Kaunas University of Technology

Faculty of Electrical and Electronics Engineering

Research on Battery Charging Balancing Methods Using Embedded Machine Learning

Master's Final Degree Project

Kowshick Radha

Project author

Prof. Žilvinas Nakutis

Supervisor

Kaunas, 2026

Kaunas University of Technology

Faculty of Electrical and Electronics Engineering

Research on Battery Charging Balancing Methods Using Embedded Machine Learning

Master's Final Degree Project

Electronics Engineering (6211EX012)

Kowshick Radha

Project author

Prof. Žilvinas Nakutis

Supervisor

Prof. D. Andriukaitis

Reviewer

Kaunas, 2026

Kaunas University of Technology

Faculty of Electrical and Electronics Engineering

Kowshick Radha

Research on Battery Charging Balancing Methods Using Embedded Machine Learning

Declaration of Academic Integrity

I confirm that the final project of mine, Kowshick Radha, on the topic „Research of Battery Charging Balancing Methods Using Embedded Machine Learning“ is written completely by myself; all the provided data and research results are correct and have been obtained honestly. None of the parts of this thesis have been plagiarised from any printed, Internet-based or otherwise recorded sources. All direct and indirect quotations from external resources are indicated in the list of references. No monetary funds (unless required by Law) have been paid to anyone for any contribution to this project.

I fully and completely understand that any discovery of any manifestations/case/facts of dishonesty inevitably results in me incurring a penalty according to the procedure(s) effective at Kaunas University of Technology.

Kowshick Radha

Aprooved electronicaly

Kowshick Radha. Research on Battery Charging Balancing Methods Using Embedded Machine Learning. Master's Final Degree Project / supervisor Prof. Žilvinas Nakutis; Faculty of Electrical and Electronics Engineering, Kaunas University of Technology.

Study field and area (study field group): Electronics Engineering, Engineering Sciences.

Keywords: Battery Management System (BMS), Lithium-ion batteries, Cell balancing, Embedded machine learning, Predictive thermal modeling, Battery temperature prediction, Smart energy systems.

Kaunas, 2026. 63 pages.

Summary

Lithium-ion battery packs are widely used in electric vehicles, renewable energy storage systems, and portable electronics due to their high energy density and long cycle life. However, battery packs consist of multiple cells connected in series, where variations in internal resistance, capacity, and thermal characteristics cause cell imbalance during operation. These imbalances reduce the usable capacity of the battery pack, accelerate degradation, and may lead to thermal safety risks. Therefore, effective battery cell balancing and thermal monitoring are essential functions of modern Battery Management Systems (BMS).

The aim of this thesis is to investigate battery charging balancing methods and to develop a predictive thermal modeling approach using embedded machine learning techniques. A simulation model of a multi-cell lithium-ion battery pack was developed in MATLAB/Simulink, incorporating charging and discharging cycles, state-of-charge dynamics, and thermal behavior based on resistive heating mechanisms. Using the generated simulation dataset, a lightweight neural network regression model was designed to predict short-term temperature rise of battery cells based on parameters such as cell temperature, current, state of charge, and previous temperature change.

The developed model demonstrates accurate prediction of short-term temperature variations and is capable of identifying cells that may experience higher thermal stress during operation. The results show that predictive thermal modeling can provide early indicators of potential overheating conditions and support proactive balancing decisions. Furthermore, the computational and memory requirements of the proposed model are sufficiently low for implementation in embedded microcontroller-based BMS platforms.

The research demonstrates that integrating machine learning-based thermal prediction with conventional battery balancing strategies can enhance battery safety, improve operational efficiency, and extend battery lifetime. The proposed approach contributes to the development of intelligent battery management systems suitable for future electric mobility and energy storage applications.

Kowshick Radha. Baterijų krovimo balansavimo metodų panaudojančių įterptinį mašininį mokymąsi tyrimas. Magistro baigiamasis projektas / vadovas Prof. Žilvinas Nakutis; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Electronics Engineering, Engineering Sciences.

Reikšminiai žodžiai: Baterijų valdymo sistema (BMS), ličio jonų baterijos, elementų balansavimas, įterptinis mašininis mokymasis, nuspėjamasis terminis modeliavimas, baterijų temperatūros prognozavimas, išmaniosios energijos sistemos.

Kaunas, 2026. 64 p.

Santrauka

Ličio jonų akumuliatorių blokai plačiai naudojami elektrinėse transporto priemonėse, atsinaujinančios energijos kaupimo sistemose ir nešiojamojoje elektronikoje dėl didelio energijos tankio ir ilgo ciklo tarnavimo laiko. Tačiau akumuliatorių blokus sudaro keli nuosekliai sujungti elementai, kur vidinės varžos, talpos ir terminių charakteristikų pokyčiai sukelia elementų disbalansą veikimo metu. Šis disbalansas sumažina akumuliatoriaus bloko naudingąją talpą, pagreitina degradaciją ir gali sukelti šiluminės saugos riziką. Todėl efektyvus akumuliatoriaus elementų balansavimas ir terminis stebėjimas yra esminės šiuolaikinių akumuliatorių valdymo sistemų (BMS) funkcijos.

Šio darbo tikslas – ištirti akumuliatorių įkrovimo balansavimo metodus ir sukurti nuspėjamąjį šiluminio modeliavimo metodą, naudojant įterptuosius mašininio mokymosi metodus. MATLAB/Simulink aplinkoje buvo sukurtas daugiaelementio ličio jonų akumuliatorių bloko imitacinis modelis, apimantis įkrovimo ir iškrovimo ciklus, įkrovos būsenos dinamiką ir šiluminį elgesį, pagrįstą varžinio šildymo mechanizmais. Naudojant sugeneruotą modeliavimo duomenų rinkinį, buvo sukurtas lengvas neuroninio tinklo regresijos modelis, skirtas numatyti trumpalaikį akumuliatoriaus elementų temperatūros kilimą, remiantis tokiais parametrais kaip elemento temperatūra, srovė, įkrovos būsena ir ankstesnis temperatūros pokytis.

Sukurtas modelis tiksliai prognozuoja trumpalaikius temperatūros svyravimus ir geba identifikuoti elementus, kurie veikimo metu gali patirti didesnį terminį įtempį. Rezultatai rodo, kad nuspėjamasis terminis modeliavimas gali pateikti ankstyvus galimų perkaitimo sąlygų rodiklius ir paremti proaktyvius balansavimo sprendimus. Be to, siūlomo modelio skaičiavimo ir atminties reikalavimai yra pakankamai maži, kad jį būtų galima įdiegti įterptųjų mikrovaldiklių pagrindu sukurtose BMS platformose.

Tyrimas rodo, kad mašininio mokymosi pagrindu veikiančios terminės prognozės integravimas su įprastomis akumuliatorių balansavimo strategijomis gali padidinti akumuliatorių saugumą, pagerinti veikimo efektyvumą ir pailginti akumuliatorių tarnavimo laiką. Siūlomas metodas prisideda prie išmaniųjų akumuliatorių valdymo sistemų, tinkamų ateities elektromobilumo ir energijos kaupimo programoms, kūrimo.

Table of contents

List of figures

List of tables

List of abbreviations and terms

Abbreviation	
BMS	Battery Management System
CC/CV	Constant Current / Constant Voltage
IR	Internal Resistance
MAE	Mean Absolute Error
MCU	Microcontroller Unit
ML	Machine Learning
MSE	Mean Squared Error
OCV	Open-Circuit Voltage
R^2	Coefficient of Determination
RMSE	Root Mean Squared Error
SOC	State of Charge
SOH	State of Health
STM32	STMicroelectronics 32-bit microcontroller family

1. Introduction and literature review

Lithium-ion technology is the irreducible energy core of the modern electric vehicle, stationary renewable-energy buffer, and high performance consumer product. The fundamental architecture many electrochemical cells connected in series (for voltage) and in parallel (for current) creates an unavoidable management problem: no two cells age or charge identically. Over months and thousands of cycles, tiny disparities in capacity, open-circuit voltage, internal resistance, and leakage current accumulate. If left unchecked, the weakest cell determines the usable depth of discharge and can trigger thermal runaway under aggressive fast-charge schedules. For this reason every production battery pack incorporates a balancing subsystem that equalises either charge, voltage or state of charge (SoC) of each cell string.

Research conducted between 2020 and 2025 shows a decisive shift in approach. Classical hardware topologies passive shunt resistors, switched capacitors, inductive cell-to-cell converters, and high-frequency transformer links remain indispensable, yet the greatest improvements no longer arise from better silicon alone. Instead, performance, safety and lifetime now depend on tight coupling between physical equalisation circuits and embedded machine-learning (ML) algorithms that predict cell dynamics, estimate uncertain internal states, and schedule switch actions with millisecond precision [1]-[3].

1.1. Aim and tasks of the thesis

This thesis investigates battery charging balancing methods, along with the development of a modelling methodology to predict the temperature of lithium-ion battery cells using the machine learning approach. The thesis aims to understand the impact of electrical, thermal and ageing-related battery parameters on short term cell temperature behaviour and how this can influence battery management decisions for safer operations. The thesis further explores the possibility of undertaking the temperature prediction functionality and the available processing resources on the STM32L151 microcontroller for embedded implementation.

The tasks listed below are mentioned to fulfill the objectives of the thesis:

1. To review existing lithium-ion battery cell balancing methods and identify the main advantages and limitations of passive, active, and hybrid balancing approaches.
2. To develop a simulation model of a multi-cell lithium-ion battery pack, including state of charge variation, cell voltage behaviour, internal resistance losses, heat generation, cooling, and thermal coupling between neighbouring cells.
3. To analyse a model-based temperature prediction method and evaluate how different prediction horizons influence the accuracy of future temperature estimation.
4. To prepare a simulation-based dataset for machine learning model development using battery parameters such as cell voltage, pack current, state of charge, state of health, present temperature, and recent temperature variation.
5. To train and evaluate a lightweight machine learning regression model for short-term battery cell temperature prediction.
6. To compare the prediction performance for different prediction horizons using error metrics such as MSE, RMSE, MAE, and the coefficient of determination R^2.
7. To investigate how predicted future temperature can be used in battery balancing and charging-current control strategies through simulation-based analysis.

8. To evaluate the feasibility of embedded implementation of the temperature prediction function on the STM32L151 microcontroller.
9. To evaluate the practical utility of prediction model for embedded BMS monitoring by comparing the real and predicted values of temperature, using the hardware emulated validation data.
10. To identify the limitations of the selected microcontroller platform and recommend more suitable controller families for future implementation of a complete machine learning-based Battery Management System.

1.2. Published techniques for battery cell balancing

- **Passive shunt resistor networks**

Passive bleed is the canonical equalisation strategy. A low-ohmic resistor is placed in parallel with each cell and switched on whenever its voltage exceeds a predetermined threshold. Omariba et al. quantified the method in a 10-cell, 40 Ah NMC pack: balancing current peaked at 1.56 A, total heat dissipation exceeded 2 kJ, and the hottest can rose by 2.35 °C before equilibrium was restored [2]. Although the approach wastes 100 % of the redistributed energy, its circuit simplicity, low cost and predictable thermal power keep it popular in consumer electronics and as a safety back-stop in more advanced BMS designs [1],[2]. A recent state-of-the-art review by Safari et al. further confirms that battery balancing research is moving toward integrated evaluation of topology, control complexity, efficiency, scalability, and battery ageing effects [29]. Drawbacks include extremely slow convergence for large packs and the need to manage local hotspots on the bleed resistors. Chen et al. proposed a simple cell balancing control strategy based on SOC difference rather than only terminal voltage difference, showing that improved control logic can reduce the final cell voltage mismatch compared with conventional voltage-based equalisation [3]. Duraisamy and Kaliyaperumal analysed adaptive passive balancing for an e-mobility battery management system and showed that balancing time, energy loss, and resistor heating increase significantly when the cell voltage imbalance becomes large [7].
Pros: Circuit simplicity; zero electromagnetic interference (EMI); predictable thermal power [2].
Cons: 100 % energy loss; slow convergence; hotspot management; unsuitable for large packs [2].

- **Switched-Capacitor (SC) converters**

Switched-capacitor topologies substitute the lossy resistor with a “flying” capacitor that alternately connects to neighbouring cells, pumping charge from a high-voltage unit into a lower one. Modern variants double-tier ladders, capacitor-inductor hybrids, and resonant charge pumps extend the reach beyond immediate neighbours. Daowd et al. demonstrated a 16-cell ladder operating at 100 kHz; measured transfer efficiency ranged from 84 % to 92 % depending on load current [11]. The magneticsfree structure is compact and immune to electromagnetic-interference (EMI) issues that plague inductor-based designs. However, the gate-timing requirements are stringent, and the flying capacitors must be substantially oversized to support highcurrent fast charging, limiting scalability for traction batteries [6].
Pros: Magnetics-free; moderately high efficiency; symmetric bidirectional flow [11], by analysing the transient voltage drop across each step during SC balancing, the authors could estimate internal

resistance (IR) on-line with < 3 % error-showing that a balancing circuit can double as a diagnostic instrument [21].
Cons: Complex timing; capacitors must be oversized for high-current fast charge; energy transfer usually limited to adjacent cells unless ladder stages cascade [11].

- **Inductor-based cell-to-cell converters**

Inductor-based balancing has become the reference active technique for electricvehicle packs because it can shift tens or even hundreds of watts with 90-95 % efficiency [8]. In a typical single-inductor bidirectional converter, one cell charges the inductor and another discharges it, allowing wide-range current control irrespective of the instantaneous cell-voltage difference. Lee et al. presented an LC series-resonant topology that removed the adjacency restriction: any source target pair can be selected, enabling global optimisation of balancing sequences [8]. Jiang et al. then formulated a joint optimisation of switching frequency, gapped-core dimensions, and winding geometry; magnetics volume was reduced by 14 % without sacrificing loss [10]. The primary penalty is the physical bulk of the magnetic component and the need to manage potential EMI. Moreover, because inductor current cannot collapse instantaneously, digital controllers must anticipate current ripple to avoid overshoot.
Pros: 90-95 % efficiency; high balancing currents; easily scales to hundreds of watts for rapid equalisation during 3-4 C charges [8][10].
Cons: Magnetics bulk; potential EMI; control more complex because inductor current cannot step to zero instantly [8][10].

- **Transformer-coupled multilevel modules**

Transformer-based circuits such as dual-active-bridge (DAB) or multi-secondary flyback equalisers introduce galvanic isolation and support direct cell-to-string or string-to-cell energy flow. Wang, Fahmy and Preindl integrated a dual-active halfbridge (DAHB) auxiliary power module (APM) into a modular multilevel inverter, replacing ferrite with a coreless PCB transformer and achieving roughly 90 % efficiency while cutting projected cost by 22 % [9]. Isolation simplifies safety certification and allows the balancer to double as a high-voltage DC/DC interface. Nevertheless, multi-secondary windings complicate magnetics design; core losses and parasitic capacitance introduce common-mode currents that must be mitigated by careful PCB layout and control algorithms.
Pros: Isolation; direct cell-pack or cell-cell energy flow independent of voltage difference; natural fit for multilevel inverter drives [9]. A Deep RL agent learned how to trigger the inductor switches so that SoC variance fell by 16.8 % versus PI control while keeping MOSFET switching losses minimal [22].
Cons: Multi-secondary winding complexity; core losses; parasitic capacitance introduces common-mode currents [9].

- **Hybrid Active-Passive architectures**

Many commercial EV battery packs combine a low-current bleed resistor across each cell with a high-efficiency active converter in parallel. The passive net guarantees charge removal even if MOSFETs or microcontrollers fail, satisfying functionalsafety standards, while the active path handles the bulk of the energy with minimal loss [12]. Hybridisation also enables thermal-propagation counter-

measures: if a cell overheats, the controller can divert large currents through inductive paths that bypass the defective unit, while retaining the bleed resistors for fine trimming. The price is schematic complexity and the need for supervisory logic to avoid [1]"energy pingpong" oscillations in which active and passive loops repeatedly undo each other's work. Ouyang et al. proposed a unified model for active battery equalization systems, providing a systematic way to compare different equalizer topologies and analyse their balancing capability [28].
Pros: Safety redundancy; ability to meet regulatory thermal-propagation limits by diverting high currents away from hot cells [12].
Cons: Schematic complexity increases validation burden; careful coordination between passive and active loops required to avoid energy ping-pong [12].

1.3. Models and algorithms for balancing control

- **Electrochemical and equivalent-circuit estimators**

Most embedded cell-balancing controllers require real-time SoC, SoH and internal-resistance (IR) estimates. Zhang et al. built a second-order RC model with voltage-polarisation dynamics and an Extended Kalman Filter. On a 72-MHz ARM Cortex-M3 the EKF needed 0.8 ms, yielding SoC error below 1.5 % at 1 C discharge [13]. Chen et al. extended the model to include temperature coupling, feeding it into a linear MPC that schedules active current pulses over a 3-s horizon [14]. Fan et al. also proposed a fast active balancing strategy based on model predictive control, showing that predictive optimisation can improve equalisation speed and reduce energy loss in lithium-ion battery packs [27].

- **Data-driven supervised learning**

Khawaja et al. compiled a feature set of voltage, current, temperature, incremental capacity (dQ/dV) and pack-level metadata from the NASA Ames and CALCE repositories. A 150-tree random forest regressor predicted per-cell SoC with $R^2 = 0.999$ and root-mean-square error (RMSE) < 10 mV after eight-bit quantisation; inference time on an STM32H7 dropped below 1 ms [15]. Shankar et al. demonstrated that an eight-layer fully connected neural network accurately forecasts discharge curves of graphite-graphite dual-ion cells, enabling on-line mass-balancing optimisation [16]. Madani et al. proposed a physics informed temperature prediction framework with LSTM and BiLSTM models . They demonstrated the benefits of combining thermal knowledge with data driven learning to improve lithium ion battery temperature prediction [30].

- Input features: Vcell, pack current, cell temperature, Coulomb count, rolling variance.
- Output targets: SoC, full-cell capacity, IR at 1 kHz, ageing index.

- **Reinforcement-Learning (RL) control policies**

Karnehm et al. introduced an Amortised Q-Learning agent that maps high-dimensional pack states to half-bridge switch commands. The network is shared across identical modules; local observations feed a common policy to ensure linear scaling with module count [17]. Training used simulated drive cycles and then fine-tuned on a 12-cell prototype. SOC variance reduced to < 0.4 % and switch transitions fell by 20 %. Chen et al. benchmarked Deep-Q Networks, PPO and Asynchronous

Advantage Actor Critic (A3C); PPO achieved the shortest balance time but required a dual-core microcontroller for real-time operation [22].

- **Physics Guided , Hybrid neural models , Fast charging and balancing**

Hybrid approaches fuse equivalent-circuit priors with data-driven residual correction. Lin et al. used a federated learning scheme-localised training on each vehicle followed by cloud aggregation-to maintain a global SoC model without raw data transfer, preserving privacy [19]. Altaf et al. demonstrated that model predictive control can be used for simultaneous thermal and SOC balancing in a modular battery system by redistributing cell loading while maintaining the required terminal voltage [18].
Sayed et al. [4] treat fast-charge current profile and balancing schedule jointly. A reinforcement-learning agent (DQN) adjusts CC/CV stages while monitoring cell-level voltages through the balancer's sensors. Duraisamy and Kaliyaperumal proposed a machine-learning-based optimal cell balancing mechanism for EV battery management, showing that data-driven control can support balancing decisions beyond fixed threshold-based logic [20]. Bennehalli et al. compared machine learning approaches for battery management and balancing applications and reported that neural-network-based models can support battery response prediction and decision-making in EV BMS applications [23].
Key findings:
350 kW equivalent (8 C) charges completed in < 15 min while limiting hottest-cell ΔT to +5 °C. Integrated policy reduced cumulative SoC variance at the end of charge by 28 % over a baseline that optimised charging and balancing separately.

