

---

# Lithium-Ion Battery System Health Monitoring and Resistance-Based Fault Analysis from Field Data Using Recursive Spatiotemporal Gaussian Processes

---

**Joachim Schaeffer**

Control and Cyber-Physical Systems  
Technical University of Darmstadt, Germany  
Massachusetts Institute of Technology  
Cambridge, MA, USA

**Eric Lenz**

Control and Cyber-Physical Systems  
Technical University of Darmstadt, Germany

**Duncan Gulla**

Control and Cyber-Physical Systems  
Technical University of Darmstadt, Germany

**Martin Z. Bazant**

Massachusetts Institute of Technology  
Cambridge, MA, USA

**Richard D. Braatz**

Massachusetts Institute of Technology  
Cambridge, MA, USA  
braatz@mit.edu

**Rolf Findeisen**

Control and Cyber-Physical Systems  
Technical University of Darmstadt, Germany  
rolf.findeisen@iat.tu-darmstadt.de

## Abstract

Health monitoring is important for the safe operation of battery systems. We use recursive spatiotemporal Gaussian processes to model the resistance of lithium iron phosphate batteries from field data. These processes scale linearly with the number of data points, allowing online monitoring. The kernels separate the time-dependent and operating-point-dependent resistance contributions. We develop probabilistic fault probabilities based on time-dependent resistance estimates. The fault analysis underlines that often, only a single cell shows abnormal behavior, consistent with weakest-link failure for cells connected in series, amplified by local resistive heating. The results further the understanding of how battery packs degrade and fail in the field and demonstrate the potential of online monitoring. The data set contains 28 battery systems returned to the manufacturer for warranty, each with eight cells in series, totaling 224 cells and 133 million data rows. The data and code are openly available.

*Disclaimer:* This extended abstract is based on [1].

## 1 Introduction

Lithium-Ion Batteries (LIBs) are essential for Electric Vehicles (EVs), grid storage, mobile applications, consumer electronics, and more. Over the last 30 years, remarkable advances have led to long-lasting cells with high energy efficiency and density [2]. Safe operation of LIBs is vital to protect life and property and to strengthen trust in LIBs. In the past, LIB fires erupted in many different applications, including EVs [3], stationary storage [4], and electric bicycles [5]. Monitoring batteries during operation is important to have a chance of detecting electrical or mechanical abuse of the system or the onset of accelerated cell degradation, which is critical to reducing the potential of such fires. Faults are abnormal events that cause the system to behave in an unintended way or stop operating. Battery system faults can be auxiliary, sensor, or battery faults. Fault detection methods

can be categorized as signal-based or model-based. Much research considers fast signal-based fault detection for battery systems [6–8]. Model-based fault detection methods complement signal-based fault detection because they are usually computationally more complex and slower. However, they can potentially detect certain faults earlier and improve robustness (see [9–15]). In particular, battery system faults, which are often due to slow battery degradation, can be addressed by a model to estimate State Of Health (SOH) batteries [16–19].

Physics-based, machine learning, and empirical methods are available to model the cycling and degradation behavior of battery cells and systems [20–27]. Choosing an appropriate model is important to optimally use the available data [20]. Physics-based battery models are often challenging to parameterize with battery field data, i.e., time series data consisting of noisy temperature, current, and voltage measurements corresponding to the system, module, and cell level [28]. Equivalent Circuit Models (ECMs) are an alternative because they are easier to parameterize with limited data [21] and are applicable to field data [16]. We apply spatiotemporal Gaussian Processes (GPs) [29, 30] to battery field data with uncontrolled operational conditions [1], that is, the device is in the hands of the customer, who can use and potentially abuse the battery system. We build on a hybrid approach that uses GPs and ECMs developed in [16] for single-cell lead-acid batteries and adapt the model to Lithium-Iron-Phosphate (LFP) battery systems. This hybrid approach estimates an operational point-dependent and a temperature-dependent resistance as two resistors in series [16]. We developed fault probabilities [1], which allow monitoring the homogeneity of cell resistances and the probability of a single cell exceeding a maximum resistance threshold. The data set contains 28 systems and 133 million data rows. The data and associated Python package *BattGP* are available as open source. This data set is the first field data set of batteries that failed in the field.

## 2 Data and Model

The data set contains data from 28 portable 24 V LFP battery systems, each with eight cells in series (224 cells in total), with approximately 160 Ah nominal capacity per cell. The specific use case of each system is unknown, but battery systems of this size are typically used as power sources for recreational vehicles, solar energy storage, and more. All battery systems in this data set showed some form of unsatisfactory behavior and were returned to the manufacturer. This can be motivated by personal judgment, Battery Management System (BMS) warnings, or customer support advice. This data set comprises a very small fraction of the batteries sold in this version. Therefore, this data set is biased and not representative of the operational data of the entire population of this system type. An improved version replaced this battery system type. The battery system manufacturer provided the data set for this study and allowed its open-source release under the condition of anonymity.

Each battery system consists of 8 prismatic cells in series. Each system has one load current sensor and each cell has one voltage sensor. The four temperature sensors are placed between adjacent cells; thus, each temperature sensor is shared by two cells (Fig. 1a). Furthermore, these battery systems have active cell balancing. The available measurements for the systems range from a single month to five years. Consequently, the number of data rows per system varies from several thousand to millions, depending on the duration of battery operation. The data set contains a total of 133 million rows. More technical specifications are described in Sec. A.7, Tab. 2. As an example, the 24.2 million data rows associated with system 8 are visualized in Sec. A.8, Fig. 5.

GPs,  $f(x) \sim \text{GP}(\mu(x), k(x, x'))$ , defined by a mean function  $\mu(x)$  and a covariance function  $k(x, x')$ , are nonparametric probabilistic models defining a distribution of functions [31]. GPs are suitable for modeling the time- and operating point behavior of batteries [1, 16, 17]. We base our analysis on a GP-ECM modeling framework that uses an ECM consisting of two series resistors modeled by two additive kernels (Fig. 1b) suggested by [16]. A Radial Basis Function (RBF) kernel models the current, State Of Charge (SOC), and temperature-dependent resistance. The resulting RBF kernel function estimates are infinitely many times differentiable, roughly in agreement with smooth experimental results [32]. A non-stationary Wiener velocity kernel models time-dependent resistance, which is expected to increase with increasing degradation. We use a linear approximate Open-Circuit Voltage (OCV) because the exact OCV was not available (Sec. A.6, Fig. 4). Any deviation from the true OCV will lead to a biased resistance; however, this bias is only a function of SOC and will therefore affect mainly the SOC dimension of the RBF kernel (Fig. 3b). Furthermore, even if a precisely measured OCV would be available at the beginning of life, the OCV is expected to change with degradation. The GP-ECM algorithm consists of the following steps for a single cell. Voltage,

