



UNIVERSIDADE FEDERAL DA INTEGRAÇÃO LATINO-AMERICANA
INSTITUTO LATINO AMERICANO DE CIÊNCIAS DA VIDA E DA NATUREZA
PROGRAMA DE PÓS-GRADUAÇÃO EM FÍSICA APLICADA

DISSERTAÇÃO DE MESTRADO

Foz do Iguaçu-Paraná

2024



UNIVERSIDADE FEDERAL DA INTEGRAÇÃO LATINO-AMERICANA
INSTITUTO LATINO AMERICANO DE CIÊNCIAS DA VIDA E DA NATUREZA
PROGRAMA DE PÓS-GRADUAÇÃO EM FÍSICA APLICADA

**Estimativa do Estado de Carga de Baterias de Íons
de Lítio Utilizando Modelos de Rede Neural
Convolutiva Unidimensional Especializados para
Segmentos de Curvas de Carga**

Jhoan Rodrigo Pérez Vargas

Orientador: Prof. Dr. Marcelo Goncalves Honnicke

Co-orientador: Dr. Daniel Augusto Cantane

Foz do Iguaçu-Paraná

2024

Estimativa do Estado de Carga de Baterias de Íons de Lítio Utilizando Modelos de Rede Neural Convolutacional Unidimensional Especializados para Segmentos de Curvas de Carga

Jhoan Rodrigo Pérez Vargas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Física Aplicada da Universidade Federal da Integração Latino-Americana (PPGFISA-UNILA) como parte dos requisitos necessários para obtenção do título de Mestre em Física.

Orientador: Prof. Dr. Marcelo Goncalves Honnicke

Co-orientador: Dr. Daniel Augusto Cantane

Banca Examinadora

Prof. Dr. Marcelo Goncalves Honnicke (Orientador)

Prof. Dr. Carlos Henrique Farias (Membro Externo)

Prof. Dr. Jose Ricardo Cezar Salgado (Membro Interno)

Foz do Iguaçu-Paraná

2024

Catálogo elaborado pelo Setor de Tratamento da Informação
Catálogo de Publicação na Fonte. UNILA - BIBLIOTECA LATINO-AMERICANA - CENTRAL

P438

Pérez Vargas, Jhoan Rodrigo.

Estimativa do estado de carga de baterias de íons de Lítio utilizando modelos de Rede Neural Convolutacional Unidimensional Especializados para segmentos de curvas de carga / Jhoan Rodrigo Pérez Vargas. - Foz do Iguaçu, 2025.

94 f.: il.

Dissertação (Mestrado) - Universidade Federal da Integração Latino-Americana, Instituto Latino-Americano de Ciências da Vida e da Natureza, Pós-Graduação em Física Aplicada. Foz do Iguaçu - PR, 2025.

Orientador: Marcelo Gonçalves Honnicke.

1. Estado de Carga. 2. Baterias de Íons de Lítio. 3. Redes Neurais Convolucionais Unidimensionais. I. Honnicke, Marcelo Gonçalves. II. Título.

CDU 620.1:004

Resumo

ESTIMATIVA DO ESTADO DE CARGA DE BATERIAS DE ÍONS DE LÍTIO UTILIZANDO MODELOS DE REDE NEURAL CONVOLUCIONAL UNIDIMENSIONAL ESPECIALIZADOS PARA SEGMENTOS DE CURVAS DE CARGA

Jhoan Rodrigo Pérez Vargas

Orientador: Prof. Dr. Marcelo Gonçalves Honnicke

Resumo da Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Física Aplicada da Universidade Federal da Integração Latino-Americana (PPGFISA-UNILA), como parte dos requisitos necessários para a obtenção do título de Mestre em Física.

A crescente demanda por técnicas avançadas de gerenciamento de baterias impulsiona a busca por métodos precisos de estimativa do Estado de Carga, do inglês: *State of Charge (SoC)*. Neste estudo, investiga-se o emprego de Redes Neurais Convolucionais Unidimensionais (**1D CNN**) para estimar o **SoC** de baterias de íons de lítio. Para tanto, foi elaborada uma estratégia que utiliza redes **1D CNN** especializadas em segmentos específicos das curvas de carga.

Foram desenvolvidos modelos especializados para três segmentos distintos das curvas de carga, além de um modelo geral para fins comparativos. Cada modelo foi treinado com conjuntos de dados provenientes de ciclos de carga, considerando variações nos tamanhos de entrada e configurações de hiperparâmetros das redes **1D CNN**.

Os resultados indicaram que a abordagem segmentada gera estimativas do **SoC** mais precisas em comparação com as metodologias descritas na literatura. A estratégia aplicada reduziu o Erro Quadrático Médio (**RMSE**) de 3,10% para 2,75%, evidenciando a eficácia da especialização.

Além disso, foi desenvolvida a aplicação `SoC_Tool`, uma ferramenta distribuível que integra processamento de dados, visualização, treinamento de modelos e estimativa do `SoC` por meio de uma interface gráfica. Tal ferramenta facilita a replicação dos resultados e o desenvolvimento de novos modelos a partir de dados adicionais.

Conclui-se que a utilização de modelos `1D CNN` especializados em segmentos das curvas de carga melhora significativamente a precisão na estimativa do `SoC` em baterias de íons de lítio.

Foz do Iguaçu - Paraná

2024

Abstract

ESTIMATION OF STATE OF CHARGE IN LITHIUM-ION BATTERIES USING SPECIALIZED ONE-DIMENSIONAL CONVOLUTIONAL NEURAL NETWORK MODELS FOR CHARGE CURVE SEGMENTS

Jhoan Rodrigo Pérez Vargas

Advisor: Prof. Dr. Marcelo Gonçalves Honnicke

Abstract of the Master's Dissertation presented to the Graduate Program in Applied Physics at the Federal University for Latin American Integration (PPGFISA-UNILA) as part of the requirements for obtaining the degree of Master in Physics.

The growing demand for advanced battery management techniques drives the search for precise methods to estimate the **SoC**. In this study, the use of **1D CNN** is investigated to estimate the **SoC** of lithium-ion batteries. To this end, a strategy was developed that employs **1D CNN** networks specialized in specific segments of the charge curves.

Specialized models were developed for three distinct segments of the charge curves, in addition to a general model for comparative purposes. Each model was trained using datasets obtained from charge cycles, considering variations in input sizes and hyperparameter configurations of the **1D CNN** networks.

The results indicated that the segmented approach yields more accurate **SoC** estimates compared to the methodologies described in the literature. The applied strategy reduced the **RMSE** from 3.10% to 2.75%, demonstrating the effectiveness of the specialization.

Furthermore, the application **SoC_Tool** was developed, a distributable tool that integrates data processing, visualization, model training, and **SoC** estimation through a graphical user interface. This tool facilitates the replication of the results and the development of new models using additional data.

It is concluded that the use of **1D CNN** models specialized in segments of the charge curves significantly improves the accuracy of **SoC** estimation in lithium-ion batteries.

Foz do Iguaçu-Paraná

2024

*“Dedico este trabalho a todos aqueles
que acreditam na força dos sonhos
e no poder da perseverança.
Às noites silenciosas que abrigaram
minhas reflexões mais profundas
e aos dias iluminados por desafios
que me fizeram crescer.*

*Àqueles que caminharam ao meu lado,
e mesmo nos momentos de incerteza,
me deram forças para seguir em frente.*

*Este é o fruto da jornada,
mas é também o começo de muitas outras.*

”

*”Não é o caminho fácil que nos define,
mas as dificuldades que superamos
e a coragem com que enfrentamos
nossos maiores desafios.”*

Desconhecido

Agradecimentos

Gostaria de agradecer a todas as pessoas que, de uma forma ou de outra, contribuíram para a realização deste trabalho:

- Aos meus pais, por todo o amor, apoio e incentivo incondicional ao longo de minha vida e durante essa jornada acadêmica.
- Aos meus irmãos, que sempre estiveram ao meu lado, oferecendo palavras de apoio e companheirismo.
- Aos meus amigos, pela amizade sincera e por sempre estarem prontos a me apoiar nos momentos mais difíceis.
- Ao meu orientador, Prof. Marcelo Gonçalves Honnicke, por toda a orientação, paciência e pelos valiosos conselhos que me ajudaram a crescer como pesquisador e profissional.
- Ao meu coorientador, Doutor Daniel Cantane, e ao meu colega Adalberto Tavares, agradeço pelo suporte, pelas contribuições essenciais e pelo compartilhamento de conhecimentos ao longo deste trabalho.
- Aos professores da banca, Prof. José Ricardo e Prof. Carlos Henrique Farias, pela sua valiosa participação neste processo formativo, pelas sugestões e contribuições que enriqueceram ainda mais este trabalho.

Lista de Siglas

1D CNN	Redes Neurais Convolucionais Unidimensionais
AI	Inteligência Artificial, do inglês: <i>Artificial Intelligence</i>
ANN	Rede Neural Artificial, do inglês: <i>Artificial Neural Network</i>
BMS	Sistema de Gerenciamento de Baterias
CA	Corrente Alternada
CALCE	Centro de Engenharia de Ciclo de Vida Avançado
CC-CV	Carga a Corrente Constante seguida por Tensão Constante, do inglês: <i>Constant Current, Constant Voltage</i>
CNN	Redes Neurais Convolucionais
DL	Aprendizado Profundo, do inglês: <i>Deep Learning</i>
DoD	Profundidade de Descarga, do inglês: <i>Depth of Discharge</i>
EC	Carbonato de Etileno, do inglês: <i>Ethylene Carbonate</i>
EIS	Espectroscopia de Impedância Eletroquímica, do inglês: <i>Electrochemical Impedance Spectroscopy</i>
ESC	Auto-Correção Aprimorada, do inglês: <i>Enhanced Self-Correcting</i>
ESS	Sistema de Armazenamento de Energia, do inglês: <i>Energy Storage System</i>
FC	Camada Totalmente Conectada, do inglês: <i>Fully Connected</i>
FEM	Força Eletromotriz
GUI	Interface Gráfica de Usuário
HDF5	<i>Hierarchical Data Format version 5</i>
HIL	<i>Hardware-in-the-Loop</i>

LCO	Óxido de Lítio-Cobalto, do inglês: <i>Lithium Cobalt Oxide</i>
LIB's	Baterias de Íons de Lítio, do inglês: <i>Lithium Ion Battery</i>
LSTM	Memória de Longo e Curto Prazo, do inglês: <i>Long Short-Term Memory</i>
ML	Aprendizado de Máquina, do inglês: <i>Machine Learning</i>
NASA	Administração Nacional da Aeronáutica e Espaço
OCV	Tensão de Circuito Aberto, do inglês: <i>Open Circuit Voltage</i>
ReLU	Unidade Linear Retificada, do inglês: <i>Rectified Linear Unit</i>
RMSE	Erro Quadrático Médio
RNN	Rede Neural Recorrente, do inglês: <i>Recurrent Neural Network</i>
SEI	Camada de Interfase Sólido-Eletrólito, do inglês: <i>Solid Electrolyte Interphase</i>
SoC	Estado de Carga, do inglês: <i>State of Charge</i>
SoE	Estado de Energia, do inglês: <i>State of Energy</i>
SoH	Estado de Saúde, do inglês: <i>State of Health</i>

Sumário

Introdução	9
1 Baterias de Íons de Lítio e Estimativa do Estado de Carga	11
1.1 Introdução	11
1.2 Fundamentos das Baterias de Íons de Lítio	12
1.2.1 Composição e Estrutura	12
1.2.2 Inovações na Química e Estrutura das Baterias	13
1.2.3 Princípio de Funcionamento	15
1.2.4 Características de Desempenho	16
1.3 Modelagem de Baterias	22
1.3.1 Modelagem Matemática	22
1.3.2 Modelagem Elétrica	24
1.4 Conceito de Estado de Carga (SoC)	28
1.4.1 Definição e Importância do SoC	28
1.4.2 Métodos Convencionais de Estimativa do SoC	29
1.4.3 Limitações dos Métodos Convencionais	32
1.5 Métodos Baseados em Inteligência Artificial para Estimativa do SoC	33
1.5.1 Introdução aos Métodos de AI Aplicados ao SoC	33
1.5.2 Fundamentos Teóricos das Técnicas de AI	34
1.5.3 Conceitos Relevantes para a Otimização de Redes Neurais	37
1.5.4 Estimativa do SoC com técnicas de AI	40
2 Estimando o Estado de Carga (SoC)	42
2.1 Objetivo	42
2.1.1 Estrutura da Rede Neural Convolutacional Unidimensional	42

2.1.2	Estrutura dos Dados de Entrada	43
2.1.3	Segmentação dos Dados e Modelos Especializados	44
2.1.4	Integração dos Modelos Especializados	46
2.2	Resumo das Etapas Metodológicas	46
3	Metodologia	48
3.1	Desenvolvimento de uma Ferramenta de Estimativa de SoC	48
3.2	Coleta e Preparação de Dados	50
3.2.1	Bases de Dados	50
3.2.2	Descrição do Dataset da CALCE	51
3.2.3	Preparação do <i>dataset</i> para Treinamento e Validação	59
3.2.4	Declaração e Configuração dos Modelos 1D CNN	61
3.2.5	Processo de Treinamento dos Modelos e Avaliação de Desempenho	63
3.2.6	Integração e Implementação dos Modelos A, B, C e F para Estimativas de SoC	66
4	Resultados	68
4.1	Aplicação Distribuível para Estimativa de SoC	68
4.1.1	Integração e Fluxo de Trabalho	73
4.1.2	Distribuição e Compatibilidade	73
4.1.3	Feedback e Logs	73
4.2	Comparação dos Modelos com Base nas Métricas de RMSE	73
4.2.1	Seleção dos Melhores Modelos por Segmento	77
4.2.2	Interpretação dos Resultados	78
4.3	Análise das Curvas de Perda de Treinamento e Validação	79
4.3.1	Observações sobre a Convergência dos Modelos	81
4.4	Combinação dos Melhores Modelos e Comparação com o Modelo F	82
4.4.1	Métricas de RMSE Obtidas	82
4.4.2	Análise dos Resultados	83
4.4.3	Interpretação dos Resultados	83
4.4.4	Considerações sobre a Confiabilidade dos Modelos	84

5	Conclusões	85
5.1	Resumo do Trabalho	85
5.2	Principais Contribuições	85
5.3	Limitações do Estudo	86
5.4	Trabalhos Futuros	86
5.5	Considerações Finais	87

Introdução

As baterias de íons de lítio têm se consolidado como a principal tecnologia de armazenamento de energia para uma ampla gama de aplicações, desde dispositivos eletrônicos portáteis até veículos elétricos e sistemas de energia em grande escala. Sua alta densidade de energia, longa vida útil e eficiência as tornam essenciais na transição para fontes de energia mais sustentáveis.

A estimativa precisa do **SoC** é fundamental para garantir a eficiência, segurança e longevidade desses sistemas. Métodos convencionais de estimativa do **SoC**, como a contagem de Coulombs e a tensão em circuito aberto, apresentam limitações significativas, especialmente sob condições operacionais dinâmicas e variáveis.

Com o avanço das técnicas de Inteligência Artificial, do inglês: *Artificial Intelligence (AI)*, abordagens baseadas em Aprendizado de Máquina, do inglês: *Machine Learning (ML)* surgem como alternativas promissoras para a estimativa do **SoC**. Em particular, as **1D CNN** demonstram capacidade de capturar padrões complexos em dados sequenciais, tornando-se ferramentas eficazes para modelar o comportamento das baterias.

Neste trabalho, desenvolvemos uma estratégia para a estimativa precisa do **SoC** de baterias de íons de lítio, utilizando modelos de **1D CNN** especializados para segmentos específicos das curvas de carga. Adaptamos a metodologia proposta por Qian *et al.*, explorando a especialização dos modelos em diferentes regiões das curvas de carga e desenvolvendo modelos especializados para três segmentos distintos (Segmentos A, B e C), capturando características específicas de cada fase do carregamento.

Comparando o desempenho dos modelos especializados com um modelo geral, avaliamos a eficácia da abordagem segmentada na melhoria da precisão das estimativas de **SoC**. Além disso, criamos a ferramenta **SoC_Tool**, uma aplicação distribuível que integra processamento de dados, visualização, treinamento de modelos e estimativa de **SoC**, facilitando a aplicação prática dos modelos desenvolvidos.

Espera-se que este trabalho contribua para o avanço das técnicas de estimativa do SoC em baterias de íons de lítio, demonstrando que a especialização de modelos 1D CNN para segmentos específicos das curvas de carga pode melhorar significativamente a precisão das estimativas. A precisão na estimativa do SoC é fundamental para o gerenciamento eficiente e seguro de baterias de íons de lítio, com implicações diretas na autonomia, vida útil e segurança de dispositivos eletrônicos e veículos elétricos.

Capítulo 1

Baterias de Íons de Lítio e Estimativa do Estado de Carga

1.1 Introdução

As Baterias de Íons de Lítio, do inglês: *Lithium Ion Battery* (**LIB's**) são fundamentais na sociedade moderna, alimentando uma ampla gama de dispositivos eletrônicos portáteis, veículos elétricos e Sistema de Armazenamento de Energia, do inglês: *Energy Storage System* (**ESS**). Sua popularidade deve-se a vantagens significativas sobre outras tecnologias de baterias, como alta densidade de energia, maior tensão nominal, longa vida útil, baixa taxa de autodescarga e eficiência de recarga. Essas características as tornam ideais para aplicações que exigem desempenho elevado em um formato compacto [1].

Com a crescente demanda global por **LIB's**, que deve atingir aproximadamente 4.700 GWh até 2030 [2], a gestão sustentável dessas baterias torna-se cada vez mais crucial. Esse crescimento acelerado, impulsionado principalmente pela eletrificação dos transportes e pela transição para energias renováveis, exige melhorias nas metodologias de estimativa do **SoC** para garantir um uso mais eficiente e seguro. Estimar com precisão o **SoC** não é apenas importante para maximizar a autonomia de dispositivos e veículos, mas também para reduzir o impacto ambiental, pois uma gestão eficiente do **SoC** contribui para a longevidade da bateria e evita desgastes prematuros.

Este capítulo apresenta uma visão geral das **LIB's**, incluindo sua composição, princípio de funcionamento e características de desempenho. Além disso, discute-se a importância do **SoC** e os métodos convencionais e avançados para sua estimativa, com ênfase em técnicas baseadas

em AI que oferecem maior precisão e adaptabilidade em condições operacionais variáveis. Dessa forma, as melhorias na precisão do cálculo de SoC representam não apenas um avanço técnico, mas também um compromisso com a sustentabilidade no contexto da expansão do mercado de baterias.

1.2 Fundamentos das Baterias de Íons de Lítio

1.2.1 Composição e Estrutura

As LIB's são compostas por vários componentes que trabalham em conjunto para armazenar e liberar energia de forma eficiente. Segundo Plett [1], os principais componentes de uma célula de íons de lítio incluem o ânodo, o cátodo, o eletrólito e o separador. A Figura 1.1 ilustra esses componentes e sua disposição em uma célula típica.

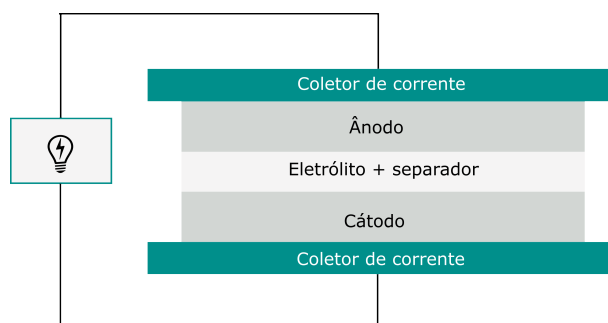


Figura 1.1: Esquema dos componentes principais de uma célula de bateria de íons de lítio, incluindo o ânodo, cátodo, eletrólito e separador. Adaptado de [3].

Ânodo

O ânodo é o eletrodo negativo da bateria durante a descarga e é tipicamente feito de grafite, devido à sua capacidade de armazenar íons de lítio de maneira estável e eficiente. A escolha do material do ânodo influencia diretamente a densidade de energia, a velocidade de carregamento e o ciclo de vida útil da bateria. Existem pesquisas em andamento sobre materiais alternativos para o ânodo, como o silício e o Titanato de Lítio, visando melhorar estes aspectos [1].

Cátodo

O cátodo é o eletrodo positivo da bateria durante a descarga e é geralmente composto de óxidos de lítio com metais de transição, como cobalto, níquel e manganês. Este componente define a capacidade e a tensão da célula. Diferentes combinações de materiais para o cátodo podem afetar o desempenho da bateria em termos de estabilidade térmica, capacidade de carga e durabilidade. Por exemplo, o Óxido de Lítio-Cobalto, do inglês: *Lithium Cobalt Oxide* (LCO) oferece alta densidade de energia e é comum em dispositivos portáteis, enquanto o Fosfato de Ferro-Lítio é popular em veículos elétricos e sistemas de armazenamento de energia pela sua maior segurança e estabilidade.

Eletrólito

O eletrólito é o meio pelo qual os íons de lítio se movem entre o ânodo e o cátodo durante os processos de carga e descarga. Geralmente é composto por uma solução de sais de lítio dissolvidos em solventes orgânicos, como o Carbonato de Etileno, do inglês: *Ethylene Carbonate* (EC) e o Carbonato de Dimetila. O eletrólito deve ter alta condutividade iônica e estabilidade eletroquímica, garantindo uma transferência eficiente de íons sem reagir diretamente com os eletrodos. A composição do eletrólito afeta o desempenho, a segurança e a durabilidade da bateria, e avanços nesse campo buscam reduzir riscos de inflamabilidade e melhorar a resistência a temperaturas extremas.

Separador

O separador é uma membrana permeável que evita o contato direto entre o ânodo e o cátodo, prevenindo curtos-circuitos dentro da célula [1]. Ele permite a passagem de íons de lítio através de seus poros durante o funcionamento da bateria. Geralmente feito de polímeros, como o polietileno ou polipropileno, o separador deve ser fino o suficiente para permitir a livre movimentação de íons, mas resistente o suficiente para suportar as condições eletroquímicas e térmicas durante a operação.

1.2.2 Inovações na Química e Estrutura das Baterias

De acordo com Fleischmann *et al.* [2], avanços significativos estão sendo feitos na composição e estrutura das LIB's para aumentar a densidade de energia, melhorar a segurança e prolongar a

vida útil das baterias. Essas inovações são impulsionadas pela crescente demanda por baterias de maior desempenho e menor impacto ambiental.

Materiais de Ânodo Avançados

Pesquisas recentes têm direcionado esforços para o desenvolvimento de ânodos à base de silício e lítio metálico como alternativas promissoras ao grafite convencional. O ânodo de silício destaca-se por sua capacidade teórica de armazenar íons de lítio muito superior, o que pode aumentar significativamente a densidade energética das baterias. Contudo, este material enfrenta desafios críticos de estabilidade estrutural, sobretudo devido à expansão volumétrica durante os ciclos de carga e descarga, comprometendo a durabilidade do eletrodo [4]. Por outro lado, os ânodos de lítio metálico apresentam uma densidade energética elevada, mas sua aplicação requer soluções para inibir a formação de dendritos, que podem representar sérios riscos à segurança dos dispositivos [5].

Baterias de Estado Sólido

A introdução de baterias de estado sólido tem sido apontada como uma inovação significativa, pois a substituição do eletrólito líquido por um sólido elimina os riscos de vazamentos e, consequentemente, melhora a segurança do sistema. Essa abordagem não só possibilita a obtenção de densidades energéticas superiores, como também aprimora a estabilidade térmica e eletroquímica das baterias, fatores essenciais para aplicações em veículos elétricos, onde restrições de espaço e peso são determinantes [2]. Ademais, a tecnologia de estado sólido promete aumentar a vida útil dos dispositivos e oferecer um desempenho consistente mesmo sob condições extremas de temperatura [6].