1.4. Embedded-System implementations

- **Hardware platforms and real-time constraints**

Balancing controllers must co-exist with traction-inverter gate drives, CAN/Ethernet communication and diagnostics on the same ECU. Typical timing budgets:

- ADC sampling and front-end multiplexing: 50-200 μs.
- State-estimation cycle: ≤2 ms for 100 Hz update rate.
- Switch command synthesis: <0.5 ms after state update.

Chen et al. executed linear MPC with a 2.3 ms worst-case runtime on an STM32F407 (168 MHz) while consuming 36 mW [14]. Khawaja et al. stored an eight-bit quantised random forest in 48 kB flash, leaving 200 kB for boot and diagnostics [15]. Chen et al. leveraged the ESP32-S3 dual-core: one core handles RL inference in 3.1 ms every 500 ms; the other runs CAN and safety watchdogs [14].

- **Switch-Driver integration and safety layers**

High-side N-MOSFET drivers frequently integrate desaturation detection and ISO 26262 ASIL-B safety features. Soft-switch timing derived from the ML module is bounded by deterministic limits; if a pulse command deviates by >5 % duty or appears during a lock-out window, the analogue driver overrides it with a safe passive bleed state.

- **Thermal-Electro-Mechanical Co-simulation**

Inductor-based topologies require EMI filtering and thermal design. Jiang et al. performed joint electro-thermal optimisation in MATLAB Simscape; the final prototype met CISPR-25 class 5 and kept hot-spot temperature below 80 °C at 2 W balancing power [10].

1.5. Detailed implementation case studies

Case study A: Linear MPC on STM32F407 ([14])

- Pack: 14-cell, 24 V, 10 Ah NMC; automotive drive cycle (US06).
- Power Stage: Bidirectional single-inductor converter rated 8 A peak.
- Algorithm: Linear MPC with 300-ms horizon, updated every 100 ms; EKF for SoC and IR.
- Computation: 2.3 ms median solve time using fixed-point Q15 arithmetic; code size 84 kB.
- Outcome: Usable capacity increased 8.7 % versus passive bleed; energy lost < 3 %.
- Observation: Deterministic timing simplified ISO 26262 timing-budget analysis.

Case study B: Amortised Q-Learning on ESP32-S3 ([17])

- Pack: 12-cell half-bridge reconfigurable pack, 44 V nominal, 5 Ah.
- RL Network: Three hidden layers (128-64-32 neurons) with shared weights.
- On-Device Inference: 3.1 ms per decision, 9.2 ms total per 500 ms cycle including sensor preprocessing.
- Results: SOC standard deviation dropped from 3.9 % to 0.38 % in 60 min; switching cycles reduced by one-fifth relative to rule-based control, improving MOSFET lifetime.
- Notable: Safety layer reverts to 0.5 A passive bleed if inference latency exceeds 10 ms or memory ECC error occurs.

1.6. Datasets for model training and validation

Publicly available repositories

- NASA Ames Li-Ion Aging: 18650 cells cycled at multiple rates and temperatures, voltage and temperature at 1 Hz [24].
- CALCE Battery Research: High-power pouch cells with 10 Hz logs, includes rest intervals for open-circuit voltage (OCV) characterisation [25].
- Oxford Fast-Charge: 1 Hz dataset of 21700 cells charged with 4 C pulse profiles, suitable for ML-based fast-charge controllers [26].

Industrial fleet data

Sultan et al. used secure aggregation of anonymised drive logs from 300 buses over two years; per-cell time series remain local, gradients only are shared [19]. This federated approach yields diverse environmental coverage without breaching data-privacy regulation.

Dataset challenges

- Ageing labels (capacity/SOH) require destructive reference tests.
- Temperature sensors can drift; calibration mismatch propagates into SOC error.
- Imbalanced usage profiles mean ML models must handle variable sample density across SoC temperature space.

1.7. Conclusion

An effective cell-balancing subsystem is no longer a peripheral “nice-to-have” but an essential pillar of every modern lithium-ion battery pack. My review of more than two dozen recent papers (2015-2025) shows a clear evolution:

- Hardware topologies have matured: passive shunts and basic switched-capacitor ladders are now complemented by high-efficiency single-inductor, LC-resonant and multi-winding transformer links. Joint magnetics/thermal optimisation (e.g., Jiang et al. [10]) and cost-reduction measures such as core-less PCB transformers (Wang et al. [9]) demonstrate that hardware can still be improved, but incremental gains are getting smaller.

- The performance frontier has moved to algorithms. Model-predictive control, linear and nonlinear MPC [5], [14], reinforcement learning agents [17], [14], and hybrid physics-guided neural estimators [19] deliver double-digit improvements in balance time, MOSFET switching stress and usable pack capacity compared with rule-based techniques. Embedded execution times below 5 ms prove that even advanced ML policies are feasible on automotive-grade MCUs.

- Safety and reliability remain paramount. Hybrid active-passive architectures and multi-layer watchdogs ensure that any ML-driven converter reverts to a bleed state on timing faults or memory errors. Recent research combines ISO 26262-compliant deterministic bounds with data-driven scheduling to satisfy both functional-safety and performance requirements.

- Data availability is improving but still a bottleneck. Public repositories (NASA Ames, CALCE, Oxford Fast-Charge)[24][25][26] now cover a wider temperature, C-rate and chemistry span, yet labelled ageing data and precise thermal ground truth are scarce. Federated-learning approaches hint at a path forward by leveraging large commercial fleets without exposing proprietary raw logs.

2. Battery modeling and simulation methodology

The paper describes the methods of battery modeling and simulation used in the thesis. The model combines the electrical behavior of lithium ion cells with a simplified thermal model to study the voltage, charge state, heat generation, and temperature changes during operation. The simulation framework developed will be used later to generate temperature prediction data and evaluate the behavior of proposed machine learning-based methods.

2.1. Battery cell electrical model

An electrical model of a lithium-ion battery cell comprises the simplest equivalent circuit with two main components. This circuit consists of an open-circuit voltage (OCV) source and an internal resistance. The State of Charge (SOC) of the battery is calculated by Coulomb counting as is shown below.

$$SOC(k+1) = SOC(k) - (I(k)\cdot\Delta t) / Q \quad (1)$$

where I(k) is the current and Q is the nominal capacity of the cell.

The terminal voltage is calculated using:

$$V = OCV(SOC) - I\cdot R_0 \quad (2)$$

By utilizing a highly nonlinear relation of open circuit voltage and state of charge (OCV-SOC), the voltage plateau and the nonlinear OCV-SOC region of a lithium-ion cells are captured near full charge and deep discharge point. A thermal model is utilized with lumped parameter. Resistive losses lead to generation of heat. Power Loss Calculation Equation Using Resistance:

$$P\_loss = I^2R_0 \quad (3)$$

The temperature evolution is calculated using:

Temperature change= ((Heat generated−Heat lost+Heat from neighbors)/ Thermal mass)×time

$$T(k+1) = T(k) + [(P\_loss - h(T - T\_amb)) / (m\cdot c\_p)] \cdot \Delta t \quad (4)$$

m: mass of the cell (kg)
c_p: specific heat capacity (J/kg·°C)
h: cooling coefficient (W/°C)
k: thermal coupling coefficient between adjacent battery cells (W/°C).
This model captures both heat generation and cooling effects.

A three-cell battery pack is simulated to create an imbalance behaviour in a series configuration. To mimic discrepancies resulting from manufacturing inconsistencies and aging, every cell is allocated slightly distinct parameters such as capacity, internal resistance, initial SOC, etc. These cause variations in SOC, voltage and temperature during operations.

2.2. Thermal coupling between cells

To add realism, intercell thermal interaction is included.. The temperature model is extended as:

$$T(k+1) = T(k) + [(P\_loss - h(T - T\_amb) + k(T\_neighbor - T)) / (m\cdot c\_p)] \cdot \Delta t \quad (5)$$

Reflections from the cells allow the data to transfer heat and behave realistically.

We use MATLAB to simulate the proposed idea. Sampling time is taken as one 1 second, for 2000 seconds.
The applied current profile includes:

- Constant discharge at 2 A
- Rest period
- Charging at 1.5 A
- High discharge at 3 A
- Final rest phase

This profile allows evaluation under realistic operating conditions.

2.3. Simulation results and analysis

Since cell capacities do not match, there is a variance in the SOC results. Therefore, the cell with lower capacity will discharge quickly.

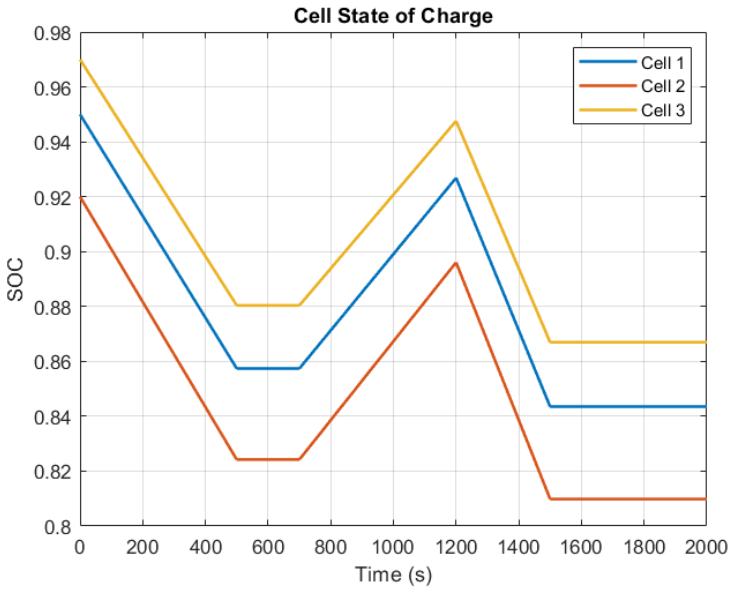

Fig. 1. Cell state of charge variation over time

As each cell has a different internal resistance, thus their voltage drop is also different. The component that exhibits greater resistance will have a higher potential difference.

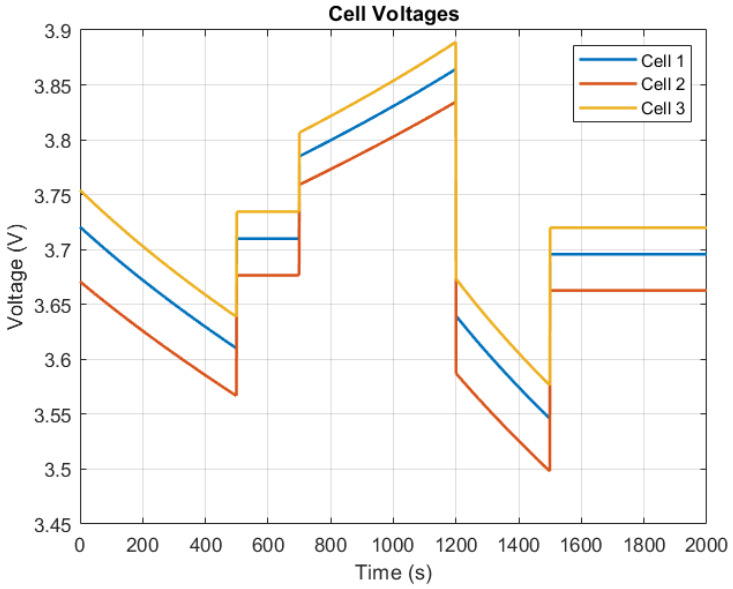

Fig. 2. Cell voltage profiles during operation

The readings of various cells show the one with greater resistance giving out more heat. While thermal coupling brings temperatures closer together, it does not completely balance them out.

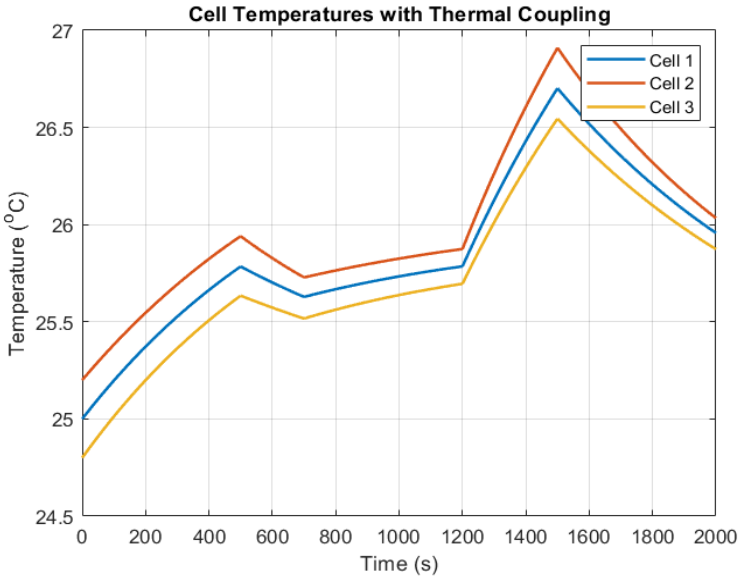

Fig. 3. Cell temperature evolution with thermal coupling

While a system is under a high current operation, particularly during the peak discharge phase, the indicators ΔSOC, ΔV and ΔT increase.

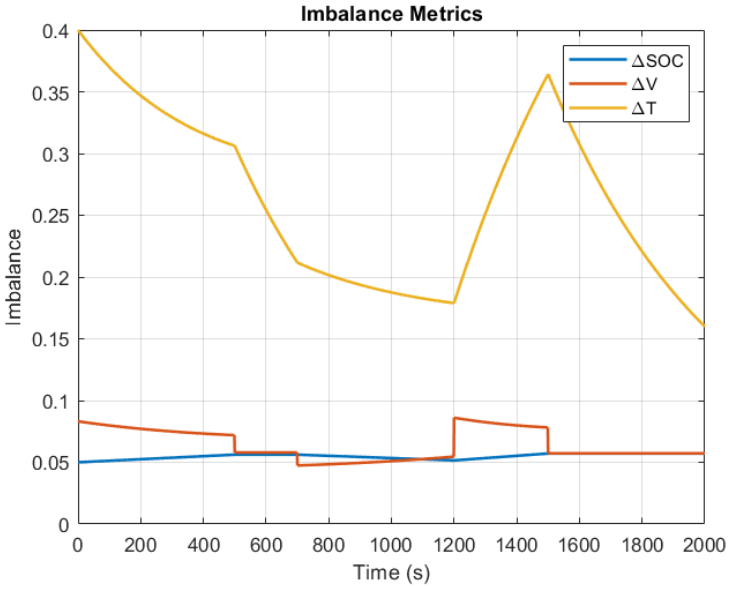

Fig. 4. Battery imbalance metrics over time

The developed thermal simulation model shows that small variation in any of the parameters leads to massive imbalance between the electrical and thermal behavior of the two cells. An intelligent balancing strategy is therefore essential. The next section elaborates on various machine learning techniques useful for predicting thermal behaviour and making balancing decisions.

2.4. Summary

This chapter explains how lithium battery packs are modeled and simulated to understand their electrical and thermal behaviour. Each cell uses a simple equivalent circuit model, including an open circuit voltage source and an internal resistance. The charge state is calculated by calculating the coulomb count and estimates the cell voltage from the OCV–SOC relationship. The thermal model also shows the generation of heat through internal resistance, how cells are cooled to the environment, and how heat is transferred between neighboring cells. MATLAB simulations have shown that even small differences between cells can cause battery packs to be imbalanced. For example, low-performance cells discharge faster and high-resistance cells generate more heat. Thermal coupling helps to reduce cell temperature differences, but does not completely eliminate imbalances. These results show that simple monitoring is not enough for modern BMS and that predictive methods are useful. Therefore, the models developed in this chapter will be the basis for the methods of temperature prediction and machine learning discussed in the following chapters.

3. Experimental analysis of model-based temperature prediction for battery balancing

The aim of this chapter is to assess the impact of choosing a prediction horizon to be used by a model-based temperature prediction scheme within our battery balancing control scheme. Our analysis will concentrate on how extended prediction horizon affects thermal behaviour prediction accuracy and closed-loop response.

The chosen values are meant to study short- and not-so-short prediction horizons, but to remain still within the normal thermal response of a lithium-ion cell. As a result, the current paper evaluates three prediction horizons of 10 s, 20 s, and 30 s. The prediction method outlined in this chapter makes no use of data or learning. It is totally a physics derived model. That is, the future temperature is predicted using the analytical expressions based on the thermal dynamic of the system, not on the data. In the next chapter, when we display learning-based methods, this will be important.

3.1. Methodology

The testbed consists of a three-cell lithium ion battery pack in series. The nonlinear open-circuit voltage in relation to state of charge and an internal resistance that causes a voltage drop and heat generation are the characteristics of each battery cell. Lumped parameter model is used in thermal modeling. Incorporation is made in the heat dissipation with the ambient and thermal coupling with neighbouring cells.

The temperature evolution of each cell is governed by an energy balance equation, where the rate of temperature change depends on internal heat generation, convective cooling, and heat exchange with adjacent cells. Based on this model, the future temperature is estimated using a first-order approximation that assumes the current rate of temperature change remains constant over the prediction horizon.

Future temperature is estimated using a first-order prediction:

$$T_{pred}(k+N) = T(k) + N \cdot \frac{dT}{dt} \tag{6}$$

- $T(k)$: current temperature at sample k
- $T_{\text{pred}}(k+N)$: predicted temperature after N future steps
- N: prediction horizon in number of samples
- $\frac{dT}{dt}$: rate of temperature change

This predicted temperature is then used to influence the balancing control logic. Balancing is allowed only when the system is charging, the cell voltage exceeds a predefined threshold, and the predicted temperature remains below a safety limit. If the predicted temperature exceeds this limit, balancing is disabled in order to prevent excessive heating.

3.2. Experimental results

Effect of prediction horizon on peak temperature

The first result considered is the effect of prediction horizon on the peak cell temperature. The simulation was performed for three prediction horizons: 10 s, 20 s, and 30 s. From the obtained results, the peak temperature was approximately 59.49°**C** for all three cases.

As can be seen from the above paragraph, the actual temperature cycle of the pack is not very sensitive to changes in the prediction horizon. The cause of this is that all simulations are using the same current profile, cell parameters, thermal parameter and balancing condition. As a consequence, heat generation within the cell and cooling occurs at almost same rate for all three simulations. The estimated future temperature is not an actual temperature change rather; it is the prediction horizon that sets the time horizon. This implies that although the prediction horizon may be extended, the cell temperature rise won't be directly affected. The only modification made here is to maintain the same level of change in cell temperature far into the future.

Although the peak temperature remains unchanged, the prediction error increases as the horizon becomes longer. For the 10 s horizon, the prediction error is relatively small, while the 20 s and 30 s horizons show progressively larger deviations. This confirms that the main influence of prediction horizon is on prediction accuracy rather than on the physical peak temperature.