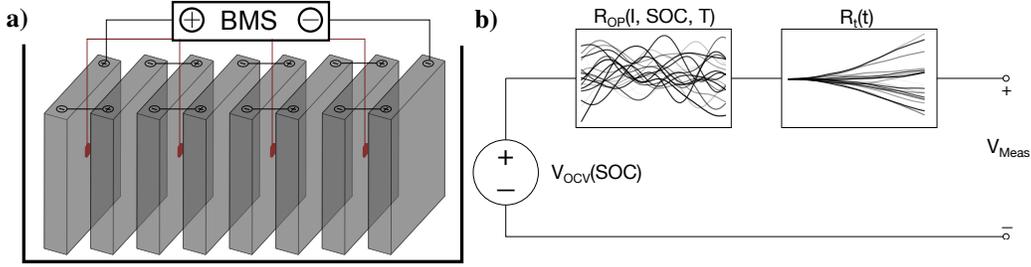


Figure 1: a) Battery system with 8 cells, temperature sensors (red) shared by two cells. b) ECM model with illustrations of random draws from the kernel functions. Adapted from [1].

current, SOC, and time are extracted and down-selected to use only data points with significant information for the learning task and to avoid sparse extreme conditions where little data are available or the simple R-R model is less suitable. We use only discharge data because, in the data set, more dynamics are present during discharging (see Tab. 1 for selection criteria). In an online setting, the ECM-GP needs to be updated continuously with new data arriving. The computational complexity of exact GPs scales with  $O(n^3)$ , where  $n$  is the number of data points, making it computationally infeasible to update the model continuously with new data arriving. Therefore, we use a recursive spatiotemporal GP, which scales linearly with the number of data points [16, 29, 30], allowing the processing of millions of data points on a laptop computer and suitable for the use in embedded BMS systems. The spatiotemporal GP approach is described mathematically in Sec. A.1. Our approach is motivated by [16] but not identical. We generate an independent model for each cell, resulting in eight models for each battery system. The hyperparameters were optimized as outlined in Sec. A.3. Identical hyperparameters are used for all battery systems because the hyperparameters characterize the system behavior, and all systems are identical in construction.

### 3 Results

Figure 2 shows the time-dependent resistance modeled by the Wiener velocity kernel at the same reference operating point ( $-15$  A, 90% SOC,  $25^\circ\text{C}$ ) for systems 6, 8 and 9, selected as a showcase for this extended abstract. Systems 6 and 8 show seasonal temperature variations. These variations pose challenges for the GP framework because the Wiener velocity kernel resistance estimates are affected by the temperature variations, with lower temperatures leading to higher resistance estimates (e.g., [32, 33]). Consequently, using the resistance time derivative for forecasting (as suggested in [16] for lead-acid batteries) appears challenging for lithium-ion batteries. Furthermore, the resistance time derivative at time  $t_0$  should not be used for forecasting as it can be significantly influenced by data  $t > t_0$ . Systems 6 and 9 show a single cell behaving differently than the remaining cells. Battery system 8 shows an increasing resistance trajectory for all cells at roughly the same time. Their resistance trajectories suggest that these systems were healthy for most of their operation, with degradation accelerating toward the end of use. These resistance patterns can have a wide variety of root causes. To further understand the cause of individual degradation, a mechanical inspection of the systems would be necessary, but this cannot be done because the returned battery systems are not physically available to the authors.

We propose to define a battery pack consisting of cells in series to have an acceptable resistance distribution if the internal resistances of each cell are within a resistance band centered around a robust estimate of the mean of the other cells. We use the Hodge-Lehman estimator [34] to estimate the location (i.e., the mean) of cell resistances. A derivation of the resistance band probabilities is shown in Sec. A.4. The spatiotemporal GP walks forward in time using a Kalman filter. To calculate the fault probabilities, we use the Kalman filter resistance estimates at time  $t_k$ , which depend only on data up to time  $t_k$  (Fig. 2 bottom). We set  $b = 0.55$  m $\Omega$  based on the resistance spread observed at the beginning of life for the investigated systems. The forward fault probabilities need a certain amount of data to settle in, here in the order of 30–200 days (Fig. 2 bottom), depending on the usage of the system. During this initial period, the GP learns the operating characteristics of the systems (Fig. 3).

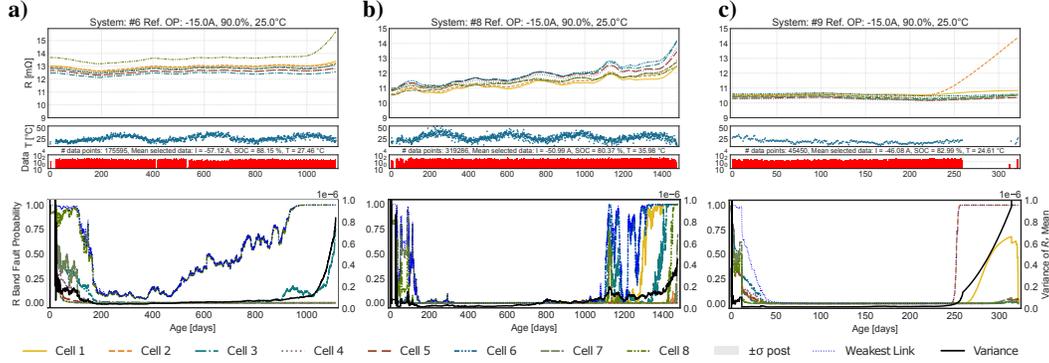


Figure 2: a) System 6, b) System 8, c) System 9. Top: Time-dependent Wiener velocity kernel resistance at  $-15$  A, 90% SOC,  $25^\circ\text{C}$ . Top middle: Average daily temperatures. Top bottom: data points. Bottom: Forward Kalman filter based resistance distribution fault probabilities for each cell in color. Variance of the GP resistance mean in black. Adapted from [1].

**System 6:** The fault probability for cell 8 is significantly higher than for the other cells even after the Kalman filter settled in after approximately 200 days. After 500 days, the fault probability of cell 8 starts increasing and crosses 0.5 shortly before 800 days. The remaining cells have a very low fault probability.

**System 8:** Around 1100–1200 days, multiple cell resistance fault probabilities and the weakest link statistic increase sharply, coinciding with the lower temperatures. Around 1300 days, the weakest link statistic reaches values close to 1, which is consistent with the increasing spread between cell resistances. System 8 reached an equivalent full cycle count of 1531 cycles, with seasonal temperature variations but a long-term upward resistance trend with increasingly inhomogeneous resistance as commonly observed by cell-to-cell variability [35].

