LiSimPack: A Python Library for Battery Pack Simulations with PyBaMM

Arjun Lakhanpal
Lakhanpxl.arjun@gmail.com
GGSIPU

Parul
jparul187@gmail.com
IGDTUW

Abstract

Battery technology has become a cornerstone of modern energy systems, enabling advancements in electric vehicles, renewable energy storage, and portable electronics. Accurate simulation tools are critical for understanding battery pack behavior under various conditions, optimizing performance, and ensuring safety. LiSimPack is a Python-based library built upon PyBaMM (Python Battery Mathematical Modeling) designed to simplify the simulation of battery packs, offering users an accessible, modular, and high-performance framework. This paper introduces LiSimPack, outlines its architecture, highlights its capabilities, and demonstrates its application through case studies. By bridging the gap between individual cell modeling and pack-level simulations, LiSimPack empowers researchers, engineers, and educators to explore advanced battery behaviors with ease.

ACM Reference Format:
Arjun Lakhanpal and Parul. 2025. LiSimPack: A Python Library for Battery Pack Simulations with PyBaMM. In . ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

The growing demand for efficient energy storage solutions has accelerated the need for advanced battery modeling tools. While many tools focus on individual cell performance, real-world applications require understanding how cells interact within a pack. PyBaMM is a powerful library for electrochemical battery modeling, but extending its capabilities to battery packs involves significant effort. LiSimPack addresses this gap by providing an intuitive interface and robust features for pack-level simulations.

Electrification of transport and energy-intensive activities is crucial for reducing carbon emissions. Batteries play a pivotal role in energy storage for electric vehicles and renewable energy applications. Understanding battery behavior, degradation, and optimization at scale is essential to meeting emission reduction targets. As improvements in individual battery chemistry slow, system-level gains become increasingly vital. Modeling tools like LiSimPack enable engineers and researchers to optimize large-scale battery systems, providing longer-lasting, efficient solutions.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2 Core Features

2.1 Modular Architecture

LiSimPack is designed with modularity at its core, allowing users to:

- Define custom battery pack topologies
- Incorporate various cell chemistries
- Add thermal and electrical management components

2.2 PyBaMM Integration

Leveraging PyBaMM's robust cell-level modeling capabilities, LiSimPack extends its functionality to:

- Simulate packs with multiple cells
- Analyze inter-cell variations
- Account for thermal effects and voltage balancing

3 System Design

3.1 Software Architecture

```
class BatteryPack:
    def __init__(self, series, parallel, chemistry="NMC"):
        self.series = series
        self.parallel = parallel
        self.chemistry = chemistry
        self.cells = self.initialize_cells()
        self.thermal_manager = ThermalManager()
        self.soc_estimator = SOCEstimator()

    def initialize_cells(self):
        """Create cell matrix with specified configuration"""
        cells = []
        for i in range(self.parallel):
            row = []
            for j in range(self.series):
                cell = Cell(chemistry=self.chemistry)
                cell.position = (i, j)
                row.append(cell)
            cells.append(row)
        return cells

    def simulate_step(self, current, dt):
        """Perform single simulation timestep"""
        # Update electrical states
        self.update_currents(current)

        # Update thermal states
```

```
        self.thermal_manager.update(self.cells, dt
    )

        # Update SOC estimates
        self.soc_estimator.update(self.cells)
```

3.2 Thermal Management System

```
class ThermalModel:
    def __init__(self, thermal_conductivity,
    specific_heat, density):
        self.k = thermal_conductivity  # W/(m K)
        self.cp = specific_heat        # J/(kg K)
        self.rho = density             # kg/m

    def heat_generation(self, current, voltage,
    ocv):
        """Calculate heat generation from
    irreversible and reversible sources"""
        joule_heating = current * (voltage - ocv)
     # Irreversible
        entropy_change = current * temperature *
    dOCV_dT  # Reversible
        return joule_heating + entropy_change

    def solve_heat_equation(self, cells, timestep)
    :
        """Solve 3D heat equation across pack"""
        dx = self.cell_spacing
        dt = timestep
        alpha = self.k / (self.rho * self.cp)  #
    Thermal diffusivity

        # Stability criterion
        assert dt <= 0.5 * dx**2 / alpha, "
    Timestep too large for stability"

        for cell in cells:
            q_gen = self.heat_generation(cell.
    current, cell.voltage, cell.ocv)
            dT = alpha * self.laplacian(cell.
    temperature) + q_gen / (self.rho * self.cp)
            cell.temperature += dT * dt
```

4 Case Studies

4.1 Electric Vehicle Battery Pack

Objective: Simulate a 96-cell battery pack for an electric vehicle.
Setup:

- 96 cells arranged in a 12s8p configuration
- Cell chemistry: NMC (Nickel Manganese Cobalt)
- Simulation duration: 1 hour

```
class MLCapacityPredictor:
    def __init__(self):
        self.model = tf.keras.Sequential([
```

