50	500	200
	layers	-	20 (n_a)	1	3	3
				$\mathcal{L}_{F-\mathrm{norm}}$		
Oxford	neurons		1000	100	250	300
	layers	-	1 (n_a)	1	1	6
NASA	neurons		1000	40	50	300
	layers	-	1 (n_a)	4	2	4

Table 1 Optimal hyperparameters in the Oxford and NASA experiments with training losses defined as $\mathcal{L}_\mathrm{F}$ and $\mathcal{L}_{\mathrm{F-norm}}$ as in Methods, Overview.

4 Results

4.1 SOH prediction

4.1.1 Oxford dataset

From the hyperparameter optimization (Table 1), we obtain 1 layer for GRU, LSTM, and PC-RNN and 300, 250, 20, 100, and 1000 neurons in GRU, LSTM, PC-RNN, neural-ODE, and ANODE, respectively. In ANODE, the optimal augmented space dimension n_a is 22. The models were trained and tested on batteries B2, B4-6, and B8. Different prediction horizons dependent on the proportion of data used for training (*i.e.* $N_{\mathrm{TP}}/\mathrm{EOL} = 50, 60, \dots, 90\%$) were analyzed. The prediction on SOH for the battery B2 at $N_{\mathrm{TP}}/\mathrm{EOL} = 70\%$ is shown in Figure 4. We can observe that all models can successfully regress the training data (solid red lines) and predict the downward degradation trend (red dashed lines). The methods' accuracy in predicting SOH is benchmarked using the $\mathrm{RMSE}_{\mathrm{SOH}}$, whose mean values and standard deviations are reported in Table 2. Overall, ANODE achieves the best performance, except at $N_{\mathrm{TP}}/\mathrm{EOL} = 70\%$, with an average $\mathrm{RMSEs}_{\mathrm{SOH}}$ lower than 3% for all prediction windows [48]. PC-RNN performs slightly worse than ANODE, and GRU shows the worst results. As expected, the model's accuracy was higher for the shorter prediction horizon, with the only exception of neural-ODE, which produces the lowest $\mathrm{RMSE}_{\mathrm{SOH}}$ (0.93%) at 70%, see Figure 5(b).

		% training data	neural-ODE	ANODE	PC-RNN	LSTM	GRU
				$\mathcal{L}_F$			
Oxford	$\mathrm{RMSE_{SOH}}$ [%]	50	3.11±2.33	**2.59±2.84**	2.83±1.68	4.07±0.64	3.92±1.04
		60	1.95±1.84	**1.40±1.26**	1.41±0.99	2.82±1.59	3.45±0.95
		70	**0.93±0.70**	1.61±1.44	1.30±0.82	1.64±0.54	2.21±0.47
		80	2.05±1.17	**1.05±1.07**	1.32±0.46	1.38±0.68	1.53±0.67
		90	3.72±2.31	**1.10±1.04**	1.20±1.12	1.47±1.06	1.41±1.09
NASA	$\mathrm{RMSE_{SOH}}$ [%]	50	5.70±3.52	5.66±5.07	4.93±3.67	**4.79±2.00**	6.34±4.07
		60	**2.51±1.34**	11.68±11.41	6.45±1.03	7.89±1.02	6.90±2.27
		70	**3.27±0.10**	4.70±0.33	6.31±1.19	6.33±1.45	3.56±2.52
		80	4.72±0.78	5.27±0.78	3.68±0.89	6.23±0.43	**3.55±1.96**
		90	1.83±0.02	**1.73±0.58**	1.96±0.21	4.17±0.09	2.86±1.50
				$\mathcal{L}_{F-\mathrm{norm}}$			
Oxford	$\mathrm{RMSE_{SOH}}$ [%]	50	2.37±0.71	2.02±1.60	**1.74±0.80**	4.73±0.65	3.99±0.57
		60	1.72±1.54	**1.64±1.03**	1.71±0.09	3.05±1.58	3.51±1.33
		70	1.13±0.71	1.77±1.52	**1.05±0.53**	2.01±0.78	1.66±0.71
		80	1.29±0.88	1.29±0.52	**0.84±0.49**	1.43±0.74	1.73±0.74
		90	1.46±0.44	1.17±1.21	**1.08±1.12**	1.40±1.13	1.79±1.25
NASA	$\mathrm{RMSE_{SOH}}$ [%]	50	4.49±0.59	10.88±11.85	9.54±9.96	**4.06±0.58**	7.57±1.57
		60	**1.76±0.75**	10.19±10.12	5.05±0.55	8.35±0.72	6.44±3.82
		70	3.97±1.07	5.36±4.17	**3.81±1.75**	5.88±0.24	4.56±2.36
		80	**2.25±0.97**	3.39±2.24	3.65±3.51	4.31±0.26	3.12±0.75
		90	**1.85±0.47**	4.59±3.81	1.87±1.49	4.97±0.40	2.39±1.70

Table 2 $\mathrm{RMSE_{SOH}}$ mean values and standard deviations on batteries B2, B4-B6, and B8 for Oxford and A2, A4 for NASA, obtained training each algorithm with the losses $\mathcal{L}_F$ and $\mathcal{L}_{F-\mathrm{norm}}$ (see Section Methods, Overview for details), respectively, at different training portions of data. The best results are in bold font.