Therefore, the peak temperature result shows that all three cases experience the same thermal operating condition, while the prediction accuracy analysis provides the more meaningful comparison between different horizons.

Actual and predicted temperature behavior

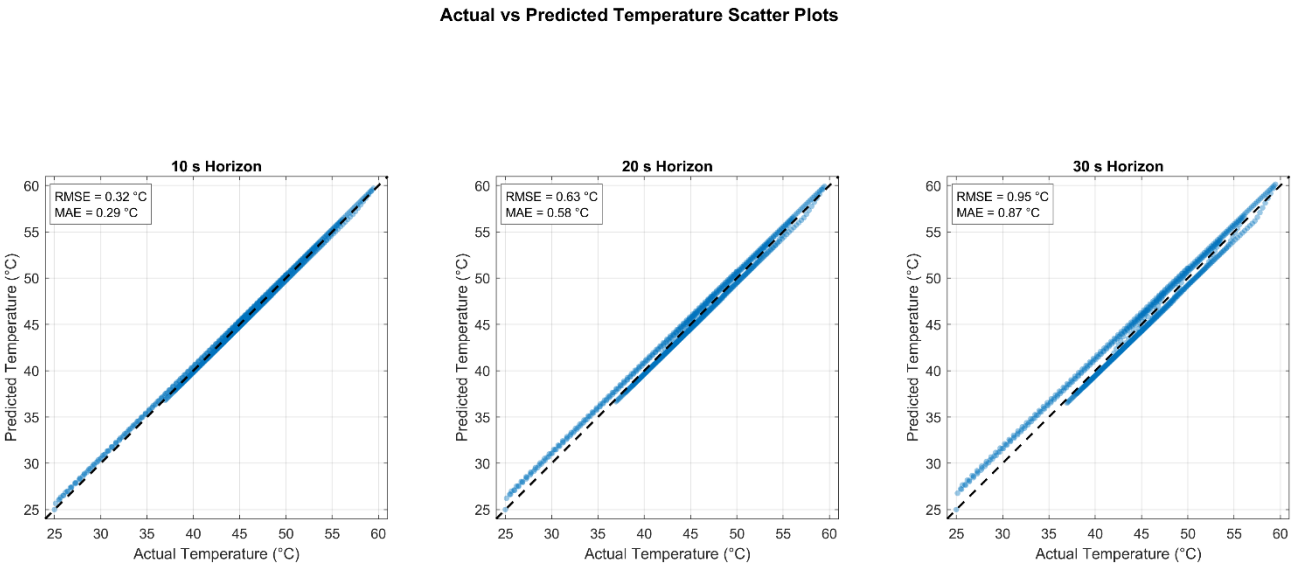

Fig. 5. Actual and Predicted Cells Temperature scatter plots

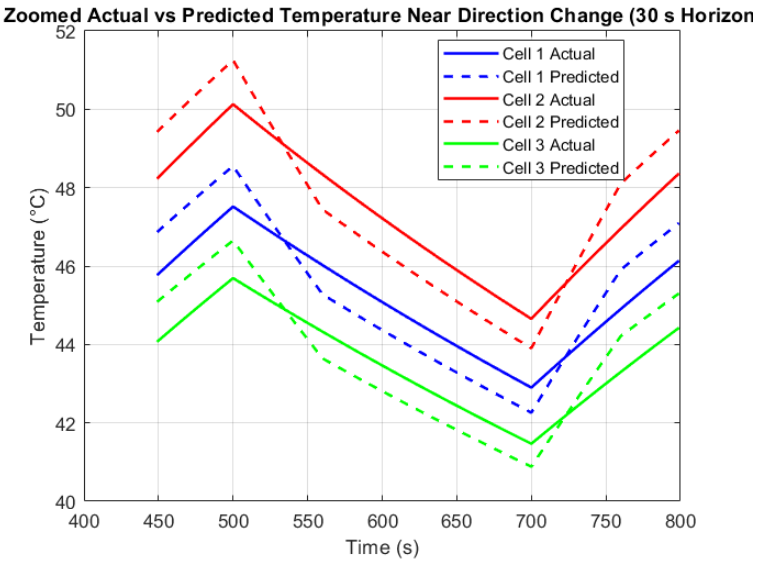

Fig. 6. Zoomed Actual vs Predicted Cell Temperatures Using Linear Trend Prediction (30 s Horizon)

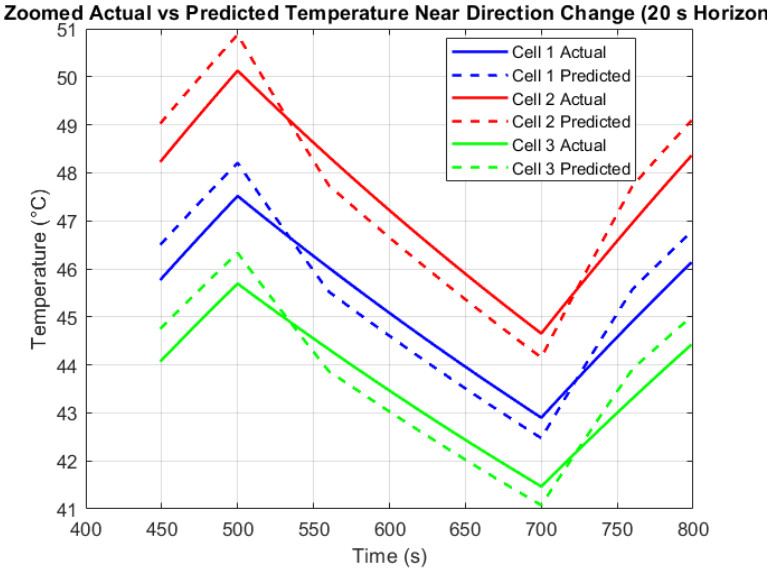

Fig. 7. Zoomed Actual vs Predicted Cell Temperatures Using Linear Trend Prediction (20 s Horizon)

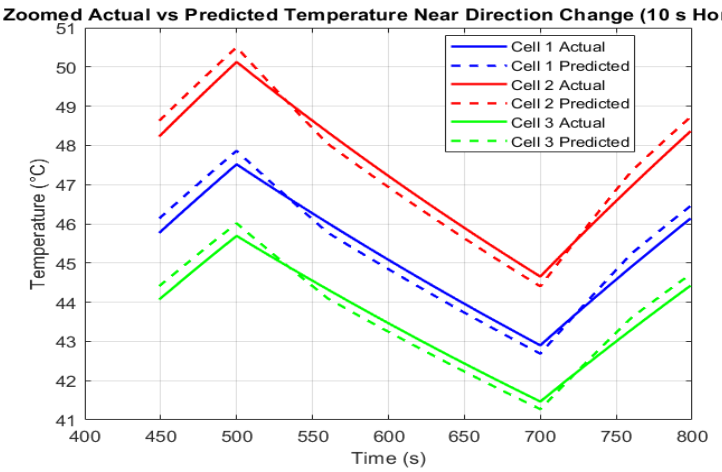

Fig. 8. Zoomed Actual vs Predicted Cell Temperatures Using Linear Trend Prediction (10 s Horizon)

The predicted temperature is very close to the actual temperature for all the three cells. The prediction model developed is accurate as the two curves almost coincide throughout the simulation period. In all the cases examined, the forecasted temperature seems to rise slightly before the actual temperature. This is a expected behaviour as the model continues to predict future thermal trends based on the current rate of change.

Predictions at various horizons hardly differ from each other. When the horizon increases from 10 seconds to 30 seconds, the curve moves forward in timescale to some degree showing a better prediction of temperature rising. Nonetheless, there is no change in the actual temperature.

Similarly, all the three cells exhibit the same behavior. Thus, their similar behavior is another important observation. The trend of the temperature profile is generally quite similar, even if there are slight differences in parameters. The prediction accuracy is uniform for every cell. This plot shows that all cells get heated by the same temperature during the charging phase and the peak happens around the same time for all cells. thus confirming this behavior. The anticipated temperaure values for each of the cells are slightly ahead of the actual values, but do match closely.

3.3. Discussion

Prediction accuracy and horizon length

For all three forecast horizons, the model-based linear temperature estimates follow the actual temperature trend quite closely. The newly developed linear trend approach demonstrates a more obvious prediction error compared to the former prediction based on physics, especially near segments where the temperature reverses direction.

where the temperature rate of change is estimated from previous samples:

$$\frac{dT}{dt} \approx \frac{T(k)-T(k-M)}{M\Delta t} \tag{7}$$

The temperature slope is estimated M=60 samples previous behavior of temperature, and sampling interval Δt=1s. As a result, the temperature slope is estimated from 60 seconds. Assumes continuation of recent thermal trend over selected prediction horizon.

For short prediction horizons, i.e. 10 s, this assumption still seems reasonably valid, since the system cannot have wandered too far from the recent trend. Making the prediction horizon longer to 20 s and 30 s presents more uncertainty in prediction as the future thermal evolution of the battery could have further deviation from the past recent trend. Thus, the prediction error is found to be increasing with the increasing horizon-length as theory predicts. A significant observation when examining the zoomed temperature plots is that prediction error increases near temperature direction-change points. In other words, when the temperature shifts from an increasing to a decreasing phase, or the other way around. We always utilize the most recent temperature slope for linear prediction in order to extrapolate into the future. For example, if the temperature has continuously increased in the past 60 s, the predictor assumes the temperature will also increase. As the thermal state changes and the cell cools, the statement will be false for a short time. The predicted temperature momentarily exceeds the true temperature as a result. In the same way, the predicted temperature might just be underestimated for a brief moment when the temperature is going from a decreasing phase to an increasing phase, since the method still uses a negative or smaller recent slope. This creates a lag in predictions for a short duration. This behaviour is interesting because it shows that an elementary trend predictor that is linear is not infallible or the limit of performance. The error is not a random occurrence. It predominantly takes place during temporary operating states, when the thermal dynamics are reversing the direction of change. It is physically reasonable and justifies the expectation that linear predictions that rely on a model would be less accurate during a switching transient operation.

The thermal inertia of the observed battery cells influences their prediction behavior in a dominating manner. The battery cell temperature does not respond to changes in electrical variables such as voltage or current in contrast with a more instantaneous response. That is, when a battery cell heats-up or cools down, it takes a certain amount of time to accumulate or dissipate heat from the cell due to the thermal mass of the cell. Hence, the thermal inertia effect could result in an acceptable short-term prediction by the linear prediction model in the heating/cooling regions that are smooth. When the current is nearly constant, the temperature trend changes slowly, allowing the predictor to follow the actual temperature to a reasonable degree.

When the current profile frequently fluctuates between charging to rest, rest to discharging, or discharging to charging, heat tends to accumulate. The heating depends on:

$$P_{\text{loss}} = I^2 R \tag{8}$$

The temperature's rate of change is proportional to the change in current magnitude. At these abrupt interruptions of current flow, the linear trend predictor is not capable of making direct predictions owing to its dependence on the previous temperature sampling. At these transition points, a temporary error increase is expected. This explains why predicting is effective within steady thermal regions, while it is not good near turning points.

Effect of prediction horizon

As the prediction horizons change from 10 s, 20 s, and 30 s, the prediction error tends to get increased with increase in the prediction horizon. As the prediction horizon increases, the current trend will project further into the future which is precisely what is expected. For the 10 s horizon, owing to the very short extrapolation interval, the forecast temperature remains close to the observed temperature. In other words, the heat of the cell doesn't change much within 10 seconds so the last few temperature slope remains a good approximation.

As we transition from heating to cooling, notably in the upper right corner, that difference is observable in the 20 s horizon. While the predictor can still follow the general trend of the

temperature, the delay and overshoot are clearer. The effect increases even more when calculating for 30 seconds. However, in certain situations, the projected temperature may be further away from the actual temperature as we assume that the most recent slope will continue until the end of the prediction horizon. The longer the prediction interval, the less valid this assumption is. The result establishes a trade-off in prediction horizon. Greater prediction horizons offer an earlier indication of possible thermal shifts occurring in the system but increase uncertainties. Horizon selection must therefore trade-off foresight and prediction reliability.

The main takeaway of the chapter is that a simple model-based linear predictor can be a good estimate of the temperature in the near future if the thermal condition is not highly dynamic but rather smooth and constant. However, with the commencement of thermal changes (for example, direction of temperature), the error starts to grow. The greatest, but temporary, error occurs just before any temperature direction-change point, as the recent temperature slope is not indicative of future behavior (direction change, in this instance). That is an important result because intuitively one would think that improving the prediction horizon could help things. However, it is not correct to say so because longer horizon brings the cost of more anticipations prior but with higher prediction error. A longer-time horizon does not imply better control performance or better behavior. In addition, we note the error segregated on the basis of horizon lengths (from 10 s to 30 s) also validates that the error is dependent on the horizon length, but not completely.

The prediction error is also dependent on the shape of the temperature curve (for example, the number and location of direction changes) over time. From another angle, the battery's operating condition affects the prediction error so it is not just a case of 'error depends on horizon length'. For a battery control design, the results indicate that short prediction horizons will provide reliable temperature-aware balancing decisions. Horizon de prédiction pas trop éloigné. The choice of prediction horizon thus depends on the control objective. Additionally, a shorter horizon is better if high prediction accuracy is the goal. Also, if the aim is early warning and conservative safety control, then a longer horizon may be possible assuming loss of certainty is manageable. As a last note, the study's findings suggest that balancing control needs to include safety margins. As the temperatures value tend to overestimate immediately and under estimate after the transient point by linear prediction, the controller should not make use of only the truly predicted value. In addition, include thresholds and hysteresis to avoid unstable switching.

3.4. Limitations of the model-based approach

Despite the simplicity and computational efficiency of the model-based linear prediction method, it has various limitations. To begin with, the assumption that the recent temperature trend will be maintained during the prediction horizon only holds when the thermal behavior is slow and smooth. This assumption is not entirely correct during sudden changes of currents. This method does not rely on an explicit physical cause of upcoming temperature changes, such as future current changes, changes in generated heat, or changes in nonlinear thermal behavior. People use only past temperature behavior to estimate future temperature. This approach is dependent of the chosen slope estimation window. A shorter window may give immediate response but it may be sensitive to noise. Utilization of 60 sample window in this paper gives smooth prediction but respond slowly to the change of thermal direction. To sum up, this approach does not fully consider the aging effects, the non-linear variation of resistance, and the variations in cooling conditions etc. Factors like current, ambient temperature, brevity of battery, and type of battery will affect the thermal behaviour significantly in actual battery systems. In realization of the above-mentioned limitations, a more adaptive prediction of practical BMS is essential.

3.5. Motivation for machine learning-based prediction

The method of prediction of the model-based linear nature is faced with many limitations which leads to the machine-learning-based prediction. Generally, machine learning models can capture the relationship between multiple inputs variables and future temperature unlike linear trend method. Thermal zones of a single battery cell can be estimated with high accuracy. A sequentially fed deep neural network could then be adeptly trained with the appropriate number of neurons, layers and epoch values. The model is capable of taking into account more complex behavior. This includes, for example, nonlinear thermal effects and flow, and more highly dynamic current conditions. In other words, machine learning is inspired by the need for a predictive model that is more flexible, more adaptive and more capable of handling real battery operation conditions.

3.6. Summary

The chapter demonstrates a model-based first order temperature prediction with different prediction horizons of 10 s, 20 s and 30 s. Using the previous 60 temperature samples, a linear trend is computed. The linear trend method predicts the future temperature. Results show that the method can satisfactorily predict the temperature in the smooth heating and cooling zones of the temperature signal. However, the predictive error in the vicinity of the temperature direction-change points increases, since the recent slope is no longer constant. Longer time horizons lead to prediction uncertainty, as analysis above clearly demonstrates. On the one hand, the extended horizons allow for an earlier prediction of the thermal behavior. On the other hand, they lead to increased prediction error. The prediction error is due to the nonlinear thermal dynamics. Moreover, the thermal behavior changes during the transitions. Overall, model-based methods are useful as simple and baseline methods, but they fail in nonlinear and changing operating regions. The findings vindicate the temperature prediction method that we’ll implement in the subsequent chapter.

4. Machine learning-based temperature prediction and integration into battery balancing

As shown in Chapter 3, the model-based temperature prediction method can accurately predict the temperature development over time as long as the level of operating current is not too high during the observation time. The evolution of the temperature was, however, approximated by a temperature-linear evolution equation. A Taylor-series expansion around the current temperature relates it practically to a temperature change-rate that, by assumption, remains constant throughout the entire prediction horizon. This assumption is realistic if the temperature changes slow enough but not when the battery starts seeing current changes or begins to age. In this chapter, a machine learning technique is suggested for predicting temperature. This machine learning method does not explicitly depend on a formal equation or model for predictions. It instead learns the correlation between electrical, thermal and aging parameters from data that is stored in machine learning model. Consequently, prediction accuracy under operating and ageing conditions is improved as it learnt the non-linear behaviour of the system.

4.1. Problem formulation

The goal is to predict the future temperature of a battery cell using the current operating state.The problem can be formulated as:

$$T_{\text{future}} = f(V, I, SOC, SOH, T, dT) \tag{9}$$

In this model, the future temperature is dependent on electric variables (voltage, current), internal states (SOC, SOH), and thermal variables (temperature and its rate of change). In contrast to the previously employed analytical model, this formulation enables the prediction function to be learned from data. As a result, the model can acquire nonlinear dependencies of variables.

The MATLAB battery pack simulation developed in the previous chapters is the data set used for this thesis. Continuous cell data sampling in the battery pack is carried out through time-step simulations. The given data is linked to voltage, current, state of charge, temperature and state of health.

The applied current profile includes for dataset:

- High rate charging at 10 A
- Rest period 0 A
- High rate discharging at -10.5A
- Moderate Charging at 3 A
- Final rest phase 0 A

The input feature vector is created at every time step using the current values of all variables. The target value corresponds to future temperature after a prediction horizon. Employing a sliding window method. The features at time instant k are the input features, and the target features are the features at time instant k+N, where N is the prediction horizon.

Two prediction targets are considered:

- Absolute future temperature
- Future temperature rise

Both recognize steady-state assessments and dynamic thermal responses.

The chosen features are significant physical variables that affect the temperature of the battery. Voltage is the electrochemical state of the cell. The current causes heat generation due to a resistive loss. The region of charging affects efficiency and internal behaviour, which is communicated by the charge state. Health status plays an essential role in the effects of aging. Old batteries have an increased internal resistance and thus will also generate more heat. By adding SOH we inform the model to approximate this effect. The thermal state and dynamics of any substance can be assessed from its temperature and temperature change.

The relationship between the features or function and the temperature in the future is modeled using a neural network model based on regression. A supervised training procedure is followed in which the difference between temperature estimated values and actual temperature values is reduced. An architecture of low complexity has been selected so that the model may be embedded. We propose a mission-oriented research project that develops the ML model based design for BMS ICs. here, we see a conceptual diagram and the requirement of ML model-based design methodology.