Revestimento Seco de Eletrodos

Uma inovação relevante na fabricação de baterias de íon de lítio é a adoção do revestimento seco para os eletrodos. Essa técnica elimina a necessidade de solventes orgânicos e dispensa o uso de fornos para secagem, reduzindo significativamente o consumo de energia durante o processo de produção. Como consequência, há uma diminuição do impacto ambiental e dos custos, o que viabiliza a produção de baterias mais sustentáveis e econômicas. Além disso, o método contribui para uma menor pegada de carbono na fabricação das LIB's [7].

Design Integrado e Cell-to-Pack

O design integrado, incluindo o conceito de **Cell-to-Pack**, elimina componentes intermediários entre as células e o módulo da bateria, resultando em uma estrutura mais leve e densa em energia. Essa abordagem maximiza a eficiência do espaço e do peso, tornando-se ideal para aplicações que demandam baterias compactas com alta capacidade. A integração direta das células ao módulo aumenta a densidade energética total e reduz a complexidade e o custo de produção [2].

1.2.3 Princípio de Funcionamento

O princípio de funcionamento das **LIB's** envolve a movimentação de íons de lítio entre o ânodo e o cátodo durante os processos de carga e descarga. Esse movimento iônico ocorre através do eletrólito e é fundamental para a armazenagem e liberação de energia. Diferente das baterias que dependem de reações químicas que alteram a composição dos eletrodos, as **LIB's** utilizam um processo de intercalação, no qual os íons de lítio são inseridos ou removidos das estruturas cristalinas dos materiais dos eletrodos sem modificar sua composição química. Esse processo resulta em maior durabilidade e eficiência energética [1].

Durante a descarga, os íons de lítio deixam o ânodo e migram através do eletrólito até o cátodo, enquanto os elétrons fluem pelo circuito externo, fornecendo energia ao dispositivo. Durante a carga, o processo é inverso: uma fonte externa de energia força os íons de lítio a se moverem do cátodo de volta para o ânodo, restaurando o estado carregado da bateria. A Figura 1.2 ilustra o mecanismo de intercalação de íons de lítio no ânodo de grafite [3].

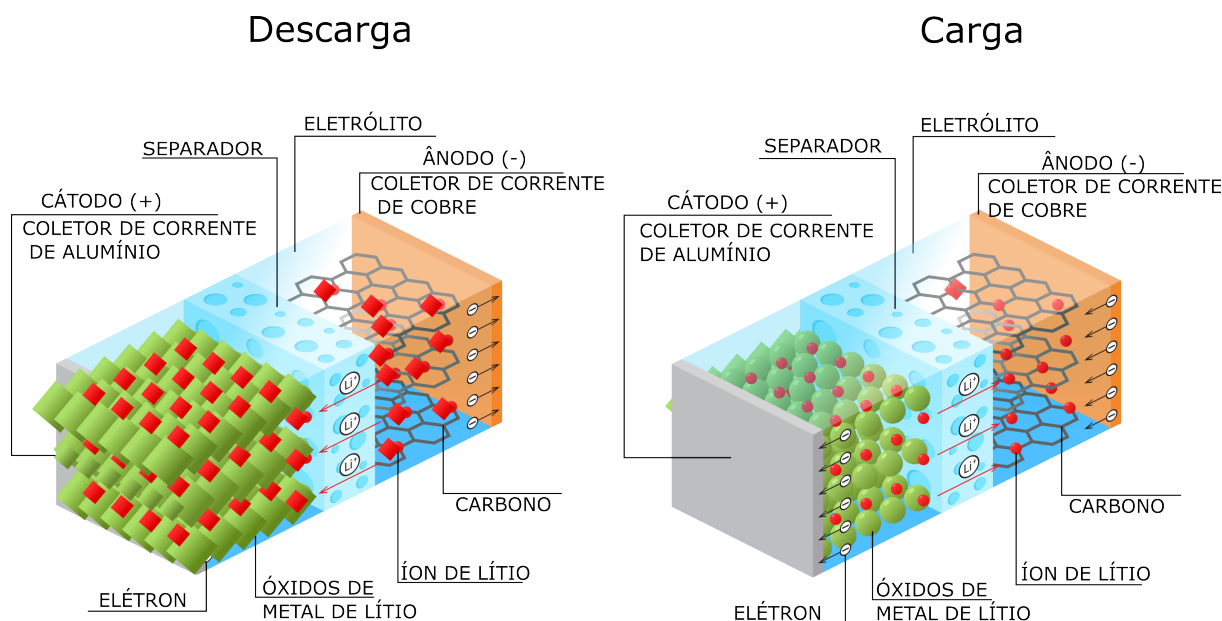


Figura 1.2: Mecanismo de intercalação de íons de lítio no ânodo de grafite durante os processos de carga e descarga em LIB's, adaptado de Fathi (2020) [3].

Esse processo permite que as LIB's alcancem altas densidades de energia e eficiência, tornando-as ideais para aplicações que exigem desempenho elevado em um formato compacto. No entanto, os materiais ativos dos eletrodos – bem como as interfaces entre eletrodos e eletrólito (incluindo a camada de Camada de Interação Sólido-Eletrólito, do inglês: *Solid Electrolyte Interphase (SEI)*) – estão sujeitos à degradação ao longo do tempo, o que pode impactar a capacidade de carga e, conseqüentemente, a vida útil da bateria.

1.2.4 Características de Desempenho

Capacidade e Ciclos de Vida

A capacidade de carga de uma LIB's refere-se à quantidade total de energia que ela pode armazenar e fornecer ao longo de seu ciclo de vida. Expressa em miliampère-hora (mAh) ou ampère-hora (Ah), a capacidade depende dos materiais e da arquitetura dos eletrodos, além da eficiência do eletrólito. Em uma célula nova, a capacidade nominal especificada pelo fabricante representa a carga máxima que a bateria pode armazenar sob condições ideais. Com o uso e à medida que a bateria é ciclada, essa capacidade diminui gradualmente devido a processos de degradação internos [1].

Ciclos de Carga e Descarga: Um ciclo de carga e descarga é completado quando a bateria é totalmente carregada e descarregada. Para analisar o desempenho da bateria durante esses ciclos, utiliza-se a técnica de carga e descarga galvanostática, que envolve a aplicação de uma corrente constante durante os processos de carga e descarga dentro de uma faixa de potencial segura. Essa técnica permite avaliar a eficiência coulumbiana da bateria, que indica a relação entre a carga extraída e a carga inserida em cada ciclo. Uma eficiência coulumbiana próxima a 100% sugere que a bateria está operando com poucas perdas ao longo dos ciclos [3].

A Figura 1.3 ilustra um exemplo de ciclo galvanostático para uma célula prismática de íons de lítio com cátodo de Óxido de Lítio-Cobalto (LiCoO_2). A célula possui uma capacidade nominal de 1100 mAh, e o gráfico de tensão versus tempo mostra as características de tensão ao longo dos ciclos.

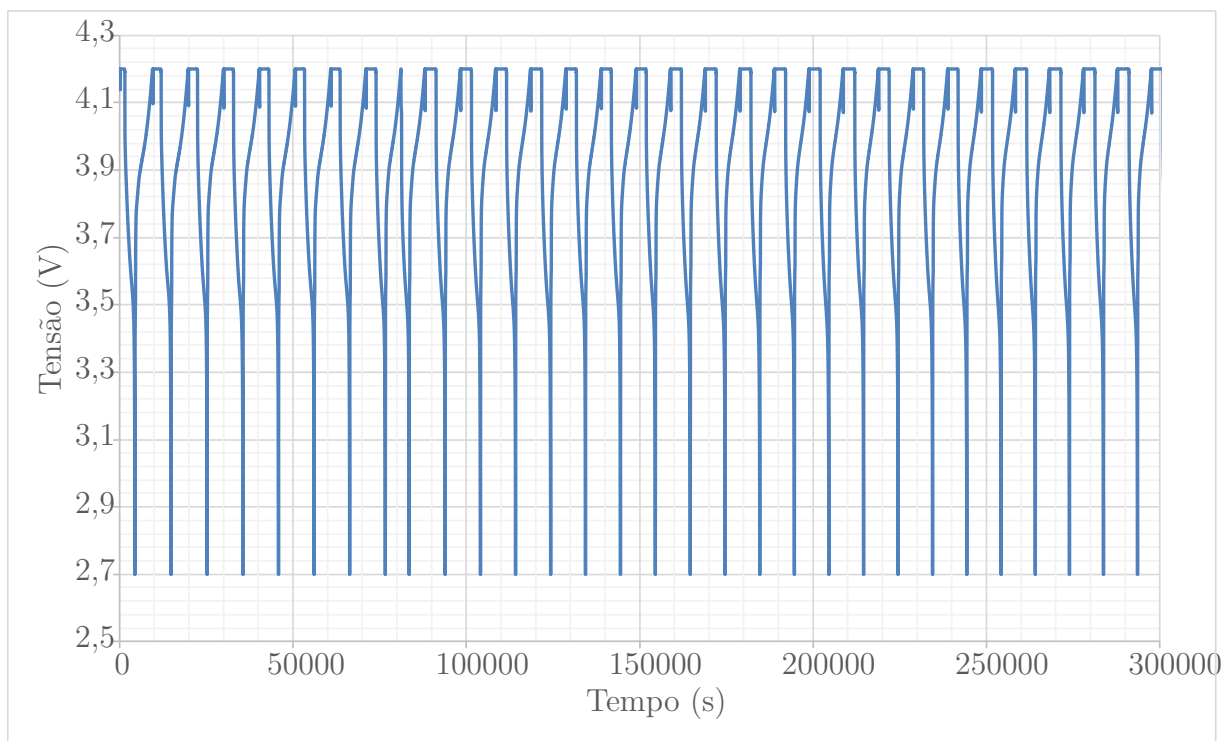


Figura 1.3: Ciclo galvanostático de uma célula prismática de íons de lítio com cátodo de LiCoO_2 , capacidade nominal de 1100 mAh, em uma janela de potencial de 2,7 V a 4,2 V a uma taxa de 1C.

Curvas de Carga e Descarga

As curvas de carga e descarga das LIB's representam a relação entre tensão, corrente e tempo durante os processos de carga e descarga, fornecendo informações detalhadas sobre o comportamento e o estado da bateria. Essas curvas são amplamente utilizadas para monitorar o estado de carga (SoC) e o Estado de Saúde, do inglês: *State of Health* (SoH), permitindo que pesquisadores e engenheiros analisem o desempenho e o ciclo de vida sob diferentes condições operacionais [3].

Para gerar essas curvas de maneira precisa e repetível, os cicladores de baterias são configurados com parâmetros programáveis, permitindo o ajuste da corrente e da tensão em cada ciclo. Com isso, é possível realizar cargas e descargas controladas, reproduzindo perfis de ciclagem específicos que incluem:

- **Carga a Corrente Constante (CC) seguida de Tensão Constante (CV):** A bateria é carregada a uma corrente constante até alcançar uma tensão máxima predeterminada, seguida por uma fase de tensão constante até que a corrente diminua a um valor limite. Esse perfil permite observar o aumento gradual de tensão e seu efeito sobre o SoC.
- **Descarga a Corrente Constante:** A bateria é descarregada a uma corrente constante até atingir uma tensão mínima predeterminada, ajudando a entender como o SoC diminui ao longo do tempo e fornecendo dados importantes para a curva de descarga [3].

Comportamento das Curvas de Tensão ao Longo do SoC

Durante a carga e descarga de uma bateria de íons de lítio, a tensão varia de forma não linear, apresentando regiões distintas com comportamentos específicos. No início da descarga, a tensão cai rapidamente, seguida por uma fase de declínio mais estável onde o SoC se mantém relativamente constante e, finalmente, uma nova queda rápida na tensão. Durante a carga, observa-se o inverso: uma rápida elevação da tensão no início, seguida por uma fase mais estável e, por fim, um aumento novamente acentuado à medida que a bateria se aproxima da plena carga. Esse padrão em três etapas detalha o SoC em diferentes condições de operação, revelando a capacidade de retenção de carga em SoCs altos ou baixos, a taxa de variação intermediária e a resistência interna da bateria.

A Figura 1.4 apresenta a curva de tensão durante a descarga da célula LTO/NMC sob diferentes taxas de corrente (C-rates). Observa-se que, com o aumento da taxa de descarga, a célula sofre um aumento na polarização, o que se reflete na queda de tensão. Esse comportamento

limita a capacidade utilizável em altas potências, afetando a eficiência da bateria. Já a Figura 1.5 mostra o comportamento da tensão durante a carga, onde a polarização é comparável àquela observada na descarga. Entretanto, a capacidade carregada é consideravelmente menor do que a descarregada, devido à limitação imposta pela curva de potencial de circuito aberto dos materiais ativos. Esses gráficos fornecem uma visão clara das limitações da célula LTO/NMC em diferentes condições de operação.[8]

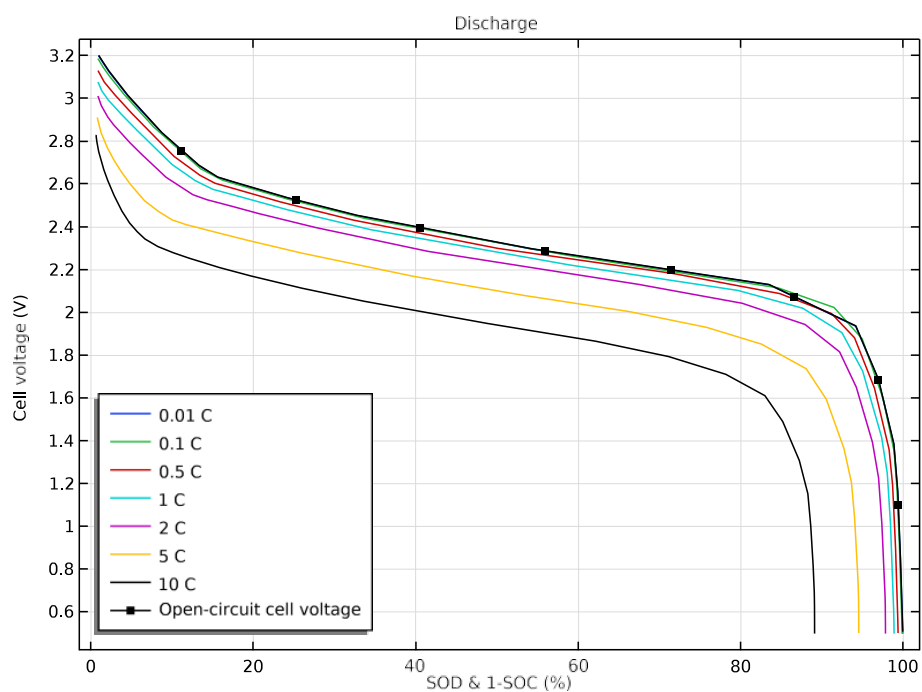


Figura 1.4: Curva de tensão durante a descarga da célula LTO/NMC sob diferentes taxas de corrente (C-rates). Fonte: COMSOL Multiphysics[8].

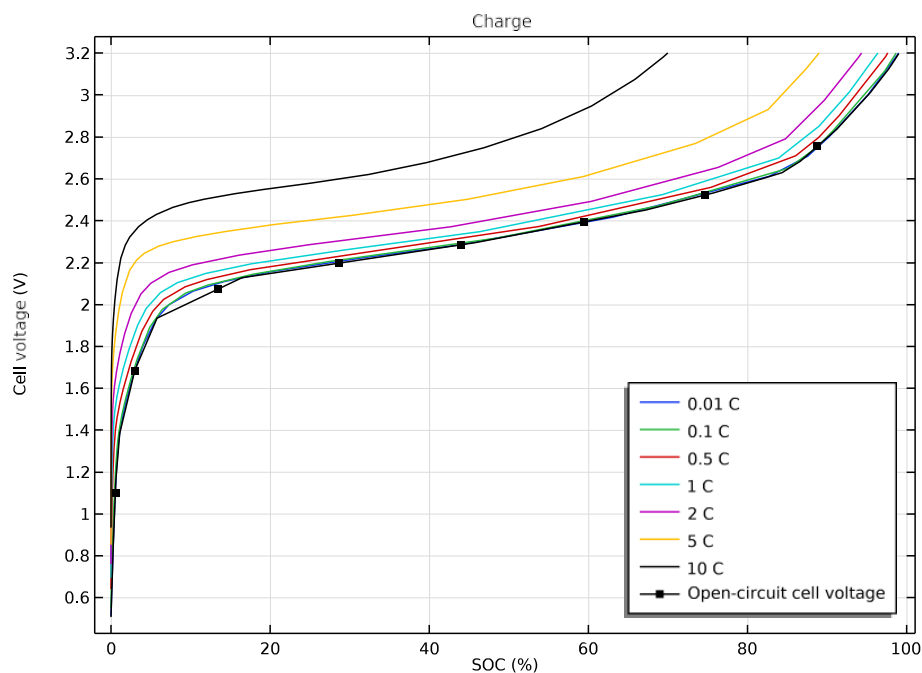


Figura 1.5: Curva de tensão durante a carga da célula LTO/NMC sob diferentes taxas de corrente (C-rates). Fonte: COMSOL Multiphysics[8].

A segmentação das curvas de tensão permite extrair informações relevantes para modelos de estimativa de SoC, pois cada segmento reflete como a bateria responde em diferentes estados de carga. Isso possibilita que os modelos analisem padrões específicos de cada faixa de SoC, melhorando a precisão das previsões em variadas condições operacionais.

Informações Obtidas a Partir das Curvas:

- **Declínio de Tensão em SoC Baixo:** Os segmentos finais da descarga mostram uma rápida queda na tensão, indicando a proximidade do esgotamento da bateria.
- **Estabilidade de Tensão em SoC Intermediários:** Em SoC intermediários, a tensão se mantém estável, fornecendo dados sobre a eficiência e estabilidade durante o uso regular.
- **Resistência Interna:** A taxa de variação da tensão em SoCs altos e baixos permite inferir a resistência interna, que tende a aumentar com o envelhecimento da bateria [9].

Fatores que Afetam o Desempenho

Diversos fatores contribuem para a degradação da capacidade e a redução da vida útil das LIB's:

Camada SEI A camada de interface sólido-eletrolítica (SEI) é uma fina camada formada sobre o ânodo durante as primeiras cargas, resultante da decomposição do eletrólito. Essa camada desempenha um papel crucial na proteção do ânodo e na estabilização das reações químicas, evitando a decomposição contínua do eletrólito. Contudo, o crescimento progressivo da SEI ao longo dos ciclos de carga e descarga aumenta a resistência interna e consome lítio ativo, comprometendo a capacidade total e a eficiência da bateria. Um modelo teórico que descreve quantitativamente essa formação e seu impacto na degradação é apresentado por Pinson e Bazant [10].

Temperatura Altas temperaturas aceleram reações químicas internas, causando degradação mais rápida dos materiais do ânodo, cátodo e eletrólito. Além disso, promovem o crescimento da SEI sobre o ânodo, aumentando a resistência interna e consumindo lítio ativo, o que reduz a capacidade total. Baixas temperaturas diminuem a condutividade do eletrólito, reduzindo a capacidade de carga e a eficiência de transferência de íons. Para minimizar o impacto da temperatura, recomenda-se operar as baterias dentro da faixa térmica ideal especificada pelo fabricante [1].

Taxa de Carga e Descarga A taxa de carga e descarga (C-RATE) afeta significativamente a capacidade e a vida útil da bateria. Altas taxas podem causar aquecimento excessivo, acelerando a degradação dos materiais internos e promovendo o crescimento de dendritos de lítio [3]. Esse fenômeno aumenta a resistência interna e compromete a segurança e o desempenho da bateria ao longo do tempo.

Profundidade de Descarga A Profundidade de Descarga, do inglês: *Depth of Discharge* (DoD) é a porcentagem da capacidade total descarregada em cada ciclo. Descarregar a bateria até o limite reduz sua vida útil em comparação com uma operação em que a profundidade de descarga é limitada (por exemplo, 20–80% da capacidade total). Operar com profundidade de descarga moderada é uma estratégia recomendada para prolongar a vida útil das LIB's [1].

Envelhecimento e Degradação dos Materiais O envelhecimento das baterias resulta em processos irreversíveis que comprometem a eficiência e a vida útil do dispositivo. No ânodo, o repetido intercalamento de íons de lítio promove o crescimento da camada de SEI – que consome lítio ativo e, aliado a estresses mecânicos, pode gerar microfaturas no material – reduzindo a

capacidade disponível. No cátodo, a dissolução dos materiais ativos e as alterações estruturais diminuem a capacidade de armazenamento e aumentam a resistência interna, afetando o desempenho global. Assim, os mecanismos de degradação dos eletrodos atuam em conjunto para reduzir a durabilidade das LIB's [1].

A compreensão dos fatores que afetam a capacidade e o ciclo de vida das LIB's permite desenvolver estratégias de gerenciamento de baterias, incluindo técnicas avançadas de estimativa do SoC. Essas estratégias ajudam a prolongar a vida útil, melhorar a segurança e otimizar o desempenho das baterias em diferentes aplicações [1].

1.3 Modelagem de Baterias

1.3.1 Modelagem Matemática

A modelagem matemática das baterias permite compreender os processos eletroquímicos que ocorrem em seu interior e prever seu comportamento em diferentes condições operacionais. Essa modelagem frequentemente se beneficia da teoria de soluções concentradas, que considera as interações entre todas as espécies presentes no sistema, conforme demonstrado em diversos estudos [11, 12, 13, 14].

Essa teoria se baseia na equação de difusão multicomponente de Stefan-Maxwell, a qual descreve o transporte de espécies impulsionado pelo gradiente do potencial [11, 14, 15, 16]. A equação (1.1) apresentada a seguir descreve esse processo de transporte, fundamental para a compreensão do comportamento das baterias.

$$c_k \nabla \mu_k = \sum_{k \neq j} K_{kj} (v_j - v_k) = RT \sum_{k \neq j} \frac{c_k c_j}{c_T D_{kj}} (v_j - v_k) \quad (1.1)$$

Em (1.1), c_k representa a concentração da espécie k , μ_k o potencial eletroquímico da espécie k , e K_{kj} os coeficientes de atrito que quantificam as interações entre as espécies k e j . O termo v_k denota a velocidade da espécie k na fase intersticial em relação a uma velocidade de referência, enquanto D_{kj} representa o coeficiente de difusão que expressa as interações entre as espécies k e j . A concentração total das espécies é dada por $c_T = \sum_k c_k$.

É importante destacar que o coeficiente de atrito pode ser diretamente relacionado às propriedades de transporte, como os coeficientes de difusão, o número de transferência (t_k^0) e a condutividade iônica/elétrica do eletrólito (κ) [16].

Em uma solução composta por um sal binário e um polímero, o transporte de espécies pode ser descrito por duas equações independentes. Considerando o polímero como espécie de referência com velocidade nula, as expressões que governam o transporte do cátion e do ânion do sal são dadas pelas equações (1.2) e (1.3), respectivamente. Essa simplificação, que assume a imobilidade do polímero, é frequentemente empregada na modelagem de eletrólitos poliméricos, permitindo uma análise mais concisa do transporte iônico no sistema.

$$N_+ = -\nu_+ D \nabla c + \frac{j t_+^0}{z_+ F} \quad (1.2)$$

$$N_- = -\nu_- D \nabla c + \frac{j t_-^0}{z_- F} \quad (1.3)$$

A análise do balanço material do sal no polímero, crucial para a compreensão da dinâmica do sistema, leva à equação (1.4), que descreve a evolução da concentração do sal ao longo do tempo. Essa equação incorpora os processos de transporte e reação que afetam a distribuição do sal no eletrólito polimérico, fornecendo uma ferramenta fundamental para o estudo do comportamento da bateria.

$$\frac{\partial c}{\partial t} + \nabla \cdot (c v_0) = \nabla \cdot \left[D \left(1 - \frac{d \ln c_0}{d \ln c} \right) \nabla c \right] - \frac{j \nabla t_+^0}{z_+ \nu_+ F} \quad (1.4)$$

Admitindo que a concentração do solvente não seja função da concentração do eletrólito, o que implica em um volume parcial molar nulo para o eletrólito e uma relação linear entre a densidade da solução e a concentração do sal [12, 17], a equação (1.4) pode ser simplificada. Nessa expressão, o operador matemático divergência ($\nabla \cdot$) é utilizado, resultando em um escalar que representa o fluxo líquido de massa do sal em um ponto do sistema.