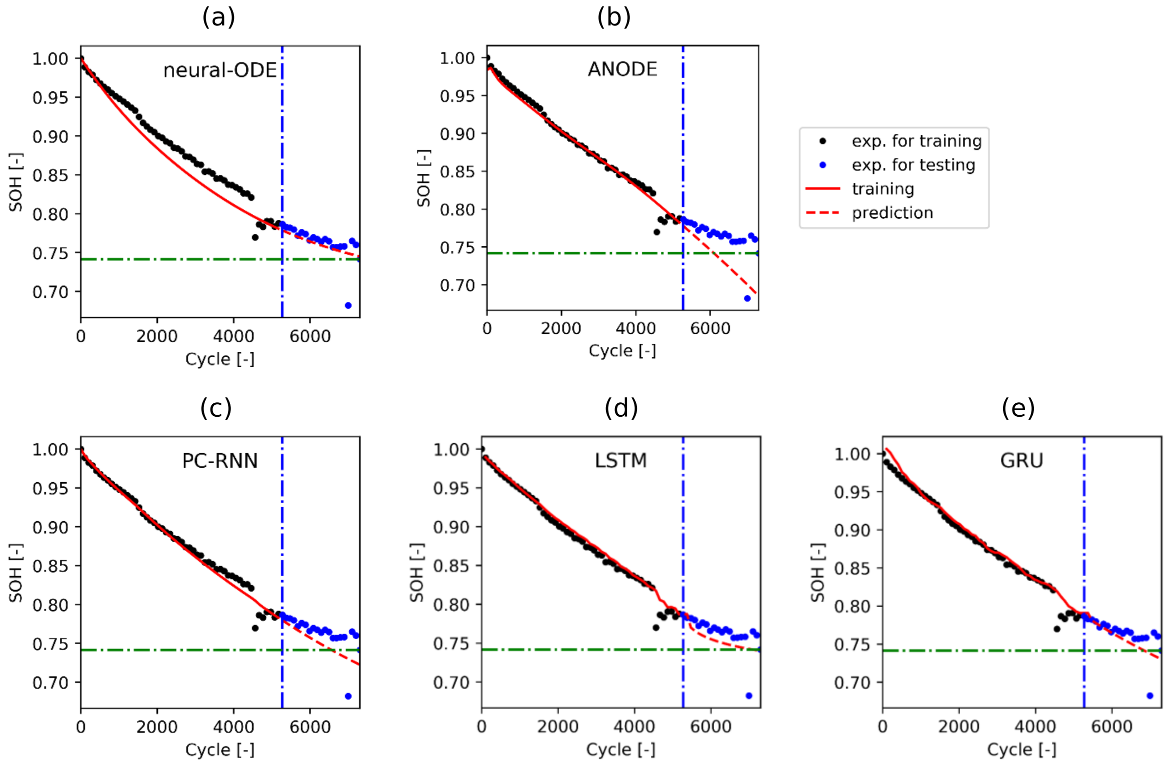


Figure 4 Examples of experimental and predicted SOH for the B2 Oxford battery with 70% as a function of the portion of data used for training.

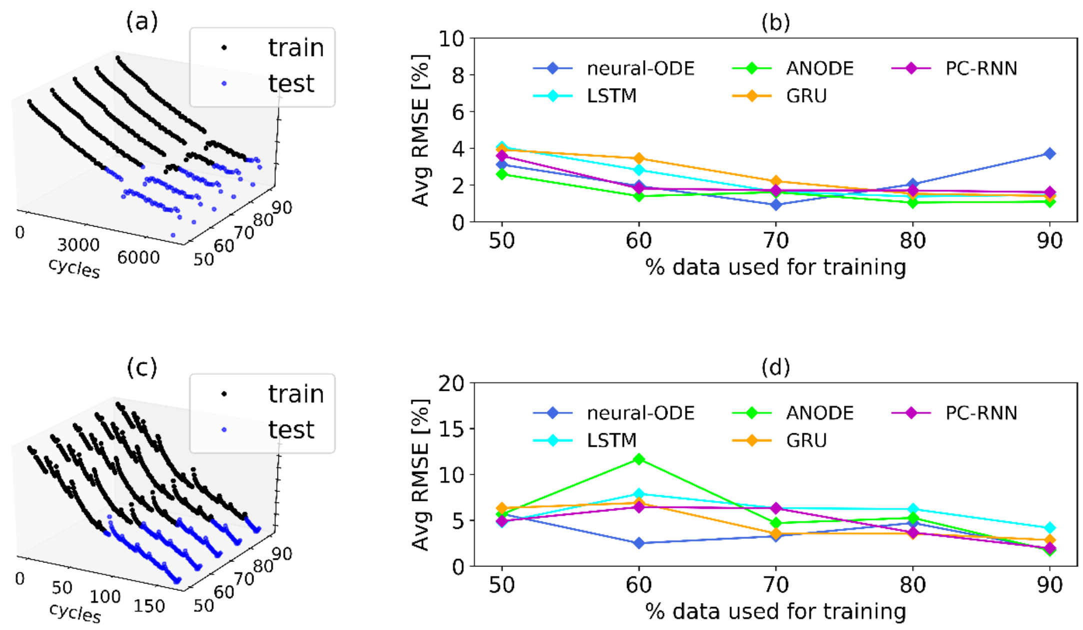


Figure 5 (a) Examples of training and testing points on the capacity curve of the B2 battery. (b) Evolution of the test error on the SOH prediction, averaged on the batteries B2, B4-6, and B8, with different portions of data used for training (50%, 60%, 70%, 80%, 90%). (c) Examples of training and testing points on the capacity curve on the A2 battery. (d) Evolution of the test error on the SOH prediction, averaged on the batteries A2 and A4, with different potions of data used for training (50%, 60%, 70%, 80%, 90%).

4.1.2 NASA dataset

The optimal hyperparameters (Table 1) are identified to be 3 layers in GRU, 1 layer in LSTM and PC-RNN, and 200, 500, 50, 50, 1000 neurons in GRU, LSTM, PC-RNN, neural-ODE, and ANODE, respectively. In ANODE $n_a = 20$. The prediction on SOH for the battery A2 at $N_{\mathrm{TP}}/\mathrm{EOL} = 70\%$ is shown in Figure 6. The $\mathrm{RMSE}_{\mathrm{SOH}}$'s of batteries A2 and A4 are reported in Table 2. The models' prediction accuracy drops compared to the Oxford results. For instance, at $N_{\mathrm{TP}}/\mathrm{EOL}$ =70%, the average $\mathrm{RMSE}_{\mathrm{SOH}}$ increases to 6.33% (LSTM), 7.48% (GRU), 4.70% (ANODE), 3.27% (PC-RNN) and 6.31% (neural-ODE). The larger errors can be attributed to the irregular patterns characteristic of the NASA degradation dataset (see paragraph Dataset). At $N_{\mathrm{TP}}/\mathrm{EOL}$ =60%, ANODE fails to reproduce the downward trend typical of battery degradation, leading to the highest average RMSEs (larger than 10%), see Figure 5(d). Despite fluctuations and the inability to capture short-term jumps, neural-ODE and PC-RNN yield the best results, see Figure 5(d). Conversely, while LSTM and GRU can fit short-term patterns, they have poorer prediction ability. Finally, we believe that the single-battery approach is not only meaningful in one-shot predictions (when there is no data to pretrain the model) but also can highlight the potential of each model. In this context, PC-RNN, neural-ODE, and ANODE models appear to be more promising.