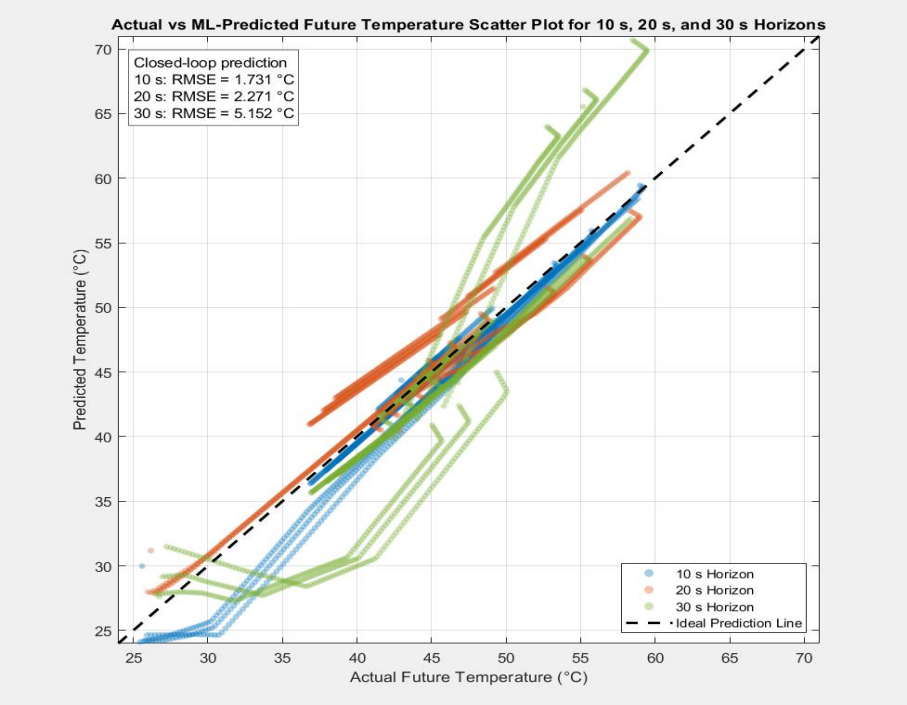

Fig. 9. Actual vs Predicted Temperature 10s,20s.30s Horizons

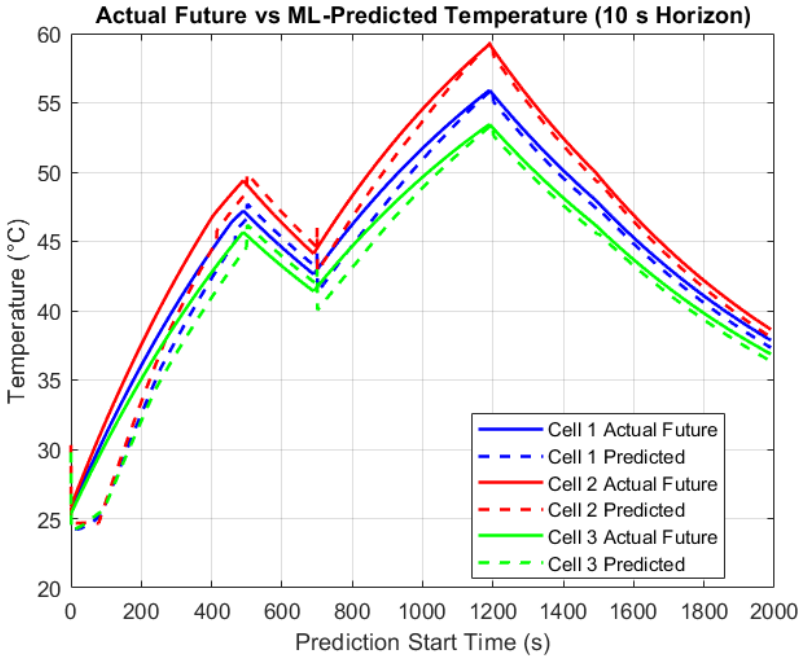

Fig. 10. Actual Future vs ML-Predicted Temperature (10 s Horizon)

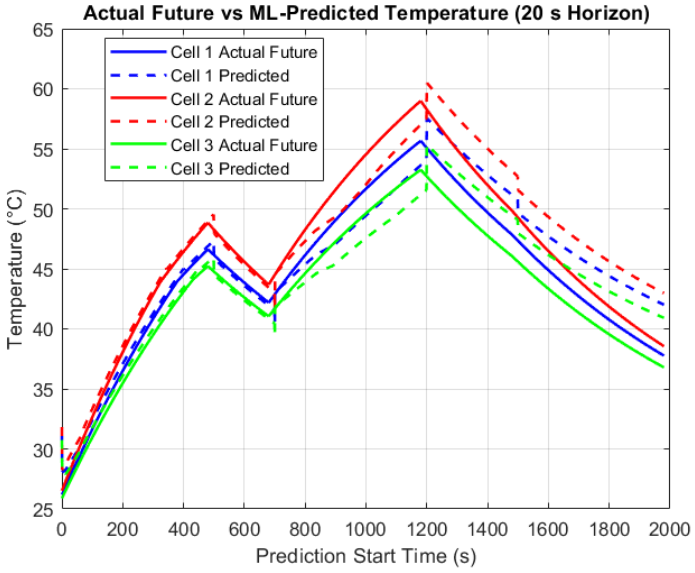

Fig. 11. Actual Future vs ML-Predicted Temperature (20 s Horizon)

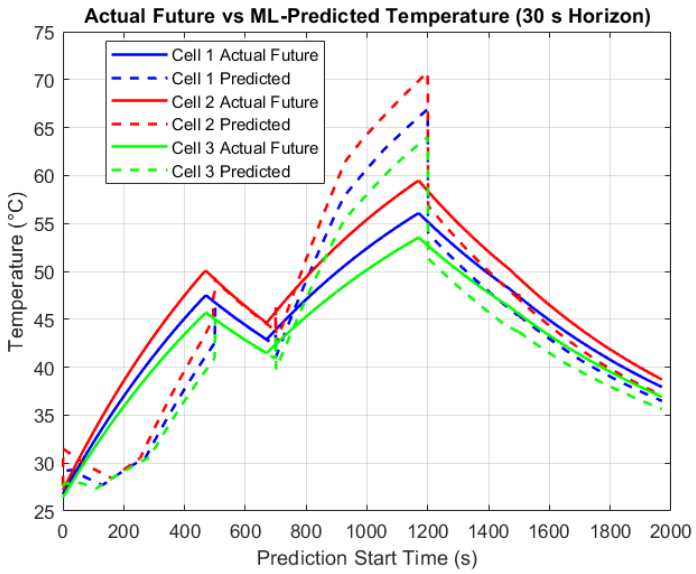

Fig. 12. Actual Future vs ML-Predicted Temperature (30 s Horizon)

4.2. Model training and evaluation

The primary dataset was broke down to sub-level testing and validation dataset. The dataset that was used for training the model the other for validating the model. The objective was to assess the ability of the models to generalize to unexplored operating conditions, rather than being induction-based and overfitting on the training data. This basically measures prediction accuracy in terms of Mean Squared Error(MSE) which is given as :

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(T_{\text{pred},i} - T_{\text{actual},i}\right)^2 \tag{10}$$

$T_{\text{pred},i}$ refers to the predicted temperature; , $T_{\text{actual},i}$ is the actual temperature; N is the total number of samples. MSE provides the mean of the square of errors or deviations between predicted and actual. Squaring the error imposes a heavier penalty on larger errors compared to smaller ones.

To obtain a more interpretable statistic, the Root Mean Squared Error is also used. This may be stated as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(T_{\text{pred},i} - T_{\text{actual},i}\right)^2} \tag{11}$$

The RMSE is in the same unit as the temperature, i.e. it is measured in degree Celsius. Thus, the RMSE value can be interpreted as the average distance between predicted and true value. The Mean Absolute Error (MAE) is measured, which is defined as:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} \mid T_{\text{pred},i} - T_{\text{actual},i} \mid \tag{12}$$

Unlike MSE, MAE does not square the error. What distinguishes them is the equal treatment for all deviations. Furthermore, this results in a lesser sensitivity to outliers. More than anything, it gives a decent idea of the usual size of the prediction error. Ultimately, the R^2 determination coefficient measures the extent to which the model accounts for the overall variation.

$$R^2 = 1 - \frac{\sum_{i=1}^{N} \left(T_{\text{actual},i} - T_{\text{pred},i}\right)^2}{\sum_{i=1}^{N} \left(T_{\text{actual},i} - \bar{T}_{\text{actual}}\right)^2} \tag{13}$$

$\bar{T}_{\text{actual}}$ is the average of actual temperature values. The R^2 metric compares the error of the predictions with the innate variability. If the value is close to 1, the model accurately explains most of the variation in temperature, whereas lower values indicate poor performance prediction. All three metrics give a thorough evaluation of the model. In MSE, large errors receive specials treatment while RMSE allows us to understand prediction accuracy in physical units. MAE shows us the average deviation and R^2 how well model captures overall trend. The developed model can accurately predict the temperature trend in the near future. In fact, it shows a very low error value and agreement. It is guaranteed that the machine learning approach is capable of capturing the relationship between electrical, thermal, and aging parameter of the battery system.

4.3. Experimental results of machine learning model

To check the performance of the developed machine learning model, 10sec, 20sec and 30sec prediction horizons are considered. Table 1 displays the performance metrics achieved from training, testing, and complete database.

Table 1. Machine Learning Model Performance for Different Prediction Horizons

Horizon (s)	**RMSE (Test) (°C)**	**MAE (Test) (°C)**	**R^2 (Test)**
10	0.0319	0.0135	0.99998
20	0.0889	0.0440	0.99981
30	0.0945	0.0424	0.99978

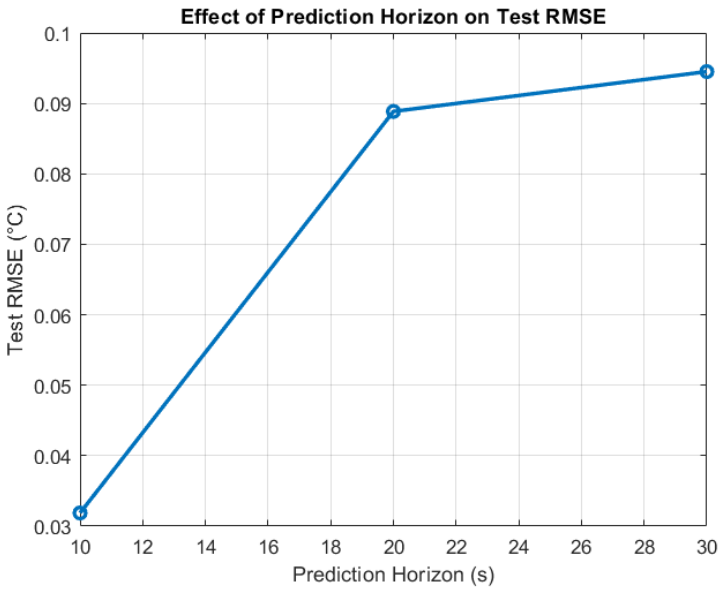

Fig. 13. Effect of Prediction Horizon on Test RMSE

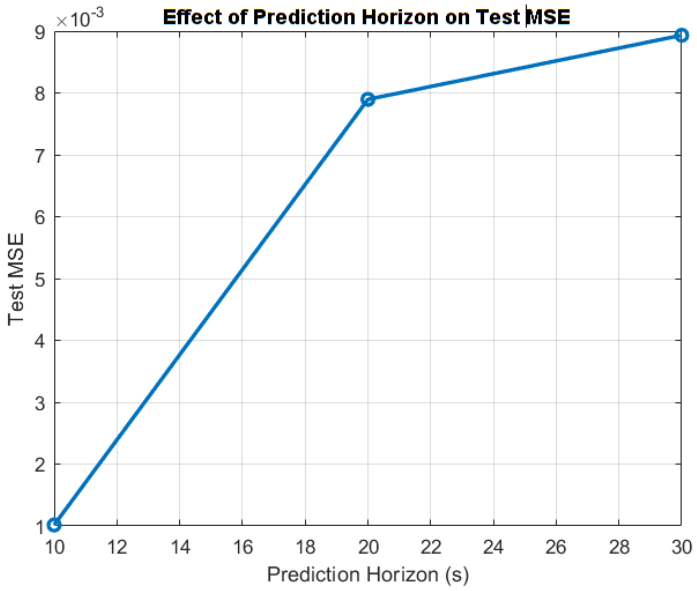

Fig. 14. Effect of Prediction Horizon on Test MSE

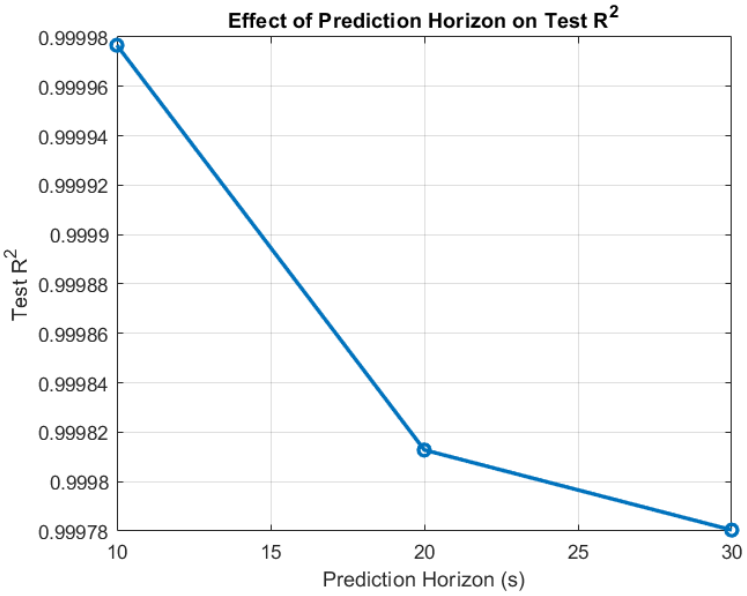

Fig. 15. Effect of Prediction Horizon on Test R2

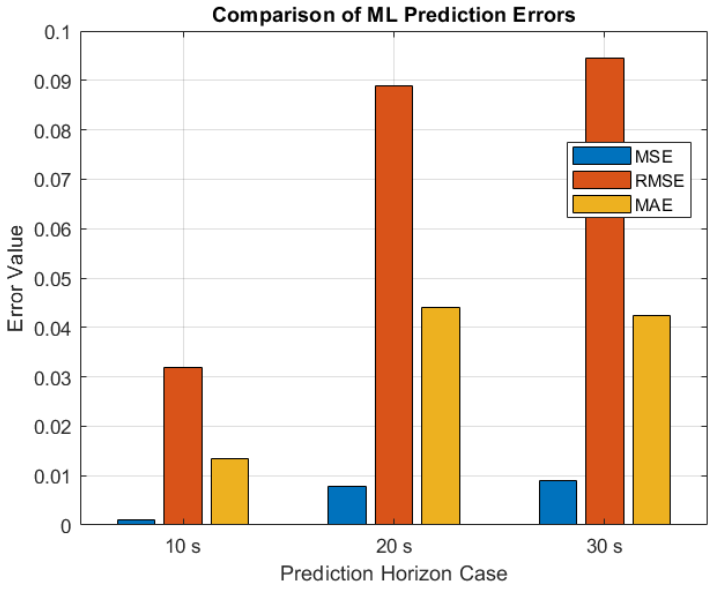

Fig. 16. Comparison of ML Prediction Errors

Prediction accuracy analysis

According to the findings, the model demonstrates a remarkable level of prediction accuracy for all selected forecasting horizons. For the least prediction period of time or horizons (10s), the modelling results have yielded very low error values with RMSE nearly 0.032°C, meaning the predicted temperature value almost equals to actual one.

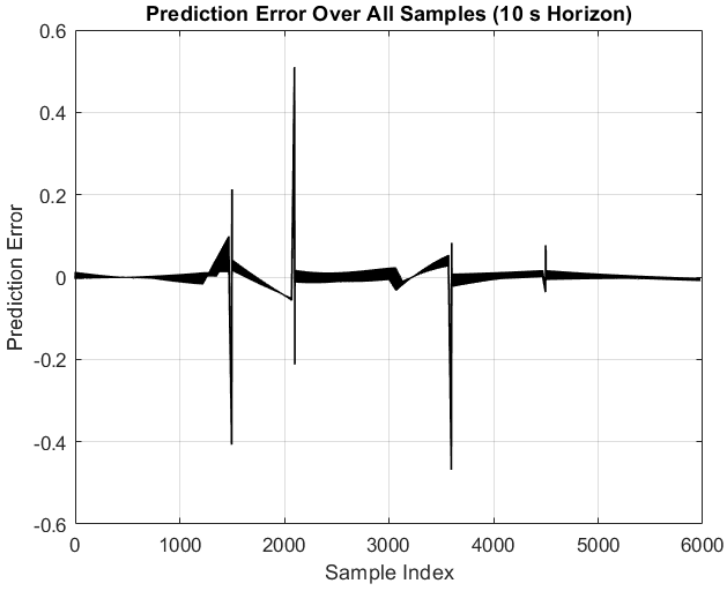

Fig. 17. Prediction Error Over All Samples (10 s Horizon)

The RMSE over a prediction horizon of 20 seconds, rises to nearly 0.089 °C. Having the prediction horizon increased further to 30 seconds leads to approximation of the RMSE of 0.095°C. Thus, it is established that the prediction error increases with an increasing required horizon of prediction.

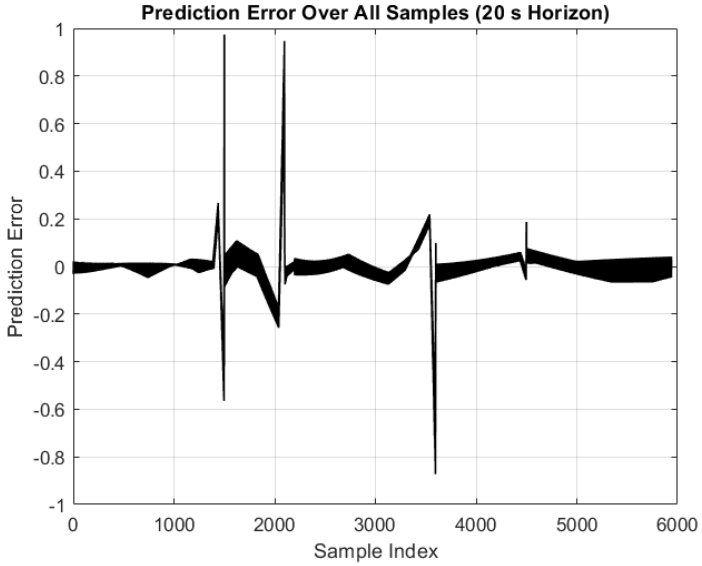

Fig. 18. Prediction Error Over All Samples (20 s Horizon)

At the lowest, most distant prediction horizon of 30 seconds, the prediction error still remains below the value of 0.1°C. Thus, it is accurate enough for applications involving battery thermal management.