A partir da introdução de um eletrodo de referência, é possível obter a densidade de corrente em função da tensão no separador, conforme a equação (1.5) [12, 13, 17]. Essa relação entre densidade de corrente e tensão é fundamental para a caracterização do comportamento elétrico da bateria.

$$j_2 = -\kappa \nabla \Phi_2 - \frac{\kappa R T}{F} \left(1 + \frac{\partial \ln f_{\pm}}{\partial \ln c} \right) \left(\frac{1}{n \nu_+} + \frac{t_+^0}{z_+ \nu_+} \right) \nabla \ln c \quad (1.5)$$

Como uma abordagem alternativa, os Modelos Baseados em Física têm sido empregados para descrever com maior detalhe os processos eletro-químicos internos que ocorrem dentro das células. Ao contrário dos modelos baseados em circuitos equivalentes, esses modelos utilizam

equações diferenciais parciais para representar variáveis eletroquímicas internas, como a concentração de lítio no eletrólito e nos eletrodos, bem como os potenciais elétricos ao longo do sistema [18]. Essas variáveis são resolvidas em um modelo acoplado que captura o transporte de íons e o equilíbrio de carga nas diferentes regiões da célula, incluindo os eletrodos e o eletrólito.

Apesar da maior precisão em prever o comportamento da célula em uma ampla gama de condições operacionais, os modelos baseados em física demandam um custo computacional elevado e podem apresentar problemas de convergência, especialmente com variações abruptas de corrente [18]. Assim, na prática, utiliza-se frequentemente uma versão simplificada desses modelos, conhecida como modelos de ordem reduzido, para melhorar a viabilidade computacional. Essa abordagem permite equilibrar a necessidade de detalhes sobre o estado eletroquímico interno com a eficiência computacional exigida para simulações mais rápidas e robustas.

1.3.2 Modelagem Elétrica

Em contraste com os modelos matemáticos e eletroquímicos, os modelos elétricos apresentam a vantagem de serem mais intuitivos e de fácil aplicação. Baseados em circuitos equivalentes compostos por elementos passivos como resistores, capacitores e, em alguns casos, indutores, além de uma fonte de tensão, esses modelos oferecem uma representação simplificada do comportamento da bateria. Aplicações práticas desses modelos em diferentes químicas de baterias de íon-lítio indicam que a escolha do modelo adequado depende da química da bateria e do perfil de corrente envolvido, com resultados satisfatórios em termos de precisão e simplicidade para gestão de baterias em sistemas reais [19]

Modelo de Resistência Interna

O modelo de resistência interna, ilustrado na Figura 1.6, é amplamente utilizado para representar o comportamento elétrico de baterias. Esse modelo consiste em uma fonte de tensão ideal com Força Eletromotriz (FEM) V_0 , que representa a Tensão de Circuito Aberto, do inglês: *Open Circuit Voltage (OCV)*, em série com uma resistência interna equivalente constante R_{int} . A tensão nos terminais da bateria, denotada por $v(t)$, é a diferença entre a FEM e a queda de tensão na resistência interna [20, 21].

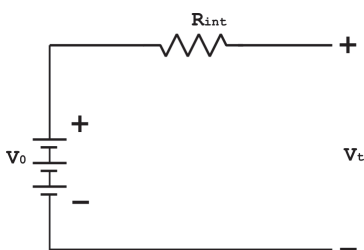


Figura 1.6: Circuito Equivalente

As equações para o cálculo são as seguintes:

$$v(t) = OCV(z(t)) - I(t)R_{\text{int}} \quad (1.6)$$

$$\frac{dz}{dt} = -\frac{I(t)}{Q} \quad (1.7)$$

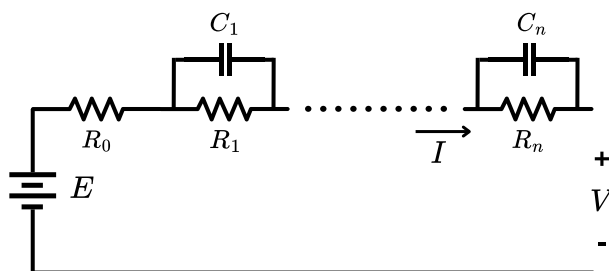
$$z(t) = z(t_0) - \frac{1}{Q} \int_{t_0}^t I(\tau) d\tau \quad (1.8)$$

Este modelo é suficiente para diversas simulações envolvendo sistemas simples; porém, outros componentes elétricos devem ser incorporados para representar fenômenos mais complexos [20].

Modelos de Thévenin

Visando aprimorar a precisão do modelo de resistência interna, comumente se adicionam pares RC (Resistor-Capacitor) em série com R_{int} . Essa adição permite a modelagem dos efeitos dinâmicos da bateria, capturando o comportamento transiente da tensão e corrente em resposta a variações de carga [22]. Esse tipo de circuito é conhecido como modelo de Thévenin de n pares RC, onde o número n de pares está diretamente relacionado à aplicação desejada [22, 21].

A Figura 1.7 apresenta a representação do modelo Thévenin com n pares RC.

Figura 1.7: Circuito Equivalente Thévenin com n pares RC

A solução deste modelo é determinada pela seguinte expressão:

$$v(t) = E - R_0 I - \sum_{k=1}^n R_k I \left(1 - e^{-\frac{t}{R_k C_k}}\right) \quad (1.9)$$

Este modelo tem aplicação prática em simulações de sistemas de baterias, como demonstrado em [23], onde um modelo de Thévenin com múltiplos pares RC foi implementado no MATLAB/Simulink para capturar com precisão o comportamento dinâmico de uma bateria de íons de lítio. Outra aplicação é apresentada por You *et al.* [24], que utilizaram modelos de Thévenin com um e três pares RC para analisar o comportamento de descarga de baterias de íons de lítio, concluindo que o modelo com três pares oferece maior precisão na simulação da tensão ao longo do tempo em comparação com dados experimentais, especialmente em condições de descarga constante.

Modelo Auto-Correção Aprimorada (ESC)

Para melhorar a precisão das previsões em condições de repouso e corrente constante, uma variação dos modelos de circuito equivalente é o modelo Auto-Correção Aprimorada, do inglês: *Enhanced Self-Correcting (ESC)*, ilustrado na Figura 1.8. Esse modelo inclui elementos adicionais para capturar os efeitos de histerese e difusão, fenômenos físicos que podem ocorrer em diversas químicas de baterias, incluindo as de íons de lítio [18].

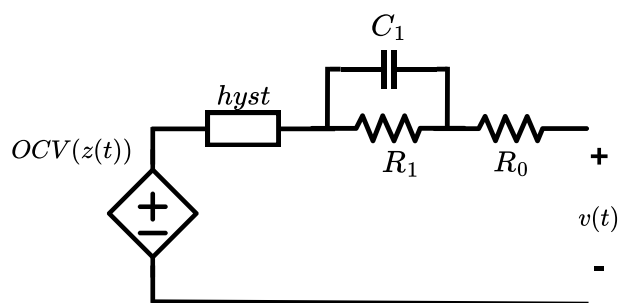


Figura 1.8: Modelo de Circuito Equivalente ESC.

No modelo ESC, a tensão terminal v_k no índice de tempo discreto k é expressa por:

$$v_k = OCV(z_k) + M h_k + M_0 s_k - \sum_i R_i i_{R,i,k} - R_0 i_k, \quad (1.10)$$

onde $OCV(z_k)$ é a tensão de circuito aberto como uma função do estado de carga z_k ; M é a tensão máxima de histerese nesta temperatura; M_0 é a histerese instantânea; e R_0 é a resistência ôhmica da célula.

A evolução do SoC e da corrente de difusão são dadas por:

$$z_{k+1} = z_k - \frac{\eta i_k \Delta t}{Q}, \quad (1.11)$$

$$i_{R,i,k+1} = e^{\left(-\frac{\Delta t}{R_i C_i}\right) i_{R,i,k} + \left(1 - e^{\left(-\frac{\Delta t}{R_i C_i}\right)}\right) i_k}, \quad (1.12)$$

onde $i_{R,i,k}$ representa a corrente nos pares de difusão, e Δt é o intervalo de tempo entre amostras. A constante adimensional η é a eficiência coulômbica da célula.

Para modelar a tensão de histerese, a seguinte equação é utilizada:

onde γ ajusta a rapidez com que o estado de histerese muda com a variação do SoC da célula, e $\text{sgn}(i_k)$ é a função sinal.

Este modelo é indicado para captar a dinâmica transiente e os efeitos de histerese em ciclos de carga e descarga de baterias, proporcionando uma simulação robusta e ajustável [18].

Modelos Baseados em Impedância

Os modelos baseados em impedância, como ilustrado na Figura 1.9, empregam a técnica de Espectroscopia de Impedância Eletroquímica, do inglês: *Electrochemical Impedance Spectroscopy* (EIS) para obter um modelo equivalente em Corrente Alternada (CA) no domínio da frequência [22]. A partir dos espectros de impedância obtidos experimentalmente, uma rede equivalente complexa (Z_{ac}) é ajustada aos dados. Essa rede, composta pela combinação de n pares RC, busca reproduzir o comportamento da impedância da bateria em função da frequência, comumente representado no diagrama de Nyquist.

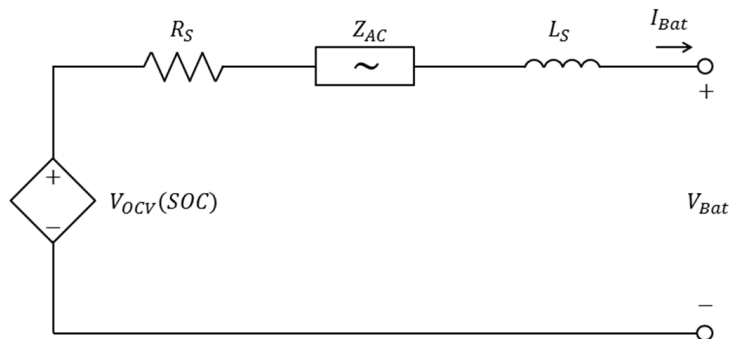


Figura 1.9: Circuito Equivalente Baseado na Impedância

Na Figura 1.10 são apresentados os resultados obtidos experimentalmente, exibidos em gráficos de Nyquist para diferentes condições experimentais de SoC a uma temperatura de 25°C [25].

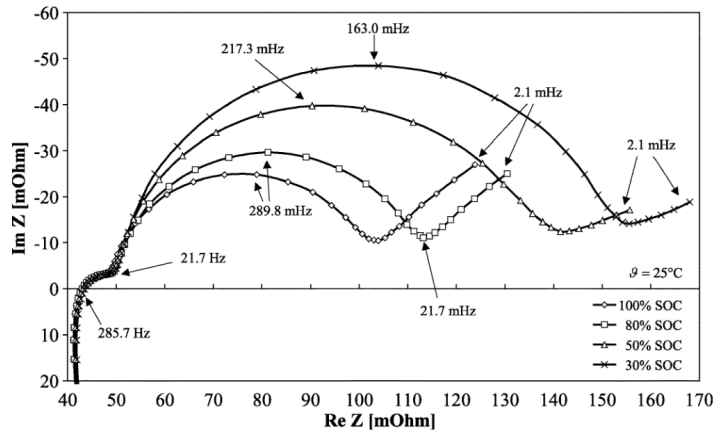


Figura 1.10: Espectros de impedância de uma bateria

Embora capazes de capturar a resposta dinâmica da bateria em CA, o processo de ajuste da rede equivalente é complexo e requer conhecimento especializado. Além disso, os modelos de impedância são válidos apenas para condições específicas de SoC, SoH e temperatura [26], o que limita sua aplicabilidade em condições operacionais variáveis.

1.4 Conceito de Estado de Carga (SoC)

1.4.1 Definição e Importância do SoC

O SoC de uma bateria representa a quantidade de carga restante em relação à sua capacidade máxima, expressa como uma porcentagem. Em termos práticos, o SoC informa o nível de energia disponível na bateria, sendo semelhante a um "indicador de combustível" para dispositivos elétricos. A definição matemática do SoC pode ser representada como:

$$SoC(t) = \frac{Q_{rem}(t)}{Q_{max}} \times 100\% \quad (1.13)$$

onde:

- $Q_{rem}(t)$ é a carga remanescente na bateria no instante t ;

- Q_{\max} é a capacidade nominal total da bateria.

O SoC varia de 0% (bateria totalmente descarregada) a 100% (bateria totalmente carregada).

A estimativa precisa do SoC é essencial para várias razões:

- **Previsão da Autonomia:** Permite prever quanto tempo a bateria pode continuar fornecendo energia antes de precisar ser recarregada.
- **Otimização do Desempenho:** Ajuda a gerenciar a energia de forma eficiente, evitando quedas de desempenho em dispositivos elétricos.
- **Segurança:** Evita condições de sobrecarga ou descarga profunda que podem danificar a bateria ou representar riscos de segurança.
- **Prolongamento da Vida Útil:** O monitoramento adequado do SoC contribui para práticas de carregamento que prolongam a vida útil da bateria.

Em aplicações práticas, o SoC é monitorado constantemente para que a bateria opere dentro de uma faixa recomendada (geralmente entre 20% e 80%) para preservar sua integridade e prolongar sua vida útil [18].

1.4.2 Métodos Convencionais de Estimativa do SoC

A estimativa do SoC em baterias é um processo complexo devido às características não lineares do sistema e à variabilidade de comportamento sob diferentes condições de operação. Métodos convencionais foram desenvolvidos para fornecer um monitoramento eficaz em diversas aplicações. Os principais métodos incluem:

Contagem de Coulombs

A contagem de coulombs é um método direto de estimativa do SoC, no qual se mede a corrente que entra e sai da bateria ao longo do tempo. A estimativa é realizada integrando a corrente e ajustando-a com base na capacidade nominal da bateria. A expressão matemática que define essa estimativa é:

$$SoC(t_1) = SoC(t_0) + \frac{1}{C_n} \int_{t_0}^{t_1} I_{\text{bat}} dt \times 100\% \quad (1.14)$$

onde $SoC(t_0)$ é o estado de carga inicial, C_n é a capacidade nominal da bateria, e I_{bat} representa a corrente da bateria, com polaridade positiva durante o carregamento e negativa durante a descarga [27].

Apesar de sua simplicidade, o método de contagem de coulombs acumula erros ao longo do tempo devido a imprecisões nas medições de corrente e variações na capacidade real da bateria devido ao envelhecimento. Em aplicações de longo prazo, essas imprecisões podem resultar em desvios significativos na estimativa do SoC, exigindo calibração periódica [18].

Tensão em Circuito Aberto (OCV)

A OCV utiliza a relação entre a tensão de repouso da bateria e o SoC, baseada em uma curva de calibração específica para a química da bateria. Essa relação é descrita matematicamente como:

$$SoC = f(OCV) \quad (1.15)$$

onde $f(OCV)$ representa uma função de calibração que mapeia a tensão em circuito aberto para o estado de carga da bateria [27].

Este método é preciso em condições de repouso, quando a bateria não está sendo carregada ou descarregada. No entanto, é sensível a variações de temperatura e ao envelhecimento das células, o que reduz sua precisão ao longo do tempo. A necessidade de repouso da bateria para medições precisas também limita o uso desse método em aplicações que requerem monitoramento constante.

Segundo [28], os modelos de OCV-SoC podem ser classificados em quatro categorias principais, cada uma com aplicações e limitações específicas:

- **Modelos Lineares:** Esses modelos são simples e de baixa complexidade computacional, o que os torna atraentes para sistemas com restrições de processamento. No entanto, a precisão desses modelos é limitada, especialmente em regiões extremas de SoC (0–30% e 80–100%), onde a relação OCV-SOC tende a ser mais não linear. São comumente usados em aplicações onde simplicidade e baixo custo são prioritários sobre a precisão.
- **Modelos Não Lineares:** Utilizam funções matemáticas mais complexas (exponenciais, logarítmicas) para melhorar a precisão, o que os torna mais adequados para capturar a relação não linear entre OCV e SoC, especialmente em baixas e altas regiões de carga. No entanto, esses modelos exigem mais recursos computacionais, o que pode ser uma

limitação em sistemas embarcados com capacidade de processamento restrita. Eles são preferidos em aplicações onde a precisão é crítica, como em sistemas de gerenciamento de baterias em veículos elétricos.

- **Modelos Híbridos:** Combinam segmentos lineares e não lineares para oferecer maior flexibilidade e precisão em toda a faixa de SoC. Esses modelos ajustam a simplicidade e precisão de acordo com a faixa de operação da bateria, podendo economizar recursos computacionais em algumas regiões enquanto mantêm a precisão em outras. São ideais para sistemas que exigem um equilíbrio entre custo computacional e precisão, sendo aplicáveis em dispositivos eletrônicos de consumo.
- **Modelos Tabulares:** Utilizam tabelas de consulta para armazenar pares de valores OCV-SOC, o que permite uma busca rápida e direta dos valores. Essa abordagem é vantajosa em sistemas com limitações de memória e processamento, pois evita cálculos complexos durante a operação. No entanto, a precisão depende da densidade de pontos armazenados na tabela e é suscetível a erros de interpolação. Esse modelo é comumente utilizado em sistemas de baixo custo ou com baixa capacidade de processamento, onde simplicidade e rapidez são mais importantes que uma extrema precisão.

Cada uma dessas categorias oferece vantagens específicas para diferentes tipos de aplicações, o que torna a escolha do modelo OCV-SOC fundamental para o desempenho do sistema de gerenciamento de baterias. A revisão de [28] também destaca que a seleção adequada depende não apenas do tipo de bateria, mas das condições operacionais, como faixa de temperatura, taxas de carga e o envelhecimento das células, que influenciam diretamente na precisão e confiabilidade do modelo.

Modelos de Circuitos Equivalentes

Os modelos de circuitos equivalentes simulam a resposta dinâmica da bateria utilizando componentes elétricos ideais, como resistores e capacitores, para representar o comportamento sob condições variadas de carga e descarga. Esses modelos são amplamente utilizados em Sistema de Gerenciamento de Baterias (BMS) devido à sua capacidade de representar com precisão fenômenos transientes e variações na resposta de tensão durante eventos de carga e descarga [18].

Os modelos mais comuns incluem o modelo de Thévenin, o modelo Parceria para a Nova Geração de Veículos e o modelo RC simplificado, cada um com características distintas em termos de precisão e complexidade computacional [27].

Filtros de Kalman e Métodos Avançados de Estimação

O filtro de Kalman é um método estatístico utilizado para a estimativa do SoC, combinando medições de corrente e tensão com um modelo dinâmico da bateria para fornecer estimativas em tempo real. Ele é eficaz para lidar com incertezas e ruídos nas medições, bem como com variações dinâmicas nas condições de operação [18].

Existem variações do filtro de Kalman desenvolvidas para melhorar a precisão em sistemas não lineares, como as baterias de íons de lítio:

- **Filtro de Kalman Estendido (EKF):** Lida com a não linearidade nas relações entre tensão, corrente e SoC aplicando uma linearização local do modelo.
- **Filtro de Kalman Unscented (UKF):** Utiliza uma técnica de transformação não linear para melhor lidar com as não linearidades severas do sistema.

Um exemplo prático de aplicação do filtro de Kalman em estimativa de SoC para baterias de íons de lítio é apresentado em [29]. Neste trabalho, os autores utilizam uma versão simplificada do modelo pseudo-bidimensional, permitindo a implementação do filtro em plataformas de simulação em tempo real, como o *Hardware-in-the-Loop* (HIL). A técnica mostrou-se eficaz para fornecer estimativas rápidas e precisas em cenários dinâmicos, mesmo em condições de alta exigência computacional.

Embora poderosos, esses métodos requerem um modelo detalhado da bateria e uma configuração cuidadosa dos parâmetros iniciais. Além disso, o processamento computacional exigido pode limitar seu uso em sistemas com recursos restritos [27].

1.4.3 Limitações dos Métodos Convencionais

Apesar de amplamente utilizados, os métodos convencionais de estimativa do SoC apresentam limitações:

- **Sensibilidade à Temperatura:** A temperatura afeta a resistência e a capacidade da bateria, impactando diretamente a precisão dos métodos de SoC.

- **Acúmulo de Erros na Contagem de Coulombs:** Erros de medição e variações na capacidade real da bateria levam ao acúmulo de erros ao longo do tempo.
- **Dependência de Curvas de Calibração em OCV:** As curvas de **OCV** podem se desatualizar devido ao envelhecimento e à temperatura, afetando a precisão da estimativa.
- **Limitações dos Modelos de Circuitos Equivalentes:** A precisão depende da correta calibração dos parâmetros, que variam com o envelhecimento e a temperatura.
- **Complexidade dos Métodos Avançados de Estimação:** Requerem maior capacidade computacional e modelagem precisa, o que pode ser limitante em dispositivos com restrições de processamento.

Essas limitações reforçam a necessidade de desenvolver métodos mais avançados e robustos, como abordagens baseadas em **AI**, que oferecem maior adaptabilidade e precisão para estimar o **SoC** em diferentes condições operacionais.

1.5 Métodos Baseados em Inteligência Artificial para Estimativa do SoC

1.5.1 Introdução aos Métodos de AI Aplicados ao SoC

A estimativa precisa do **SoC** das baterias de íons de lítio é essencial para a eficiência e segurança em diversas aplicações, desde dispositivos móveis até veículos elétricos. Os métodos convencionais de estimativa de **SoC**, como a contagem de coulombs e o uso de curvas de tensão em circuito aberto, apresentam limitações em ambientes dinâmicos e podem ser sensíveis a variáveis como temperatura e envelhecimento da bateria. Para superar essas limitações, técnicas de **AI**, como Rede Neural Artificial, do inglês: *Artificial Neural Network* (**ANN**), Redes Neurais Convolucionais (**CNN**) e Rede Neural Recorrente, do inglês: *Recurrent Neural Network* (**RNN**), têm emergido como abordagens robustas para a estimativa de **SoC**.

1.5.2 Fundamentos Teóricos das Técnicas de AI

Redes Neurais Artificiais

Uma ANN é um modelo computacional inspirado no funcionamento dos neurônios do cérebro [30]. Uma ANN é composta por uma série de neurônios ou nós conectados entre si, e cada neurônio é ativado ou inibido em função do sinal que recebe de outros neurônios. Os neurônios são agrupados em camadas, e cada camada é dedicada a uma tarefa específica de processamento da informação. Por exemplo, a camada mais simples em termos de estrutura é a Camada Totalmente Conectada, do inglês: *Fully Connected* (FC). Essas camadas conectam cada um dos nós da entrada da camada com cada um dos nós da saída da camada.

Uma ANN é composta por três tipos de camadas 1.11: uma camada de entrada, que recebe uma representação numérica dos dados; camadas ocultas, que realizam transformações nos dados, normalmente não lineares; e uma camada de saída, que produz uma previsão [31].