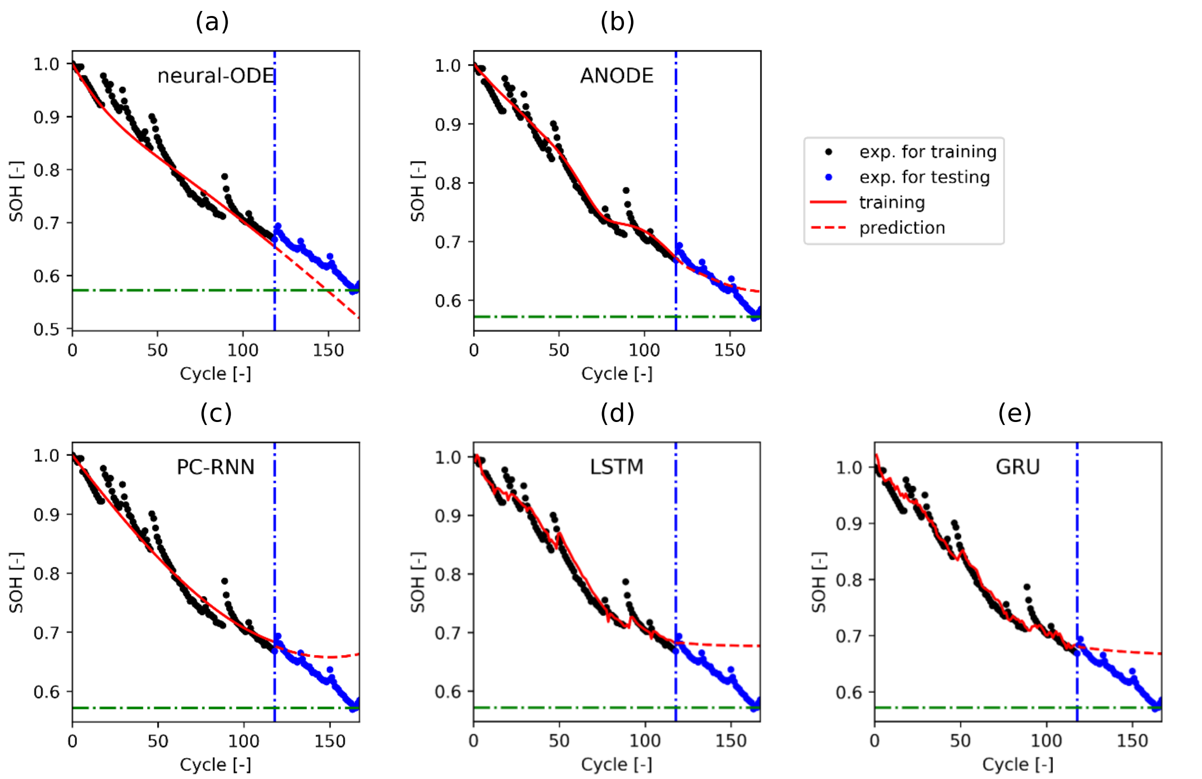


Figure 6 Examples of experimental and predicted SOH for the A2 NASA battery with 70% as a portion of data used for training.

4.2 EOL prediction

The EOL analysis is based on the same experiments described in the previous sections. The accuracy of the models with respect to EOL prediction is assessed using the DEOL (see Equation 6), whose values and confidence intervals are shown in Figure 7. The ODE-based models show good ability in predicting the EOL in both Oxford and NASA datasets, especially in case of shorter predictions (*i.e.* $N_{\mathrm{TP}}/\mathrm{EOL} = 70, 80, 90\%$), see Figure 7(a-j). Specifically for the Oxford datasets, median DEOLs are below 10% in neural-ODE and ANODE. Conversely, at 50% and 60%, only ANODE maintains good accuracy, see Figure 7(a-e). For the NASA datasets (Figure 7(f-j)), the best performances are achieved by the neural-ODE model. In short, ODE-based models underestimate the battery life compared to LSTM and GRU, which is advantageous for circumstances in which prudence is preferred.