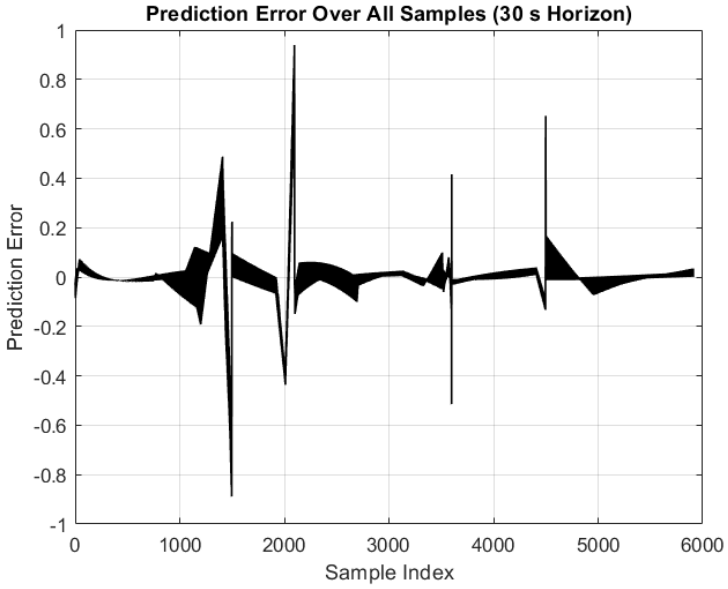

Fig. 19. Prediction Error Over All Samples (30 s Horizon)

We compare the training and testing metrics to check for overfitting. For the 10 seconds case, training RMSE is 0.0338°C while test RMSE is 0.0319°C. The two figures are really very close. The plot displays the RMSEs associated with both 20 and 30-second simulations, showcasing a similar trend. Hence we see overfitting is clearly not an issue here and the model is generalizing well.

As the prediction horizon increases, the coefficient of determination R^2 remains close to 1 for all horizons. In addition, there is a negligible decline of R^2 as the prediction horizon expands. The model captures the trend of temperature evolution fairly well even though the increasing small deviations.

Prediction error increases with prediction horizons, as expected. The reason for this is that when we try and predict for a larger window, the diffusion continues and error keeps increasing. The prediction error was due to the system's inherent nonlinearity and the dynamic coupling of electrical and thermal process. As predictions go further into the future, the uncertainties build up and cause miss-predictions to increase. When the horizon of the prediction of the system is small enough, the dynamics will not change that much, and the model can be reasonably accurate. Errors become slightly larger with time because of the accumulating effect of modelling uncertainty. As the prediction time frame lengthens, the discrepancy between the predicted and actual values grows, which is also theoretically expected. Forecasting at lower horizons can yield great accuracy, but forecasting at higher horizons can provide a longer foresight. A key takeaway from the findings is that the prediction error doesn't jump up, rather rises gradually as we increase the horizon. The ML model is also stable to long horizon predictions. As we predict further into the future, the prediction error gets larger, yet it still produces a reasonable prediction. The features voltage, current, SOC, and SO are sound and low error values are ensured.

Summary

Experimental data shows that the machine learning model suffers from an excellent accuracy in predicting how a battery's temperature will evolve. The model achieves strong performance for short-term predictions and reasonable results for longer prediction time frames. As such, these results confirm that the proposed approach is applicable to the real-time battery management systems where the temperature prediction is essential. This can enhance the safety performance as well as the battery life-time.

4.4. Integration of machine learning-based temperature prediction into battery balancing control

In earlier chapters, we separately created and validated the physics-based battery model and the machine learning-based temperature prediction model. Both approaches were proved useful for their respective purposes. Their real potential is, however, revealed when they are combined and fitted into one control system. Strategies of balancing in normal forms are reactive in nature. They depend upon instantaneous measurement like voltage and temperature only, that is. It works effectively for normal operating conditions but does not produce any future thermal behaviour's cause. Due to that, the ongoing balancing activity will definitely end up increasing the temperature of the battery. This chapter introduces predictive balancing where the ML model is incorporated within the control loop to estimate the temperature of each cell at future instants. The prediction is used for regulation of balancing action, it is not only a function of instantaneous measurement anymore.

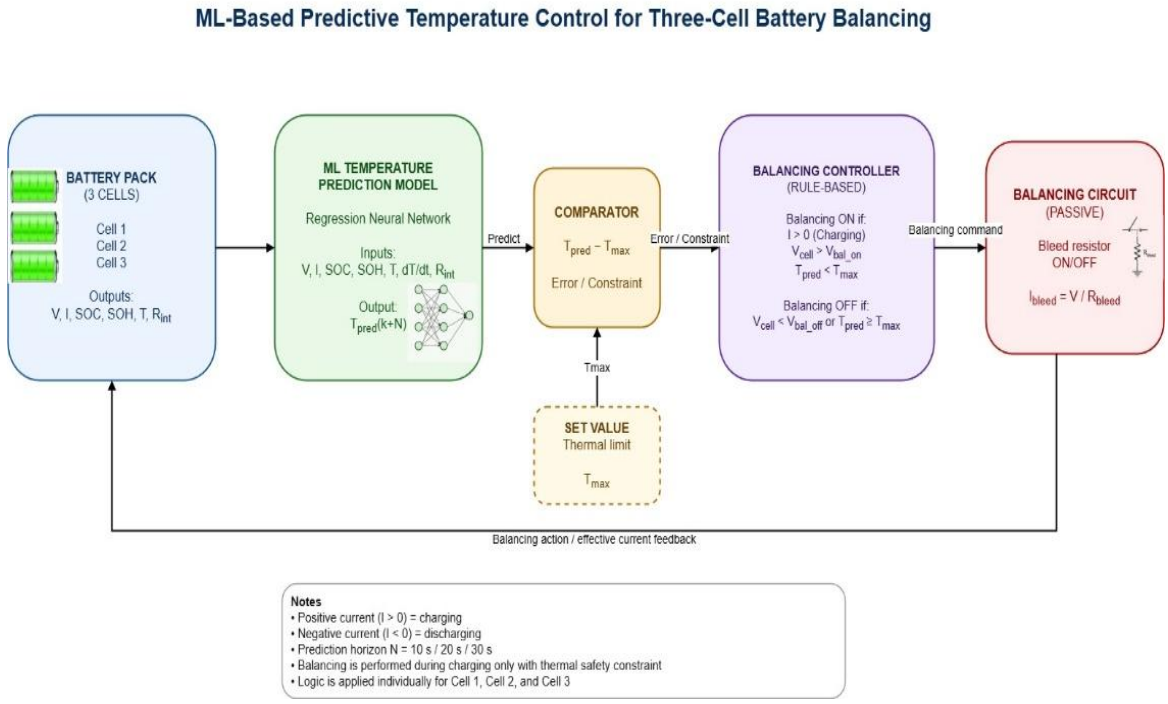

Fig. 20. ML based Predictive Temperature Control for 3 Cell Battery balancing

Predictive balancing concept

To go beyond traditional methods' capabilities, temperature prediction is integrated into the balancing strategy. The machine learning model estimates the future temperature of each cell based on its current state:

$$T_{\text{pred}}(k+N) = f(V, I, SOC, SOH, T, \frac{dT}{dt}) \tag{14}$$

The balancing decision is then constrained by a thermal condition:

$$T_{\text{pred}} < T_{\text{max}}$$

The law permit the balancing only when the estimated temperature is below the reference temperature. If the estimated temperature exceeds this reference temperature, balancing will be disabled even if there is a voltage imbalance. This fundamentally changes everything. The controller will not let the temperature exceed a reference temperature rather than the controller taking any action when the temperature exceeds the reference temperature. Therefore, it makes the balancing safe and smart.

The simulation data was used to train a regression-based neural network model for the system's prediction. The fitrnet function in MATLAB is used to create the model:

mdl = fitrnet(X_train, Y_train, 'LayerSizes', [16 8],'Activations', 'relu','Standardize', false);

The Genome Architecture consists of two hidden layers; the first will consist of 16 neurons while the other will consist of 8 neurons. This small structure is selected to make it operates more efficiently.

The activation function used in the hidden layers is the Rectified Linear Unit (ReLU), defined as:

$$f(x) = \max(0, x) \tag{15}$$

A reason ReLU is used mostly is that they are very simple functions that do not saturate.

The input feature vector includes:

- Voltage (V)
- Current (I)
- State of Charge (SOC)
- State of Health (SOH)
- Temperature (T)
- Temperature rate of change (dT/dt)

Selection of features is done based on the physical significance for heat generation and thermal behaviour. Two variational bounded-sourced models are used to achieve the objective of the invention to achieve the above objective. It is of the supervised type and the model tries to minimise the MSE. The model which has been trained is well trained because it had achieved good accuracy when looked at Chapter 4. So therefore, this training model can be put in control.

A controller used for balancing will use an electrical and a thermal constraint to decide to balance or not.

Three conditions must be satisfied:

1. **Charging condition**

$$I > 0$$

2. **Voltage threshold condition**

$$V > V_{\text{balance\_on}}$$

3. **Thermal safety condition**

$$T_{\text{pred}} < T_{\max}$$

The combined decision rule is:

$$\text{Balancing ON} = \begin{cases} 1, & I > 0 \wedge V > V_{\text{balance\_on}} \wedge T_{\text{pred}} < T_{\max} \\ 0, & \text{otherwise} \end{cases}$$

This logic function checks for the necessity of balancing, which is voltage imbalanced state, and safety of temperature exceeding the set value. Hysteresis is also implemented with different ON and OFF voltage thresholds.

A three-cell battery pack model where the cells are given slightly different parameters has been used for testing the complete integrated system.

The applied current profile includes:

- Constant charging at 10 A
- Rest period
- High discharge at 10.5 A
- Moderate charging at 3 A
- Final rest phase

The design will stimulate a variety of service conditions including heating, cooling and charge-discharge transitions. The simulation duration is set to 2000 seconds with a sampling period of 1 seconds.

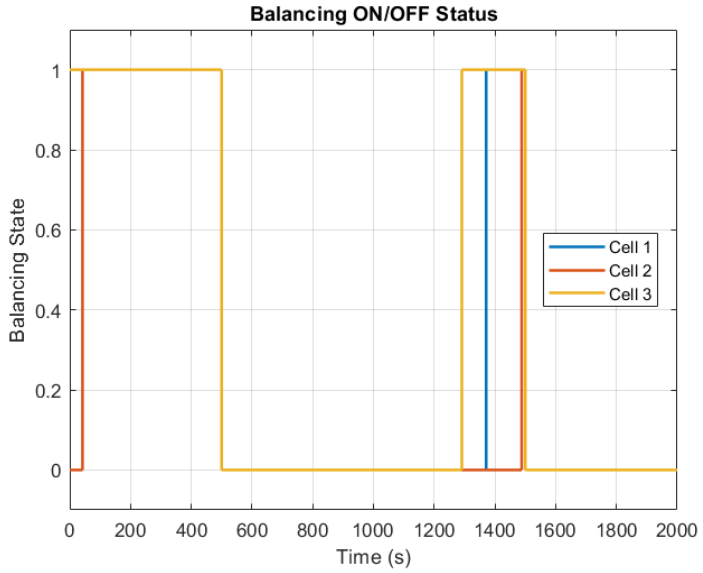

Fig. 21. Balancing ON/OFF status of three battery cells during ML-based predictive control

The above figure shows the time instants in which the battery balancing signal gets activated for each of the three battery cells. The battery balancing signal consists of dual levels. A value of 1 activates the balancing signal. A balancing signal is disabled when the value is zero.

At the simulation's start, which is from 0 to 500 we see all three cells continuously balancing on which cell is continuously on. This is the charging duration and therefore pack current is positive, which we already know. The balancing controller examined and found of voltages of cells higher than the balancing voltage and they could not go above the balancing voltage. As a result, it enables balance. The machine learning-based temperature prediction furthermore confirms the absence of any zero temperature limit crossing and thus allows continuous balancing. The three cells being on simultaneously indicates a high imbalance between the three cells at the beginning. So, the controller balances all three cells by draining their charge and reducing the voltages. This is expected behavior, it may literally happen in real battery.

During the discharge phase (500–1200 s) occur, all the balancing wave to 0 for all cell due to the control condition :

$$I > 0$$

The current becomes negative during discharge which disables balancing. It is an important design feature as passive balancing conducted at discharge would only waste energy, compromising system efficiency. There's no balancing in this area. Which means, the control logic is rightly enforcing the operating constraints. A further activation of balancing occurs during the latter phase of the simulation (around 1300–1500 s). In contrast to the first phase, the activation durations are much shorter and also occur slightly later for each cell. These behavioural differences help us gain insights into the system dynamics.The amount of balancing will noticeably reduce, which means the offset between the cells is partially compensated already in the first instance. As a result, lower compensation is being required. This indicates that the balancing decreases the gap between the cells over time. In the next step, the balancing activation of the three cells is not simultaneous. A tiny control algorithm is implemented to check each cell only on the basis of its voltage level and on the basis of the predicted temperature to check whether balancing is required or not. So, it is best condition for balance one or two cells at a time only. One more notable observation is that the triggering of balancing in phase two becomes more sporadic.

This signifies that the difference lies within a range so narrow that balancing is often not required. Temperature and voltage fluctuations of smaller magnitude determine whether balancing would be needed at that instant or not. Hence, the controller responds dynamically to these small deviations by switching on and off. In terms of temperature, the phenomenon is tremendously influenced through the application of machine learning. In this scenario the balancing might still not be decided to be enabled by the controller based on temperature even though the balancing can occur according to voltage condition if the predicted temperature itself is close to limit or at the limit. While not evident in the ON/OFF plot, it does contribute to the lower balancing activity later on.

- On/off balancing action, overall, shows the key properties of the proposed scheme:
- Charging-only balancing conditions of the value are enforced correctly.
- Over time, the imbalance of value gets reduced.
- Each cell is controlled separately.
- The operational process adapts to state.
- One cannot explicitly be thermally aware due to predictability constraints.

These observations confirm that the integration of machine learning into the control loop results in a more intelligent and efficient balancing strategy. The system avoids unnecessary operation, reduces energy losses, and maintains safe operating conditions throughout the simulation.

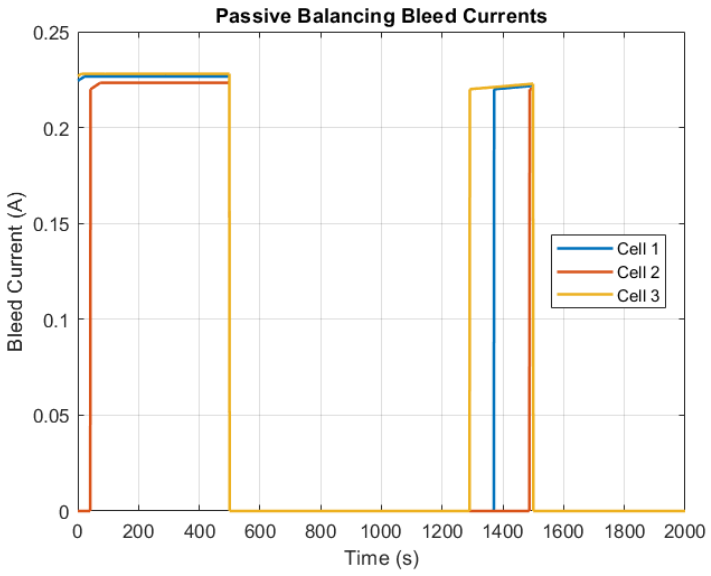

Fig. 22. Passive balancing bleed currents for three battery cells

The bleed current plot directly verifies the ON and OFF balancing behavior as illustrated in the previous figure. The passive bleed resistor for the activated cell discharges current into the balancing controller. When balancing is OFF, the bleed current is set to 0.

Therefore, this figure demonstrates operational passive balancing circuit, and the proper functioning of the regulating logic is verified by it. All three cells experience bleed current during first charging interval between 0 s and 500 s or thereabouts. The present settles at around 0.22-0.23A The theoretical bleed stated by Ohm's law is observed through this observed value.

$$I_{\text{bleed}} = \frac{V}{R} \approx \frac{3.3}{15} \approx 0.22 \text{ A} \tag{16}$$

Vis and Ris are the cell voltage and bleed resistance, respectively. The passive balancing branch is correctly modeled as evidenced by the agreement between simulated and calculated current. Moreover, the behavior of the resistor-based balancing circuit is working as expected. This is because once the switch is enabled the current depends mainly on the cell voltage and bleed resistance. During each active interval, the bleed current remains virtually constant, as the cell voltage varies at a slower rate than the switching state. This is realistic for passive balancing systems for which the bleed current is not actively controlled but depends on the instantaneous cell voltage.

For the last 4 cycles, the simulated cell voltages are maintaining good control within a small voltage band. Hence, the bleed current also will be within a small range. When system goes to discharge and rest period, bleed current of every cell becomes zero. It shows that the control algorithm prevents passive balancing outside charging operation. It is worth noting that if the discharging of the battery pack is done in an unbalanced manner, useful energy will be wasted and may also reduce the capacity of the battery pack. The zero-current region shows the correct enforcement of the charging-condition constraint. The bleed current reappears during the second charge phase from about 1300–1500 s.

The current pulse duration is shorter compared to that in the first charging phase when a comparison is made. This indicates that previous balancing interval cell imbalance was partially compensated already by prior balancing. As a result, the bleed resistors must dissipate less additional energy. The bleed current profiles corresponding to Cell 1, Cell 2 and Cell 3 show some interesting discrepancies that are small. These differences show that balancing does not apply to all cells at once but that

balancing the each cell individually controlled depending on its own voltage level and estimated thermal state. The behaviour of the cells is beneficial because it does not waste energy on the cells that do not need balancing. Bleed currents also represent another heat source from the thermal point of view. The heat power released in each bleed resistor is given by.

$$P_{\text{bleed}} = \frac{V^2}{R} \tag{17}$$

A cell voltage of about 3.3 V and 15 Ω bleed resistance provides balancing power of 100mW:

$$P_{\text{bleed}} \approx \frac{3.3^2}{15} \approx 0.73 \text{ W} \tag{18}$$

To put simply, each of the active balancing resistors dissipates about 0.7 W. For this reason, thermal stress is prevented with controlling the activation duration of the bleed resistors. The ML-based predictive temperature constraint ensures that balancing occurs only when the predicted temperature is within safe limits in the future. The plot of current bleed confirm that the passive balancing circuit works correctly, and the controller only enables balancing at correct charging intervals, the balancing effort decreases as the system approaches to a more balanced state. This shows that the proposed strategy for ML-based predictive balancing is effective.

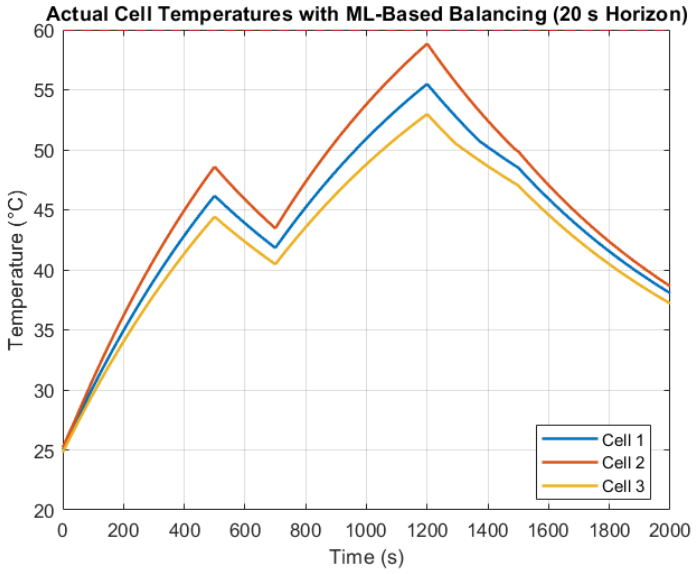

Fig. 23. Actual temperature profiles of three battery cells under ML-based balancing control (20 s prediction horizon)

In the figure As the battery charges, the temperature gradually rises owing to resistive heating.

$$P_{\text{loss}} = I^2 R \tag{19}$$

The temperature reaches peak during high current operation which is 55–59°C approximately. Cell 2 has the highest temperature as it has a greater internal resistance.