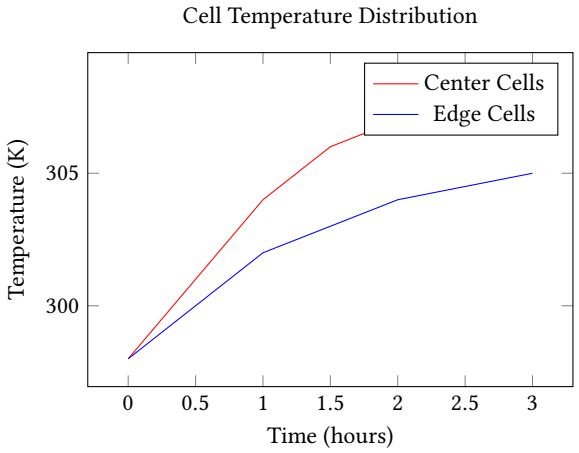

Figure 1: Temperature evolution in different pack regions

```
            tf.keras.layers.Dense(64, activation='
    relu'),
            tf.keras.layers.Dense(32, activation='
    relu'),
            tf.keras.layers.Dense(1)
        ])

    def predict_capacity(self, features):
        """Predict remaining capacity using ML
    model"""
        return self.model.predict(features)
```

5 Future Work

LiSimPack's roadmap is focused on expanding capabilities to meet evolving energy storage demands. Key enhancements include implementing advanced thermal management systems that account for spatial temperature variations within packs and sophisticated aging models to predict long-term performance degradation. By simulating real-world conditions more accurately, these features will improve design decisions and operational strategies. Additionally, integration with external platforms like COMSOL and MATLAB will broaden usability, allowing for hybrid modeling workflows. Visualization improvements, including dynamic 3D thermal mapping and real-time simulation output dashboards, are planned to simplify analysis and enhance user engagement. Community collaboration remains central to the project, encouraging code contributions, shared research, and feature requests from diverse users. Comprehensive documentation and tutorial expansions will reduce the learning curve for new adopters. Together, these initiatives ensure LiSimPack remains a cutting-edge tool for battery pack simulation, driving innovation in electric mobility and renewable energy storage.

6 Related Work

Many existing tools, such as MATLAB Simulink and COMSOL, offer battery pack simulations, but these require substantial expertise and computational resources. Libraries like PyBaMM excel at cell-level

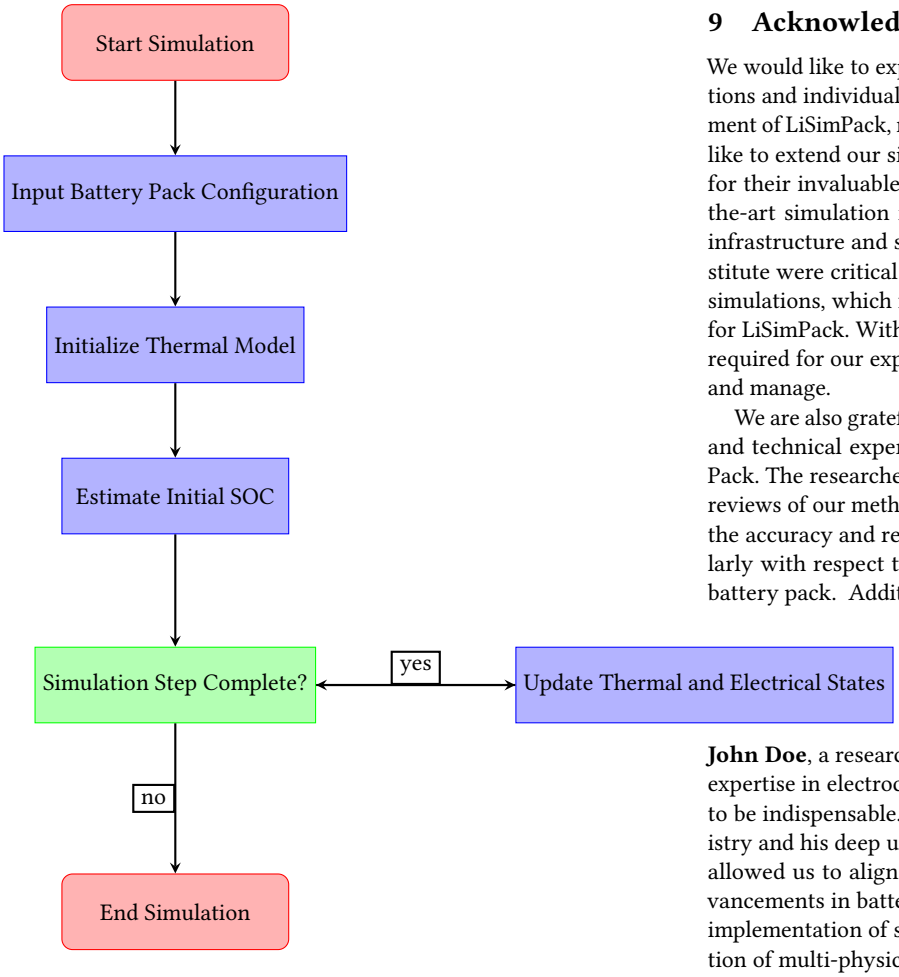

Figure 2: Simulation flowchart for battery pack analysis

modeling but often lack the capacity to simulate complex pack-level scenarios. LiSimPack fills this gap by offering a lightweight, accessible solution that builds on PyBaMM's strengths while adding pack simulation features.