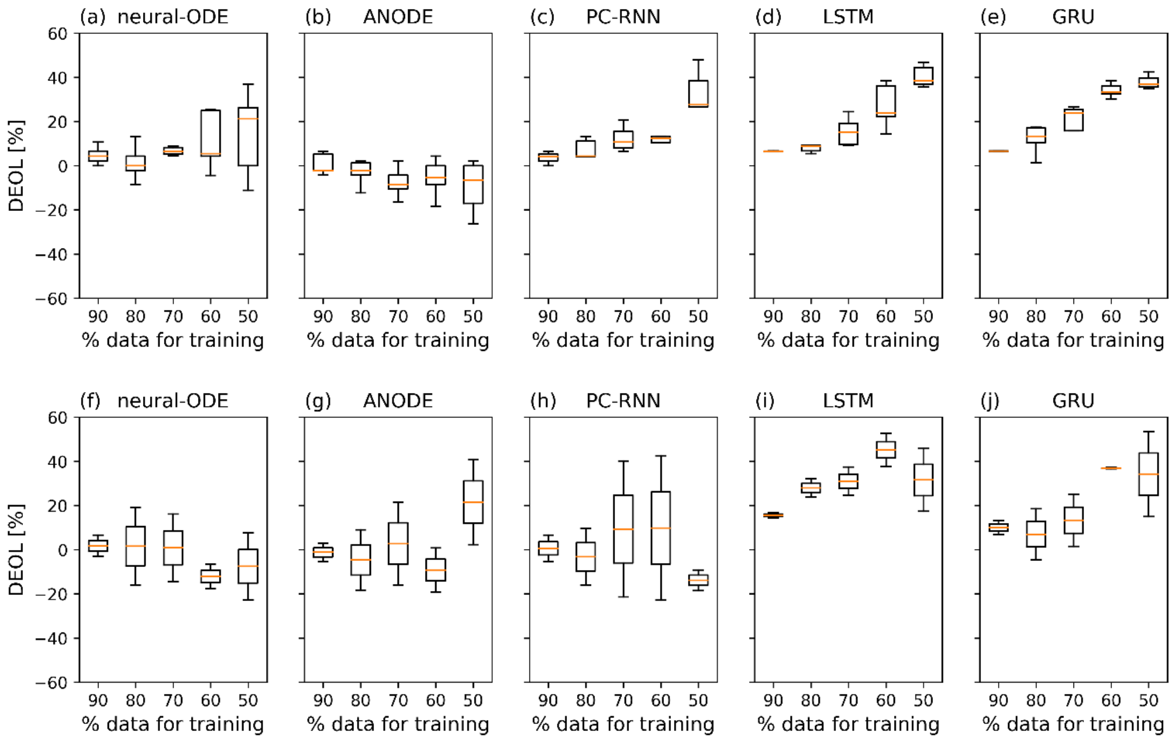


Figure 7 DEOL in Oxford (a-e) and NASA (f-j) experiments, training losses defined as $\mathcal{L}_F$ in Methods, Overview.

4.3 Influence of voltage *versus* capacity features

Since the training losses, $\mathcal{L}$, play a key role in the learning process, we aimed at understanding how results change with the loss term. Therefore, we defined $\mathcal{L}_{F-\mathrm{norm}}$ (see Equation 4), where an identical aggregate weight is assigned to capacity and cumulative time/voltage features. In the Oxford experiments, the hyperparameter optimization (see Table 1) leads to 1 layer in LSTM, 6 layers in GRU, 1 layer in PC-RNN, and respectively 250, 300, 100, 100, and 1000 neurons in LSTM, GRU, PC-RNN, neural-ODE, and ANODE. In ANODE $n_a = 1$. In the NASA experiments, the hyperparameter optimization leads to 2 layers in LSTM, 4 layers in GRU, 4 layers in PC-RNN, and respectively 50, 300, 40, 50, 100 neurons in LSTM, GRU, PC-RNN, neural-ODE, and ANODE. In ANODE $n_a = 1$. The $\mathrm{RMSE}_{\mathrm{SOH}}$ are listed in Table 2. The EOL analysis is reported in Figures 8(a-e) and (f-j), for Oxford and NASA datasets, respectively. From the results in Table 2, and comparing the results obtained by using $\mathcal{L}_F$, only a few cases showed marginal improvement (*i.e.* PC-RNN at 50, 70, 80, and 90% or ANODE at 50%). Neural-ODE improves slightly for all prediction windows, except at 70%. LSTM and GRU exhibit small changes. From the DEOL in Figure 8(a-e), we can observe that only PC-RNN shows improvement. Neural-ODE, ANODE, LSTM, and GRU show overall slightly worst performances. With reference to the NASA results in Table 2, we observe a slight decrease of $\mathrm{RMSE}_{\mathrm{SOH}}$ in $\mathcal{L}_{F-\mathrm{norm}}$ relative to $\mathcal{L}_F$ for ANODE at 60, and 80%, LSTM at 50, 70, and 80%, and PC-RNN in all cases except at 50%, neural-ODE at 50, 60, and 80%, and GRU at 60, 80, and 90%. Overall, the results do not suggest a strong trend. In reference to EOL analysis, see Figure 8(f-j), neural-ODE, PC-RNN, and ANODE are the most conservative. Interestingly, for PC-RNN, DEOL is negative in all cases.