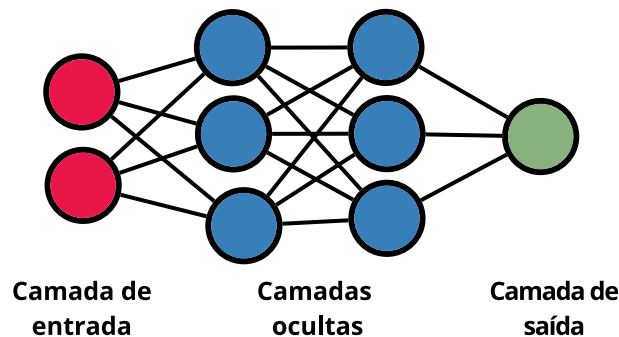


Figura 1.11: Estrutura de uma ANN

Existem vários tipos de redes neurais, entre elas as redes *feed-forward*, uma rede progressiva tal que a saída de um neurônio se conecta com outro neurônio da camada seguinte sempre na direção da esquerda para a direita (como a ANN da Figura 1.11). A Figura 1.12 mostra uma rede neural artificial simples conhecida como perceptron, que recebe um vetor de dados como entrada e o multiplica por um vetor de pesos. Em seguida, os valores ponderados pelos pesos são somados, e sobre este valor é aplicada uma função de ativação. A função de ativação determina se o perceptron "se ativa" ou não. Se o perceptron se ativa, ele devolve uma saída com valor 1; caso contrário, devolve uma saída com valor 0. Os valores do vetor de pesos podem ser ajustados para que o perceptron aprenda a classificar os dados de entrada. Para isso, fornece-se um conjunto de dados de treinamento etiquetados, onde essas etiquetas podem ser valores 0 ou 1.

O perceptron usa esses dados para avaliar sua capacidade de classificar corretamente a entrada e, posteriormente, com a ajuda de algumas métricas de erro, ajustar seus pesos, de modo a melhorar sua precisão. Uma ANN formada por várias unidades que replicam o comportamento do perceptron forma uma rede chamada perceptron multicamada.

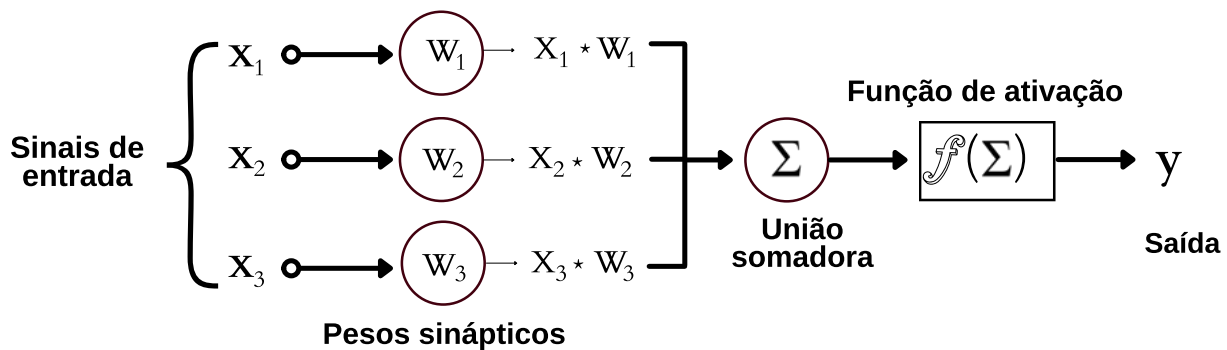


Figura 1.12: Exemplo de um Perceptron

É importante notar que a função de ativação desempenha um papel fundamental na rede, pois determina se a informação recebida por um neurônio é propagada para a próxima camada da rede [32]. Existem vários tipos de função de ativação, sendo os mais comuns: sigmoide, Unidade Linear Retificada, do inglês: *Rectified Linear Unit* (ReLU) e *tanh* [33] (Figura 1.13).

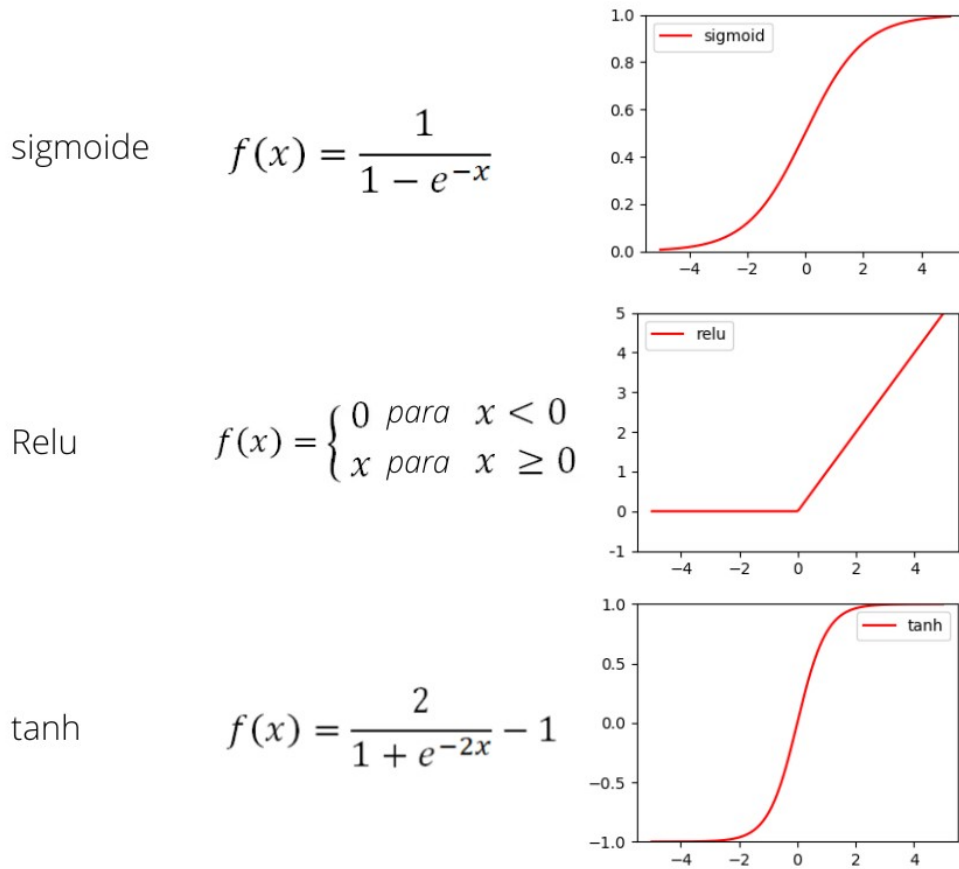


Figura 1.13: Funções de ativação

Para proporcionar maior clareza sobre o funcionamento da função de ativação, observe a Figura 1.14, que exemplifica a aplicação de uma função ReLU após a execução de uma operação de convolução. Note que a aplicação da função de ativação não afeta a dimensão da matriz obtida pela convolução.

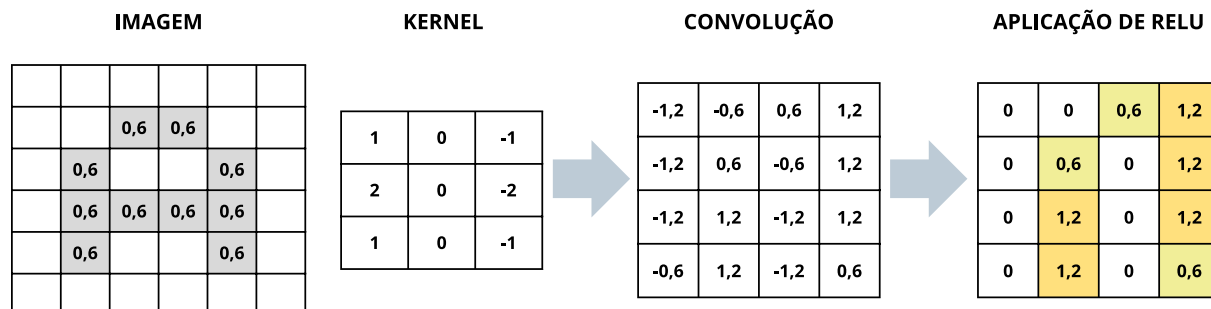


Figura 1.14: Aplicação da função de ativação

Outra função de ativação de grande interesse é a função de ativação Softmax, também

conhecida como Função Exponencial Normalizada.

$$\text{softmax}(x_j) = \frac{e^{x_j}}{\sum_{k=1}^K e^{x_k}} \quad (1.16)$$

Essa função recebe vetores de números reais como entradas e os normaliza em uma distribuição de probabilidade proporcional aos exponenciais dos números de entrada. Assim, a aplicação da Softmax assegura que a soma de todas as probabilidades atribuídas seja igual a 1 [34]. Essa função de ativação é de grande importância na resolução de problemas de classificação multiclasse, pois fornece uma distribuição de probabilidade necessária para estimar a classe à qual um novo dado pertence.

1.5.3 Conceitos Relevantes para a Otimização de Redes Neurais

Função de Custo

A função de custo, também conhecida como *Loss Function*, é uma medida de quão bem o modelo está ajustado aos dados. Trata-se de uma função de erro que é minimizada durante o treinamento do modelo. No contexto de classificação, a função de perda de entropia cruzada (*Cross Entropy Loss*), expressa matematicamente na Equação 1.17, é a mais comumente utilizada [35]. Essa função diminui à medida que a probabilidade prevista converge para a etiqueta real, medindo o desempenho de um modelo de classificação cuja saída prevista é um valor de probabilidade entre 0 e 1.

$$\text{Loss} = - \sum_{i=1}^N y_i \log(\hat{y}_i) \quad (1.17)$$

É importante notar que, para modelos de classificação, o vetor y é do tipo *one-hot*, um vetor que contém um único elemento igual a 1 e todos os outros elementos iguais a 0. Esses vetores *one-hot* são frequentemente utilizados para representar dados categóricos, como etiquetas. Por exemplo, se tivermos um conjunto de dados com três etiquetas A, B e C, podemos representar cada etiqueta como um vetor *one-hot*, onde A seria representado como [1, 0, 0], B como [0, 1, 0] e C como [0, 0, 1]. Nesse contexto, y atua como uma delta de Kronecker na Equação 1.17, somando apenas as probabilidades atribuídas à etiqueta real do objeto a classificar. Dessa forma, a *Loss* retorna uma média das previsões realizadas pelo modelo no conjunto de dados avaliado.

Descida do Gradiente

A descida do gradiente é um algoritmo de otimização utilizado para minimizar uma função de custo. Esse algoritmo baseia-se no cálculo da derivada da função de custo em relação aos parâmetros do modelo. Os parâmetros são atualizados na direção do gradiente descendente para minimizar a função de custo.

Iterações, Tamanho de Lote e Épocas

Uma iteração é um ciclo de aprendizado que inclui uma passagem por todos os dados de entrada e uma atualização dos pesos. O tamanho do lote, ou *batch size*, é o número de exemplos de treinamento usados em cada iteração, geralmente estabelecido em potências de 2 [36] devido à relação com a quantidade de memória necessária para armazenar os dados. As épocas representam o número de iterações completas necessárias para processar todo o conjunto de dados. Por exemplo, se temos 1000 dados e definimos um *batch size* de 200, serão realizadas 5 iterações por época de treinamento.

Optimizadores

Os otimizadores são utilizados para atualizar os parâmetros do modelo em cada iteração do algoritmo de descida do gradiente. Para isso, seleciona-se uma taxa de aprendizado que controla o quanto os pesos mudam em cada iteração. Os otimizadores mais comuns são a Descida de Gradiente Estocástico e a Descida de Gradiente com Momento. O Descida de Gradiente Estocástico calcula a derivada da função de custo em relação a cada parâmetro do modelo, atualizando os pesos na direção oposta ao gradiente para minimizar a função de custo.

Taxa de Aprendizado

A taxa de aprendizado, ou *Learning Rate*, é um hiperparâmetro importante no aprendizado profundo. Ela controla quanto o modelo deve ser ajustado em resposta ao erro estimado cada vez que os pesos são atualizados. Se a taxa de aprendizado for muito alta, o modelo pode ultrapassar o alvo e não convergir. Se for muito baixa, o modelo pode demorar muito para convergir [37]. Estratégias comuns incluem usar uma taxa de aprendizado menor nas primeiras épocas e aumentá-la gradualmente ao longo do treinamento. Normalmente, é recomendado começar com uma taxa de aprendizado pequena e aumentá-la caso o treinamento não esteja

convergindo.

Retropropagação

A retropropagação de erros (*Backpropagation*) é um método de aprendizado para ANN. O algoritmo “propaga” o erro da saída da rede para trás através das camadas, permitindo ajustar os pesos das sinapses para minimizar o erro. O processo ocorre em duas etapas principais. Primeiro, no processo *Feed-Forward*, um valor é estabelecido na camada de entrada, e cálculos são executados através das camadas ocultas da rede até que a resposta seja gerada na camada de saída [38]. Em seguida, o valor obtido é comparado ao valor desejado, calculando-se o erro da rede. Esse erro é propagado para trás na rede por meio de diferenciação parcial [39], resultando em um ajuste nos pesos das sinapses, onde a magnitude do ajuste é influenciada pela taxa de aprendizado. Esse processo é o que permite o aprendizado da rede.

Sobreajuste

Em geral, o sobreajuste (ou *overfitting*) é um problema comum no aprendizado de máquina. Ele ocorre quando um modelo aprende os detalhes e o ruído dos dados de treinamento de tal forma que afeta negativamente seu desempenho em novos dados [40]. Isso significa que o modelo apresenta bom desempenho nos dados de treinamento, mas não generaliza bem para novos dados. Existem várias estratégias para evitar o sobreajuste, entre elas:

- **Parada precoce:** Interromper o treinamento do modelo antes que ele aprenda o ruído presente nos dados.
- **Aumento de dados:** Adicionar mais dados ao modelo para torná-lo mais estável.
- **Seleção de características:** Identificar as características mais importantes dos dados de treinamento e eliminar as irrelevantes ou redundantes.
- **Regularização:** Aplicar uma “penalização” aos parâmetros de entrada com coeficientes maiores, limitando, assim, a variação no modelo.
- **Métodos de conjunto:** Os métodos de aprendizado em conjunto são compostos por um conjunto de classificadores, como árvores de decisão, e suas previsões são combinadas para identificar o resultado mais frequente.

1.5.4 Estimativa do SoC com técnicas de AI

Os avanços na estimativa do SoC de baterias de íons de lítio têm sido impulsionados por métodos de ML e Aprendizado Profundo, do inglês: *Deep Learning* (DL), conforme discutido em diversos estudos recentes. No trabalho de Ng *et al.* [41], técnicas de ML foram empregadas para estimar tanto o SoC quanto o SoH, com destaque para a precisão das redes neurais e máquinas de vetores de suporte na predição de vida útil e degradação da capacidade. Essa abordagem permitiu uma avaliação eficaz dos parâmetros da bateria, considerando a complexidade das reações internas e os efeitos ambientais.

Richardson *et al.* [42] propuseram o uso de regressão por processos gaussianos para estimar a capacidade da bateria em tempo real, sem a necessidade de transformar dados em curvas de capacidade incremental. Este método demonstrou uma redução significativa no erro de previsão, com potencial aplicação em sistemas que requerem monitoramento contínuo, como veículos elétricos.

Estudos focados em predição de vida útil também mostraram avanços importantes. O trabalho de Severson *et al.* [43] utilizou dados de ciclo de carga para prever a vida útil das baterias antes que a degradação da capacidade se tornasse aparente. A análise dos dados de 124 células comerciais de íons de lítio possibilitou uma previsão precisa da vida útil, destacando o papel dos modelos baseados em dados para aplicações de longo prazo.

Outros estudos abordaram o uso de redes neurais convolucionais para estimativa de capacidade. Qian *et al.* [44] desenvolveram um modelo que utiliza segmentos aleatórios das curvas de carga para prever a capacidade da bateria. Esse método se mostrou eficaz para estimativas em segmentos mais longos e com menores estados de carga, reforçando a aplicabilidade de DL na captura de padrões complexos.

Em relação à automação de métodos tradicionais, Vibhuti [45] apresentou um modelo baseado em MATLAB para automatizar a estimativa do SoC a partir de dados experimentais. Essa solução simplifica o processo de monitoramento e facilita a integração com BMS.

Além disso, uma revisão abrangente realizada por Cui *et al.* [46] destaca os avanços e limitações das técnicas baseadas em redes neurais para a estimativa de SoC, discutindo a necessidade de algoritmos que reduzam a complexidade computacional e melhorem a generalização.

Por fim, Ragone *et al.* [47] combinaram simulações automotivas e modelos multi-físicos para gerar dados representativos, que foram usados para treinar modelos de ML aplicados ao SoC de veículos elétricos. Essa abordagem permite que os modelos de ML considerem fatores como a

potência do motor e a resistência aerodinâmica, proporcionando uma estimativa mais robusta em condições reais.

Capítulo 2

Estimando o Estado de Carga (SoC)

2.1 Objetivo

O objetivo deste trabalho é desenvolver uma estratégia para a estimativa do SoC de baterias de íons de lítio, adaptando o método proposto por Qian *et al.* (2021) [44]. A abordagem emprega 1D CNN para estimar a capacidade das baterias a partir de conjuntos de dados das curvas de carga, fundamentando-se nos resultados de Qian *et al.*, que demonstraram a eficácia da 1D CNN para essa tarefa.

Em uma comparação abrangente com diferentes métodos de estimativa de capacidade, como SVM, FNN, 2D CNN, LSTM e GPR, Qian *et al.* observaram que a 1D CNN alcançou os menores valores de RMSE, evidenciando sua superioridade em termos de precisão. Este estudo, visa aprimorar a precisão na estimativa do SoC através da especialização das 1D CNN para diferentes segmentos da curva de carga, testando configurações variadas de hiperparâmetros e tamanhos distintos dos conjuntos de dados de entrada.

A seguir, detalharemos as etapas metodológicas principais, destacando as adaptações realizadas para este estudo. A metodologia completa será explicada no capítulo 3, garantindo uma compreensão aprofundada dos processos envolvidos.

2.1.1 Estrutura da Rede Neural Convolutacional Unidimensional

A estrutura 1D CNN proposta por Qian *et al.* consiste em três camadas convolucionais, seguidas de uma camada de pooling global e duas camadas totalmente conectadas. Conforme ilustrado na Figura 2.1, cada camada convolutacional aplica filtros de tamanho fixo para extrair características

dos dados, identificando padrões nas curvas de carga associados à capacidade da bateria. Os dados de entrada para a rede incluem segmentos aleatórios de três curvas: a curva de tensão de carga, a curva diferencial de tensão de carga e a curva de corrente de carga. Estes segmentos são extraídos de ciclos de carga das baterias, permitindo que o modelo 1D CNN estime a capacidade com base em dados obtidos sob diferentes condições de carga, o que aumenta sua robustez e adaptabilidade.

Para garantir uma normalização consistente dos dados, cada segmento de entrada é processado para ter o mesmo comprimento e é dividido em amostras igualmente espaçadas. As três curvas (tensão, diferencial de tensão e corrente) são então organizadas em uma matriz de entrada de dimensões $m \times 3$, onde m representa o número de amostras por segmento, com colunas I , V e DV .

Qian *et al.* utilizam um algoritmo de otimização por enxame de partículas com peso linear decrescente para ajustar os hiperparâmetros das camadas convolucionais, incluindo o número de filtros e o tamanho dos kernels, maximizando o desempenho da rede. Neste estudo, consideramos três configurações de hiperparâmetros. Duas dessas configurações são baseadas nos hiperparâmetros otimizados por Qian *et al.*, enquanto uma última é ajustada aleatoriamente para explorar o impacto no desempenho.



Figura 2.1: Estrutura 1D CNN. Adaptada de Qian *et al.* (2021) [44]

2.1.2 Estrutura dos Dados de Entrada

Como ilustrado na Figura 2.2, os conjuntos de dados de entrada (I , V , DV) para o modelo 1D CNN são extraídos das curvas de carga das baterias de íons de lítio, incluindo:

- **Curva de Tensão de Carga:** Variação da tensão ao longo do tempo durante o carrega-

mento da bateria.

- **Curva de Tensão Diferencial:** Derivada da tensão em relação à capacidade da bateria.
- **Curva de Corrente de Carga:** Resposta de corrente durante a fase de carga a tensão constante.

Esse conjuntos de dados possuem n dados, assim, cada entrada para a rede consiste em uma matriz $n \times 3$, com dados normalizados para estabilizar o treinamento.

A abordagem de Qian *et al.* utiliza a variável k para definir a quantidade de dados extraídos de cada ciclo de carga para o conjunto de treinamento, permitindo investigar o impacto do tamanho do conjunto sobre a precisão da estimativa de capacidade. Os resultados indicam que conjuntos mais longos, com valores maiores de k , melhoram a precisão, reduzindo **RMSE**. Qian *et al.* observaram que uma entrada com $n = 55$ apresentou o melhor desempenho.

Neste estudo, foram testados tamanhos variados (20, 40, 55 e 60) para avaliar esses achados.

2.1.3 Segmentação dos Dados e Modelos Especializados

Conforme discutido no Capítulo 1 e ilustrado na Figura 2.2 (b), durante a carga de uma bateria de íons de lítio, a tensão varia de forma não linear, apresentando regiões com comportamentos específicos. Esse padrão permite dividir a curva de carga em três etapas principais, cada uma refletindo diferentes condições de operação e **SoC** da bateria.

Qian *et al.* investigaram o impacto das posições dos conjuntos de dados de entrada na estimativa da capacidade de baterias de íons de lítio. Foram selecionadas três posições distintas dos segmentos (denominados segmentos A, B e C), representando as três etapas mencionadas anteriormente. Eles observaram que, devido à natureza altamente não linear das curvas de carga, a capacidade estimada depende significativamente da posição dos conjuntos de dados de entrada nos modelos. O estudo revelou que conjuntos no início da curva de carga (Segmento A) geraram estimativas de capacidade mais precisas, enquanto conjuntos na fase final (Segmento C) apresentaram **RMSE** superiores a 7,5%.

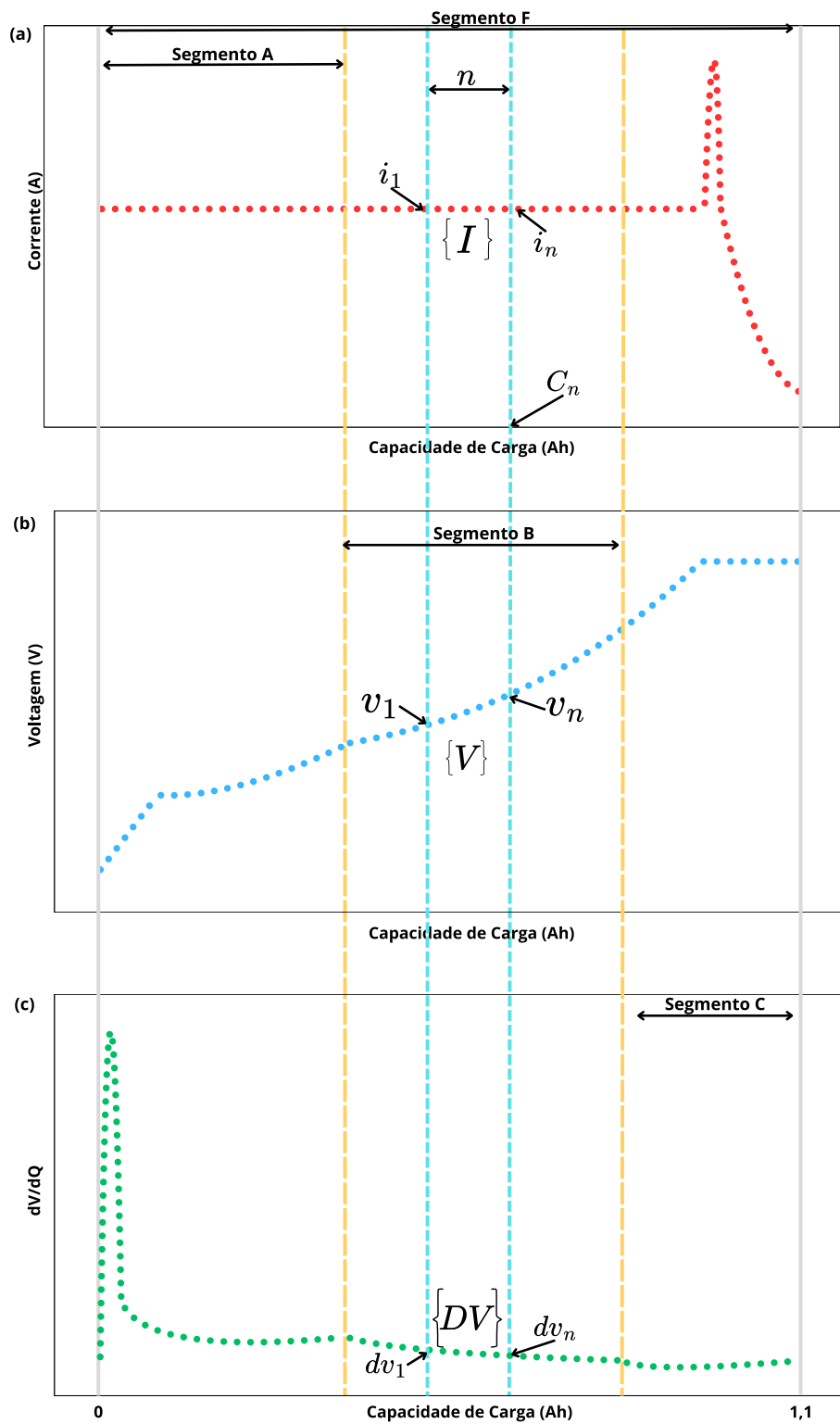


Figura 2.2: Dados segmentados a partir das curvas de carga em um único ciclo de carga. (a): Dados da curva de corrente de carga; (b): Dados da curva de tensão de carga; (c): Dados da curva de tensão diferencial de carga. Adaptada de Qian *et al.* (2021) [44]

Neste estudo, consideram-se relevantes os achados de Qian *et al.*, uma vez que a posição relativa do conjunto de dados de entrada afeta a precisão da estimativa. Portanto, propõem-se redes **1D CNN** especializadas, treinadas individualmente com conjuntos de dados de cada segmento de carga. Inicialmente, foram desenvolvidos três modelos de **1D CNN**, baseados na estrutura proposta por Qian *et al.* e configurados com diferentes conjuntos de hiperparâmetros, resultando nos Modelos 1, 2 e 3.