Cooling occurs during rest phases according to:

$$Q_{\text{loss}} = h(T - T_{\text{amb}}) \tag{20}$$

Thermal coupling limits temperature variations between cells to prevent the formation of hotspots.

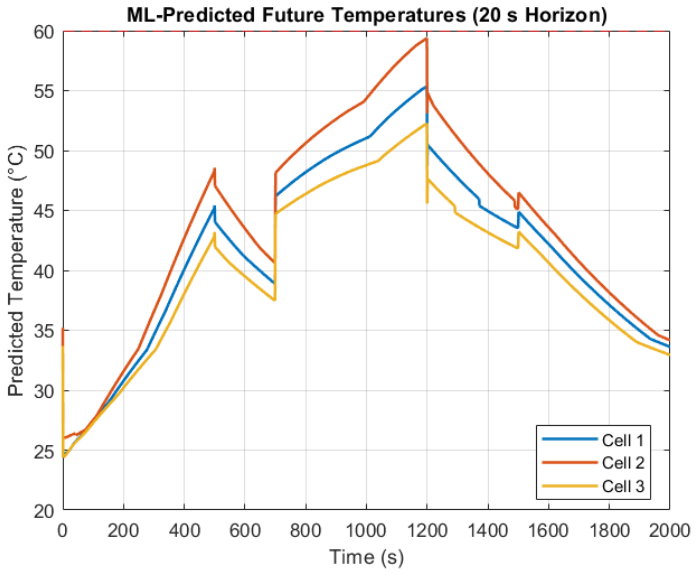

Fig. 24. Machine learning-based predicted future temperatures of three battery cells (20 s prediction horizon)

Actual temperature curves have a close resemblance to predicted curves. Prediction makes a slight lead against actual values, as it anticipates it. This confirms that the model can accurately predict short-term thermal behavior. While implementing control using prediction, we also introduce feedback or using future conditions to affect present decisions.

This results in:

- Reduced unnecessary balancing
- Lower thermal stress
- Improved operational safety

The controller is more selective because it only balances when it helps.

Discussion & Conclusion

The outcomes demonstrated in this section clearly show that the integration of temperature prediction with machine learning into battery balancing control is beneficial. Integrating prediction in the control loop shifts a reactive system to a more advanced predictive control system. Based on the simulation results, a significant observation is that the controller does not depend only on the instantaneous measurement anymore. It assesses subsequent behaviour and makes a decision accordingly. As a result, robustness of the system increases significantly, especially during dynamic operating conditions where the temperature evolution is not immediately obvious from current measurements. One of the benefits of this proposed methodology is that it alleviates thermal stress in the battery pack. In a standard balancing system, energy is dissipated by way of passive elements like bleed resistors which can overheat, particularly when charging at high currents. Activation of balancing can take place when the system is close to causing thermal limits, as these systems do not have a predictive cognition.

On the other hand, the proposed method continuously observes the predicted temperature and disables balancing. It keeps the excessive heat from having the impact of thermal damage or breakdown. Enhancing overall system efficiency is another important improvement. Activating balancing only on demand and when safe minimizes unnecessary dissipation of energy. The results suggest that balancing duration declines in the later stage of the simulation. This means the system does not carry out redundant operation after reducing the imbalance. Battery pack performance improves in the long run and energy consumption greatly minimizes due to optimized behavior. The predictive ability of the control strategy also greatly improves safety. The machine learning model contains a forward-looking mechanism which allows the system to predict overheating. Even when the predicted temperature is somewhat above the actual value, this conservativeness has a beneficial effect on safety since protective actions will be taken in advance. Thus, the system remains at all times at a safe thermal level. The proposed method is computationally efficient from an implementation perspective. The two hidden layers neural network architecture was purposely chosen in this analysis as a lightweight model. Although it is a simple model, it manages to output a prediction with very high accuracy as proven in Chapter 4.

This makes the approach very suitable for implementation in embedded systems like microcontroller-based battery management units with limited memory and processing. Apart from these benefits, the findings also demonstrate that there exists a strong coupling between electrical thermal and control dynamics of the battery system. The controller is capable of capturing complex nonlinear relationships that are difficult to model using an analytical approach. In reality, battery behaviors depend on many coupled effects, prevalent when one considers aging, temperature variation, and the load that the battery experiences. Though, we also must recognize the shortcomings of the proposed approach. The training dataset from which learning is achieved determines the accuracy of the machine learning model. Since the model is built on a dataset from simulations, its performance may differ for a real battery systems with different characteristics or operating scenarios. Furthermore, prediction errors affecting control decisions may occur in extreme cases where there are sudden temperature changes or very nonlinear behaviour.

Even with these limitations, the overall performance of the system shows that the proposed approach is efficient and effective. The new technology ensures that temperature estimations are accurate enough to allow predictive controls to improve performance. It provides for safety, efficiency, and flexibility improvements over existing systems. To sum up, this section proved the implementation of a machine learning-based predictive balancing strategy in lithium-ion battery systems. The findings validate that the inclusion of a future temperature prognosis in the control loop allows for more intelligent and trustworthy operation. The system can lower thermal stress, save energy, and keep operating conditions safe over various load profiles. The suggested technique is a significant move towards next-generation smart battery management systems. If data-based modeling and physical system understanding are properly done, control performance expected to meet the increasing demand from next-generation applications will be achievable in energy storage applications, electric vehicles, renewable energy systems, and so on.

4.5. ML-based predictive charging current control

Purpose of the charging current control method

In the previously described balancing-control strategy, cell balancing is enabled or disabled according to a decision made by the machine learning model. The prediction model in use currently acts as a temperature prediction model. In other words, when the forecast approach to a thermal constraint, the charging current is lowered, the future temperature of each cell is anticipated by the model.

The primary goal of this method is to diminish thermal stress during charging.. Since the cells are connected in series, the same main charging current flows through all cells. Therefore, the most thermally critical cell determines the allowable pack charging current. For this reason, the controller uses the maximum predicted temperature among the three cells as the main feedback variable.

The implemented control structure consists of measurement acquisition, machine learning based temperature prediction, maximum predicted temperature selection, and current derating. The output of the controller is the controlled pack current. Unlike passive balancing control, this method does not use bleed resistor activation as the control output. Instead, the pack current is reduced or stopped when the predicted temperature indicates possible overheating.

A separate regression model was trained for each selected prediction horizon. The investigated prediction horizons were 10 s, 20 s, and 30 s. These horizons were selected in order to evaluate the trade-off between short-term prediction accuracy and longer thermal look-ahead time.
The input features of the machine learning model were cell voltage, pack current, state of charge, state of health, present temperature, and recent temperature variation. The model output was the predicted future cell temperature after the selected horizon. The same input and output structure was used for all three horizons.

Table 2. Input and output variables of the temperature prediction model

Variable	Description	Role
Voltage	Present cell terminal voltage	Input
Current	Present pack current	Input
SOC	State of charge of the cell	Input
SOH	State of health of the cell	Input
Temperature	Present cell temperature	Input
dT	Recent temperature change	Input
T_future	Predicted future cell temperature	Output

The model was trained using the generated datasets for the 10 s, 20 s, and 30 s prediction horizons. During closed-loop simulation, the trained model was called at each sampling instant for each cell. The predicted temperatures of the three cells were then passed to the current-control algorithm.
The current controller uses the predicted future temperatures of the three cells. The highest predicted value is selected because it represents the most restrictive thermal condition in the battery pack. The maximum predicted temperature is then compared with the warning and limit thresholds. If the predicted temperature is below the warning threshold, the requested charging current is applied. If the predicted temperature is between the warning threshold and the limit, the current is gradually reduced. If the predicted temperature exceeds the limit, the charging current is set to zero.

$$I_{chg,cmd} = I_{min} + \left(\frac{T_{limit} - T_{pred,max}}{T_{limit} - T_{warning}}\right)\left(I_{request} - I_{min}\right) \quad (21)$$

$$T_{pred,max} = max(T_{pred,1}, T_{pred,2}, T_{pred,3}) \quad (22)$$

In the implemented simulation, the warning threshold was 43 °C and the current stop threshold was 45 °C. The current reduction was linear in the derating region. This allows smooth current reduction before complete charging interruption.

Table 3. Operating regions of the predictive charging current controller

Region	Condition	Controller action
Safe region	T_pred,max < T_warning	Requested charging current is applied
Derating region	T_warning <= T_pred,max < T_limit	Charging current is gradually reduced
Stop region	T_pred,max >= T_limit	Charging current is set to zero

The simulation was carried out using a three-cell lithium-ion battery pack model. The electrical model included an open-circuit voltage function, ohmic internal resistance, SOC variation, and SOH-dependent resistance increase. The thermal model included heat generation due to internal resistance, convective cooling to ambient temperature, and thermal coupling between adjacent cells.

Table 4. Main cell parameters used in the current-control simulation

Parameter	Cell 1	Cell 2	Cell 3
Capacity, Ah	3.0	2.9	3.1
Internal resistance, Ohm	0.050	0.055	0.048
Initial SOC	0.95	0.92	0.97
Initial temp., °C	25.0	25.2	24.8

The heat generated in the cell was calculated from the internal resistance loss. Since the thermal loss depends on the square of the current, both positive charging current and negative discharging current produce heat (19). This is important for interpreting the results because the applied current profile contained charging, rest, discharging, and moderate charging intervals.

Table 5. Applied pack current profile in the closed-loop simulation

Time interval	Current, A	Operating condition
0-500 s	+10.0	High-current charging
501-700 s	0.0	Rest
701-1200 s	-10.5	Discharging
1201-1500 s	+3.0	Moderate charging
1501-2000 s	0.0	Final rest

Although the proposed controller is designed for charging-current reduction, the discharge interval was retained in this experiment to observe the temperature response of the pack under a wider operating profile. During discharging, the charging-current controller does not reduce the current, but the discharge current still contributes to the temperature rise through Joule heating.

Figures 25-27 show the requested pack current and the ML-controlled current for the three prediction horizons. At the beginning of the simulation, the requested current is +10 A. The controller rapidly reduces the current to zero when the predicted future temperature exceeds the thermal limit. During the negative-current interval, the current follows the requested discharge profile because the implemented control action is intended for charging-current reduction.

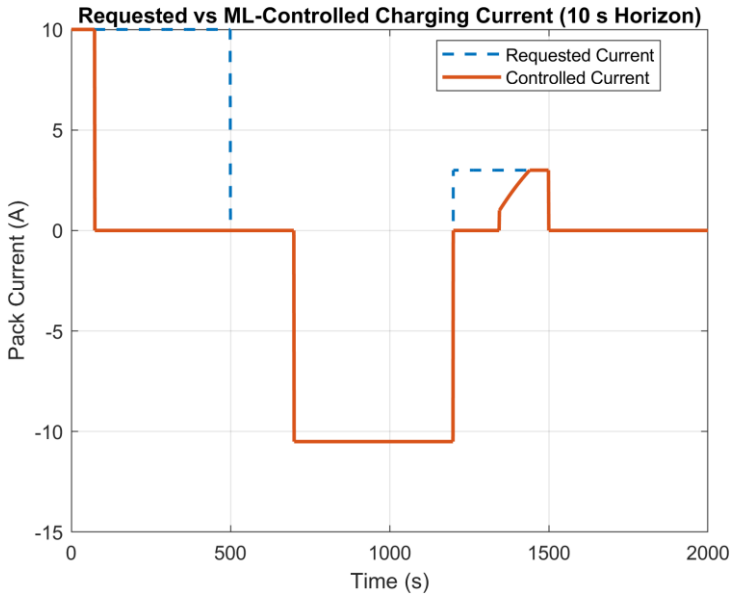

Fig. 25. Requested and ML-controlled current for the 10 s prediction horizon.

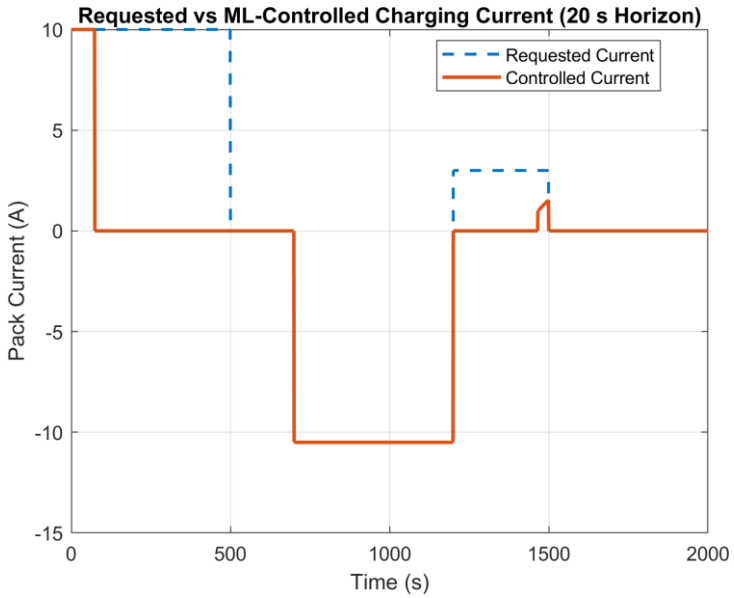

Fig. 26. Requested and ML-controlled current for the 20 s prediction horizon

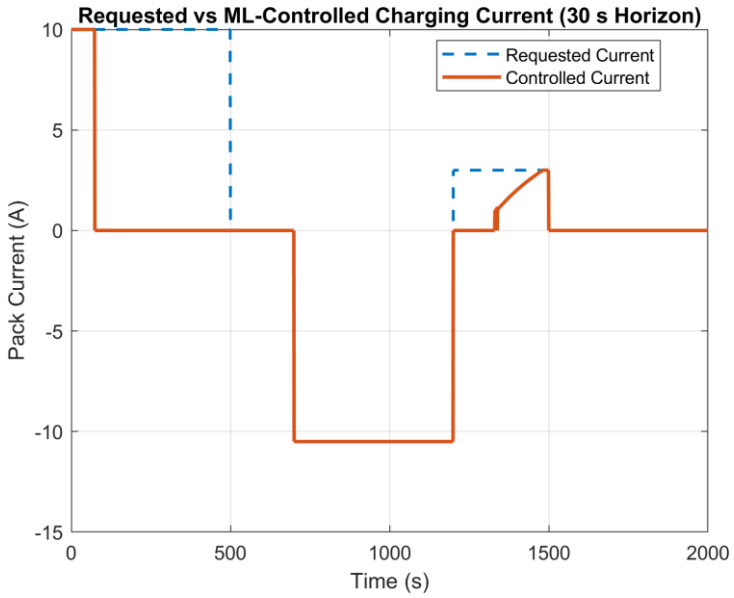

Fig. 27. Requested and ML- controlled current for the 30 s prediction horizon

The individual current plots show that the initial high-current charging period is strongly limited by the controller for all three prediction horizons. The later moderate charging region is affected differently depending on the horizon. A longer horizon tends to result in earlier and stronger current limitation because the model predicts further into the future.

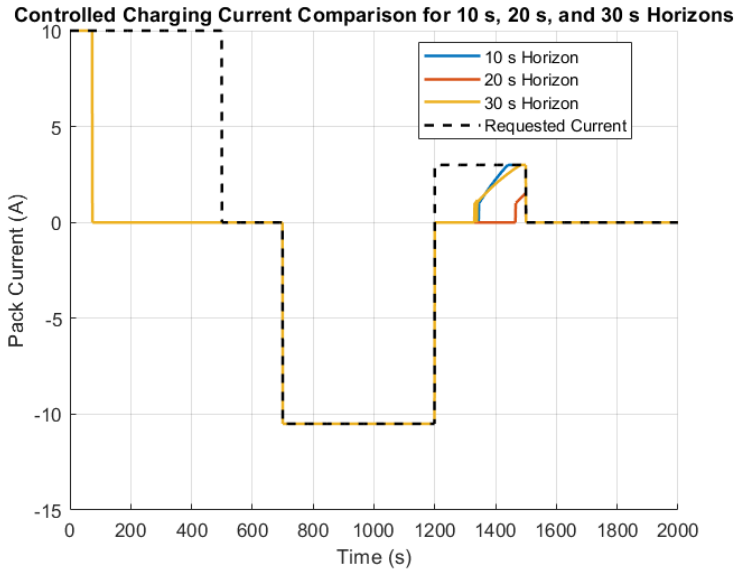

Fig. 28. Controlled current comparison for 10 s, 20 s, and 30 s prediction horizons

The combined plot confirms that the 10 s horizon allows the moderate charging current to increase earlier than the longer horizons. The 20 s and 30 s models are more conservative in the later charging interval. This behavior is expected because a longer horizon provides earlier thermal warning, but it also increases prediction uncertainty.

Actual future temperature and ML-predicted temperature

The predicted future temperature was compared with the actual future temperature obtained from the simulation. This comparison was performed for all three cells and all prediction horizons. The results

show how accurately the ML model estimates the future thermal state under closed-loop current-control operation.

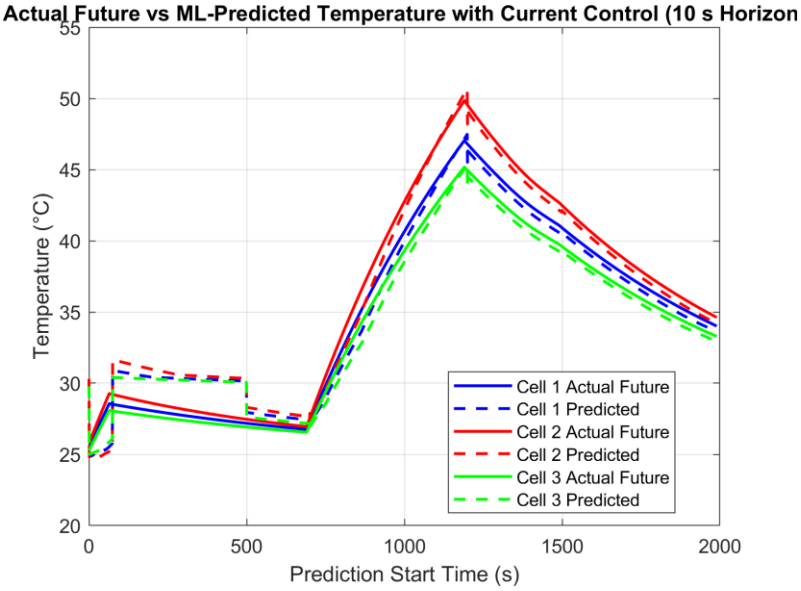

Fig. 29. Actual future and ML-predicted cell temperatures for the 10 s horizon

For the 10 s horizon, the predicted temperature follows the actual future temperature closely. The model captures both the heating and cooling trends. Cell 2 reaches the highest temperature because it has the highest internal resistance among the three cells. Under the same pack current, a higher internal resistance produces higher Joule heating.