**System 9:** The fault probabilities increase quickly around day 250 for cell 2. In contrast, the other cells have a low fault probability, except for cell 1, which might be affected because it shares a temperature sensor with cell 2. The operating characteristics, i.e.,  $R(I, \text{SOC}, T)$ , modeled by the RBF kernel (Fig. 3), are consistent with physical expectations and experimental data from [32]. The current-dependent resistance shows a downward trend with increasing discharge current [36, 37]. The SOC dependent resistance is fairly flat. Some unexpected small upward and downward trends can be attributed to the linear pseudo OCV (Sec. A.6, Fig. 4). The temperature characteristics show increasing resistance at low and very high temperatures, which is in line with physical expectations (for more details, see SI of [1]). Here, we'd like to point out further challenges. First, the influence of seasonal temperature variations on the Wiener velocity kernel currently prevents using resistance derivatives for forecasting. Second, reducing the time and data it takes for the Kalman filter to settle in is important. Pre-initializing the matrices of the spatiotemporal GP based on laboratory test data could be a possible pathway. Third, the cells share common operating characteristics; however, there are also cell-to-cell variations and sensor bias affecting the operating characteristics (Figs. 3, 2 and [35]). Coupling the individual cell GPs could further improve robustness.

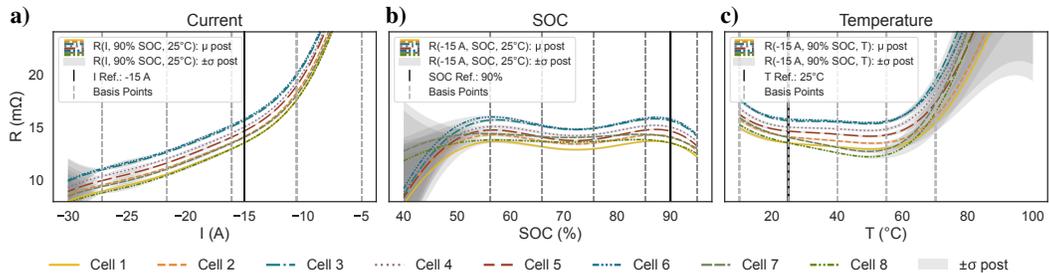


Figure 3: Operating characteristics for all cells of system 8 at  $t = 1476$  days, using 319,278 data points for each cell. a) Current dependency, b) SOC dependency, c) Temperature dependency.

## 4 Conclusion

Health monitoring is important for the safe operation of battery systems. We use a recursive spatiotemporal GP-ECM framework to analyze faults from battery field data. The estimated time-dependent resistance is one possible SOH metric for lithium-ion battery packs. Furthermore, we developed resistance fault probabilities using the individual cell resistance of the cells in a pack, which is suitable for early online monitoring. The results show that often, a single cell with abnormal performance can cause the end of a system's use and suggest that such faults can be detected with the proposed framework. Abnormal performance can occur after heavy use due to degradation or other issues. Furthermore, the results support that if a cell has a higher resistance early on, this can already be an indicator that this cell will age faster than the remaining cells. Physically, this could occur by enhanced resistive heating of the most degraded cell, thereby accelerating its degradation [38], which has been observed in the case of lithium-ion batteries connected in series [39].

**Data and Code Availability** The associated Python software *BattGP* is available on GitHub: <https://github.com/JoachimSchaeffer/BattGP>. The dataset is available on Zenodo: <https://zenodo.org/records/13715694>.

## References

- [1] Joachim Schaeffer, Eric Lenz, Duncan Gulla, Martin Z. Bazant, Richard D. Braatz, and Rolf Findeisen. Gaussian process-based online health monitoring and fault analysis of lithium-ion battery systems from field data. *Cell Reports Physical Science*, 5(11), 2024.
- [2] George E. Blomgren. The development and future of lithium ion batteries. *Journal of The Electrochemical Society*, 164(1):A5019–A5025, 2016.
- [3] Peiyi Sun, Roeland Bisschop, Huichang Niu, and Xinyan Huang. A review of battery fires in electric vehicles. *Fire Technology*, 56:1361–1410, 2020.
- [4] Dong-Hyeon Im and Ji-Bum Chung. Social construction of fire accidents in battery energy storage systems in korea. *Journal of Energy Storage*, 71:108192, 2023.
- [5] Winnie Hu. How New York plans to regulate e-bikes in the wake of deadly fires, Sept. 2023. URL <https://www.nytimes.com/article/ebike-laws-nyc.html>.
- [6] Xiaosong Hu, Kai Zhang, Kailong Liu, Xianke Lin, Satadru Dey, and Simona Onori. Advanced fault diagnosis for lithium-ion battery systems: A review of fault mechanisms, fault features, and diagnosis procedures. *IEEE Industrial Electronics Magazine*, 14(3):65–91, 2020.
- [7] Qingsong Wang, Binbin Mao, Stanislav I Stoliarov, and Jinhua Sun. A review of lithium ion battery failure mechanisms and fire prevention strategies. *Progress in Energy and Combustion Science*, 73:95–131, 2019.
- [8] Manh-Kien Tran and Michael Fowler. A review of lithium-ion battery fault diagnostic algorithms: Current progress and future challenges. *Algorithms*, 13(3):62, 2020.
- [9] Zeyu Chen, Rui Xiong, Jinpeng Tian, Xiong Shang, and Jiahuan Lu. Model-based fault diagnosis approach on external short circuit of lithium-ion battery used in electric vehicles. *Applied Energy*, 184:365–374, 12 2016.
- [10] Kai Zhang, Xiaosong Hu, Yonggang Liu, Xianke Lin, and Wenxue Liu. Multi-fault detection and isolation for lithium-ion battery systems. *IEEE Transactions on Power Electronics*, 37(1): 971–989, 2021.
- [11] Chandrani Sadhukhan, Swarup Kumar Mitra, Suvanjan Bhattacharyya, Eydah Almatrafi, Bahaa Saleh, and Mrinal Kanti Naskar. Modeling and simulation of high energy density lithium-ion battery for multiple fault detection. *Scientific Reports*, 12(1):9800, 2022.
- [12] Yang Zhao, Peng Liu, Zhenpo Wang, Lei Zhang, and Jichao Hong. Fault and defect diagnosis of battery for electric vehicles based on big data analysis methods. *Applied Energy*, 207:354–362, 2017.