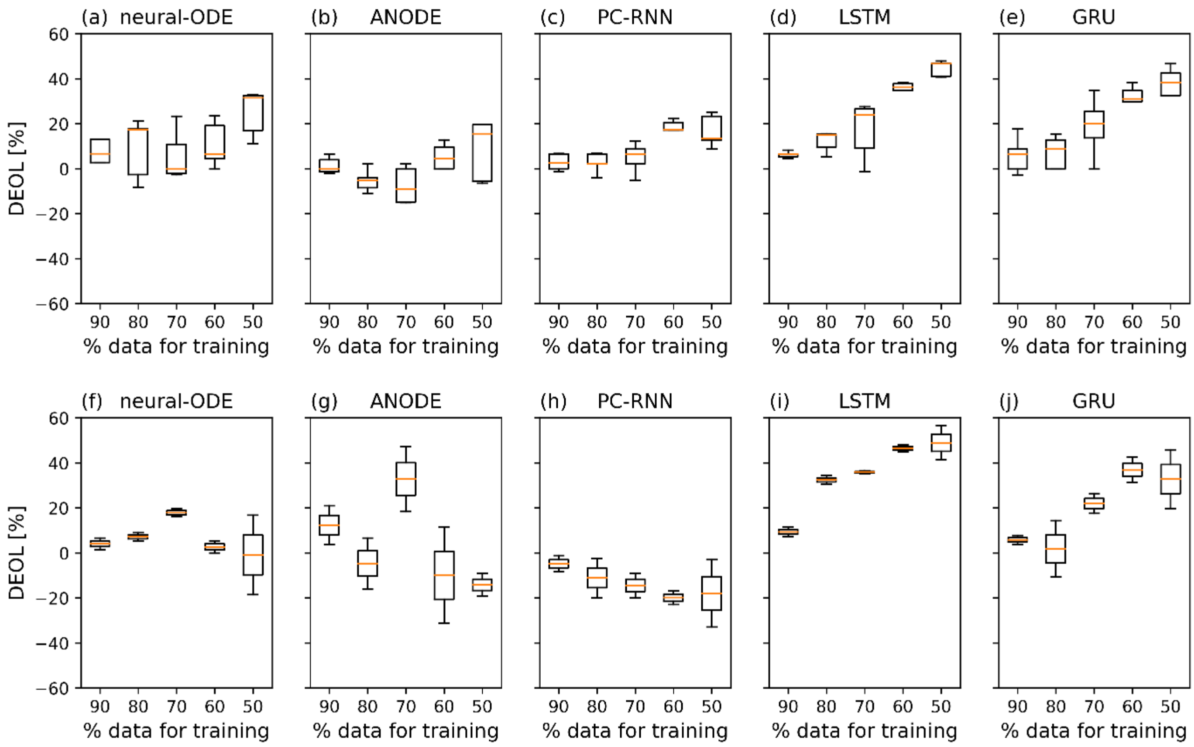
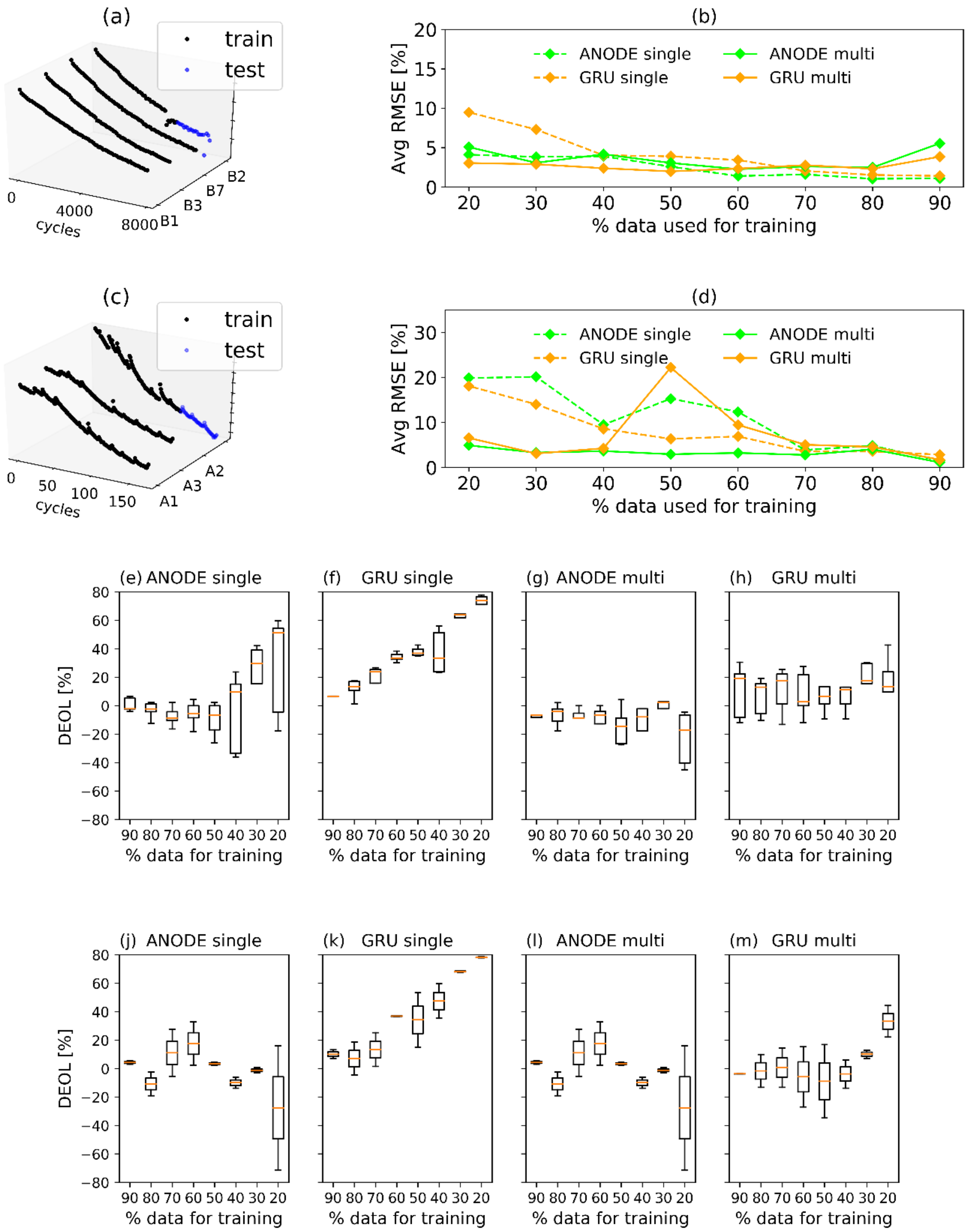


Figure 8 DEOL in Oxford (a-e) and NASA (f-j) experiments, training losses defined as $\mathcal{L}_{F-\mathrm{norm}}$ in Methods, Overview.

5 Multi-battery approach

We aimed at extending the prognostic to longer prediction horizons, by learning from multiple degradation patterns and fully aged cells. Thus, we analyze here earlier stages (*i.e.* $N_{\mathrm{TP}}/\mathrm{EOL}$ =20, 30, 40%). For a preliminary study, only ANODE and GRU methods are benchmarked. In Figure 9(b) and (d) the average $\mathrm{RMSE}_{\mathrm{SOH}}$ is Oxford and NASA experiments, respectively. From both datasets, we conclude that training multiple batteries significantly improves the SOH estimation. In the NASA case with GRU, the average $\mathrm{RMSE}_{\mathrm{SOH}}$ drops from 18.08% to 6.55% at 20%, from 14.02% to 3.10% at 30%, and from 8.58% to 4.25% at 40% (Figure 9(d)). Likewise, with ANODE, the average $\mathrm{RMSE}_{\mathrm{SOH}}$ decreases from 8.36% to 4.91% at 20%, from 8.62% to 4.48% at 30%, and 5.85% to 5.25% at 40%. The same is valid for GRU from the Oxford data (Figure 9(b)). Interestingly, ANODE seems to give similar results for both single-battery and multi-battery experiments, even for long predictions (average $RMSE_{SOH}$ always below 5.55% in Oxford cases). In general, the multi-battery training brings major advantages for long predictions but a small degradation for shorter-term horizons, perhaps due to the bias imposed. Significant improvements are also observed in the EOL estimation, see Figure 9(e-m), for both ANODE and GRU in the Oxford and NASA experiments and long-term predictions horizons.