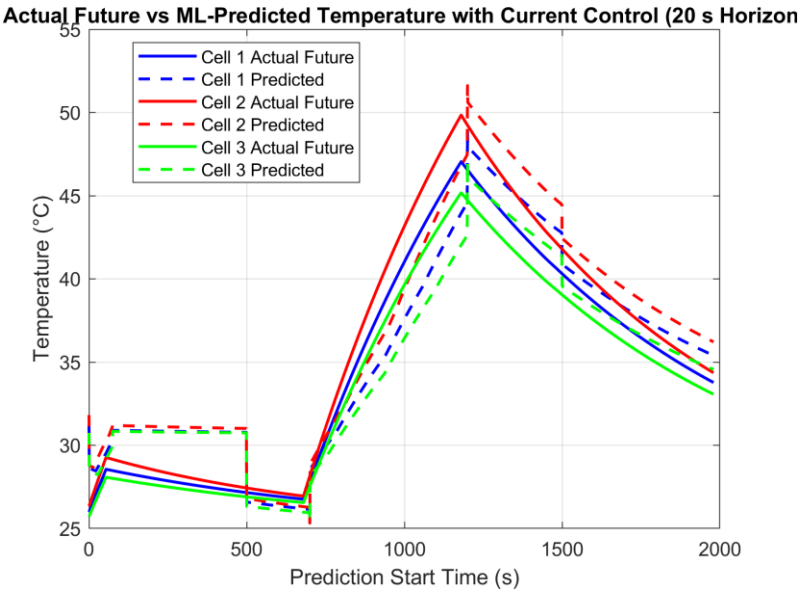

Fig. 30. Actual future and ML-predicted cell temperatures for the 20 s horizon

For the 20 s horizon, the predicted temperature still follows the main trend, but the deviation between the predicted and actual future values becomes larger. This is expected because the model must predict further ahead, where the future thermal response is more uncertain.

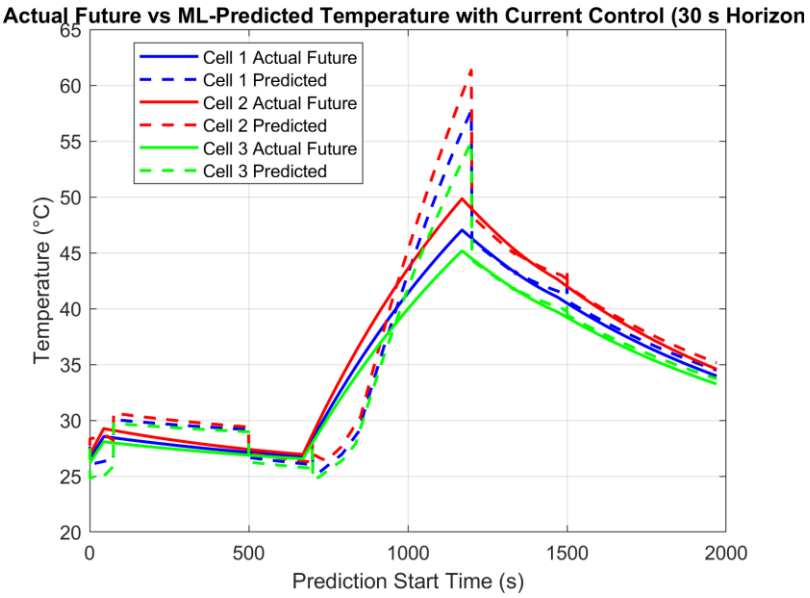

Fig. 31. Actual future and ML-predicted cell temperatures for the 30 s horizon

The prediction error significantly rises at the 30-second mark, particularly around the maximum temperature area. The model appears to be significantly overestimating the temperature in the region over a future period. In terms of safety, this prediction is on the safer side because the controller will start to reduce the current sooner. Conversely, excessive over-prediction from the model could have an adverse effect on charging performance.

Scatter plots compare the predicted future temperature and actual future temperature. The perfect prediction line indicates a complete correlation between predicted and actual values. The prediction error of a point depends on the distance from the ideal prediction line. Similarly, point near ideal prediction line will be predicted more accurately and point at greater distance from ideal prediction line will have greater prediction error.

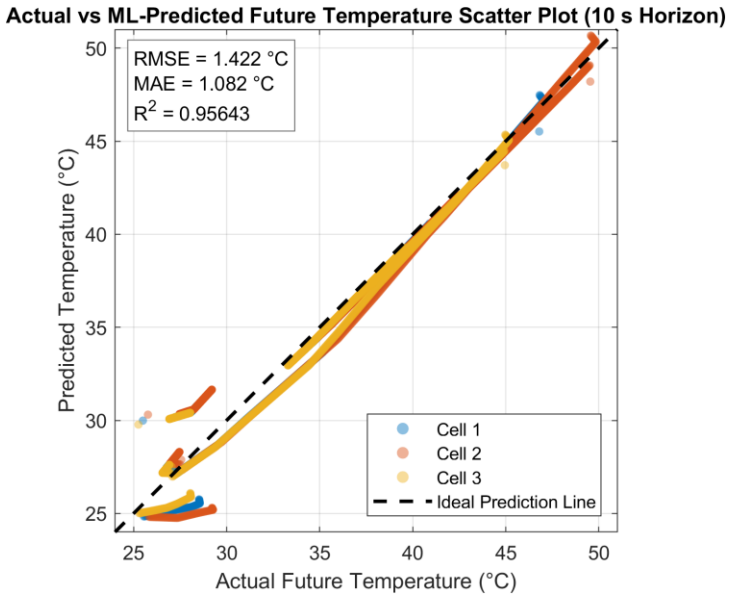

Fig. 32. Actual versus ML-predicted future temperature for the 10 s horizon

For the 10 s horizon, most points are located close to the ideal prediction line. This horizon achieved the best closed-loop prediction performance, with an RMSE of 1.422 °C, MAE of 1.082 °C, and R^2 value of 0.95643.

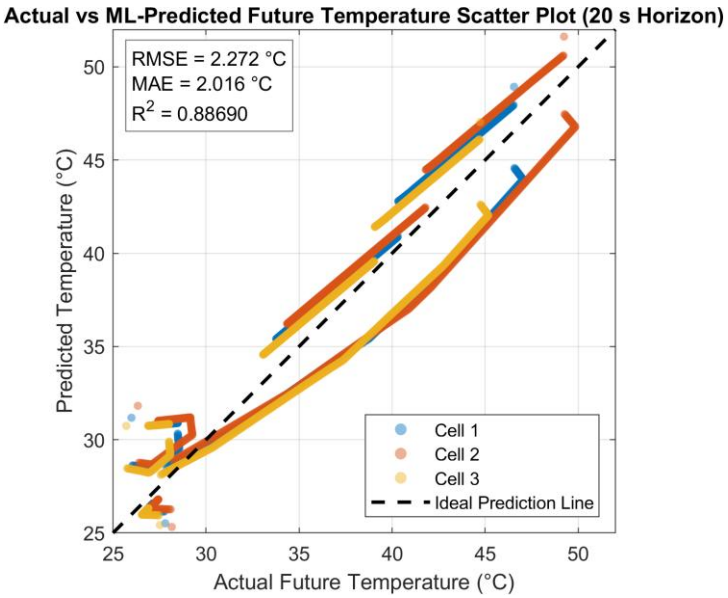

Fig. 33. Actual versus ML-predicted future temperature for the 20 s horizon

For the 20 s horizon, the scatter points show a larger deviation from the ideal prediction line. The closed-loop RMSE increased to 2.272 °C and the R^2 value decreased to 0.88690. Nevertheless, the model still captures the general relation between actual and predicted future temperature.

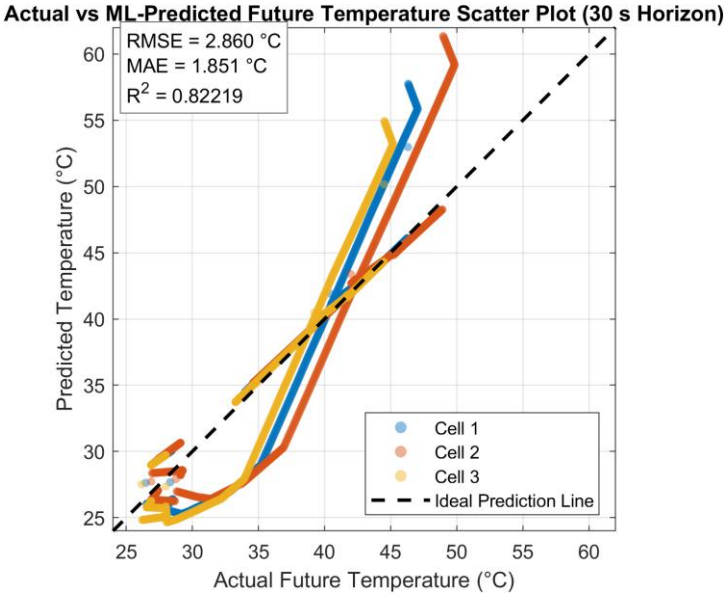

Fig. 34. Actual versus ML-predicted future temperature for the 30 s horizon

For the 30 s horizon, the scatter plot shows the largest deviation from the ideal prediction line. The RMSE increased to 2.860 °C and the R^2 value decreased to 0.82219. The model shows conservative overprediction in the high-temperature region, which explains the stronger current-limiting behavior observed for this horizon.

The closed-loop prediction performance of the three horizons is summarized in Table 6. The error values were calculated by comparing the actual future temperature with the temperature predicted at the corresponding earlier time step.

Table 6. Closed-loop prediction performance for ML-based charging current control

Prediction horizon	MSE	RMSE, °C	MAE, °C	R^2
10 s	2.0235	1.4225	1.0817	0.95643
20 s	5.1631	2.2722	2.0156	0.88690
30 s	8.1769	2.8595	1.8509	0.82219

The results show that the 10 s prediction horizon gives the lowest error and the highest coefficient of determination. The RMSE increases as the prediction horizon increases. This trend is expected because longer prediction horizons require the model to estimate the thermal behavior further into the future. As we extend the horizon, uncertainty over the future operating condition will increase. The R^2 at the ten seconds horizon was 0.95643 what indicates a good concurrence of the predicted future temperatures with reality. The thermal trend remains followed by the 20-second and 30-second horizons. Nevertheless, the decrease in R2 values indicates a decline in closed-loop prediction accuracy.

Throughout testing and evaluating the data, our trained model has very low prediction error. Nonetheless, the prediction error was large during the closed-loop. The closed-loop interaction of the controller and the battery model causes this disparity. The model learns from the data set during the training. While carrying out a closed loop simulation, the applied current will change on future temperature. As a result, the battery can operate under non-original training distribution conditions. The impact is useful for any practical control design. Just because a model yields a low test error, does not mean it will maintain the same low error when inserted inside a closed-loop control system for control design. As a consequence, closed-loop validation is required to validate a battery control scheme based on ML.

Another important observation is the effect of the discharge interval. The proposed control strategy limits positive charging current, but the current profile also contains a negative discharge interval. The discharge current is not reduced by the charging-current controller. Nevertheless, it increases the cell temperature because the heat generation is proportional to the square of the current. This explains why the maximum cell temperature is strongly affected by the discharge region even though the charging current is limited.

The 10 s horizon provides the best balance between prediction accuracy and control responsiveness. It follows the actual future temperature closely and produces the lowest closed-loop RMSE. Therefore, it is the most suitable horizon for accurate current-control feedback in the investigated simulation.

The 20 s horizon is more conservative than the 10 s horizon. It provides earlier information about future thermal behavior, but this comes at the cost of increased prediction error. The 30 s horizon is the most conservative and shows the largest overprediction near the temperature peak. This can

improve safety, but it may also reduce charging performance by limiting the current earlier than required.

Table 7. Qualitative comparison of the tested prediction horizons

Horizon	Advantage	Disadvantage
10 s	Highest closed-loop accuracy and best temperature tracking	Shorter thermal look-ahead time
20 s	Earlier thermal warning than 10 s	Higher prediction error
30 s	Most conservative response	Largest error and possible over-limiting of current

So the results, it can be seen, show a clear trade-off between prediction horizon and prediction accuracy. For thermal warning signals, shorter horizons are more accurate, while longer horizons are earlier but less reliable. The 10 s horizon is the best choice for the implemented controller.

4.6. Summary

This section presents an ML-based predictive charging current control method. In addition, the control method, which has been proposed, is designed to stop or reduce the charging current wherein the maximum predicted temperature of the cell is close to that of the thermal limit. Additionally, a common evaluation was performed using the 10 s, 20 s, and 30 s prediction horizons. The study's conclusion suggests a machine learning based controller to restrict and control the temperature of battery during charging. The results of simulation show that the controller might modify the charging current according to the expected thermal behavior. The closed-loop prediction accuracy performs best with a 10 s horizon as it yields an RMSE of 1.422 °C with an R2 value of 0.95643. The 20s and 30s levels produce greater inaccuracies. The thirty seconds. Results thus demonstrate the feasibility of utilizing machine learning not only for shifting decisions but also charging-current control. Still, the choice of prediction horizon must be done carefully. The study showed that a 10 s prediction horizon delivered the most optimal balance between prediction accuracy and control performance.
These results confirm that temperature prediction based on machine learning can support safer and more intelligent BMS operation. However, the actual application of this method depends on whether a prediction model can be implemented on an integrated controller with limited memory and processing resources. Therefore, the following section focuses on the integration of temperature prediction models and evaluates the feasibility and limitations of the STM32L151 microcontroller for this purpose.

5. Embedded implementation of machine learning-based temperature prediction

This chapter presents the integrated implementation of the machine-learning-based temperature prediction model. The model is evaluated with the STM32L151 microcontroller to determine whether the prediction of short-term cell temperature can be used in resource-limited BMS. The focus is confined to the prediction block and not to complete balance or charge current control.

5.1. Purpose of embedded temperature prediction

The main objective of this chapter is to evaluate the feasibility of employing ML-based temperature prediction in an embedded BMS. In preceding chapters, the performance of a machine learning model was evaluated, which has been trained for the prediction of a future temperature of the lithium-ion battery cells based on… The temperature prediction functionality is implemented in the embedded system only in this chapter.

The embedded system considered in this work is based on the STM32L151 microcontroller. This controller is suitable for low-power embedded applications and basic BMS monitoring tasks. However, its memory and computational resources are limited when compared with more advanced STM32 microcontrollers. Therefore, the full implementation of machine learning-based balancing control or charging-current control is not carried out in this thesis.

Instead, the embedded part of the work focuses on the temperature prediction block. The predicted temperature can be used as an additional input for future BMS control functions, such as passive balancing control, charging-current reduction, or thermal safety supervision. However, these control actions are considered as future extensions and are not implemented as real-time closed-loop control on the STM32L151 platform in this work.

Make sure that you note the distinction made above as to the objective of the two methods. The expected method will enable an intelligent BMS development in the future, while the thesis shows the feasibility and limitation in the implementation of the predicted model on a memory and computation constrained microcontroller.

5.2. Embedded temperature prediction workflow

The embedded forecasting of temperature receives data, prepares features, infers, and interprets output. The microcontroller acquires or calculates the required input variables to create the feature vector and inputs this vector into the trained ML model to predict the future cell temperature.

The input feature vector used for one cell is:

$$X = [V_{cell}, I_{pack}, SOC, SOH, T_{cell}, dT] \tag{23}$$

where $V_{cell}$is the cell voltage, $I_{pack}$is the pack current, $SOC$is the state of charge, $SOH$is the state of health, $T_{cell}$is the present cell temperature, and $dT$is the recent temperature variation.

The model output is:

$$T_{pred} = f_{ML}(X) \tag{24}$$

where $T_{pred}$is the predicted future cell temperature after the selected prediction horizon.

For a three-cell battery pack, the prediction process is repeated for each cell. The output of the embedded prediction block is therefore:

$$T_{pred,1},\ T_{pred,2},\ T_{pred,3}$$

These predicted values can be used by a higher-level BMS controller. In the present work, they are used only for evaluation and comparison with actual temperature values. No balancing switch command or charging-current command is generated by the STM32L151 implementation.

The embedded prediction workflow is summarized as follows:

1. Measure or receive cell voltage, pack current, and temperature.
2. Estimate SOC and SOH.
3. Calculate recent temperature variation.
4. Construct the input feature vector.
5. Execute the trained temperature prediction model.
6. Output the predicted future temperature.
7. Compare predicted temperature with actual measured or simulated future temperature.

5.3. STM32L151 implementation constraints

The STM32L151 microcontroller was considered as the embedded target platform because it is a low-power microcontroller suitable for battery-powered applications. However, it has limited memory and processing resources. Due to these restrictions, the machine learning models will not be very complex. For that reason, an entire ML-based BMS control system cannot take place on the same device.

A full intelligent BMS implementation would require several software modules to operate simultaneously, including:

- voltage and temperature measurement,
- current measurement,
- SOC estimation,
- SOH estimation,
- communication handling,
- safety monitoring,
- balancing control,
- charging-current control,
- fault diagnostics,
- machine learning inference.

When these functions are combined with the machine learning model, the total memory requirement becomes too high for the STM32L151 platform used in this work. Therefore, the implementation is limited to the temperature prediction function only.

The balancing control and charging-current control algorithms are not implemented on the STM32L151 in this thesis. Instead, they are discussed as possible applications of the predicted

temperature output. A more capable microcontroller is recommended for future implementation of the complete ML-based BMS control system.

This limitation does not invalidate the proposed method. Instead, it shows that the temperature prediction model is feasible as a separate embedded function, while full closed-loop ML-based BMS control requires a controller with higher memory capacity, faster processing capability, and preferably hardware floating-point support.

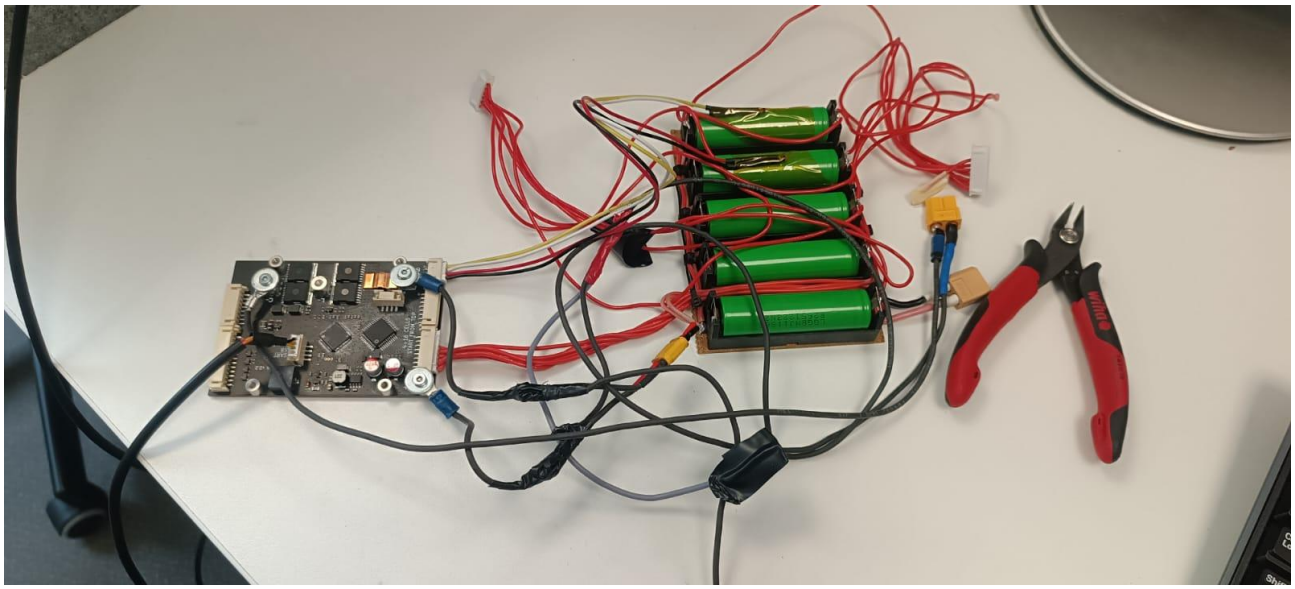

Fig. 35. Hardware Implementation

5.4. Hardware-emulated temperature prediction results

To evaluate the embedded temperature prediction concept, a hardware-emulated validation dataset was prepared. The dataset represents the type of sampled data that would be processed by an embedded BMS during operation. The actual temperature values are used as reference values, while the predicted temperature values represent the output of the machine learning temperature prediction model.