Para cada modelo considerado, realiza-se o treinamento em cada segmento: uma rede para o Segmento A, outra para o Segmento B, outra para o Segmento C e, adicionalmente, uma para o Segmento F, que representa a utilização da curva completa de carga. A inclusão do Segmento F replica o trabalho de Qian *et al.*, permitindo a comparação direta dos resultados deste estudo com os obtidos originalmente.

Considerando-se três modelos e cada um treinado para os segmentos A, B, C e F, obtém-se um total de doze modelos a serem treinados. Além disso, cada modelo será treinado com diferentes tamanhos de dados (20, 40, 55 e 60 amostras), totalizando 48 **1D CNNs** treinadas.

Após o treinamento, avalia-se o desempenho de cada modelo com base no **RMSE**. Selecionam-se os melhores modelos para cada segmento específico (A, B e C) e o modelo com melhor desempenho para o Segmento F, garantindo a maior precisão possível na estimativa de cada etapa da curva de carga.

2.1.4 Integração dos Modelos Especializados

Posteriormente, implementamos um método de integração que seleciona o modelo apropriado com base no nível atual de tensão da bateria. Dependendo da faixa de tensão observada, os dados são direcionados para o modelo correspondente (A, B ou C) para realizar a estimativa do **SoC** e, em seguida, avaliar seu desempenho em comparação com o modelo F.

2.2 Resumo das Etapas Metodológicas

As principais etapas metodológicas deste estudo são:

1. **Exploração e Tratamento dos Dados:** Coleta e pré-processamento das curvas de carga das baterias de íons de lítio.
2. **Segmentação das Curvas de Carga e Obtenção dos Conjuntos de Dados:** Divisão

das curvas de carga em segmentos A, B, C e F, extração dos conjuntos de dados e divisão desses conjuntos em treinamento, validação e teste.

3. **Configuração dos Modelos:** Definição de três modelos de [1D CNN](#) com configurações distintas de hiperparâmetros.
4. **Treinamento dos Modelos:** Treinamento de cada configuração dos modelos A, B, C e F com os dados de treinamento e validação, utilizando tamanhos variados dos conjuntos de dados de entrada (20, 40, 55 e 60).
5. **Avaliação e Seleção dos Melhores Modelos:** Avaliação do desempenho de cada modelo em termos de [RMSE](#) e seleção dos modelos com melhor desempenho para cada segmento.
6. **Integração das Estimativas:** Implementação de um método que seleciona o modelo apropriado com base no nível atual de tensão, utilizando o modelo especializado correspondente para fornecer a estimativa do [SoC](#) e comparação com o modelo F.

Capítulo 3

Metodologia

Nesta seção, detalhamos os procedimentos e técnicas utilizados para a preparação dos dados, o treinamento e a integração dos modelos especializados [1D CNN](#) para a estimativa do [SoC](#) de baterias de íons de lítio, bem como a metodologia para a comparação dos resultados da abordagem utilizada neste trabalho com os resultados do trabalho original de Qian *et al.* 2021.

3.1 Desenvolvimento de uma Ferramenta de Estimativa de SoC

Para a execução da metodologia proposta, foi desenvolvida uma aplicação personalizada em Python¹, destinada à estimativa do [SoC](#) de baterias de íons de lítio. Esta ferramenta se destaca como um diferencial significativo do presente estudo, oferecendo uma interface intuitiva e funcionalidades para treinar e utilizar modelos [1D CNN](#).

A aplicação conta com uma Interface Gráfica de Usuário (GUI) desenvolvida em Tkinter², proporcionando uma experiência de usuário amigável. Além disso, foi projetada para ser facilmente adaptável a diferentes fontes de dados, permitindo a importação e processamento de conjuntos de dados (*datasets*) variados sem a necessidade de modificações substanciais no código-

¹Python Software Foundation. Python Language Reference, versão 3.10. Disponível em: <https://www.python.org/>

²Tkinter é a biblioteca padrão para criação de interfaces gráficas em Python. Para mais informações, consulte a documentação oficial: <https://docs.python.org/3/library/tkinter.html>.

fonte. Essa flexibilidade garante que a ferramenta possa ser utilizada em futuras pesquisas com dados provenientes de ciclos de carga e descarga, permitindo a avaliação e o ajuste dos modelos em diferentes cenários operacionais.

Obtenção de Modelos 1D CNN Personalizados: A ferramenta facilita a criação de modelos 1D CNN personalizados em poucos passos. Na Interface Gráfica de Usuário (GUI) os usuários podem configurar hiperparâmetros, iniciar treinamentos e visualizar resultados diretamente, sem necessidade de conhecimentos avançados em programação ou aprendizado de máquina.

Processamento Paralelo e Aceleração com GPU: Para otimizar o tempo de processamento e treinamento dos modelos, a aplicação implementa técnicas de paralelismo utilizando múltiplas threads e processos. Além disso, a integração com CUDA³, permite a utilização eficiente da GPU, acelerando significativamente as operações de treinamento e inferência. Para o desenvolvimento deste estudo, foi utilizada a versão 12.4 do CUDA⁴. O hardware utilizado para este trabalho incluiu:

- **CPU:** AMD Ryzen 7 5700G com Radeon Graphics
 - Velocidade de base: 3,80 GHz
 - Núcleos: 8
- **GPU:** NVIDIA GeForce RTX 4060 Ti
 - Memória dedicada da GPU: 8,0 GB
 - Memória compartilhada da GPU: 7,9 GB
 - Memória total da GPU: 15,9 GB
- **Memória RAM:** 16,0 GB
 - Velocidade: 3200 MT/s
 - Fator de forma: DIMM

Estrutura Modular: A aplicação foi desenvolvida com uma arquitetura modular, permitindo a inclusão fácil de novas funcionalidades e a manutenção eficiente do código. Cada

³CUDA é uma plataforma de computação paralela e API de programação desenvolvida pela NVIDIA. Para mais informações, consulte: <https://developer.nvidia.com/cuda-toolkit>.

⁴Disponível em: <https://developer.nvidia.com/cuda-12-4-0-download-archive>

módulo é responsável por uma parte específica do processo, como carregamento de dados, pré-processamento, treinamento de modelos e visualização de resultados. Desta forma, poderão-se acrescentar módulos tanto para implementar outros modelos para estimativa de SoC quanto para implementar módulos para estimativa de outros parâmetros como o Estado de Energia, do inglês: *State of Energy* (SoE) ou SoH.

Integração com Anexos: Todo o código-fonte da aplicação foi organizado e documentado nos Anexos deste trabalho. Tanto as rotinas de tratamento e manipulação de dados quanto os *scripts* responsáveis pelo treinamento e avaliação dos modelos encontram-se no Apêndice ??.

Documentação e Manutenção: A documentação completa da aplicação, incluindo instruções de uso e exemplos de execução, está disponível no Apêndice ??. Isso garante que futuros pesquisadores possam utilizar e adaptar a ferramenta conforme necessário, promovendo a continuidade e expansão das pesquisas nesta área.

3.2 Coleta e Preparação de Dados

3.2.1 Bases de Dados

A literatura destaca a importância de *datasets* bem documentados e acessíveis para garantir a replicabilidade e a validação dos modelos de ML aplicados a baterias. Segundo uma revisão apresentada por Dos Reis *et al.*[48], há uma variedade de *datasets* públicos que incluem informações sobre envelhecimento cíclico, capacidade de carga/descarga e condições operacionais variadas, provenientes de fontes como Administração Nacional da Aeronáutica e Espaço (NASA), Centro de Engenharia de Ciclo de Vida Avançado (CALCE) e Toyota Research Institute. Esses *datasets* foram organizados por variáveis experimentais, como temperatura, DoD, química das células – isto é, a composição e a arquitetura dos materiais ativos e passivos que definem o comportamento eletroquímico e a estabilidade das baterias – e taxas de carga, sendo diretamente aplicáveis ao cálculo de SoH e SoC, pois fornecem informações para avaliar a degradação ao longo do tempo e ajustar modelos a diferentes cenários operacionais.

Para o presente estudo, optou-se pelo uso do *dataset* fornecidos pelo CALCE⁵, da Universi-

⁵Center for Advanced Life Cycle Engineering (CALCE), Universidade de Maryland. <https://calce.umd.edu/>.

dade de Maryland, que oferece acesso a dados experimentais sobre baterias de íon-lítio ao longo de ciclos de carga e descarga.

3.2.2 Descrição do Dataset da CALCE

O *dataset* da CALCE⁶ utilizado neste estudo inclui apenas baterias do grupo CS2, especificamente as células identificadas como CS2_35, CS2_36, CS2_37 e CS2_38. Esse dataset específico foi escolhido pela possibilidade de comparar os resultados obtidos com os estudos de Qian *et al.*[49], que utilizaram esses dados para avaliar o desempenho de modelos de 1D CNN aplicados a estimativas de SoC.

As especificações dessas células são as seguintes:

- **Capacidade Nominal:** 1.1 Ah
- **Química da Célula:** Cátodo de LiCoO_2
- **Peso:** 21,1 g (sem circuito de segurança)
- **Dimensões:** 5,4 mm x 33,6 mm x 50,6 mm
- **Configuração:** Enrolamento tipo "jellyroll" ao longo do eixo longitudinal. Essa configuração refere-se à disposição interna dos eletrodos, separador e coletores de corrente que são enrolados de forma contínua, semelhante a um "jellyroll". Essa técnica otimiza a área de contato entre os componentes e permite um design compacto e eficiente, sendo comumente empregada em células cilíndricas.
- **Procedimento de Carga e Descarga:** Todas as células CS2 foram carregadas utilizando um protocolo padrão de Carga a Corrente Constante seguida por Tensão Constante, do inglês: *Constant Current, Constant Voltage* (CC-CV) com uma taxa de corrente constante de 0,5C até que a tensão atingisse 4,2V. Em seguida, a tensão de 4,2V foi mantida até que a corrente de carga caísse para abaixo de 0,05A. A tensão de corte de descarga foi ajustada para 2,7V. As Figuras 3.1 e 3.2 apresentam o comportamento da corrente e da tensão no tempo durante os primeiros ciclos de carga e descarga da célula 35.

⁶Os dados podem ser acessados em: <https://calce.umd.edu/battery-data>

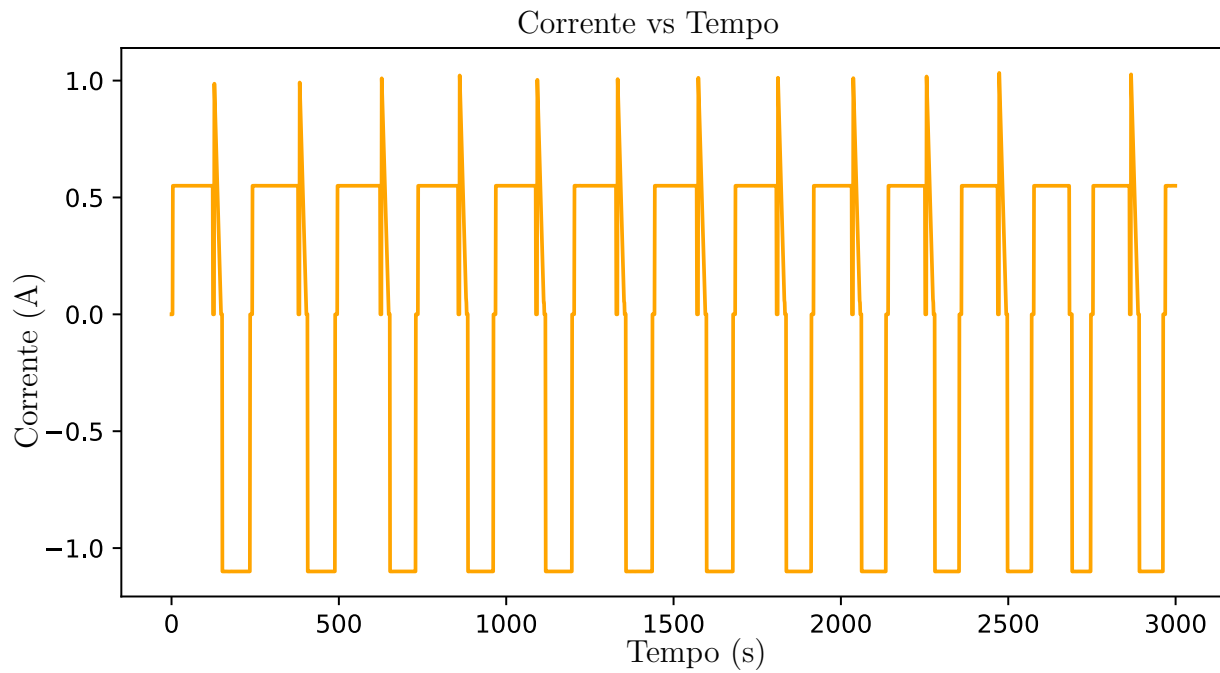


Figura 3.1: Comportamento da Corrente ao Longo do Tempo.

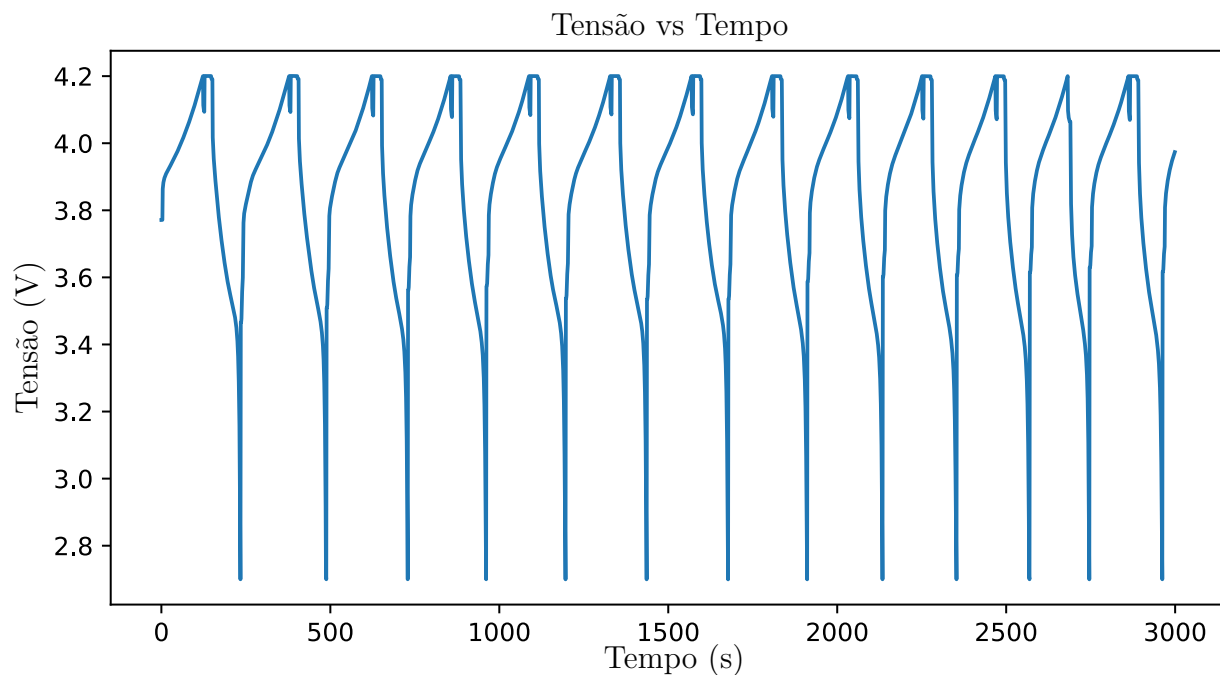


Figura 3.2: Comportamento da Tensão ao Longo do Tempo.

Estrutura de Diretórios dos Dados

Os dados das células CS2.35, CS2.36, CS2.37 e CS2.38 estão organizados em uma estrutura de diretórios conforme apresentado na Figura 3.3.

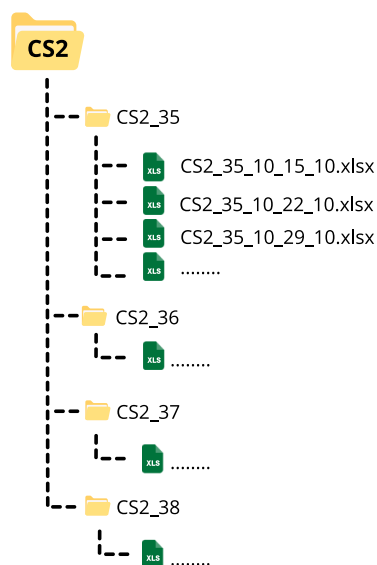


Figura 3.3: Estrutura dos dados experimentais

Cada arquivo Excel representa uma sessão de teste realizada em uma data específica, seguindo o padrão de nomenclatura `CS2-<célula_número>-<data>.xlsx`, onde `<data>` indica o dia, mês e ano do teste (por exemplo, `10_15_10` para 15 de outubro de 2010).

Tratamento e Seleção de Dados

Para garantir a integridade e relevância dos dados utilizados nas análises e modelagem, foram aplicadas diversas etapas de tratamento e seleção. O conjunto de dados de baterias foi manipulado por meio de rotinas em Python, que realizaram a leitura, filtragem e correção dos dados armazenados nos arquivos Excel. Essas etapas garantiram que apenas os ciclos de carga relevantes fossem considerados e corrigiram o comportamento cumulativo indesejado de algumas variáveis, como a *Charge Capacity*.

As etapas do tratamento e seleção de dados foram as seguintes:

1. Leitura dos Arquivos Excel:

- Todos os arquivos Excel das células CS2.35, CS2.36, CS2.37 e CS2.38 foram identificados e lidos, selecionando as colunas principais para análise, incluindo *Voltage (V)*, *Current (A)*, *Cycle Index*, *Step Index* e *Charge Capacity (Ah)*.
- Apenas os ciclos relevantes para o processo de carga foram considerados, eliminando ciclos iniciais e quaisquer dados que não fossem necessários para a análise.

2. Correção da Capacidade de Carga:

- Os dados de *Charge Capacity (Ah)* originalmente apresentavam um comportamento acumulativo, onde o valor aumentava de forma contínua ao longo dos ciclos, como observado na Fig 3.4. Esse comportamento foi ajustado para refletir apenas a capacidade de carga real em cada ciclo.
- A capacidade foi redefinida calculando a diferença na carga acumulada para cada ciclo, o que garante que a coluna de capacidade represente o valor específico de cada ciclo, em vez de uma soma acumulada.

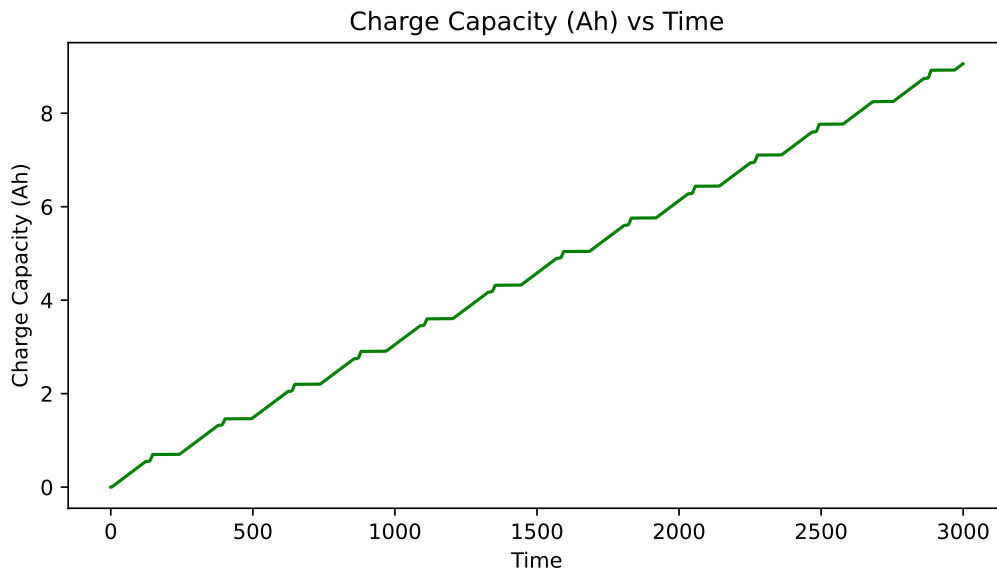


Figura 3.4: Comportamento acumulativo da capacidade de carga

3. Cálculo de dV/dQ :

- Para replicar o trabalho do Qian *et al.*, foi calculada a relação entre a variação de tensão (dV) e a variação de capacidade de carga (dQ) para cada ponto de dados dentro dos ciclos selecionados. Essa relação (dV/dQ) é relevante para a análise da dinâmica da bateria e fornece informações sobre a resistência interna e eficiência do processo de carga.

4. Concatenação dos Arquivos por Célula e Data:

- Após a leitura e manipulação de cada arquivo, os dados foram concatenados em um único *DataFrame*⁷, por célula. A numeração do índice de ciclo (*Cycle Index*) foi ajustada para garantir a continuidade entre os arquivos.
- Os dados finais foram organizados de forma que cada célula tivesse seu conjunto de ciclos completo e ordenado, prontos para análise.

5. Separação dos Dados por Ciclo:

- Uma vez obtido cada *DataFrame*, cada ciclo foi extraído separadamente e armazenado em um dicionário, com o objetivo de permitir uma análise individual dos ciclos e facilitar o acesso a dados específicos.
- Cada ciclo de carga foi armazenado como um *DataFrame* independente, que contém as colunas principais (*Voltage (V)*, *Current (A)*, *dV/dQ*, *Charge Capacity (Ah)*).

6. Armazenamento em Formato HDF5:

- Os dados finais foram salvos em um arquivo no formato *Hierarchical Data Format version 5 (HDF5)*⁸. Este formato permite uma compactação eficiente e facilita o acesso rápido aos dados.
- Como observado na Figura 3.5, para cada célula, foi criado um grupo no arquivo **HDF5**, e cada ciclo foi armazenado como um conjunto de dados separado dentro do grupo correspondente contendo as variáveis principais necessárias para a análise (*Charge Capacity (Ah)*, *Current (A)*, *Voltage (V)*, *dV/dQ*). Esta organização permite recuperar rapidamente os dados de um ciclo específico, o que reduz o tempo de leitura e processamento nas etapas subsequentes.