329 **Figure 9** (a, c) Example of the multi-battery approach applied to the Oxford and NASA batteries (b, d). Comparison of averaged $RMSE_{SOH}$ between single- and multi-battery approaches for the Oxford and NASA experiments, with ANODE and GRU. (e-h) DEOL in Oxford and (j-m) NASA experiments.

6 Conclusions

We investigated several ODE-based machine learning models for battery SOH and EOL predictions. These models included the infinitely deep neural networks neural-ODE and ANODE. We also discretized the underlying ODE as a PC-RNN, where the forward Euler scheme was followed by an explicit correction. These new models were benchmarked against established algorithms (*i.e.* LSTM and GRU). $\mathrm{RMSE_{SOH}}$ and DEOL were chosen as metrics of performances on predictions. ODEs-based (neural-ODE and ANODE) and PC-RNN algorithms outperformed LSTM and GRU in more than 80% of experiments, achieving average errors of 1% in SOH estimation on batteries at 70% of their total cycle life. The multi-battery analysis shows high accuracy even for very early predictions (*i.e.* average error of 4% in SOH estimation on batteries at 20% of their cycle lifetime. From the DEOL results, we can observe that PC-RNN, neural-ODE, and ANODE, mostly underestimate battery remaining life. No specific trends are identified when changing weights of the input features in the training losses. The accuracy of results varies considerably within the two datasets under study. All the algorithms showed lower performances when applied to the NASA experiments, which are characterized by greater stochasticity. Although the presented results are promising, both neural-ODE and ANODE can be further enhanced to allow more robust solutions [70]. More broadly, ODE-based codes have high potential in physics-informed network applications [71], in which physical constraints can be added in the learning algorithm to model complex "grey-box" systems.

List of acronyms and symbols

Acronym	Description
A1-A4	Batteries from the NASA dataset [56]
ANODE	Augmented neural ordinary differential equations
B1-B8	Batteries from the Oxford repository [57]
EOL	End-of-life
GRU	Gated recurrent unit
LSTM	Long-short-term memory
neural-ODE	Neural ordinary differential equations
RNN	Recurrent neural network
PC-RNN	Predictor Corrector-RNN
RUL	Remaining useful life
SOH	State of health

Variable	Description	Unit
DEOL	$\frac{\left(\widehat{\mathrm{EOL}}-\mathrm{EOL}\right)}{\mathrm{EOL}}$	-
$F(\boldsymbol{y})$	Function describing a dynamical system ($=F(\boldsymbol{y},t)$)	-
$F(\boldsymbol{y},\boldsymbol{\theta}(t))$	Continuous function describing a dynamical system in infinitely deep neural networks	-
$F(\boldsymbol{y}_k,\boldsymbol{\theta})$	Discretized function describing a dynamical system in recurrent neural networks	-
$\mathcal{L}$	Training losses	-
$\mathcal{L}_F$	$\sum_{k=1}^{\mathrm{N_{TP}}}\lvert\hat{\boldsymbol{y}}_t-\boldsymbol{y}_t\rvert^2$	-
$\mathcal{L}_{F-\mathrm{norm}}$	$\sum_{k=1}^{\mathrm{N_{TP}}}[\left(\widehat{\mathrm{SOH}}_k-\mathrm{SOH}_k\right)^2+\frac{1}{\mathrm{N_v}}\left\lvert\widehat{\boldsymbol{V}}_k-\boldsymbol{V}_k\right\rvert^2]$	-
$\mathrm{N_V}$	Number of time/voltage features	-
$\mathrm{N_{test}}$	Number of testing points	-
$\mathrm{N_{TP}}$	Number of training points	-
$\mathrm{RMSE_{SOH}}$	Root mean squared error on SOH	-
Q_k	Capacity of a battery at cycle k	Ah
Q_0	Capacity of a fresh battery	Ah
SOH_k	State of health of a battery at cycle k ($=Q_k/Q_0$)	-
k	Cycle number	-
$\boldsymbol{V}_k$	Vector of charging times *versus* voltage at cycle k	-
$\boldsymbol{y}$	State of a continuous dynamical system	-
$\dot{\boldsymbol{y}}$	Evolution of a battery states with time	-
$\boldsymbol{y}_0$	Initial state of a dynamic system at time zero	-
$\boldsymbol{y}_k$	State of a dynamical system at discrete times k ($=\begin{pmatrix}\mathrm{SOH}_k\\ \boldsymbol{V}_k\end{pmatrix}$	-
$\boldsymbol{\theta}(t)$, $\boldsymbol{\theta}$	Networks parameter for the continuous and discretized cases	-
$\widehat{(\cdot)}$	Predicted quantities	-

Acknowledgments

The authors gratefully acknowledge the Research Grant Council of Hong Kong (RGC Ref No. 16201820 and 16206019) and the Hong Kong Innovation and Technology Fund (grant number UIM/369) for financial support. S. Pepe and E. Quattrocchi kindly acknowledge the support of the Hong Kong Ph.D. Fellowship Scheme.