- [13] Akash Samanta, Sumana Chowdhuri, and Sheldon S. Williamson. Machine learning-based data-driven fault detection/diagnosis of lithium-ion battery: A critical review. *Electronics*, 10(11):1309, 2021.
- [14] Michael Schmid, Hans-Georg Kneidinger, and Christian Endisch. Data-driven fault diagnosis in battery systems through cross-cell monitoring. *IEEE Sensors Journal*, 21(2):1829–1837, 2020.
- [15] Jingyuan Zhao, Heping Ling, Junbin Wang, Andrew F. Burke, and Yubo Lian. Data-driven prediction of battery failure for electric vehicles. *Iscience*, 25(4):104172, 2022.
- [16] Antti Aitio and David A. Howey. Predicting battery end of life from solar off-grid system field data using machine learning. *Joule*, 5(12):3204–3220, 2021.
- [17] Antti Aitio, Dominik Jöst, Dirk Uwe Sauer, and David A. Howey. Learning battery model parameter dynamics from data with recursive gaussian process regression, 2023. arXiv preprint, <https://arxiv.org/abs/arXiv:2304.13666>.
- [18] Rui Xiong, Linlin Li, and Jinpeng Tian. Towards a smarter battery management system: A critical review on battery state of health monitoring methods. *Journal of Power Sources*, 405:18–29, 2018.
- [19] M. Bercibar, I. Gandiaga, I. Villarreal, N. Omar, J. Van Mierlo, and P. Van den Bossche. Critical review of state of health estimation methods of li-ion batteries for real applications. *Renewable and Sustainable Energy Reviews*, 56:572–587, 2016.
- [20] Joachim Schaeffer, Giacomo Galuppini, Jinwook Rhyu, Patrick A. Asinger, Robin Droop, Rolf Findeisen, and Richard D. Braatz. Cycle life prediction for lithium-ion batteries: Machine learning and more. In *Proceedings of the American Control Conference*, pages 763–768, 2024. <https://arxiv.org/abs/2404.04049>.
- [21] Ulrike Krewer, Fridolin Röder, Eranda Harinath, Richard D Braatz, Benjamin Bedürftig, and Rolf Findeisen. Dynamic models of li-ion batteries for diagnosis and operation: A review and perspective. *Journal of The Electrochemical Society*, 165(16):A3656–A3673, 2018.
- [22] Venkatasailanathan Ramadesigan, Paul WC Northrop, Sumitava De, Shriram Santhanagopalan, Richard D Braatz, and Venkat R Subramanian. Modeling and simulation of lithium-ion batteries from a systems engineering perspective. *Journal of the Electrochemical Society*, 159(3):R31, 2012.
- [23] John Newman and William Tiedemann. Porous-electrode theory with battery applications. *AIChE Journal*, 21(1):25–41, 1975.
- [24] Marc Doyle, Thomas F. Fuller, and John Newman. Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell. *Journal of the Electrochemical Society*, 140(6):1526–1533, 1993.
- [25] Joachim Schaeffer, Eric Lenz, William C Chueh, Martin Z Bazant, Rolf Findeisen, and Richard D Braatz. Interpretation of high-dimensional linear regression: Effects of nullspace and regularization demonstrated on battery data. *Computers & Chemical Engineering*, 180:108471, 2024.
- [26] Kristen A. Severson, Peter M. Attia, Norman Jin, Nicholas Perkins, Benben Jiang, Zi Yang, Michael H. Chen, Muratahan Aykol, Patrick K. Herring, Dimitrios Fraggedakis, Martin Z. Bazant, Stephen J. Harris, William C. Chueh, and Richard D. Braatz. Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy*, 4(5):383–391, 2019.
- [27] Joachim Schaeffer, Paul Gasper, Esteban Garcia-Tamayo, Raymond Gasper, Masaki Adachi, Juan Pablo Gaviria-Cardona, Simon Montoya-Bedoya, Anoushka Bhutani, Andrew Schiek, Rhys Goodall, Rolf Findeisen, Richard D. Braatz, and Simon Engelke. Machine learning benchmarks for the classification of equivalent circuit models from solid-state electrochemical impedance spectra. *Journal of The Electrochemical Society*, 170(6):060512, 2023.

- [28] Valentin Sulzer, Peyman Mohtat, Antti Aitio, Suhak Lee, Yen T. Yeh, Frank Steinbacher, Muhammad Umer Khan, Jang Woo Lee, Jason B. Siegel, Anna G. Stefanopoulou, and David A. Howey. The challenge and opportunity of battery lifetime prediction from field data. *Joule*, 5(8):1934–1955, 2021.
- [29] Simo Särkkä, Arno Solin, and Jouni Hartikainen. Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013.
- [30] Marco F. Huber. Recursive Gaussian process: On-line regression and learning. *Pattern Recognition Letters*, 45:85–91, 2014.
- [31] Christopher K. I. Williams and Carl Edward Rasmussen. *Gaussian Processes for Machine Learning*, volume 2. MIT Press, Cambridge, MA, USA, 2006.
- [32] D. Anseán, V. M. García, M. González, J. C. Viera, C. Blanco, and J. L. Antuña. Dc internal resistance during charge: Analysis and study on lifepo<sub>4</sub> batteries. In *World Electric Vehicle Symposium and Exhibition (EVS27)*, pages 1–11. IEEE, 2013.
- [33] Sebastian Ludwig, Ilya Zilberman, Max F. Horsche, Tim Wohlers, and Andreas Jossen. Pulse resistance based online temperature estimation for lithium-ion cells. *Journal of Power Sources*, 490:229523, 2021.
- [34] J. L. Hodges Jr. and E. L. Lehmann. Estimates of location based on rank tests. *The Annals of Mathematical Statistics*, 34(2):598 – 611, 1963.
- [35] David Beck, Philipp Dechent, Mark Junker, Dirk Uwe Sauer, and Matthieu Dubarry. Inhomogeneities and cell-to-cell variations in lithium-ion batteries, a review. *Energies*, 14(11), 2021.
- [36] Peng Bai and Martin Z. Bazant. Charge transfer kinetics at the solid–solid interface in porous electrodes. *Nature Communications*, 5(1):3585, 2014.
- [37] Martin Z. Bazant. Unified quantum theory of electrochemical kinetics by coupled ion–electron transfer. *Faraday Discussions*, 246:60–124, 2023.
- [38] Radu Gogoana, Matthew B. Pinson, Martin Z. Bazant, and Sanjay E. Sarma. Internal resistance matching for parallel-connected lithium-ion cells and impacts on battery pack cycle life. *Journal of Power Sources*, 252:8–13, 2014.
- [39] Kuan-Cheng Chiu, Chi-Hao Lin, Sheng-Fa Yeh, Yu-Han Lin, Chih-Sheng Huang, and Kuo-Ching Chen. Cycle life analysis of series connected lithium-ion batteries with temperature difference. *Journal of Power Sources*, 263:75–84, 2014.
- [40] David Duvenaud. *Automatic Model Construction with Gaussian Processes*. PhD thesis, University of Cambridge, UK, 2014.
- [41] Arno Solin. *Stochastic Differential Equation Methods for Spatio-Temporal Gaussian Process Regression*. PhD thesis, Aalto University, Helsinki, Finland, 2016. URL <https://aalto.doc.aalto.fi/handle/123456789/19842>. Aalto University publication series Doctoral Dissertations, 50/2016.
- [42] Simo Särkkä and Arno Solin. *Applied Stochastic Differential Equations*. Cambridge University Press, U.K., 2019.
- [43] Emil Julius Gumbel. *Statistics of Extremes*. Columbia University Press, New York, 1958.
- [44] Ronald Aylmer Fisher and Leonard Henry Caleb Tippett. Limiting forms of the frequency distribution of the largest or smallest member of a sample. *Mathematical Proceedings of the Cambridge Philosophical Society*, 24(2):180–190, 1928.
- [45] Jia-Liang Le, Zdeněk P. Bažant, and Martin Z. Bazant. Lifetime of high-k gate dielectrics and analogy with strength of quasibrittle structures. *Journal of Applied Physics*, 106(10), 2009.