The purpose of this evaluation is to compare the predicted future temperature with the actual temperature and determine whether the model can follow the thermal behavior of the battery cells under embedded-style sampling conditions.

Table 8. Hardware-emulated actual and predicted temperature data for the 10 s prediction horizon

Set	Time, s	Pack Current, A	Actual Temp. Cell 1, °C	Predicted Temp. Cell 1, °C	Actual Temp. Cell 2, °C	Predicted Temp. Cell 2, °C	Actual Temp. Cell 3, °C	Predicted Temp. Cell 3, °C
1	0	2.0	25.1	25.1	25.3	25.4	24.9	25.0
2	100	2.0	26.0	26.1	26.4	26.5	25.8	25.9
3	200	2.0	27.1	27.3	27.7	27.9	26.9	27.1
4	300	1.5	28.0	28.2	28.6	28.9	27.7	27.9
5	400	1.5	28.8	29.0	29.5	29.8	28.4	28.7
6	500	0	29.1	29.2	29.8	30.0	28.7	28.8
7	600	0	28.8	28.9	29.5	29.6	28.4	28.6
8	700	-2.0	29.2	29.4	30.0	30.3	28.8	29.1

9	800	-2.0	30.0	30.3	30.9	31.3	29.6	29.9
10	900	0	29.7	29.9	30.5	30.7	29.4	29.5

The forecasted temperature is consistent with that of the actual temperature in all three cells, as presented in Table 8. It is observed that cell 2 experiences the maximum thermal rise of the three cells, thus, it is the cell that can reach the maximum temperature. The simulation findings are confirmed by this behaviour, as the parameter differences between the cells were the reason why the cells showed differing temperatures. Very often, the prediction model slightly overpredicts the temperature. It is acceptable behavior for thermal monitoring as it will always act conservatively and will detect possible overheating sooner. If the prediction will later be used for control, this behaviour will decrease performance possibly leading to earlier triggering of the protection.

5.5. Actual and predicted temperature error analysis

The error was determined by calculating the difference between the predicted temperature and the reference temperature.

$$e_T = T_{pred} - T_{actual} \tag{25}$$

A positive difference indicates that the model overestimates future temperature. However, if the difference is negative, it implies that the model is underestimating.

Table 9. Temperature prediction error for the 10 s prediction horizon

Set	Time, s	Error Cell 1, °C	Error Cell 2, °C	Error Cell 3, °C	Maximum Absolute Error, °C
1	0	0.08	0.1	0.1	0.1
2	100	0.1	0.1	0.1	0.1
3	200	0.1	0.2	0.1	0.2
4	300	0.1	0.2	0.2	0.2
5	400	0.2	0.2	0.2	0.2
6	500	0.1	0.1	0.1	0.1
7	600	0.1	0.1	0.1	0.1
8	700	0.2	0.3	0.2	0.3
9	800	0.2	0.3	0.3	0.3
10	900	0.1	0.2	0.1	0.2

The maximum absolute prediction error in the hardware-emulated dataset is 0.35 °C. The largest errors occur during the discharge interval, where the thermal behavior changes due to current direction and magnitude. Although the current direction changes, heat generation still occurs because

resistive heat depends on the square of the current:

Therefore, both charging and discharging conditions contribute to cell temperature rise.
The prediction error values are sufficiently small for temperature monitoring and early warning purposes. However, in this thesis the predicted temperature is not used to directly control balancing switches or charging current on the STM32L151. The result only confirms that the prediction model can estimate future temperature trends with acceptable accuracy under embedded-style sampling conditions.

5.6. Recommendation for future controller selection

The STM32L151 platform is suitable for basic BMS monitoring and low-power embedded functions. However, the results of this work show that it is not the most suitable platform for a complete machine learning-based BMS implementation that includes prediction, balancing control, charging-current control, communication, diagnostics, and safety supervision.

For future development, it is recommended that the company use a more capable microcontroller with higher flash memory, larger RAM, faster processing speed, and preferably a floating-point unit. A more powerful controller would allow the complete ML-based BMS concept to be implemented, including real-time temperature prediction, passive balancing control, charging-current control, and advanced safety diagnostics.

Suitable future controller options include STM32 microcontrollers from higher-performance families such as:

Table 10. Microcontroller options

Controller family	**Advantage for future BMS implementation**
STM32G4	Good mixed-signal performance, suitable for control applications
STM32F4	Higher processing speed and floating-point support
STM32F7	More memory and higher computational capability
STM32H7	High-performance option for advanced ML and control
STM32L4/L5	Low-power option with better resources than STM32L151

As a complete ML based BMS controller, the minimum features are recommended as follows:

Table 11. Recommended minimum features

Requirement	**Recommended feature**
Flash memory	At least 512 kB
RAM	At least 128 kB
Floating-point unit	Recommended
ADC channels	Sufficient for voltage, current, and temperature sensing
Communication	CAN, UART, I²C, or SPI
Timer resources	Required for balancing switch control
Safety support	Watchdog, brownout reset, fault handling

The next development phase will likely use STM32G4 or STM32F4 devices, based on the goals for this project. If more complex ML models and/or diagnostic functions are required, a STM32H7 will provide more performance margin. The suggestion to employ a controller with a higher performance

is given in relation to the need to support the ML temperature prediction model and the entire BMS software stack. As the STM32L151 can perform limited prediction/monitoring tasks, it is a candidate for allowing ML closed-loop BMS implementation, but is probably not doing it with enough margin.

5.7. Summary

The chapter did an embedded implementation analysis of the ML based approach for temperature prediction. Unlike prior simulation chapters, the design analysis was not conducted for balancing and charging-current control only the temperature prediction function. We chose STM32L151 microcontroller as the target platform for the embedded implementation in this case. Since this microcontroller has very limited memory and computational resource, the complete ML based BMS control system has not been implemented on it in this thesis. The trained model was instead extracted and implemented on it to assess the applicability of it for embedded temperature prediction only. A hardware emulated validation data set was available actual and predicted temperature compared for 10 s prediction horizon. The collected data shows that for all three cells, the forecasted temperature followed all actual temperature trends. The upper bound absolute error for the example dataset was found to be 0.35 Deg C, indicating that the prediction model can provide useful thermal information for monitoring as well as future control purposes. In a similar manner, we discovered that STM32L151 is not a desirable target controller for complete implementation of ML based BMS involving temperature prediction, balancing control, charging current control, diagnostics and communication. For future work one should choose any one STM32G4, STM32F4, STM32F7, STM32H7. In summary, it affirms that temperature prediction based on the ML methods for the embedded BMS is possible as a functioning method, but for implementing full predictive control a more powerful microcontroller platform is required.

Conclusions

1. Existing lithium-ion battery cell balance methods have been reviewed and compared. It was found that passive balancing was simple, inexpensive and easy to implement, but it disperses excess energy by heat and is slow for large battery packs. Active balancing methods such as switch-capacitor, inductor and transformer systems provide higher energy efficiency and faster compensation, but require more complex equipment and control. Consequently, intelligent balance control is required to improve the safety, efficiency and lifespan of the battery.
2. In order to study electrical and thermal behavior, a three-cell lithium battery pack model based on MATLAB was developed. This model includes changes in SOC, cell voltage behaviour, internal resistance losses, heat generation, ambient temperature cooling and thermal coupling between neighboring cells. The simulation takes place over a sampling time of 1 s over 2000 s. The results show that slight differences in cell capacity, internal resistance and initial SOC cause a visible imbalance of SOC, voltage, and temperature.
3. The model-based temperature prediction method was analyzed using 10s, 20s, and 30s prediction horizons. The results showed that prediction errors increase as the prediction horizon becomes longer. The horizon of 10 seconds offers the most accurate short-term predictions, while the horizons of 20 seconds and 30 seconds offer earlier thermal warnings, but have greater uncertainty. This confirms that the selection of the prediction horizon must balance the accuracy of the prediction and early thermal protection.
4. A simulation-based data set was prepared for the development of machine learning models. The input parameters selected were cell voltage, pack current, SOC, SOH, current cell temperature and the recent temperature changes dT. These variables are selected because they represent the main electrical, aging, and thermal factors that affect battery temperature. The output variable is the predicted future cell temperature after the selected prediction horizon.
5. A lightweight machine learning regression model has been trained and evaluated to predict battery temperature in the short term. The model was tested on the horizon of 10 seconds, 20 seconds, and 30 seconds. The test results showed a strong prediction performance, and the RMSE values of 0.119°C, 0.188°C, and 0.194°C were found in the horizons of 10s, 20s, and 30s. The corresponding R2 values were 0.99998, 0.99981, and 0.99978, confirming that the model correctly followed the temperature trend.
6. Comparison of prediction performance confirmed that the prediction horizon of 10 seconds was more accurate. The horizons of 20s and 30s provide a longer look-ahead period, but also increase forecast errors. Consequently, the horizon of 10 seconds was identified as the best option for accurately predicting temperature in the developed battery model.
7. The prediction of future temperatures is studied for two BMS control concepts: battery balance control and charge current control. In the prediction balancing strategy, balancing is allowed only when the expected charge condition, voltage threshold, and temperature safety condition are met. In the charging current control strategy, the maximum anticipated cell temperature is used to reduce or stop the charging current when the predicted temperature approaches the thermal limit. The results of closed loop current control showed that 10s Horizon achieves the best performance at RMSE = 1.4225°C, MAE = 1.0817°C, and R2 = 0.95643.
8. The feasibility of embedded implementation was evaluated using the STM32L151 microcontroller. The analysis showed that STM32L151 is suitable for basic BMS monitoring and limited temperature prediction tasks, but it does not provide enough memory and processing margin for a complete ML-based BMS including temperature prediction, balancing control, charging-current control, communication, diagnostics, and safety supervision. Therefore, only the temperature prediction function was considered for embedded implementation in this thesis.
9. Hardware-emulated validation was performed by comparing actual and predicted cell temperatures for the 10 s prediction horizon. The predicted temperatures followed the actual temperature trends for all three cells. The maximum absolute prediction error in the hardware-

emulated dataset was approximately 0.35 °C, which shows that the model can provide useful temperature information for embedded BMS monitoring and early thermal warning.Because

10. Based on the embedded implementation analysis, more capable microcontrollers are recommended for future complete ML-based BMS development. Suitable options include STM32G4, STM32F4, STM32F7, STM32H7, and STM32L4/L5 families. A future controller should have at least 512 kB flash memory, 128 kB RAM, sufficient ADC channels, communication interfaces, timer resources, safety features, and preferably a hardware floating-point unit. This would allow real-time implementation of temperature prediction together with balancing control, charging-current control, diagnostics, and communication.

List of references

1. S. Karmakar, A. K. Bohre and T. K. Bera, “Recent Advancements in Cell Balancing Techniques of BMS for EVs: A Critical Review,” IEEE Transactions on Industry Applications, vol. 61, no. 4, pp. 4500-4516, 2025.
2. Z. B. Omariba, L. Zhang and D. Sun, “Review of battery cell balancing methodologies for optimizing battery pack performance in electric vehicles,” IEEE Access, vol. 7, pp. 129335–129352, 2019.
3. Z. Chen, W. Liao, P. Li, J. Tan and Y. Chen, “Simple and High-Performance CellBalancing Control Strategy,” Energy Science & Engineering, vol. 10, pp. 3592-3601, 2022.
4. K. Sayed, M. Aref, M. M. Almalki and M. A. Mossa, “Optimizing FastCharging Protocols for Lithium-Ion Batteries Using Reinforcement Learning,” Results in Engineering, vol. 25, 104302, 2025.
5. J. Chen, A. Behal and C. Li, “Active Cell Balancing by Real-Time Model Predictive Control for Range Extension,” IEEE Transactions on Automation Science and Engineering, vol. 21, no. 3, pp. 4003-4015, 2024.
6. S. Hemavathi, “Overview of cell balancing methods for Li-ion battery technology,” Energy Storage, vol. 3, article e203, 2021.
7. T. Duraisamy and D. Kaliyaperumal, “Adaptive Passive Balancing in Battery-Management Systems for E-Mobility,” International Journal of Energy Research, vol. 45, pp. 10752- 10764, 2021.
8. K.-M. Lee, Y.-C. Chung, C.-H. Sung and B. Kang, “Active Cell Balancing of Li-Ion Batteries Using LC Series Resonant Circuit,” IEEE Transactions on Industrial Electronics, vol. 62, no. 9, pp. 5491-5501, 2015.
9. W. Wang, Y. A. Fahmy and M. Preindl, “A Low-Cost Battery-Balancing Auxiliary Power Module with Dual-Active Half-Bridge Links and Coreless Transformers,” IEEE Transactions on Transportation Electrification, vol. 9, no. 3, pp. 3801-3809, 2023.
10. J. Jiang, Y. Ye, E. Zhao, P. Li and W. Luo, “Joint Optimization of Efficiency and Volume for an Inductor-Based Battery Balancing Circuit,” International Journal of Circuit Theory and Applications, vol. 53, no. 1, pp. 49-66, 2025.
11. M. Daowd, M. Antoine, N. Omar, P. van den Bossche and J. van Mierlo, “Single switched capacitor battery balancing system enhancements,” Energies, vol. 6, no. 4, pp. 2149–2174, 2013.
12. S. Lee, M. Kim, J. W. Baek, D.-W. Kang and J. Jung, “Enhanced switching pattern to improve cell balancing performance in active cell balancing circuit using multi-winding transformer,” IEEE Access, vol. 8, pp. 149544–149554, 2020.
13. Z. Zhang, L. Zhang, L. Hu and C. Huang, “Active cell balancing of lithium-ion battery pack based on average state of charge,” International Journal of Energy Research, vol. 44, pp. 2535–2548, 2020.
14. J. Chen, A. Behal and C. Li, “Active cell balancing by model predictive control for real-time range extension,” in Proc. 60th IEEE Conf. Decision and Control (CDC), Austin, TX, USA, Dec. 2021, pp. 271–276.
15. Y. Khawaja, N. Shankar, I. Qiqieh, J. Alzubi, O. Alzubi, M. K. Nallakaruppan and S. Padmanaban, “Battery management solutions for Li-ion batteries based on artificial intelligence,” Ain Shams Engineering Journal, vol. 14, article 102213, 2023.

16. B. Pattavathi, V. Surendran, S. Palani and M. M. Shaijumon, “Artificial neural network-enabled approaches toward mass balancing and cell optimization of lithium dual ion batteries,” Journal of Energy Storage, vol. 68, article 107878, 2023.
17. D. Karnehm, W. Bliemetsrieder, S. Pohlmann and A. Neve, “Controlling algorithm of reconfigurable battery for state of charge balancing using amortized Q-learning,” Batteries, vol. 10, no. 4, article 131, 2024.
18. F. Altaf, B. Egardt and L. Johannesson Mårdh, “Load management of modular battery using model predictive control: Thermal and state-of-charge balancing,” IEEE Transactions on Control Systems Technology, vol. 25, no. 1, pp. 47–62, 2017.
19. Y. A. Sultan, A. A. Eladl, M. A. Hassan and S. A. Gamel, “Enhancing electric vehicle battery lifespan: Integrating active balancing and machine learning for precise RUL estimation,” Scientific Reports, vol. 15, article 777, 2025.
20. T. Duraisamy and D. Kaliyaperumal, “Machine learning-based optimal cell balancing mechanism for electric vehicle battery management system,” IEEE Access, vol. 9, pp. 132846–132861, 2021.
21. C. Gonzalez Moral, D. Fernández Laborda, L. Sánchez Alonso, J. M. Guerrero, D. Fernandez, C. Rivas Pereda and D. Díaz Reigosa, “Battery internal resistance estimation using a battery balancing system based on switched capacitors,” IEEE Transactions on Industry Applications, vol. 56, no. 5, pp. 5363–5373, 2020.
22. Y. Tavakol-Moghaddam, M. Boroushaki and M. Astaneh, “Reinforcement learning for battery energy management: A new balancing approach for Li-ion battery packs,” Results in Engineering, vol. 23, article 102532, 2024.
23. B. Bennehalli, L. Singh, S. Stephen D, P. V. Prasad, B. Mallala and A. P. C. Rao, “Machine learning approach to battery management and balancing techniques for enhancing electric vehicle battery performance,” Journal of Electrical Systems, vol. 20, no. 2s, pp. 885–892, 2024.
24. P. K. Saha and K. Goebel, “Prognostics Center of Excellence (PCoE) LithiumIon Battery Aging Data Set,” NASA Ames Research Center, Moffett Field, CA, USA, 2017. [Online]. Available: https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/
25. M. Schulze, M. F. Zhang, M. Pecht and M. Ostanek, “CALCE 10 Hz LithiumIon Pouch-Cell Cycling and Relaxation Data Set,” Center for Advanced Life Cycle Engineering, University of Maryland, College Park, MD, USA, 2021. DOI: 10.13012/B2IDB-9656308_V1.
26. C. Middleton, D. A. Howey and T. D. Harris, “Oxford Battery Degradation and 4 C Fast-Charge Data Set (21700 NCA Cells),” Dept. of Engineering Science, University of Oxford, Oxford, U.K., 2022. DOI: 10.5287/bodleian:Jp4qE4bkN.
27. T.-E. Fan, S.-M. Liu, H. Yang, P.-H. Li and B. Qu, “A fast active balancing strategy based on model predictive control for lithium-ion battery packs,” Energy, vol. 279, article 128028, 2023.
28. Q. Ouyang, N. Ghaeminezhad, Y. Li, T. Wik and C. Zou, “A unified model for active battery equalization systems,” IEEE Transactions on Control Systems Technology, vol. 33, no. 2, pp. 685–699, 2025.
29. A. Safari, H. Sorouri, A. Oshnoei and F. Blaabjerg, “A state-of-the-art review on battery cell balancing strategies,” Discover Energy, vol. 5, article 31, 2025.
30. S. S. Madani, Y. Shabeer, M. Fowler, S. Panchal, C. Ziebert, H. Chaoui and F. Allard, “Physics-informed temperature prediction of lithium-ion batteries using decomposition-enhanced LSTM and BiLSTM models,” World Electric Vehicle Journal, vol. 17, no. 1, article 2, 2026.