Bibliography

1. Wang, Y., C. Zhang, and Z. Chen, *A method for joint estimation of state-of-charge and available energy of LiFePO4 batteries.* Applied Energy, 2014. **135**: p. 81-87.
2. Zhang, X., et al., *An on-line estimation of battery pack parameters and state-of-charge using dual filters based on pack model.* Energy, 2016. **115**: p. 219-229.
3. Boretti, A., *Dependent performance of South Australian wind energy facilities with respect to resource and grid availability.* Energy Storage, 2019. **1**(6): p. e97.
4. IEA. *Battery storage is (almost) ready to play the flexibility game*. [Commentary] 2019; Available from: https://www.iea.org/commentaries/battery-storage-is-almost-ready-to-play-the-flexibility-game, (accessed 31 March 2021).
5. Zhang, W., X. Li, and X. Li, *Deep learning-based prognostic approach for lithium-ion batteries with adaptive time-series prediction and on-line validation.* Measurement, 2020. **164**: p. 108052.
6. Wang, Q., Y. Jiang, and Y. Lu, *State of health estimation for lithium-ion battery based on D-UKF.* International Journal of Hybrid Information Technology, 2015. **8**(7): p. 55-70.
7. Xiong, R., et al., *A double-scale, particle-filtering, energy state prediction algorithm for lithium-ion batteries.* IEEE Transactions on Industrial Electronics, 2018. **65**(2): p. 1526-1538.
8. Zou, Y., et al., *Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicles.* Journal of Power Sources, 2015. **273**: p. 793-803.
9. He, W., et al., *State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation.* International Journal of Electrical Power & Energy Systems, 2014. **62**: p. 783-791.
10. Miao, Q., et al., *Remaining useful life prediction of lithium-ion battery with unscented particle filter technique.* Microelectronics Reliability, 2013. **53**(6): p. 805-810.
11. Su, X., et al., *Interacting multiple model particle filter for prognostics of lithium-ion batteries.* Microelectronics Reliability, 2017. **70**: p. 59-69.
12. Tian, Y., et al., *A combined method for state-of-charge estimation for lithium-ion batteries using a long short-term memory network and an adaptive cubature Kalman filter.* Applied Energy, 2020. **265**: p. 114789.
13. Xing, Y., et al., *An ensemble model for predicting the remaining useful performance of lithium-ion batteries.* Microelectronics Reliability, 2013. **53**(6): p. 811-820.
14. Yang, F., et al., *State-of-charge estimation of lithium-ion batteries using LSTM and UKF.* Energy, 2020. **201**: p. 117664.
15. Berecibar, M., et al., *Critical review of state of health estimation methods of Li-ion batteries for real applications.* Renewable and Sustainable Energy Reviews, 2016. **56**: p. 572-587.
16. Liu, Z., et al., *Particle learning framework for estimating the remaining useful life of lithium-ion batteries.* IEEE Transactions on Instrumentation and Measurement, 2017. **66**(2): p. 280-293.
17. Hu, X., et al., *Battery lifetime prognostics.* Joule, 2020. **4**(2): p. 310-346.
18. Anton, J.C.A., et al., *Support vector machines used to estimate the battery state of charge.* IEEE Transactions on Power Electronics, 2013. **28**(12): p. 5919-5926.
19. Klass, V., M. Behm, and G. Lindbergh, *A support vector machine-based state-of-health estimation method for lithium-ion batteries under electric vehicle operation.* Journal of Power Sources, 2014. **270**: p. 262-272.

20. Weng, C., et al., *On-board state of health monitoring of lithium-ion batteries using incremental capacity analysis with support vector regression.* Journal of Power Sources, 2013. **235**: p. 36-44.
21. Wei, J., G. Dong, and Z. Chen, *Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression.* IEEE Transactions on Industrial Electronics, 2017. **65**(7): p. 5634-5643.
22. Li, H., D. Pan, and C.P. Chen, *Intelligent prognostics for battery health monitoring using the mean entropy and relevance vector machine.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2014. **44**(7): p. 851-862.
23. Wang, D., Q. Miao, and M. Pecht, *Prognostics of lithium-ion batteries based on relevance vectors and a conditional three-parameter capacity degradation model.* Journal of Power Sources, 2013. **239**: p. 253-264.
24. Zhou, Y., et al., *A novel health indicator for on-line lithium-ion batteries remaining useful life prediction.* Journal of Power Sources, 2016. **321**: p. 1-10.
25. Richardson, R.R., M.A. Osborne, and D.A. Howey, *Gaussian process regression for forecasting battery state of health.* Journal of Power Sources, 2017. **357**: p. 209-219.
26. He, W., et al., *Prognostics of lithium-ion batteries based on Dempster–Shafer theory and the Bayesian Monte Carlo method.* Journal of Power Sources, 2011. **196**(23): p. 10314-10321.
27. Hu, X., et al., *Battery health prognosis for electric vehicles using sample entropy and sparse Bayesian predictive modeling.* IEEE Transactions on Industrial Electronics, 2015. **63**(4): p. 2645-2656.
28. Tagade, P., et al., *Deep Gaussian process regression for lithium-ion battery health prognosis and degradation mode diagnosis.* Journal of Power Sources, 2020. **445**: p. 227281.
29. Mansouri, S.S., et al., *Remaining useful battery life prediction for UAVs based on machine learning.* IFAC-PapersOnLine, 2017. **50**(1): p. 4727-4732.
30. Li, Y., et al., *Random forest regression for online capacity estimation of lithium-ion batteries.* Applied Energy, 2018. **232**: p. 197-210.
31. Khumprom, P. and N. Yodo, *A data-driven predictive prognostic model for lithium-ion batteries based on a deep learning algorithm.* Energies, 2019. **12**(4): p. 660.
32. Hong, J., Z. Wang, and Y. Yao, *Fault prognosis of battery system based on accurate voltage abnormity prognosis using long short-term memory neural networks.* Applied Energy, 2019. **251**: p. 113381.
33. Abdel-Nasser, M. and K. Mahmoud, *Accurate photovoltaic power forecasting models using deep LSTM-RNN.* Neural Computing and Applications, 2019. **31**(7): p. 2727-2740.
34. Cao, J., Z. Li, and J. Li, *Financial time series forecasting model based on CEEMDAN and LSTM.* Physica A: Statistical Mechanics and its Applications, 2019. **519**: p. 127-139.
35. Muzaffar, S. and A. Afshari, *Short-term load forecasts using LSTM networks.* Energy Procedia, 2019. **158**: p. 2922-2927.
36. Siami-Namini, S., N. Tavakoli, and A.S. Namin, *A comparison of ARIMA and LSTM in forecasting time series*. in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2018. IEEE.
37. Zhao, Z., et al., *LSTM network: a deep learning approach for short-term traffic forecast.* IET Intelligent Transport Systems, 2017. **11**(2): p. 68-75.
38. Bian, C., et al., *State-of-charge sequence estimation of lithium-ion battery based on bidirectional long short-term memory encoder-decoder architecture.* Journal of Power Sources, 2020. **449**: p. 227558.