- [46] Elias Barbers, Friedrich Emanuel Hust, Felix Emil Arthur Hildenbrand, Fabian Frie, Katharina Lilith Quade, Stephan Bihn, Dirk Uwe Sauer, and Philipp Dechent. Exploring the effects of cell-to-cell variability on battery aging through stochastic simulation techniques. *Journal of Energy Storage*, 84:110851, 2024.

## A Appendix

*Disclaimer:* The derivations and mathematical description below are copied from [1]. The other parts of this appendix are also based on [1].

### A.1 GPs

GPs are a flexible modeling framework that excels in the case of limited data by making a point estimate and modeling the covariance associated with the prediction. GPs are nonparametric probabilistic models fully defined by a mean function  $\mu(x)$  and a covariance function  $k(x, x')$  where  $x, x' \in \mathcal{X}^D$ . A GP is usually written as

$$f(x) \sim \text{GP}(\mu(x), k(x, x')), \quad (1)$$

making it explicit that GPs describe a distribution of functions. The GP posterior predictions are normally distributed, i.e., any marginal distribution of a GP is Gaussian. Furthermore, all joint distributions associated with a finite number of elements of the index set are multivariate normal distributions (e.g., [31] for further information). Assuming that we have  $n_o$  noisy observations  $(x_{o,i}, y_{o,i})$  with  $y_{o,i} = f(x_{o,i}) + \epsilon_i$ , where  $\epsilon_i$  is Gaussian noise with variance  $\sigma_n^2$ , the predictive GP equations are

$$\mu_{|o}(X_*) = K(X_*, X_o)[K(X_o, X_o) + \sigma_n^2 I]^{-1} y_o \quad (2)$$

$$\Sigma_{|o}(X_*) = K(X_*, X_*) - K(X_*, X_o)[K(X_o, X_o) + \sigma_n^2 I]^{-1} K(X_o, X_*), \quad (3)$$

where  $X_* = [x_{*,1} \cdots x_{*,n_*}]$  denotes the  $n_*$  test locations,  $X_o = [x_{o,1} \cdots x_{o,n_o}]$  denotes the training locations with responses  $y_o^T = [y_{o,1} \cdots y_{o,n_o}]$ , and  $K(X_1, X_2)$  denotes the covariance matrix that is constructed by applying the kernel function to all pairs of column vectors from  $X_1$  and  $X_2$  (for a full derivation, see [31]).

The training of a GP refers to the choice of kernel function and the optimization of associated hyperparameters, usually based on optimizing the marginal likelihood of the training data. Subsequently, the posterior distribution can be calculated for points of interest by inference, using (2, 3).

*SE Kernel:* The Squared Exponential (SE) kernel is a smooth, infinitely many times differentiable kernel which only depends on the distance of data points and is given for one-dimensional  $x$  by

$$k_{\text{SE}}(x, x') = \sigma_{\text{SE}}^2 \exp\left(\frac{-|x - x'|^2}{2l^2}\right). \quad (4)$$

We use three input dimensions – current, SOC, and temperature – and combine three one-dimensional RBF kernels to

$$k_{\text{SE,3ARD}}(x, x') = \prod_{d=1}^3 \sigma_{\text{SE,d}}^2 \exp\left(\frac{-|x_d - x'_d|^2}{2l_d^2}\right) = \sigma_{\text{SE,3}}^2 \exp\left(\sum_{d=1}^3 \frac{-|x_d - x'_d|^2}{2l_d^2}\right), \quad (5)$$

which is known as the SE-ARD kernel (ARD denotes “automatic relevance detection” because each length scale represents the importance of its associated direction; for more information, see [40]). There are four hyperparameters associated with the three-dimensional SE-ARD kernel: the output scale  $\sigma_{\text{SE,3}}^2$ , the length scale associated with the current dimension  $l_1 = l_I$ , the length scale associated with the SOC dimension  $l_2 = l_{\text{SOC}}$ , and the length scale associated with the temperature dimension  $l_3 = l_T$ .

*Wiener Velocity Kernel:* The Wiener Velocity (WV) model corresponds to the integrated Wiener process [41]. The WV covariance function or kernel is defined by

$$k_{\text{WV}}(t, t') = \sigma_{\text{WV}}^2 \left( \frac{\min^3\{t, t'\}}{3} + |t - t'| \frac{\min^2\{t, t'\}}{2} \right). \quad (6)$$

The WV kernel has one hyperparameter: the WV output scale,  $\sigma_{\text{WV}}^2$ .

The resulting kernel is

$$k(x, t, x', t') = k_{\text{SE,3ARD}}(x, x') + k_{\text{WV}}(t, t'). \quad (7)$$

Including the noise variance  $\sigma_n^2$ , there are six hyperparameters which define the characteristics of the GP used in this article.

**Spatiotemporal GPs** A large number of data points are recorded during the operation of a battery. Ideally, all samples that fulfill the data selection criteria are used, but a classical exact GP prevents this due to the unfavorable scaling of compute and memory usage. To address this issue, we follow [30, 41] and leverage the fact that we can interpret the GP as a spatiotemporal GP. For lead-acid batteries, a spatiotemporal GPs approach has been shown in [16].

A temporal GP, i.e., a GP that depends on only one variable, can be written as a Kalman filter [29, 41], which makes it scale linearly with the number of data points, with the restriction that the data points must be given in a sorted manner. Here, the variable is actually time; therefore, this does not impose further restrictions.

A spatiotemporal GP [41] is a GP that depends on multiple variables, of which one can be regarded as the time variable, the others being the spatial variables,

$$x = \begin{bmatrix} t \\ x_s \end{bmatrix}.$$

This concept leads to an infinite-dimensional Kalman filter that propagates the mean and variance functions over the time points. When using only a finite number of spatial vectors, this eventually leads to a classical, finite-dimensional Kalman filter [41].