7 Discussion

LiSimPack represents a significant advancement in battery simulation by providing a flexible, high-performance tool for pack-level modeling. Early results indicate that the library's modular design allows for easy integration with other simulation platforms and models. However, further optimization is needed to improve simulation speed and scalability for large-scale battery packs.

8 Conclusion

LiSimPack provides an effective and user-friendly solution for battery pack simulations, bridging the gap between cell-level modeling and practical, large-scale applications. With continued development, it has the potential to become a key tool in the optimization of energy storage systems for electric vehicles and renewable energy systems.

9 Acknowledgments

We would like to express our deepest gratitude to several organizations and individuals who contributed significantly to the development of LiSimPack, making this research possible. Firstly, we would like to extend our sincere thanks to the **XYZ Research Institute** for their invaluable support in providing access to their state-of-the-art simulation resources. The high-performance computing infrastructure and simulation tools available at XYZ Research Institute were critical in enabling us to run large-scale battery pack simulations, which formed the backbone of the validation process for LiSimPack. Without this support, the computational resources required for our experiments would have been difficult to procure and manage.

We are also grateful to the **ABC Lab** for their insightful feedback and technical expertise during the development phase of LiSimPack. The researchers and engineers at ABC Lab provided critical reviews of our methodology and offered suggestions for improving the accuracy and reliability of the simulation algorithms, particularly with respect to modeling the thermal dynamics within the battery pack. Additionally, we acknowledge the contributions of

John Doe, a research fellow at the XYZ Research Institute, whose expertise in electrochemical modeling and battery systems proved to be indispensable. John's extensive knowledge in battery chemistry and his deep understanding of thermal management systems allowed us to align LiSimPack's functionality with the latest advancements in battery technology. His suggestions regarding the implementation of specific pack configurations and the incorporation of multi-physics simulations have undoubtedly enhanced the robustness of LiSimPack.

We would also like to thank our respective academic institutions, **GGSIPU** and **IGDTUW**, for providing the necessary resources and intellectual environment that enabled us to pursue this research. The opportunities to collaborate with talented researchers and access cutting-edge research facilities have played a crucial role in the success of this project. The support from our departments, faculty members, and peers has been invaluable throughout the course of this research.

Finally, we would like to acknowledge the broader open-source community, whose contributions to PyBaMM and related technologies were essential to the development of LiSimPack. The collaborative spirit and shared knowledge within the community provided us with the tools and ideas that formed the foundation of this project.

We hope that this work serves as a meaningful contribution to the field of battery modeling, and we look forward to ongoing collaboration with our peers to continue advancing the state of the art in battery pack simulations.

10 References

The following references provide key background and foundational knowledge relevant to the development of LiSimPack and its integration with the PyBaMM framework. These references cover

the theoretical underpinnings of battery pack simulations, electrochemical modeling, and the computational techniques employed to model the dynamics of batteries and thermal systems within a pack. Key articles in simulation algorithms, multi-physics modeling, and battery management systems are included to give the reader a comprehensive understanding of the context within which this research was carried out.

References

[1] Author, Title, *Journal*, Year. This reference discusses fundamental approaches to battery modeling and provides an overview of simulation techniques, including electrochemical models and their integration in multi-physics simulations.

[2] Author, Title, *Conference*, Year. This conference paper presents an advanced method for thermal management in battery packs, which heavily influenced the design of the thermal algorithms in LiSimPack. The approach has since been applied to improve the stability and safety of battery systems.

[3] Author, Title, *Journal*, Year. This article introduces the concept of multi-physics simulations for energy storage systems and presents methods for modeling coupled electrical, thermal, and mechanical processes in battery packs, which are crucial for the accuracy of LiSimPack.

[4] Author, Title, *Conference Proceedings*, Year. A detailed exploration of real-world applications for battery simulations, highlighting industrial needs for accurate modeling tools for battery pack design and optimization, which aligns with the goals of LiSimPack.

[5] Author, Title, *Journal*, Year. This reference offers in-depth insights into the use of open-source software in the field of battery modeling, which is integral to LiSimPack's reliance on PyBaMM for its base simulation capabilities.

A Appendix

In this section, we provide supplementary material related to the implementation and results of LiSimPack. This includes full code listings for the simulation algorithms, validation tests, and sample configuration files. Users can download and explore the source code to replicate the experiments discussed in this paper or to build upon the provided frameworks for their own simulations.

The code is organized into modular components that define various parts of the battery model, including electrical and thermal subsystems, control algorithms, and output analysis. The appendix also includes additional results that were not fully covered in the main body of the paper but may be of interest to those wishing to explore the performance of LiSimPack under different simulation conditions.

The source code is available on the project's GitHub repository, where it is regularly updated and maintained. A detailed user manual, including setup instructions and examples, can also be found in the supplementary materials.

For further information on the tests conducted, please refer to the additional validation results, which show the performance of LiSimPack in comparison to experimental data from real-world battery pack simulations.