⁷Um *DataFrame* é uma estrutura de dados tabular amplamente utilizada na biblioteca **pandas**, composta por linhas e colunas, similar a uma planilha.

⁸HDF5 (*Hierarchical Data Format version 5*) é um formato de arquivo projetado para armazenar e organizar grandes volumes de dados, permitindo compactação eficiente e acesso rápido. Para mais informações, consulte: <https://www.hdfgroup.org/solutions/hdf5/>.

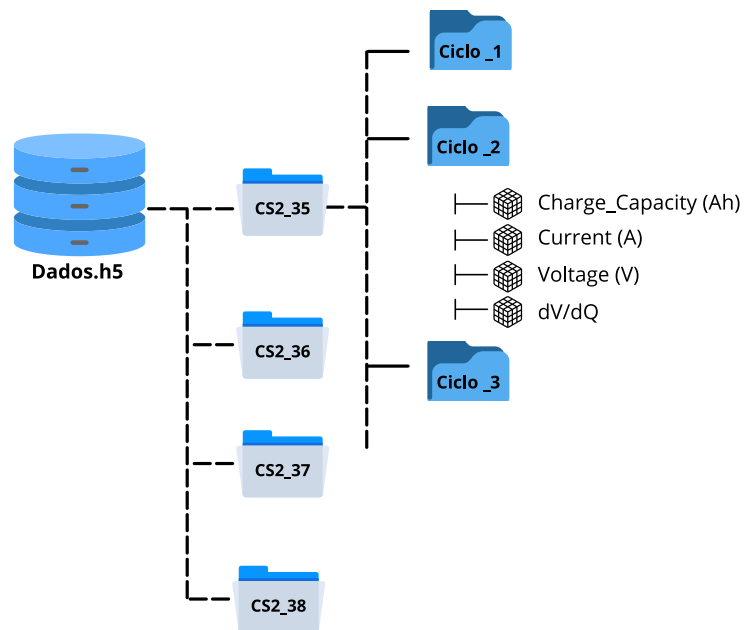


Figura 3.5: Estrutura dos dados no arquivo HDF5

Geração de Gráficos a Partir dos Dados Estruturados

A seguir, são apresentados exemplos de gráficos gerados a partir dos dados armazenados, o que ilustra diferentes aspectos do comportamento das baterias ao longo dos ciclos de carga.

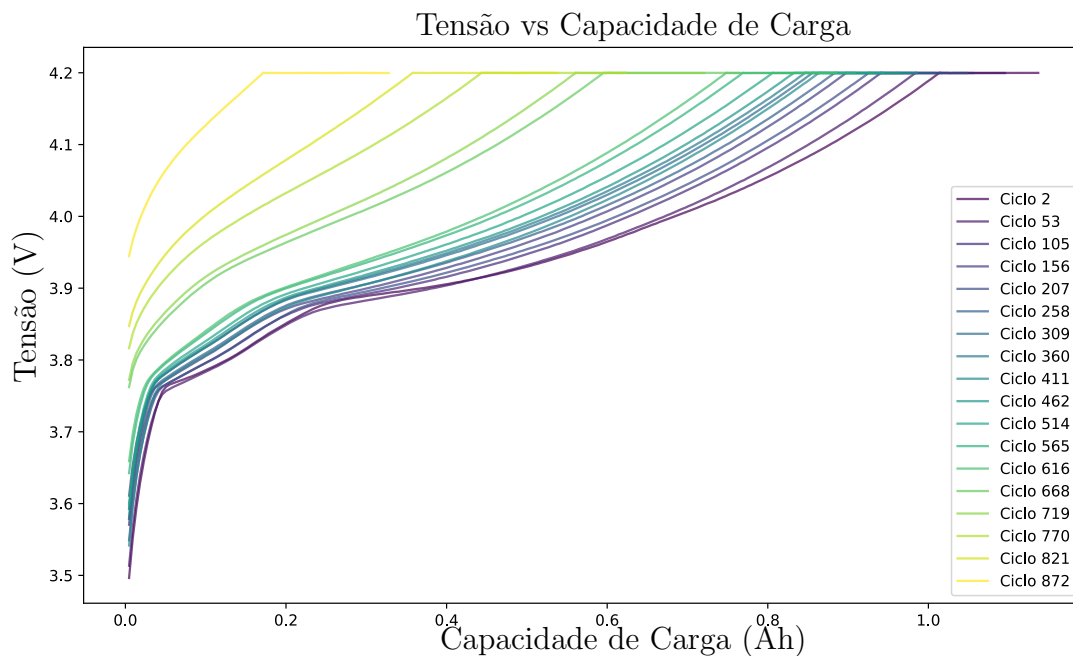


Figura 3.6: Curvas de Carga: Tensão vs. Capacidade em Diferentes Ciclos

No gráfico da Figura 3.6, observamos a relação entre a tensão e a capacidade de carga ao longo dos ciclos de carga. Com o aumento do número de ciclos, a curva de voltagem da bateria apresenta variações, especialmente nas fases de carga completa. Essas mudanças podem ser interpretadas como um indicador de degradação gradual da bateria, isto é, o processo de deterioração dos materiais e das propriedades eletroquímicas internas com o uso, que resulta na diminuição da eficiência e da capacidade máxima de carga.

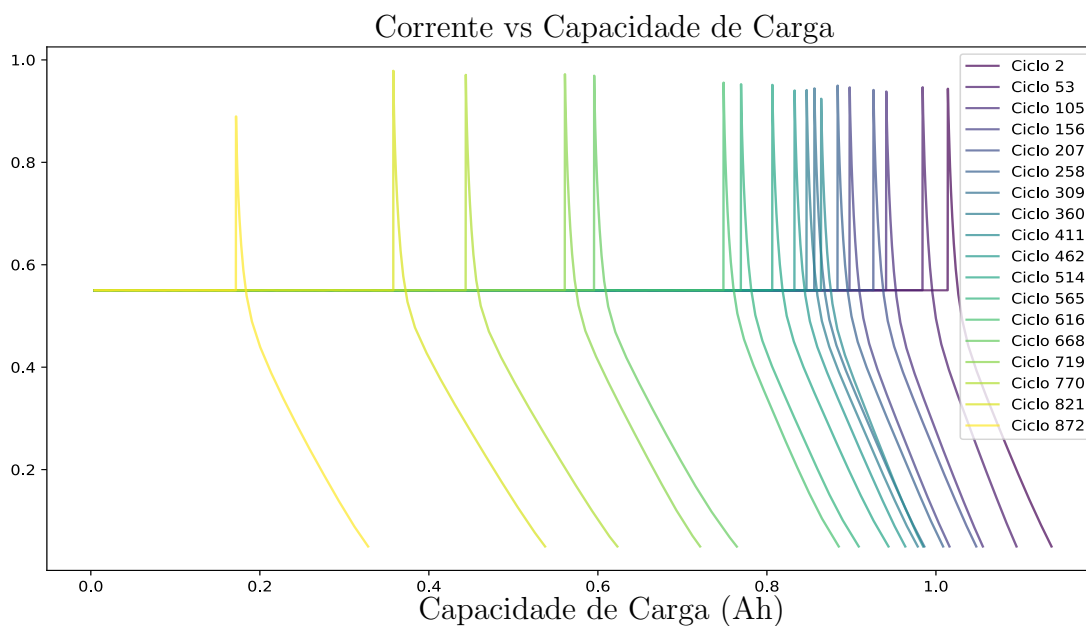


Figura 3.7: Curvas de Carga: Corrente vs Capacidade em Diferentes Ciclos

A Figura 3.7 ilustra a relação entre a corrente e a capacidade para diferentes ciclos de carga. Este gráfico mostra que, durante a fase inicial, a corrente se mantém constante, seguida de uma queda abrupta conforme a bateria se aproxima da carga completa. A mudança gradual nas curvas ao longo dos ciclos reflete uma possível redução na capacidade de carga e alterações nas características de carga da bateria, sinalizando possíveis perdas no seu desempenho com o uso.

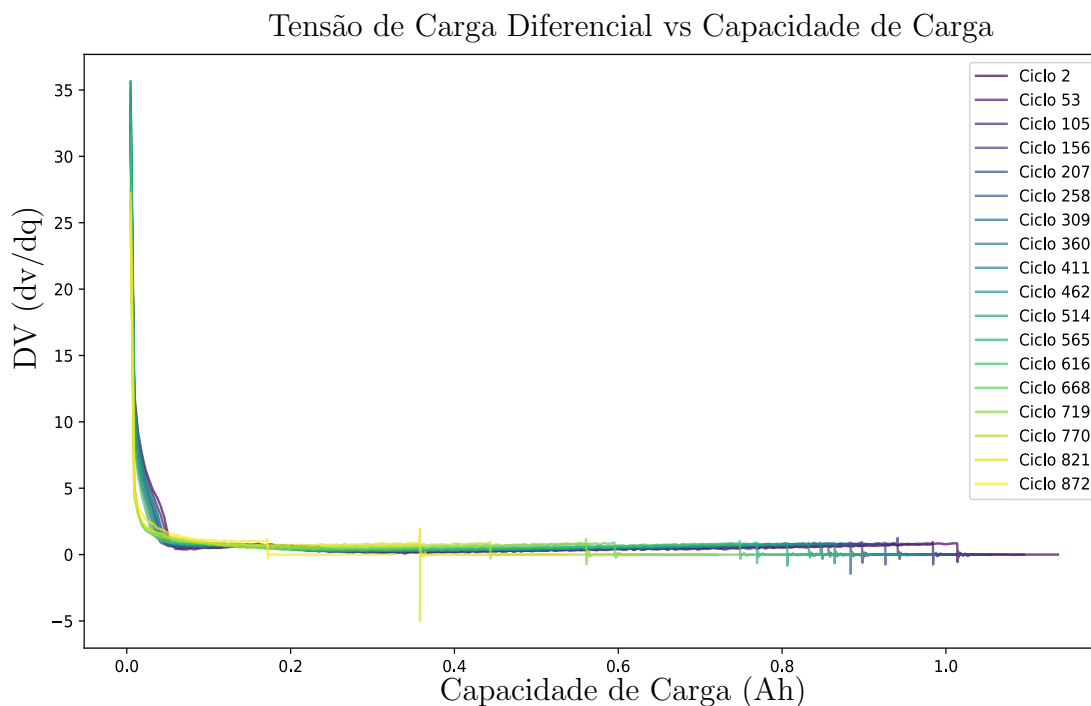


Figura 3.8: Curvas de Carga: dV/dQ vs Capacidade em Diferentes Ciclos

O gráfico da Figura 3.8 mostra a derivada da tensão em relação à capacidade de carga (dV/dQ) ao longo dos ciclos. Esta métrica é útil para identificar as mudanças nas reações químicas internas da bateria em diferentes estágios de carga. O pico inicial e a estabilização subsequente representam reações específicas que ocorrem durante o carregamento. À medida que os ciclos avançam, observa-se uma alteração na forma e intensidade dos picos, o que pode indicar degradação ou aumento da resistência interna da bateria ao longo do tempo.

Esses gráficos permitem analisar como o desempenho da bateria evolui com o uso, possibilitando a identificação de sinais de degradação e perda de capacidade ao longo dos ciclos.

Ao alimentar as [1D CNN](#) com esses dados, é possível ensinar o modelo a reconhecer padrões de degradação e características de desempenho específicas que indicam o nível de carga em diferentes condições. As variações nas três curvas ao longo dos ciclos de carga permitem ao modelo capturar relações complexas entre a voltagem, corrente, e as alterações químicas internas da bateria, o que resulta em uma estimativa de [SoC](#) mais precisa.

3.2.3 Preparação do *dataset* para Treinamento e Validação

Para treinar e validar os modelos de 1D CNN, foram processados os dados previamente tratados para obter múltiplos *datasets* com diferentes quantidades de amostras, especificamente 20, 40, 55 e 60 dados. Esses conjuntos foram extraídos para cada segmento A, B, C e F, totalizando 16 conjuntos. Os conjuntos foram obtidos considerando apenas os ciclos com *Cycle_Index* menores que 100, pois, como se observa nas Figuras 3.6 e 3.7, os ciclos posteriores exibem mudanças características que refletem a degradação da célula ou bateria.

Esse limite foi adotado para evitar que o modelo fosse "corrompido" por padrões de degradação acentuados, concentrando o aprendizado nas condições de funcionamento inicial da bateria. Dessa forma, o modelo pode capturar com maior precisão os padrões de SoC sem ser influenciado pelas variações decorrentes do envelhecimento, melhorando a generalização e a robustez da estimativa de SoC. No entanto, nada impede que outras faixas de ciclos sejam utilizadas para treinar o modelo, permitindo o cálculo do SoC para células em estágios mais avançados de vida útil, onde a degradação já é mais acentuada. Essa abordagem pode ser útil para desenvolver modelos capazes de estimar o SoC em baterias que já passaram por vários ciclos de carga.

Extração e Divisão dos Dados

Como mencionado anteriormente, foram gerados *datasets* para cada combinação de segmento e quantidade de amostras, correspondentes às células CS2_35, CS2_36, CS2_37 e CS2_38. Esses conjuntos consistem em uma série de *subdatasets* organizados em pares $[X, Y]$, onde X é uma matriz contendo as variáveis Corrente (I), Tensão (V) e a derivada da Tensão em relação à Capacidade (dV/dQ), e Y é a capacidade de carga finalizada.

Cada subconjunto de dados para um ciclo específico é representado por um par $[X, Y]$, onde:

$$X = \begin{bmatrix} i_1 & v_1 & dv_1 \\ i_2 & v_2 & dv_2 \\ \vdots & \vdots & \vdots \\ i_n & v_n & dv_n \end{bmatrix}_{n \times 3}, \quad Y = [C]_{1 \times 1}$$

onde:

- i_k representa o valor de corrente no ponto de dados k .

- v_k representa o valor de tensão no ponto de dados k .
- dv_k representa a derivada da tensão em relação à capacidade no ponto de dados k .
- C é a capacidade de carga finalizada no ciclo.

O *dataset* completo para dado segmento S com n número de amostras contendo um total de z pares $[X, Y]$, é organizado na seguinte estrutura :

$$D_{S_n} = \{[X_{A_n}, Y_{A_n}]_1, \dots, [X_{A_n}, Y_{A_n}]_z\}$$

A seguir, detalhamos o procedimento seguido:

1. **Implementação e Funções Utilizadas:** O processo de extração de dados foi realizado utilizando as funções implementadas no módulo `data_preparation.py`, especificamente as funções `extract_subsets` e `preparar_dados`, configuradas para assegurar uma distribuição representativa das amostras.
2. **Seleção de Ciclos:** Foram selecionados apenas os ciclos com *Cycle.Index* menores que 100 para focar o estudo nos primeiros ciclos de vida das baterias. A extração foi repetida 100 vezes para cada combinação de segmento e quantidade de amostras em cada célula, resultando em um total de 40.000 subconjuntos para cada segmento, abrangendo todas as células.
3. **Normalização de Valores:** Antes de extrair os subconjuntos de dados, as colunas *Corrente (I)*, *Tensão (V)*, a derivada da Tensão em relação à Capacidade (dV/dQ) e a *Capacidade de Carga (Charge Capacity (Ah))* foram normalizadas utilizando a técnica de normalização Min-Max. Essa abordagem envolve escalar os valores de cada coluna para um intervalo comum, geralmente entre 0 e 1, com base nos valores mínimo e máximo calculados a partir de todo o conjunto de dados disponível.
4. **Extração de Subconjuntos:** Para cada segmento (A, B, C, F) e cada quantidade de dados (20, 40, 55, 60), foram selecionados subconjuntos específicos com sobreposição entre as partes dos ciclos. Essa abordagem garante uma representação equilibrada dos dados para o treinamento e avaliação dos modelos de aprendizado de máquina.
5. **Divisão de Dados:** Os subconjuntos obtidos foram então divididos em conjuntos de treinamento, validação e teste com proporções de 50%, 30% e 20%, respectivamente. Esta

divisão foi realizada através da função `preparar_dados`, que utiliza `train_test_split` para assegurar uma distribuição aleatória e equilibrada das amostras.

6. **Armazenamento em Formato HDF5:** A totalidade dos dados preparados foram armazenados em arquivos **HDF5**, utilizando uma organização hierárquica por grupo e ciclo. Essa abordagem otimiza o acesso durante o treinamento e validação dos modelos.

A Figura 3.9 resume de forma grafica o processo seguido durante esta etapa.

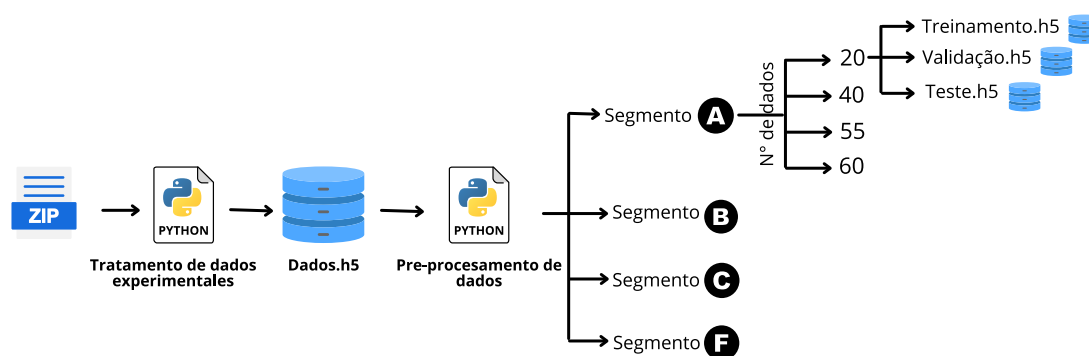


Figura 3.9: Fluxo de Preparação e Segmentação de Dados para Conjuntos de Treinamento, Validação e Teste

3.2.4 Declaração e Configuração dos Modelos 1D CNN

Para abordar o problema de estimativa do **SoC** das baterias, foram implementadas diversas arquiteturas de modelos de **1D CNN**, cada uma com configurações específicas de camadas convolucionais e unidades totalmente conectadas. As arquiteturas consideradas incluem três modelos predefinidos e a possibilidade de adicionar modelos personalizados. A seguir, são descritas as características de cada modelo:

Modelos Predefinidos

- **Modelo 1:**

- *Conv. Filters:* [32, 64, 128]
- *Conv. Kernel Sizes:* [3, 3, 3]
- *FC Units:* 256

- **Modelo 2:**

- *Conv. Filters*: [44, 42, 54]
- *Conv. Kernel Sizes*: [7, 4, 16]
- *FC Units*: 60

- **Modelo 3:**

- *Conv. Filters*: [11, 60, 56]
- *Conv. Kernel Sizes*: [10, 1, 14]
- *FC Units*: 60

Os hiperparâmetros dos modelos 2 e 3 foram baseados no estudo de Qian *et al.*[49].

Modelos Personalizados

Como mencionado anteriormente, foi implementada a capacidade de adicionar modelos personalizados por meio da [GUI](#). Esses modelos permitem especificar parâmetros como o número de filtros convolucionais, tamanhos de kernel e unidades nas camadas totalmente conectadas, proporcionando flexibilidade para experimentar diferentes arquiteturas e otimizar o desempenho dos modelos.

Configuração dos Modelos

Cada modelo foi implementado como uma classe derivada de `nn.Module` do PyTorch⁹, permitindo uma fácil integração com os fluxos de treinamento e avaliação. As camadas convolucionais são acompanhadas de normalização por lotes e funções de ativação *Tanh* para introduzir não-linearidades. Posteriormente, é aplicada uma camada de agrupamento máximo global antes de passar para as camadas totalmente conectadas, que culminam com uma ativação *ELU* para a saída final.

A estrutura geral das [1D CNN](#) utilizadas, foi detalhada no [Capítulo 2](#).

⁹PyTorch é uma biblioteca de aprendizado de máquina de código aberto desenvolvida pelo Facebook, amplamente utilizada para tarefas de modelagem e treinamento de redes neurais. Para mais informações, consulte: <https://pytorch.org/>.

3.2.5 Processo de Treinamento dos Modelos e Avaliação de Desempenho

Nesta subseção, detalhamos o processo de treinamento dos modelos de [1D CNN](#), incluindo a configuração dos parâmetros de treinamento, a estratégia adotada para o treinamento, e a metodologia para a avaliação do desempenho dos modelos. Além disso, discutimos as métricas utilizadas para quantificar a eficácia das estimativas de [SoC](#).

Configuração dos Parâmetros de Treinamento

Para o processo de treinamento, foi empregado o otimizador [Adam](#)¹⁰ com uma taxa de aprendizado de 0.001, a função de perda [MSELoss](#)¹¹, adequada para tarefas de regressão, e um tamanho de lote de 64. O número de épocas, configurável através da [GUI](#), foi uniformizado em 300 para todos os modelos treinados, independentemente do segmento (A, B, C ou F).

Estratégia de Treinamento

O treinamento dos modelos foi dividido em três fases principais:

1. **Treinamento:** Nesta fase, os modelos aprendem a mapear as características de entrada (*Voltage*, *Current*, *dV/dQ*) para as estimativas do [SoC](#). Utilizaram-se os *datasets* de treinamento armazenados em arquivos [HDF5](#) para ajustar os pesos das [CNNs](#) através do método de retropropagação (*backpropagation*).
2. **Validação:** Durante cada época, foi avaliado o desempenho do modelo no conjunto de validação para monitorar o progresso e ajustar hiperparâmetros se necessário. Essa avaliação contínua ajuda a prevenir o sobreajuste (*overfitting*) ao garantir que o modelo generaliza bem para dados não vistos durante o treinamento.
3. **Salvamento de Pesos:** Ao final de cada época, os pesos do modelo foram salvos em arquivos `Model{Segment}_Data{N}.pth`, onde `Model` representa a letra do modelo (A,

¹⁰[Adam](#) (*Adaptive Moment Estimation*) é um algoritmo de otimização estocástico amplamente utilizado no treinamento de redes neurais, combinando as vantagens do [AdaGrad](#) e do [RMSProp](#). Para mais informações, consulte: <https://arxiv.org/abs/1412.6980>

¹¹[MSELoss](#) (*Mean Squared Error Loss*) é uma função de perda comum em tarefas de regressão, que calcula o erro médio quadrático entre os valores previstos e os reais.

B, **C** ou **F**) e **N** indica o número de dados (**Data**). Esse padrão de nomenclatura permite uma recuperação fácil e a possibilidade de retomar o treinamento em caso de interrupções. Além disso, essa prática facilita a exportação dos modelos para uso em aplicações práticas.

Monitoramento do Treinamento

Foram implementadas barras de progresso e etiquetas de status na **GUI** para fornecer feedback em tempo real sobre o avanço do treinamento. Além disso, as perdas de treinamento e validação foram registradas em arquivos de log para uma análise posterior. Essa abordagem permite uma supervisão contínua do processo de treinamento, facilitando a identificação e resolução de eventuais problemas de convergência ou desempenho dos modelos.