39. Wu, Y., et al., *State of health estimation for lithium-ion batteries based on healthy features and long short-term memory.* IEEE Access, 2020. **8**: p. 28533-28547.
40. Zhang, Y., et al., *A LSTM-RNN method for the lithium-ion battery remaining useful life prediction*. in *2017 Prognostics and System Health Management Conference (PHM-Harbin)*. 2017. IEEE.
41. Zhang, Y., et al., *Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries.* IEEE Transactions on Vehicular Technology, 2018. **67**(7): p. 5695-5705.
42. Zhao, R., et al., *A compact unified methodology via a recurrent neural network for accurate modeling of lithium-ion battery voltage and state-of-charge*. in *2017 IEEE Energy Conversion Congress and Exposition (ECCE)*. 2017. IEEE.
43. Liu, Y., G. Zhao, and X. Peng, *Deep learning prognostics for lithium-ion battery based on ensembled long short-term memory networks.* IEEE Access, 2019. **7**: p. 155130-155142.
44. Yu, S., et al., *A domain adaptive convolutional LSTM model for prognostic remaining useful life estimation under variant conditions*. in *2019 Prognostics and System Health Management Conference (PHM-Paris)*. 2019. IEEE.
45. Zhang, H., et al., *Implementation of generative adversarial network-CLS combined with bidirectional long short-term memory for lithium-ion battery state prediction.* Journal of Energy Storage, 2020. **31**: p. 101489.
46. Chemali, E., et al., *Long short-term memory networks for accurate state-of-charge estimation of Li-ion batteries.* IEEE Transactions on Industrial Electronics, 2018. **65**(8): p. 6730-6739.
47. Hannan, M.A., et al., *A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: challenges and recommendations.* Renewable and Sustainable Energy Reviews, 2017. **78**: p. 834-854.
48. Ng, M.-F., et al., *Predicting the state of charge and health of batteries using data-driven machine learning.* Nature Machine Intelligence, 2020: p. 1-10.
49. Xiong, R., L. Li, and J. Tian, *Towards a smarter battery management system: A critical review on battery state of health monitoring methods.* Journal of Power Sources, 2018. **405**: p. 18-29.
50. Vidal, C., et al., *Machine learning applied to electrified vehicle battery state of charge and state of health estimation: State-of-the-art.* IEEE Access, 2020. **8**: p. 52796-52814.
51. Chen, R.T., et al., *Neural ordinary differential equations.* Advances in Neural Information Processing Systems, 2018: p. 6571-6583.
52. Dupont, E., A. Doucet, and Y.W. the, *Augmented neural ODEs*. in *Advances in Neural Information Processing Systems*. 2019.
53. Goodfellow, I., Y. Bengio, and A. Courville, *Deep learning MIT press (2016)*.
54. Chung, J., et al., *Empirical evaluation of gated recurrent neural networks on sequence modeling.* arXiv preprint arXiv:1412.3555, 2014.
55. Hochreiter, S. and J. Schmidhuber, *LSTM can solve hard long time lag problems*. in *Advances in neural information processing systems*. 1997.
56. Saha, B. and K. Goebel, *Battery data set*. Available from: http://ti.arc.nasa.gov/project/prognostic-data-repository (accessed 31 March 2021).
57. Birkl, C., *Oxford battery degradation dataset 1*, in *Long term battery ageing tests of 8 Kokam (SLPB533459H4) 740 mAh lithium-ion pouch cells*. 2017, University of Oxford, (accessed 31 March 2021).
58. Richardson, R.R., et al., *Gaussian process regression for in situ capacity estimation of lithium-ion batteries.* IEEE Transactions on Industrial Informatics, 2019. **15**(1): p. 127-138.

59. Watrin, N., B. Blunier, and A. Miraoui. *Review of adaptive systems for lithium batteries state-of-charge and state-of-health estimation*. in *2012 IEEE Transportation Electrification Conference and Expo (ITEC)*. 2012. IEEE.
60. Severson, K.A., et al., *Data-driven prediction of battery cycle life before capacity degradation.* Nature Energy, 2019. **4**(5): p. 383-391.
61. Knauf, A., *Ordinary differential equations*, in *Mathematical Physics: Classical Mechanics*. 2018, Springer Berlin Heidelberg: Berlin, Heidelberg. p. 31-60.
62. He, K., et al., *Deep residual learning for image recognition*. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016.
63. Teh, Y., A. Doucet, and E. Dupont, *Augmented neural ODEs.* Advances in Neural Information Processing Systems 32 (NIPS 2019), 2019. **32**(2019).
64. Gholami, A., K. Keutzer, and G. Biros, *Anode: Unconditionally accurate memory-efficient gradients for neural odes.* arXiv preprint arXiv:1902.10298, 2019.
65. Elman, J.L., *Finding structure in time.* Cognitive science, 1990. **14**(2): p. 179-211.
66. Gers, F.A., J. Schmidhuber, and F. Cummins, *Learning to forget: continual prediction with LSTM.* 1999.
67. Yang, F., et al., *State-of-charge estimation of lithium-ion batteries based on gated recurrent neural network.* Energy, 2019. **175**: p. 66-75.
68. Pontryagin, L.S., *Mathematical theory of optimal processes*. 2018: Routledge.
69. *PyTorch implementation of differentiable ODE solvers*. Available from: https://github.com/rtqichen/torchdiffeq , (accessed 31 March 2021).
70. Tuor, A., J. Drgona, and D. Vrabie, *Constrained neural ordinary differential equations with stability guarantees.* arXiv preprint arXiv:2004.10883, 2020.
71. Raissi, M., P. Perdikaris, and G.E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations.* Journal of Computational Physics, 2019. **378**: p. 686-707.