We cannot restrict the actual measured data locations to a finite set. Therefore, we combine the spatiotemporal approach with the recursive approach from Huber [30]. In this approach, the spatial part of the GP is not represented exactly, but is represented by  $n_b$  predefined basis vectors  $x_{s,b,i}$  that are collected into

$$X_b = [x_{s,b,1} \quad x_{s,b,2} \quad \cdots \quad x_{s,b,n_b}].$$

Combining these two approaches leads to the following procedure, which is similar but not identical to the method used in [16]. The difference is mentioned below.

The state  $z$  of the Kalman filter,

$$z = \begin{bmatrix} z_t \\ z_s \end{bmatrix},$$

comprises two parts.  $z_t$  corresponds to the representation of the temporal kernel. In the case of the Wiener velocity kernel, it contains two scalar values; the first represents the mean value, and the second is the time derivative of the mean value [41].  $z_s$  corresponds to the spatial kernel, and for the Huber approach [30], these are the mean values at the specified basis vectors  $X_b$ .

**Evaluation** Given the state  $z_{k|k}$  and its covariance matrix  $P_{k|k}$  at the time  $t_k$  and built with all information available until  $t_k$ , the mean and covariance at this time point can be calculated by

$$\mu_{|k}(X_q, t_k) = H z_{k|k} \quad (8)$$

$$\Sigma_{|k}(X_q, t_k) = K_{qq} + H P_{k|k} H^T - H_s K_{bb} H_s^T. \quad (9)$$

The mean and the covariance of the GP at  $n_q$  spatial vectors  $x_{s,q,i}$  are collected into  $X_q$  similar to  $X_b$ . Furthermore, we use the abbreviations

$$K_{bb} \in \mathbb{R}^{n_b \times n_b}, \quad (K_{bb})_{i,j} = k_s(x_{s,b,i}, x_{s,b,j}),$$

$$K_{qq} \in \mathbb{R}^{n_q \times n_q}, \quad (K_{qq})_{i,j} = k_s(x_{s,q,i}, x_{s,q,j}),$$

for the covariance at the basis vectors and the covariance at the queried locations, respectively.

The measurement matrix

$$H = [H_t \quad H_s] \quad (10)$$

is partitioned according to  $z$ . The temporal part

$$H_t = \begin{bmatrix} 1 & 0 \\ \vdots & \vdots \\ 1 & 0 \end{bmatrix}, \quad (11)$$

reflects that for the Wiener velocity kernel with the chosen representation, the first associated state corresponds to the mean. The spatial part is

$$H_s = K_{qb}K_{bb}^{-1}, \quad (12)$$

with

$$K_{qb} \in \mathbb{R}^{n_q \times n_b}, \quad (K_{qb})_{i,j} = k_s(x_{s,q,i}, x_{s,b,j})$$

given by [30]. Here, our approach differs from the approach described in [16]. In (12), only the inverse of  $K_{bb}$  is needed, which is known a priori and can be calculated offline. In [16], the corresponding term includes an additional part that depends on the current state; thus, the inverse must be calculated in each step anew.

**Prediction step** The prediction step of the Kalman filter is performed identically to [16],

$$\begin{aligned} z_{k|k-1} &= A(T_s)z_{k-1|k-1} \\ P_{k|k-1} &= A(T_s)P_{k-1|k-1}A^T(T_s) + Q(T_s) \end{aligned}$$

with

$$P(0) = \begin{bmatrix} P_t(0) & 0 \\ 0 & P_s(0) \end{bmatrix}, \quad A(T_s) = \begin{bmatrix} A_t(T_s) & 0 \\ 0 & I \end{bmatrix}, \quad Q(T_s) = \begin{bmatrix} Q_t(T_s) & 0 \\ 0 & 0 \end{bmatrix},$$

where  $T_s = t_k - t_{k-1}$  is the length of the time step that may be different for each prediction step. The structure of these matrices shows the decoupling of the two parts, i.e., the spatial state is not influenced by the time step. However, due to the correction step, the covariance matrix will not retain its initial block diagonal structure.

For the Wiener velocity kernel (6), which is used as temporal kernel, the corresponding matrices are

$$P_t(0) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad A_t(T_s) = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, \quad Q_t(T_s) = \sigma_{\text{WV}}^2 \begin{bmatrix} T_s^3/3 & T_s^2/2 \\ T_s^2/2 & T_s \end{bmatrix}.$$

The initial value for the spatial covariance matrix is  $P_s(0) = K_{bb}$ . [41, 16]

**Correction step** When new measurements  $(X_{m,k}, y_{m,k})$  arrive, the Kalman filter performs a correction step

$$\begin{aligned} z_{k|k} &= z_{k|k-1} + K_k(y_{m,k} - H_k z_{k|k-1}) \\ P_{k|k} &= P_{k|k-1} - K_k H_k P_{k|k-1}. \end{aligned}$$

$X_{m,k}$  is a matrix that contains the locations of the  $n_{m,k}$  measurements,  $y_{m,k}$  is a vector giving the measured output for each location in  $X_{m,k}$ , and  $H_k$  is the measurement matrix build as in (10) with  $X_q = X_{m,k}$ . The Kalman gain  $K_k$  is calculated by

$$K_k = P_{k|k-1} H_k^T (\Sigma_{|k}(X_{m,k}) + \sigma_n^2 I)^{-1}, \quad (13)$$

where  $\Sigma_{|k}(X_{m,k})$  is given by (9) with  $X_q = X_{m,k}$ . This step differs from [16] by the different evaluation mentioned above and another measurement model.

*Remark 1:* If  $n_{m,k} = 0$ , i.e., no measurement data were acquired within the last time step, the correction step can be skipped, and the algorithm proceeds with the next prediction step. Alternatively, the algorithm could wait until new data is available and then perform a larger prediction step.

*Remark 2:* In a standard Kalman filter setup, the correction step can be performed iteratively for each group of measurements that is not correlated with measurements outside this group. Consequently, for the case here, where we model the measurement noise with the covariance matrix  $\sigma_n^2 I$ , i.e., all measurements are uncorrelated, each measurement could be processed individually, thus rendering the matrix inverse in (13) a scalar inversion. However, due to some simplifications in the approach of [30] that led to the filter equations, processing each measurement individually or multiple measurements together in a minibatch is not analytically equivalent for this application. Based on case studies, it seems to be advantageous to add measurements in larger groups.