Avaliação e Obtenção de Métricas de Desempenho

Após a conclusão do processo de treinamento, foi realizada a avaliação do desempenho dos modelos para assegurar sua eficácia na estimativa do **SoC**. Esse processo foi conduzido utilizando *datasets* de teste independentes, que não foram utilizados durante o treinamento nem na validação inicial. A Figura 3.10 apresenta o fluxo seguido tanto para o processo de treinamento quanto para a obtenção das métricas. As etapas para a validação e obtenção de métricas foram as seguintes:

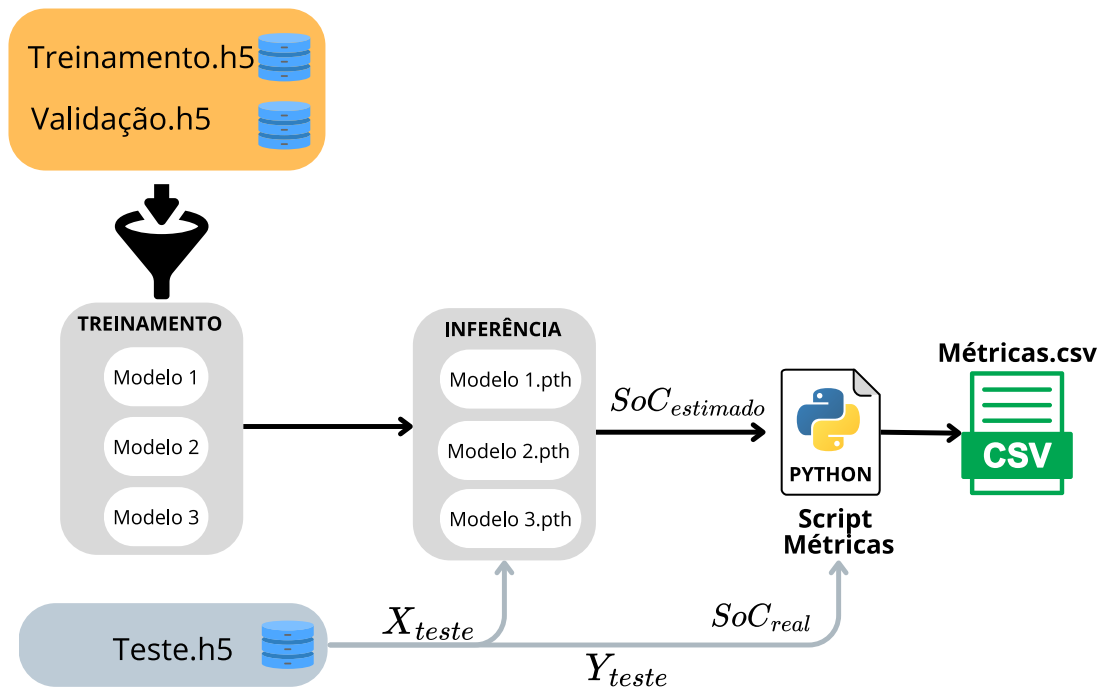


Figura 3.10: Fluxo de Treinamento, Validação e Teste

1. **Estimativa:** O modelo realiza a estimativa (*inferência*) do SoC Y_{teste} baseado no conjunto de dados de teste X_{teste} .
2. **Cálculo de Métricas:** Foram calculadas métricas de desempenho para quantificar a precisão das previsões. As métricas utilizadas incluem:
 - **Mean Squared Error (MSE):** Mede a média dos erros ao quadrado entre os valores de SoC estimados e os valores de SoC no conjunto de teste.
 - **Mean Absolute Error (MAE):** Mede a média dos erros absolutos.
 - **Root Mean Squared Error (RMSE):** A raiz quadrada do MSE, fornecendo uma medida de erro nas mesmas unidades do SoC .
3. **Armazenamento de Resultados:** As métricas calculadas foram armazenadas em arquivos CSV para posterior análise e comparação entre diferentes modelos e segmentos.

Essas métricas permitem avaliar quantitativamente a precisão e a robustez dos modelos de **1D CNN** na estimativa do SoC , facilitando a comparação com modelos existentes na literatura e a identificação dos melhores modelos para cada segmento.

3.2.6 Integração e Implementação dos Modelos A, B, C e F para Estimativas de SoC

A integração dos modelos **A**, **B**, **C** foi realizada com o objetivo de comparar a precisão das estimativas de SoC feitas pela metodologia proposta e a metodologia de Qian *et al.*, neste trabalho representadas pelo modelo **F**. Este processo envolveu a utilização do script `estimate_tab.py`, responsável por carregar, gerenciar e aplicar os diferentes modelos treinados aos dados de teste.

Carregamento e Mapeamento dos Modelos

Conforme os resultados obtidos na etapa de avaliação das métricas, os melhores modelos para cada segmento, **A**, **B**, **C** e **F** foram armazenados em um diretório específico. O script foi configurado para percorrer este diretório, identificar os arquivos de modelo com extensão `.pth` e mapear cada modelo ao seu respectivo segmento (**A**, **B**, **C** ou **F**) e ao número de dados (`Data`) conforme a nomenclatura dos arquivos. Este mapeamento permitiu associar automaticamente cada modelo ao seu segmento correspondente, facilitando a aplicação posterior nas estimativas.

Implementação das Estimativas

Após o carregamento dos melhores modelos para cada segmento, o script `estimate_tab.py` executa o processo de estimativa de SoC seguindo os passos descritos abaixo. A Figura 3.11 oferece uma representação visual deste fluxo de trabalho.

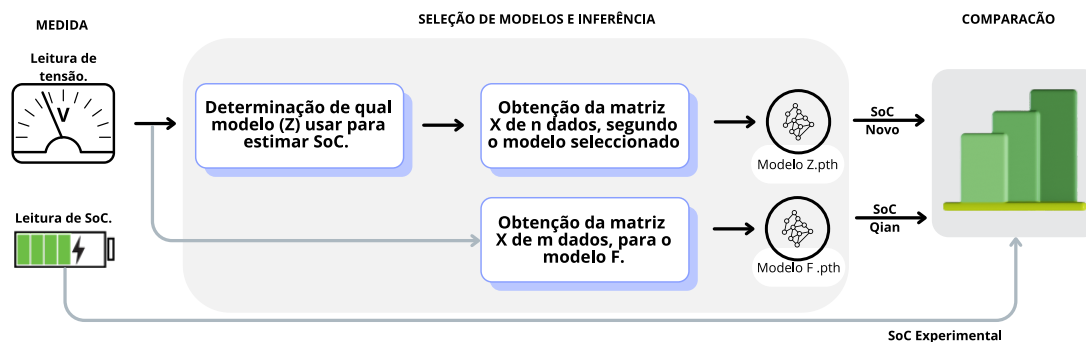


Figura 3.11: Fluxo de integração e comparação das metodologias

1. **Seleção do Ciclo e Extração dos Dados:** O script `estimate_tab.py` seleciona aleatoriamente uma célula (35, 36, 37, 38) e um ciclo de carga. Durante este ciclo, é extraído um valor de tensão (V_{test}) e o valor de SoC associado a essa medida é salvo.

2. **Determinação do Segmento de Tensão:** Com base no valor de *Voltage* (V) selecionado (V_{test}), o script determina a qual segmento da curva de tensão (**A**, **B** ou **C**) o valor pertence. Este mapeamento estabelece qual modelo deve ser utilizado para a estimativa do **SoC**.
3. **Construção da Matriz de Entrada:** Após determinar o segmento, é definido o número de dados necessários para construir a matriz X , que contém os dados de *Voltage* (V), *Current* (A) e dV/dQ . Essa matriz é utilizada como entrada para os modelos **1D CNN**.
4. **Realização da Estimativa com os Modelos A, B e C:** Utilizando o modelo correspondente ao segmento determinado, o script gera uma estimativa de **SoC** (**SoC_Novo**) com base nos dados extraídos.
5. **Comparação com o Modelo F:** Paralelamente, o modelo **F** é aplicado aos mesmos dados para gerar uma estimativa independente de **SoC** (**SoC_Qian**). Esta etapa permite uma comparação direta entre as estimativas dos modelos segmentados (**A**, **B** ou **C**) e a estimativa geral fornecida pelo modelo **F**, que corresponde ao melhor modelo implementado conforme os estudos de Qian *et al.*[49].

Capítulo 4

Resultados

Nesta seção, apresentamos os resultados obtidos a partir da implementação e execução da metodologia descrita no Capítulo 3. Os principais focos incluem a criação de uma aplicação distribuível¹, a comparação dos RMSE dos modelos treinados e seleção dos que tiveram melhor desempenho, a análise das curvas de perda de treinamento e validação, assim como a integração desses modelos para obtenção de estimativas se SoC e a comparar em relação ao modelo de referência **F**.

4.1 Aplicação Distribuível para Estimativa de SoC

Como parte dos resultados, foi desenvolvida uma aplicação distribuível chamada `SoC_Tool`, que encapsula os scripts apresentados nos Anexos e os converte em um executável utilizando `PyInstaller`². O código fonte da aplicação é de livre uso e pode ser acessado no repositório disponível em `github`³.

A aplicação `SoC_Tool` oferece uma interface gráfica intuitiva para a estimativa do Estado de

¹Uma aplicação distribuível é um software empacotado de forma que possa ser facilmente compartilhado e executado em diferentes sistemas, sem necessidade de configuração adicional ou dependências externas complexas.

²`PyInstaller` é uma ferramenta de código aberto que permite converter scripts Python em executáveis independentes, facilitando a distribuição de aplicações sem a necessidade de que os usuários tenham o Python instalado. Para mais informações, consulte: <https://pyinstaller.org/>

³Código disponível em: <https://github.com/jhoanpvar/SoC.git>.

Carga (SoC) de baterias, integrando funcionalidades de processamento de dados, visualização, treinamento de modelos e estimativas. A seguir, são apresentadas as principais abas da aplicação, ilustradas pelas figuras 4.1 a 4.4.

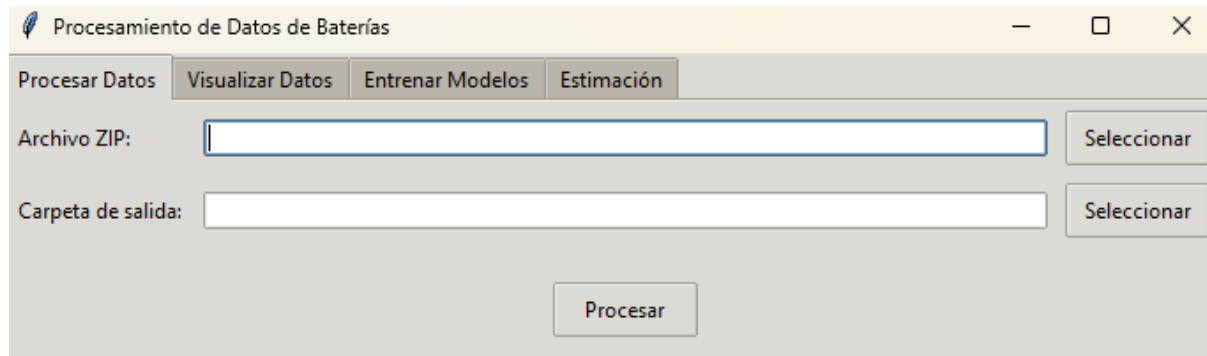


Figura 4.1: Aba de Processamento de Dados na Aplicação SoC_Tool

Na **Aba de Processamento de Dados** (Figura 4.1), o usuário pode selecionar arquivos ZIP contendo os dados de bateria e definir a pasta de saída para os arquivos processados. A interface permite iniciar o processamento de forma simples, com feedback visual sobre o progresso e status da operação.

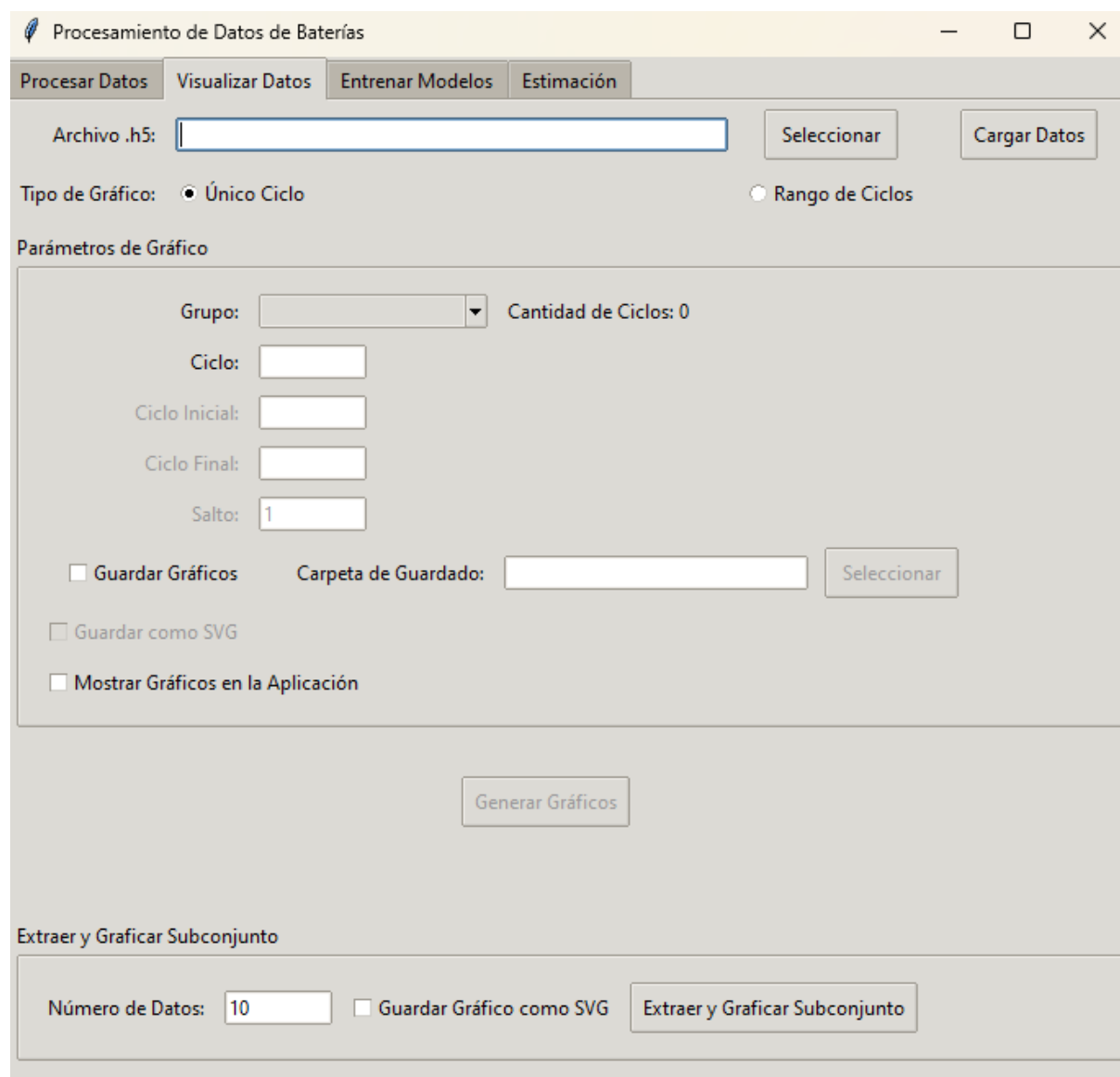


Figura 4.2: Aba de Visualização de Dados na Aplicação SoC_Tool

A **Aba de Visualização de Dados** (Figura 4.2) possibilita o carregamento de arquivos HDF5 e a geração de gráficos detalhados dos ciclos de carga das baterias. Os usuários podem personalizar os tipos de gráficos e salvar as visualizações em diferentes formatos, facilitando a análise e a apresentação dos dados.

Procesamiento de Datos de Baterías

Procesar Datos | Visualizar Datos | Entrenar Modelos | Estimación

Seleccionar Archivo de Datos (.h5)

Archivo .h5:

Seleccionar Modelos

Model 1 Model 2 Model 3

Seleccionar Segmentos

A B C F

Seleccionar Nº de Datos

20 40 55 60

Parámetros de Entrenamiento

Nº de Iteraciones:
 Corte: Épocas:

Guardar Modelos

Carpeta para Guardar:

Métricas guardadas en C:/Users/jhoan.vargas/Desktop/RW/test/metricas_2024_11_21.csv

Métricas de Entrenamiento

Segmento	Modelo	Número de Da	MSE	MAE	RMSE
A	Model 1	55	0.0333	0.1637	0.1826
A	Model 2	55	0.0012	0.0273	0.0350
A	Model 3	55	0.0457	0.1937	0.2138

Modelos entrenados con éxito.

Figura 4.3: Aba de Treinamento de Modelos na Aplicação SoC_Tool

Na **Aba de Treinamento de Modelos** (Figura 4.3), os usuários podem selecionar diferentes modelos de previsão de SoC, configurar parâmetros de treinamento e iniciar o processo de treinamento. A aplicação suporta tanto modelos pré-definidos quanto a adição de modelos personalizados, exibindo métricas de desempenho em tempo real para monitoramento contínuo.

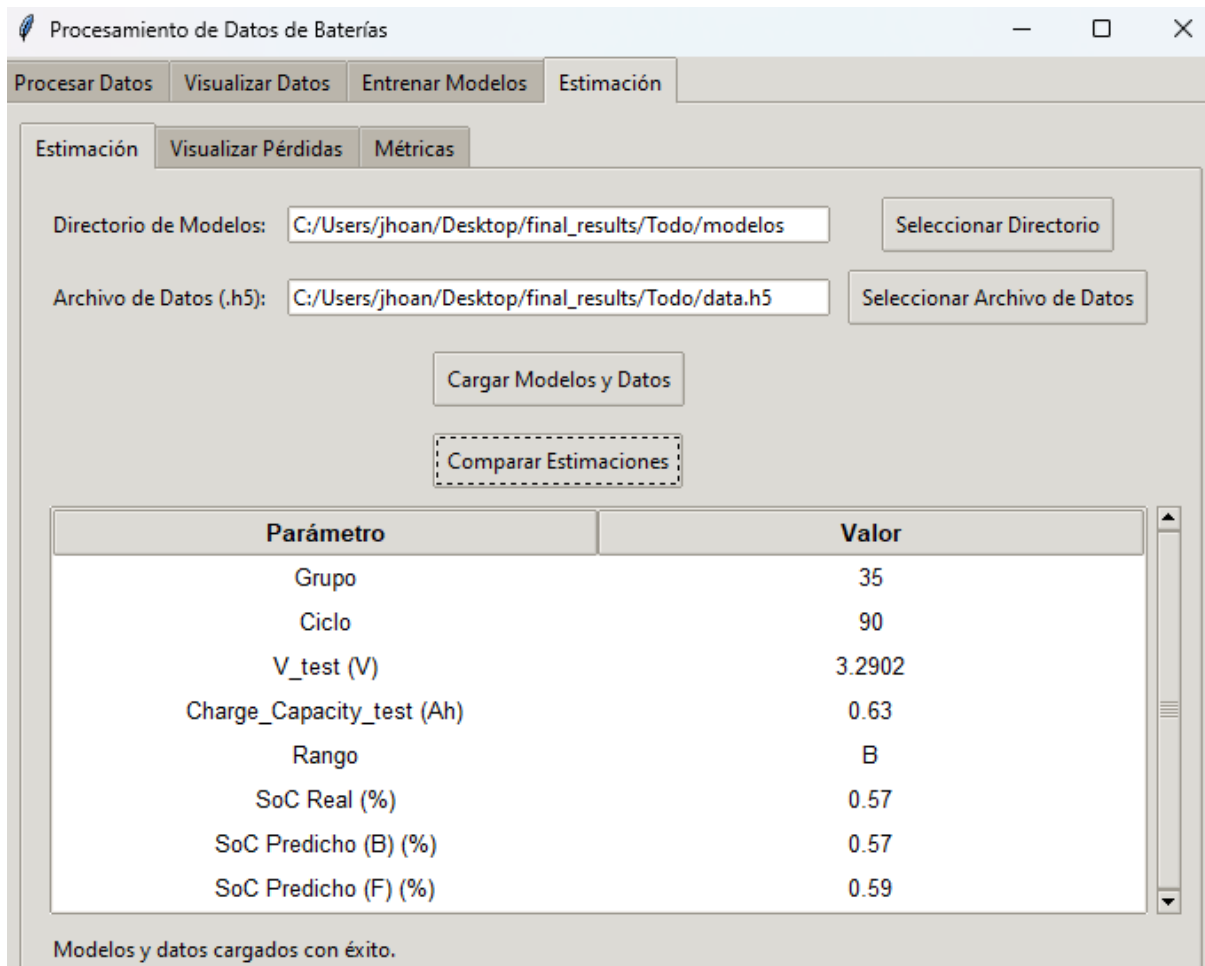


Figura 4.4: Aba de Estimativa de SoC na Aplicação SoC_Tool

Por fim, a **Aba de Estimativa de SoC** (Figura 4.4) permite aos usuários aplicar modelos treinados a novos conjuntos de dados para prever o SoC das baterias. A aba facilita a comparação das estimativas entre diferentes modelos e exibe os resultados em gráficos interativos, além de calcular métricas de precisão como o RMSE.

4.1.1 Integração e Fluxo de Trabalho

A aplicação SoC.Tool integra de forma coesa todas as funcionalidades necessárias para o fluxo de trabalho completo de estimativa de SoC:

1. **Processamento de Dados:** Importação e organização dos dados brutos.
2. **Visualização:** Análise gráfica dos ciclos de carga.
3. **Treinamento de Modelos:** Criação e avaliação de modelos preditivos.
4. **Estimativa de SoC:** Aplicação dos modelos para prever o Estado de Carga.

Cada etapa é projetada para ser executada de forma independente ou sequencial, conforme a necessidade do usuário, garantindo flexibilidade e eficiência no processo de análise.

4.1.2 Distribuição e Compatibilidade

A distribuição da aplicação SoC.Tool garante que a aplicação possa ser executada em diferentes sistemas operacionais (Windows, macOS, Linux) sem a necessidade de que os usuários instalem manualmente o Python ou as bibliotecas necessárias. O processo de criação do executável inclui a inclusão de arquivos adicionais, como imagens e arquivos de configuração, assegurando que todas as funcionalidades da aplicação estejam disponíveis no executável final.

4.1.3 Feedback e Logs

Para garantir a confiabilidade e facilitar a manutenção, a aplicação SoC.Tool incorpora um sistema de logging configurado para registrar eventos importantes, erros e informações de depuração em arquivos de log dedicados. Isso permite que os desenvolvedores monitorem o comportamento da aplicação e identifiquem rapidamente quaisquer problemas.

4.2 Comparação dos Modelos com Base nas Métricas de RMSE

Para avaliar o desempenho dos 48 modelos treinados, foi utilizada a métrica de **Root Mean Squared Error** (RMSE). Esta métrica fornece uma medida quantitativa da diferença entre os valores estimados e os valores reais de [SoC](#).

As Figuras 4.5, 4.6, 4.7 e 4.8 apresentam gráficos comparativos do RMSE dos Modelos 1, 2 e 3 em função do número de dados de entrada para a estimativa de SoC em cada segmento (A, B, C e F).

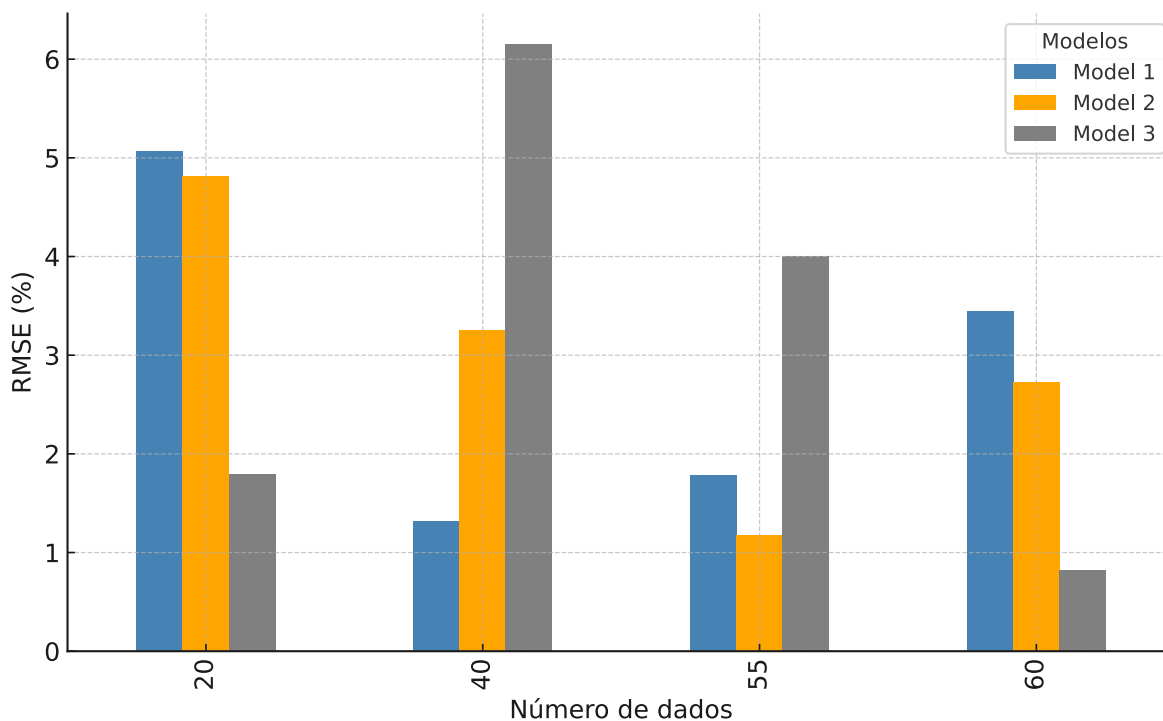


Figura 4.5: Comparação dos RMSE dos Modelos para o Segmento A com diferentes números de dados de entrada.