**Rauch-Tung-Striebel smoother** Rauch-Tung-Striebel (RTS) smoother allows calculating the state and covariance and thus the estimates  $\mu_{|n}(X_q, t_k)$  and  $\Sigma_{|n}(X_q, t_k)$  at all time points  $t_k$  under consideration of the complete measured data of the  $n$  time points. This smoother starts with the variables  $z_{n|n}$  and  $P_{n|n}$  of the last Kalman filter correction step and iterates backward using [42]

$$z_{k|n} = z_{k|k} + G_k(z_{k+1|n} - z_{k+1|k}) \quad (14)$$

$$P_{k|n} = P_{k|k} + G_k(P_{k+1|n} - P_{k+1|k})G_k^T \quad (15)$$

for  $k = n - 1, n - 2, \dots$  with

$$G_k = P_{k|k}A(t_{k+1} - t_k)P_{k+1|k}^{-1}. \quad (16)$$

The output values  $\mu_{|n}(X_q, t_k)$  and their variances  $\Sigma_{|n}(X_q, t_k)$  for each time point  $t_k$  can be calculated by (8) and (9) by replacing  $z_{k|k}$  with  $z_{k|n}$  and  $P_{k|k}$  with  $P_{k|n}$ .

For evaluating equations (14), (15) and (16), the results  $z_{k|k}$  and  $P_{k|k}$  of the Kalman filter are used. Thus, these values must be stored while performing the forward filtering pass.  $z_{k+1|k}$  and  $P_{k+1|k}$  could either be stored also or calculated anew from  $z_{k|k}$  and  $P_{k|k}$  by reevaluating the prediction step.

**Implementation** The batteries are mostly sampled with 5 s, which, with the very small magnitude of sensible parameters for the output scale of the Wiener velocity kernel, leads to numerical issues in performing the prediction steps.

Therefore, we define a sampling time for the updates (1 hour) and process all data captured within this interval at the same update time point. If no data were recorded within this hour, we only perform a new prediction step without the update to set an evaluation point for the later backward smoothing pass.

**Selection of Basis Vectors** The selection of basis vectors can be done in different ways. Similarly to [16], we found that using k-means to select the basis vectors is effective. We used 60 basis vectors selected with k-means. However, in an online setting, there will be no data available at the beginning. Therefore, as an alternative, we propose to place vectors linearly within the range of the data selection criteria and set a constraint based on the length scale of the associated measurement (for more details, see SI of [1]).

## A.2 Data Selection

The data selection criteria (Tab. 1) are applied to each cell individually.

Table 1: Data Selection Criteria

Charging/Discharging	Discharge only
Temperature (°C)	$10 < x < 100$
Current (A)	$-80 < x < -5$
SOC (%)	$40 < x < 95$

## A.3 Hyperparameter Tuning

All battery systems contain the same prismatic cell type with the same specifications. Therefore, the aim is to find a single hyperparameter set that can be used for all cells. We optimized the marginal likelihood for all cells of systems 6 and 8 individually, using an exact GP and 20k data points for each cell. Systems 6 and 8 have the highest equivalent full cycle count and were operated over large temperature range. Subsequently, we chose the median of each hyperparameter, assuming that the parameters are mutually independent. We do not formulate an optimization problem for the spatiotemporal GP but use the hyperparameters found by the exact GP.

#### A.4 Fault Probability Derivation

The Hodge-Lehman estimator is defined as the median of the mean of all possible pairs of cell resistances, including self-pairs,

$$S_i = \{1, 2, \dots, n\} \setminus \{i\} \quad (17)$$

$$\hat{\theta}_{R_i}(t) = \text{med}_{j \leq k} \frac{R_j(t) + R_k(t)}{2}, \quad j, k \in S_i, \quad (18)$$

where  $n = 8$ , the number of cells. The probability of a cell  $i$  to be outside a resistance band that is  $2b$  wide is

$$p(F_{R_i}(t)) = p(R_i(t) > \hat{\theta}_{R_i}(t) + b) + p(R_i(t) < \hat{\theta}_{R_i}(t) - b). \quad (19)$$

The GP provides the mean and probabilities needed for calculating  $p(F_{R_i}(t))$ . We propose to define the probability of a pack resistance fault,  $p(F_{R_p})$ , as

$$p(F_{R_p}(t)) = 1 - \prod_i (1 - p(F_{R_i}(t)(i))) \quad (20)$$

according to the true “weakest link” failure statistics for cells connected in series [43–45]. In the limit of rare, independent failures dominated by a single cell, the pack resistance fault can be approximated by

$$p(F_{R_p}(t)) \approx \max_i p(F_{R_i}(t)). \quad (21)$$

Similarly, battery packs can be faulty or at the end of their life, if cells exceed an upper resistance threshold,

$$p(F_{R_{m_i}}(t)) = p(R_i(t) > c) \quad (22)$$

$$p(F_{R_m}(t)) = 1 - \prod_i (1 - p(F_{R_{m_i}}(t))). \quad (23)$$

However, in the following analysis, we only focus on the resistance distribution because the resistance at  $t = 0$  differs significantly between systems (Fig. 2).

#### A.5 Software and Hardware Details

The full GP, which was used only for hyperparameter tuning, is solved on an NVIDIA A100 GPU with 80 GB RAM. We use the Python GPytorch framework, allowing us to solve an exact GP with 40k data points in 11 seconds on average. The recursive spatiotemporal GP is solved on a MacBook Pro with a 10-core M1 chip, taking approximately 22 seconds for the forward Kalman filter and 6 seconds for the RTS smoother for a single cell of system 8 with 35k time steps, 320k data points, and 61 basis points.

## A.6 Open Circuit Voltage Approxiamtion

We use a linear pseudo OCV roughly corresponding to the flat region of LFP.

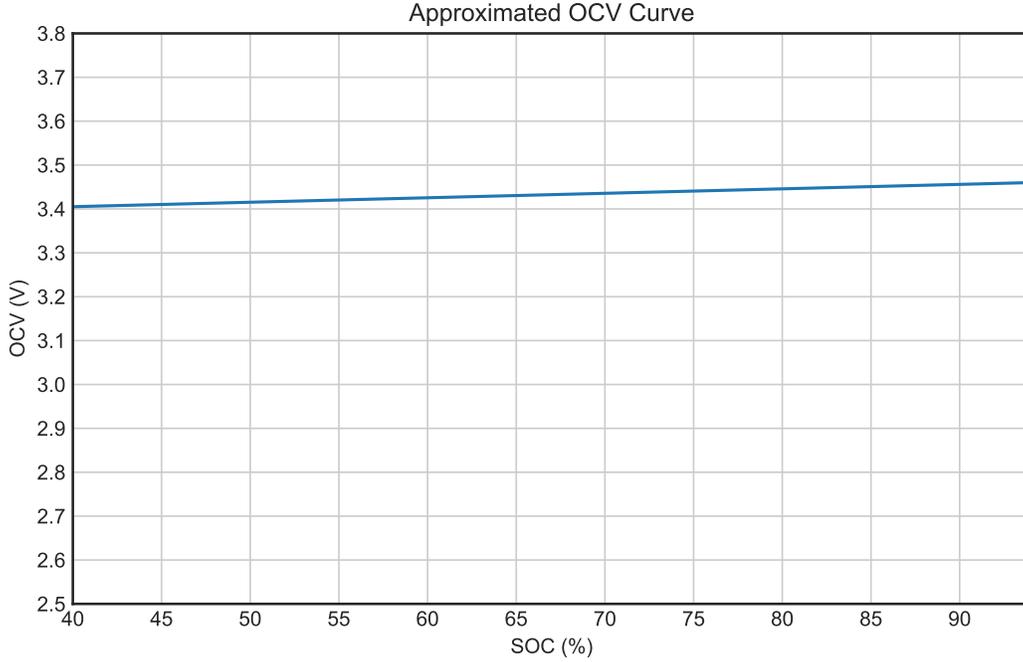


Figure 4: Approximation of the flat region of the LFP OCV. Reproduced from [1].

## A.7 Battery System Technical Details

Table 2: Technical specification of the battery system and summary of the associated data.

Technical Specification		Data Set	
Nominal voltage	24 V	Number of systems	28
Nominal capacity	$\approx 160$ Ah	Total number of cells	224
# Cells	8 (series)	Total rows of data	133 M
# Current Sensors	1	Median of measurement intervals	5 s
# Voltage Sensors	9		
# Temperature Sensors	4		
# Cell Balancing Current Sensors	8		

## A.8 Data Visualization

Figure 5 visualizes 24.2 million rows of data associated with battery system 8, which was operated for approximately five years. The BMS was switched on in November 2016, but frequent usage only started about a year later. The end of continuous usage is around December 2021. However, the system kept logging data for a couple more months afterward. The temperature profile shows seasonal variations with higher temperatures during the northern hemisphere summer months. Furthermore, voltage measurements and estimated SOC show that the system was primarily operated between 60 to 100% SOC with occasional discharges below 40%. Around September 2020, the usage pattern changes, as can be seen by the current and SOC patterns. The mean subtracted cell voltages,

$$\tilde{u}_i(t) = u_i(t) - \frac{1}{n-1} \sum_{j, j \neq i}^n u_j(t), \quad (24)$$

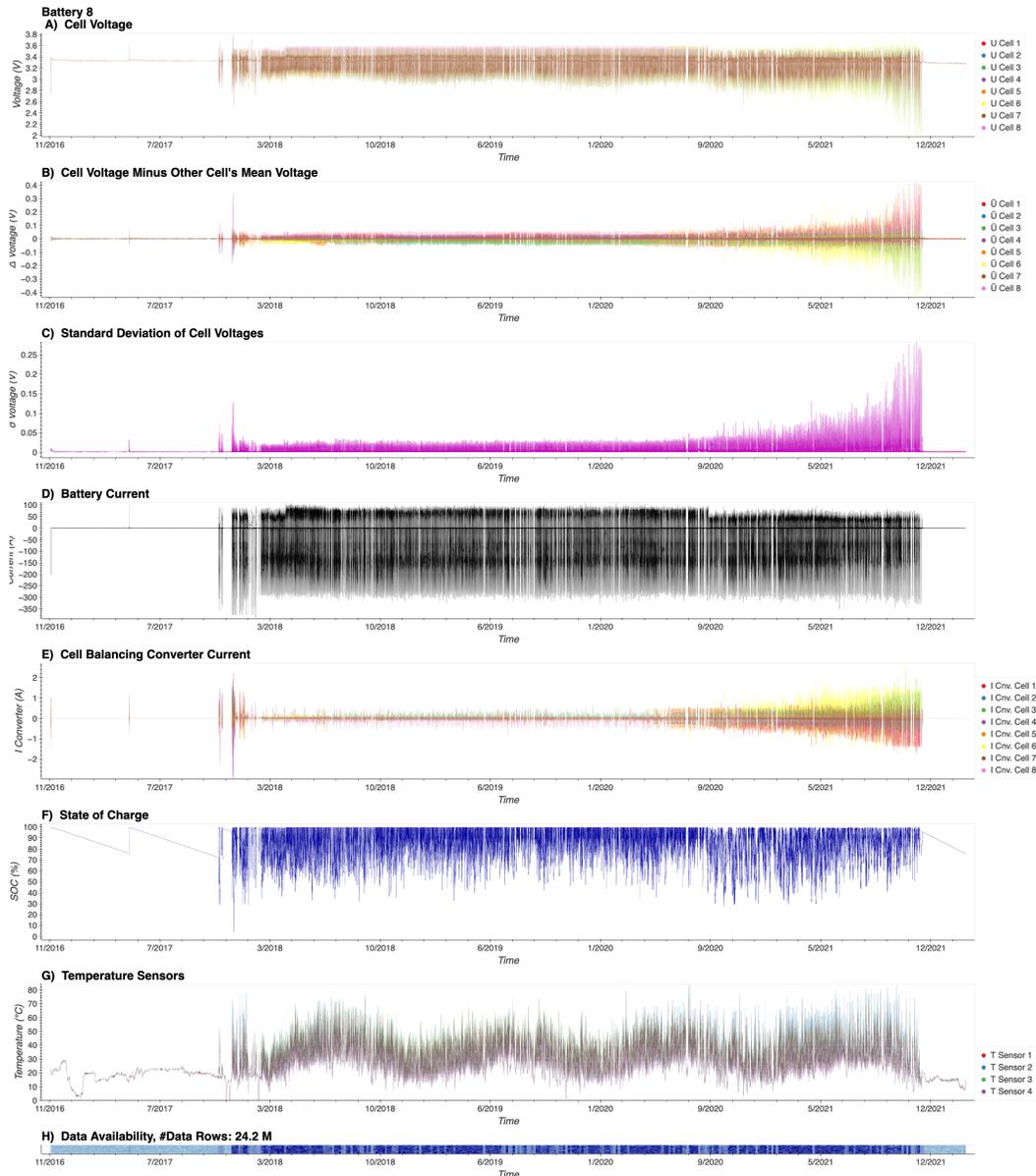


Figure 5: Data visualization of battery system 8. a) Cell voltages, b) Cell voltages with mean of other cells subtracted, c) Cell voltage standard deviation, d) Battery system current, e) Cell balancing converter current, f) State of charge, g) Temperatures (each temperature sensor neighbors two cells, h) Data availability. Reproduced from [1].

where  $u_i(t)$  is the voltage of cell  $i$  at time  $t$ , and  $n$  is the number of cells, show average deviations below 0.1 V for the first two years of operation (Fig. 5b). With increasing usage, i.e., increasing charge throughput and time and therefore also degradation, the average mean subtracted voltages increase, an indicator that individual cells age differently, likely due to cell-to-cell variations (e.g., [35, 46]), and as a consequence, the system is less balanced. To summarize, the data of system 8 shows heavy usage over five years, totaling about 1531 equivalent full cycles. However, based on the data visualization, it is challenging to understand further how cells degraded, whether certain cells degraded more than others, or when the system might fail.