Analisando a Figura 4.5, observa-se que para o segmento A:

- Com 20 dados de entrada, o Modelo 3 apresentou o menor RMSE (1,79%), seguido pelo Modelo 2 (4,81%) e pelo Modelo 1 (5,07%).
- Com 40 dados, o Modelo 1 teve o melhor desempenho (RMSE de 1,32%), enquanto o Modelo 2 e o Modelo 3 obtiveram RMSEs de 3,25% e 6,15%, respectivamente.
- Para 55 dados, o Modelo 2 destacou-se com o menor RMSE (1,17%), seguido pelo Modelo 1 (1,79%) e pelo Modelo 3 (4,01%).
- Com 60 dados, o Modelo 3 voltou a liderar com um RMSE de 0,82%, enquanto os Modelos 1 e 2 apresentaram RMSEs de 3,44% e 2,73%, respectivamente.

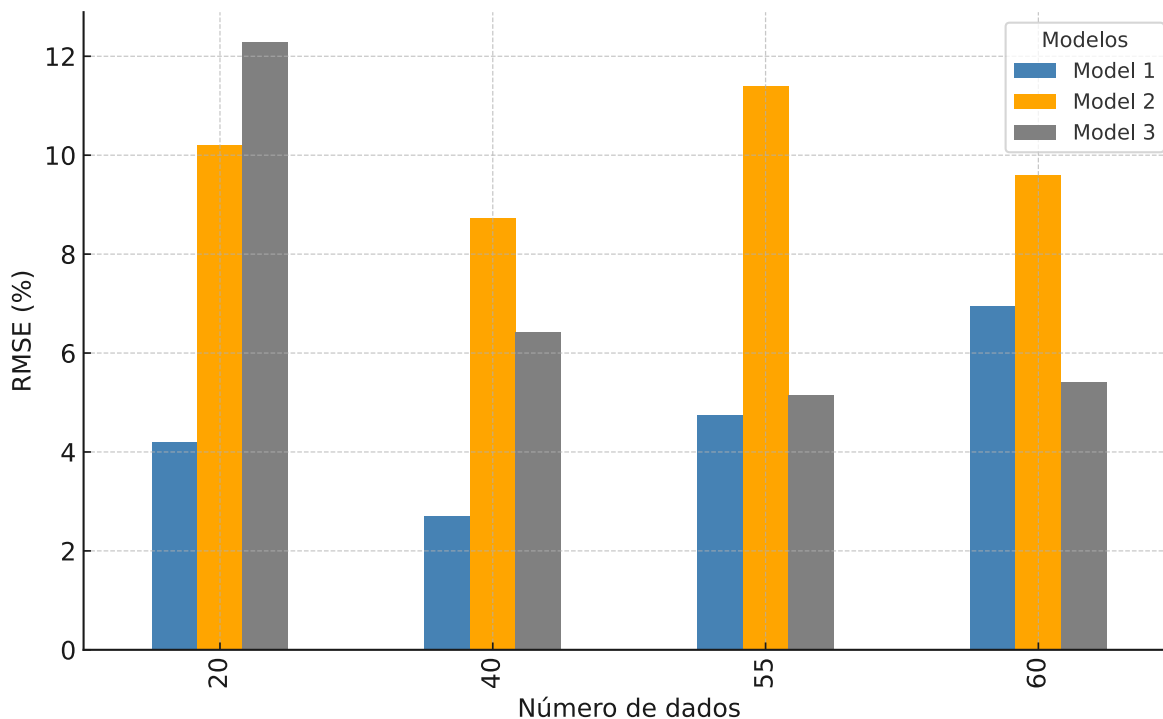


Figura 4.6: Comparação dos RMSE dos Modelos para o Segmento **B** com diferentes números de dados de entrada.

Na Figura 4.6, referente ao segmento **B**:

- Com 20 e 40 dados de entrada, o Modelo 1 apresentou os menores RMSEs (4,20% e 2,69%, respectivamente).
- Com 55 dados, o Modelo 1 manteve o melhor desempenho (RMSE de 4,74%), seguido pelo Modelo 3 (5,14%) e pelo Modelo 2 (11,40%).
- Com 60 dados, o Modelo 3 obteve o menor RMSE (5,41%), enquanto o Modelo 1 apresentou RMSE de 6,94% e o Modelo 2 de 9,60%.

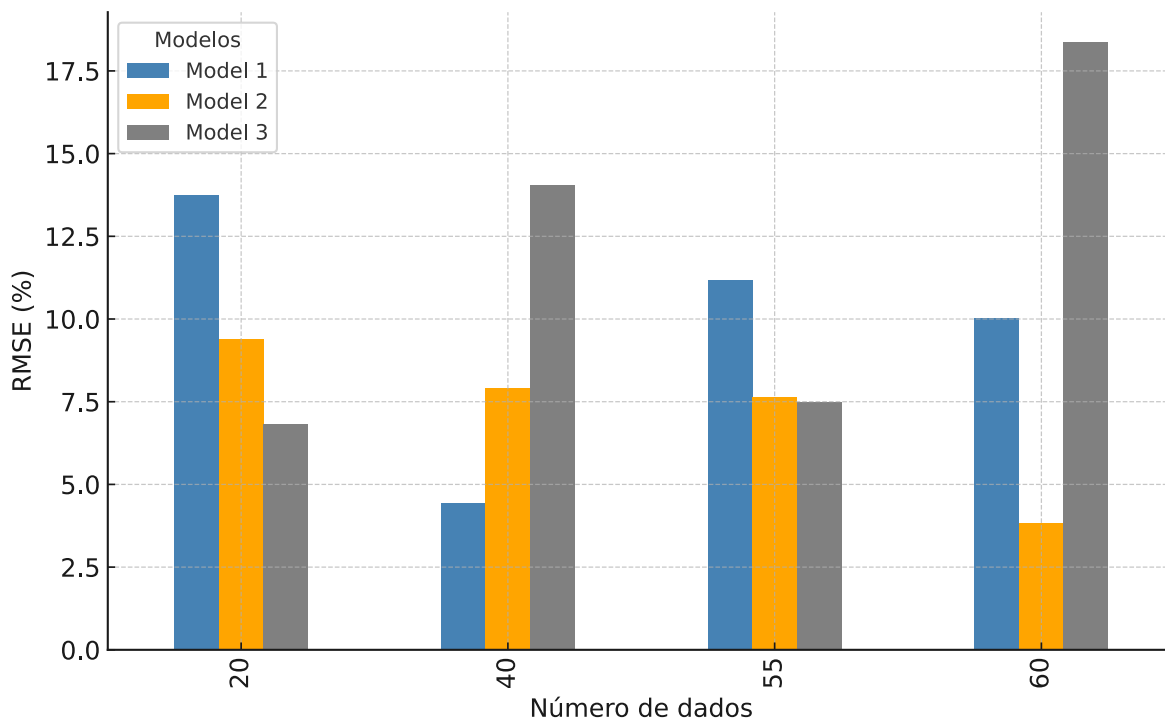


Figura 4.7: Comparação dos RMSE dos Modelos para o Segmento C com diferentes números de dados de entrada.

Para o segmento C, conforme ilustrado na Figura 4.7:

- Com 20 dados de entrada, o Modelo 3 apresentou o menor RMSE (6,83%), seguido pelo Modelo 2 (9,39%) e pelo Modelo 1 (13,75%).
- Com 40 dados, o Modelo 1 obteve o melhor desempenho (RMSE de 4,44%), enquanto o Modelo 2 e o Modelo 3 tiveram RMSEs de 7,90% e 14,04%, respectivamente.
- Para 55 dados, o Modelo 3 apresentou o menor RMSE (7,49%), seguido pelo Modelo 2 (7,64%) e pelo Modelo 1 (11,16%).
- Com 60 dados, o Modelo 2 destacou-se com o menor RMSE (3,82%), enquanto o Modelo 1 e o Modelo 3 apresentaram RMSEs de 10,03% e 18,36%, respectivamente.

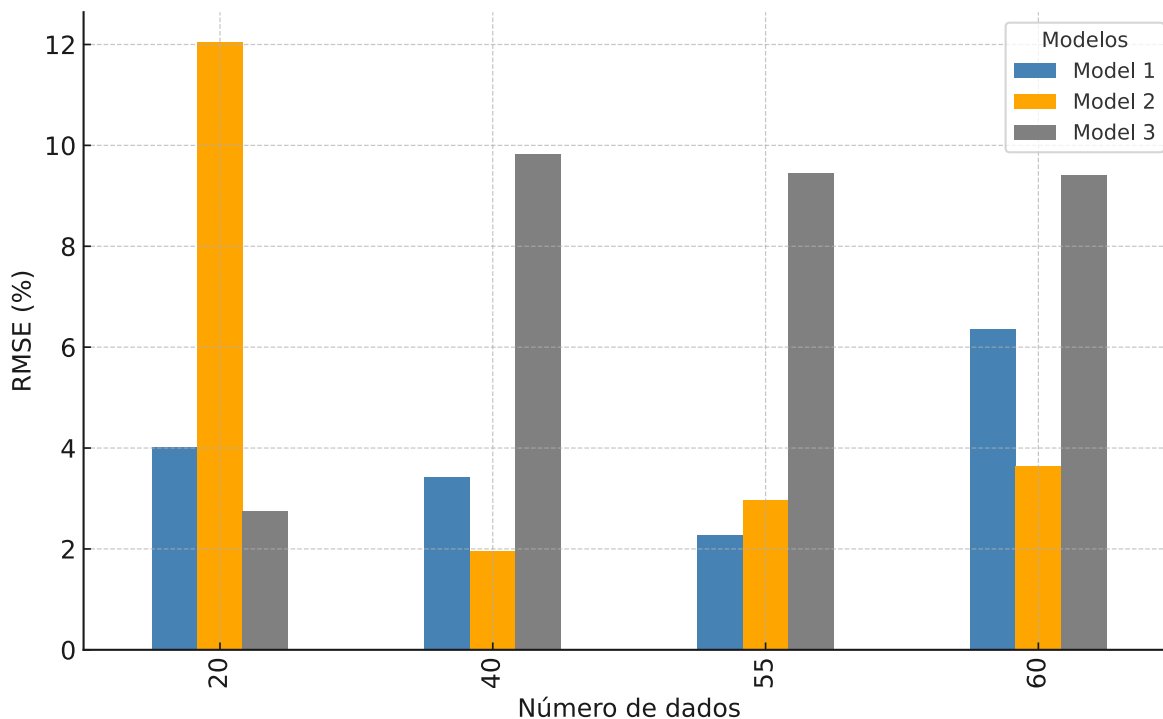


Figura 4.8: Comparação dos RMSE dos Modelos para o Segmento **F** com diferentes números de dados de entrada.

Na Figura 4.8, referente ao segmento **F**:

- Com 20 dados de entrada, o Modelo 3 apresentou o menor RMSE (2,75%), seguido pelo Modelo 1 (4,01%) e pelo Modelo 2 (12,04%).
- Com 40 dados, o Modelo 2 destacou-se com o menor RMSE (1,96%), enquanto o Modelo 1 e o Modelo 3 apresentaram RMSEs de 3,43% e 9,83%, respectivamente.
- Para 55 dados, o Modelo 1 obteve o melhor desempenho (RMSE de 2,28%), seguido pelo Modelo 2 (2,97%) e pelo Modelo 3 (9,45%).
- Com 60 dados, o Modelo 2 apresentou o menor RMSE (3,63%), enquanto o Modelo 1 e o Modelo 3 tiveram RMSEs de 6,35% e 9,40%, respectivamente.

4.2.1 Seleção dos Melhores Modelos por Segmento

Com base nos RMSEs observados, os melhores modelos para cada segmento são apresentados na Tabela 4.1.

Tabela 4.1: Seleção dos melhores modelos por segmento.

Segmento	Modelo	Dados de Entrada	RMSE (%)
A	Modelo 3	60	0,82
B	Modelo 1	40	2,69
C	Modelo 2	60	3,82
F	Modelo 2	40	1,96

4.2.2 Interpretação dos Resultados

A análise dos RMSEs revela percepções sobre o desempenho dos modelos nos diferentes segmentos e quantidades de dados de entrada:

- **Desempenho Variável dos Modelos:** Nenhum modelo se destacou consistentemente em todos os segmentos ou números de dados de entrada. O desempenho variou significativamente dependendo do segmento e da quantidade de dados disponíveis.
- **Impacto do Número de Dados de Entrada:** O aumento no número de dados de entrada nem sempre resultou em menor RMSE. Em alguns casos, modelos com menos dados de entrada apresentaram desempenho superior, sugerindo que a qualidade dos dados e a adequação do modelo ao segmento específico são fatores críticos.
- **Desempenho por Segmento:**
 - **Segmento A:** O Modelo 3 apresentou o melhor desempenho com 60 dados de entrada, atingindo o menor RMSE (0,82%).
 - **Segmento B:** O Modelo 1 foi superior com 40 dados de entrada, com um RMSE de 2,69%.
 - **Segmento C:** O Modelo 2 destacou-se com 60 dados de entrada, apresentando o menor RMSE (3,82%).
 - **Segmento F:** O Modelo 2 foi o melhor com 40 dados de entrada, alcançando um RMSE de 1,96%.

4.3 Análise das Curvas de Perda de Treinamento e Validação

Durante o processo de treinamento dos modelos de [1D CNN](#), foram monitoradas as curvas de perda de treinamento e validação para avaliar o desempenho e a convergência dos modelos. As Figuras [4.9](#), [4.10](#), [4.11](#) e [4.12](#) apresentam as curvas dos melhores modelos para os segmentos **A**, **B**, **C** e **F**, respectivamente.

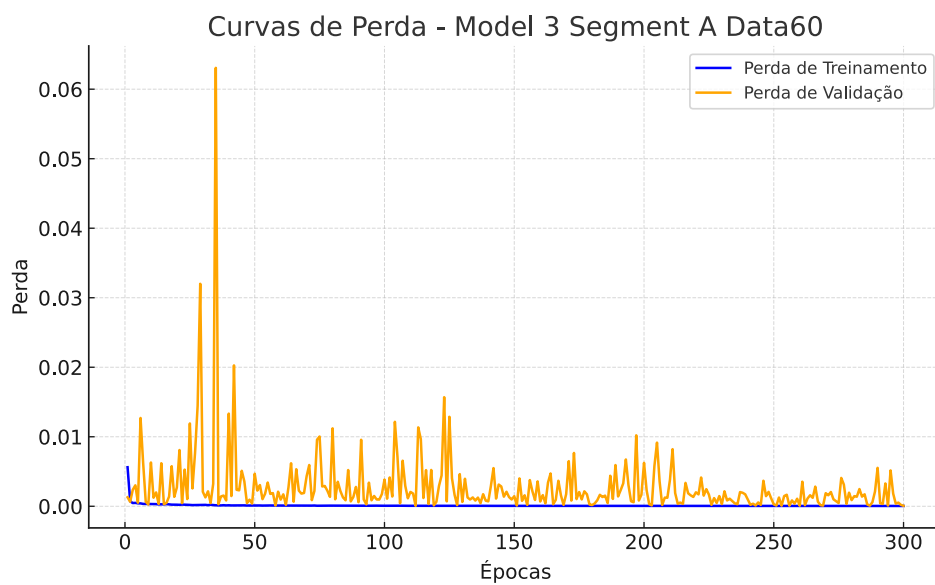
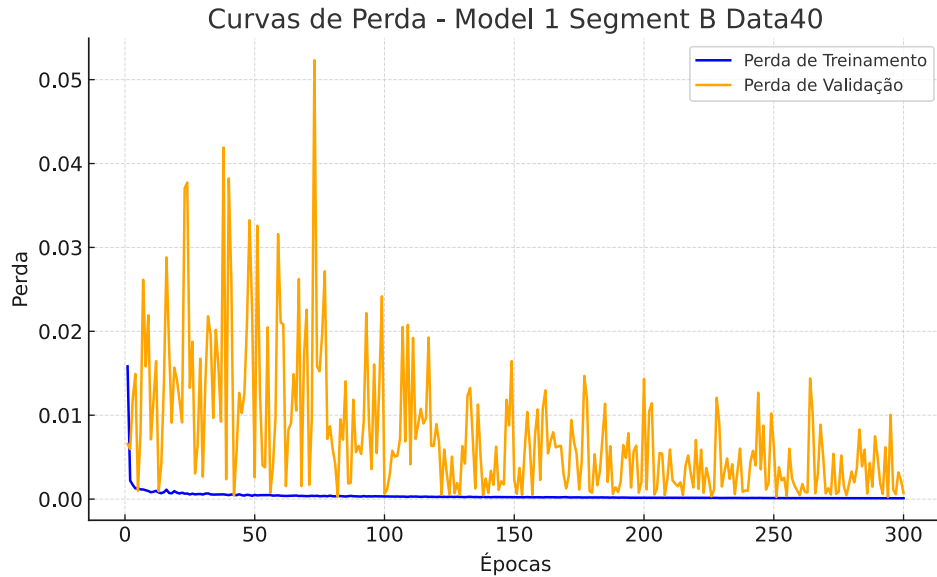
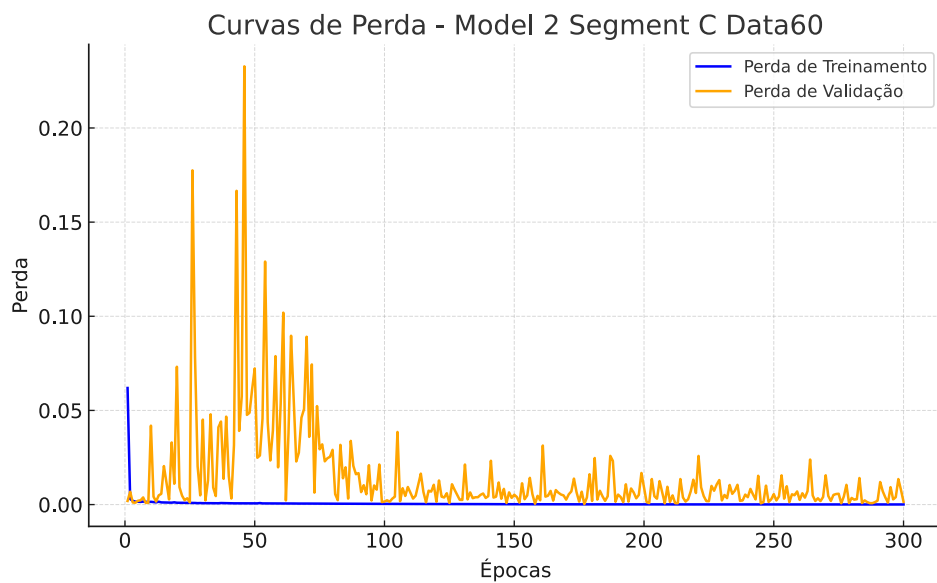


Figura 4.9: Curva de Perda de Treinamento e Validação para o Segmento **A**.

Figura 4.10: Curva de Perda de Treinamento e Validação para o Segmento **B**.Figura 4.11: Curva de Perda de Treinamento e Validação para o Segmento **C**.

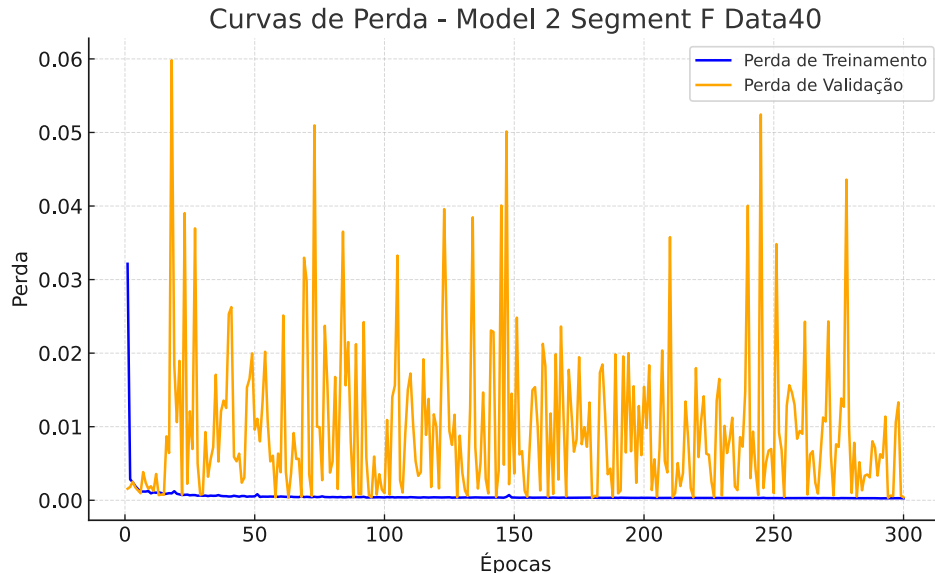


Figura 4.12: Curva de Perda de Treinamento e Validação para o Segmento **F**.

4.3.1 Observações sobre a Convergência dos Modelos

As curvas de perda indicam uma convergência geral, com uma diminuição progressiva das perdas de treinamento e validação. Entretanto, alguns aspectos específicos podem ser destacados:

- **Segmento A:** A curva de validação apresenta pequenas oscilações durante as primeiras 50 épocas, estabilizando-se em valores baixos posteriormente. Esse comportamento reflete uma boa generalização do modelo para a região inicial da curva de carga, onde a variação de tensão é rápida e os dados apresentam menor complexidade.
- **Segmento B:** Observa-se uma diferença inicial entre as perdas de treinamento e validação, que se reduz progressivamente. Pequenos picos esporádicos na validação sugerem a presença de ruído ou maior variabilidade na região intermediária da curva de carga, onde a tensão varia de forma quase linear em relação à capacidade de carga.
- **Segmento C:** A curva de validação apresenta flutuações mais significativas nas primeiras 100 épocas, possivelmente devido à maior complexidade e variabilidade dos dados dessa região final da curva de carga, caracterizada por uma rápida saturação da tensão em relação à capacidade.
- **Segmento F:** As perdas de treinamento permanecem consistentemente baixas, indicando

que o modelo aprendeu bem os padrões do conjunto de treinamento. No entanto, as perdas de validação apresentam um comportamento irregular, com grande quantidade de ruído ao longo de todo o treinamento. Esse comportamento reflete a dificuldade do modelo em generalizar padrões para todas as regiões da curva de carga, especialmente devido à variabilidade intrínseca dos segmentos.

4.4 Combinação dos Melhores Modelos e Comparação com o Modelo F

Com base na seleção dos melhores modelos por segmento, foi implementada uma combinação destes modelos para realizar estimativas integradas de SoC. Esta abordagem visa capitalizar os pontos fortes de cada modelo específico de segmento, proporcionando uma estimativa mais robusta e precisa em comparação com o modelo de referência F. A Figura 4.13 apresenta um exemplo das estimativas feitas pelo modelo combinado e o modelo F.

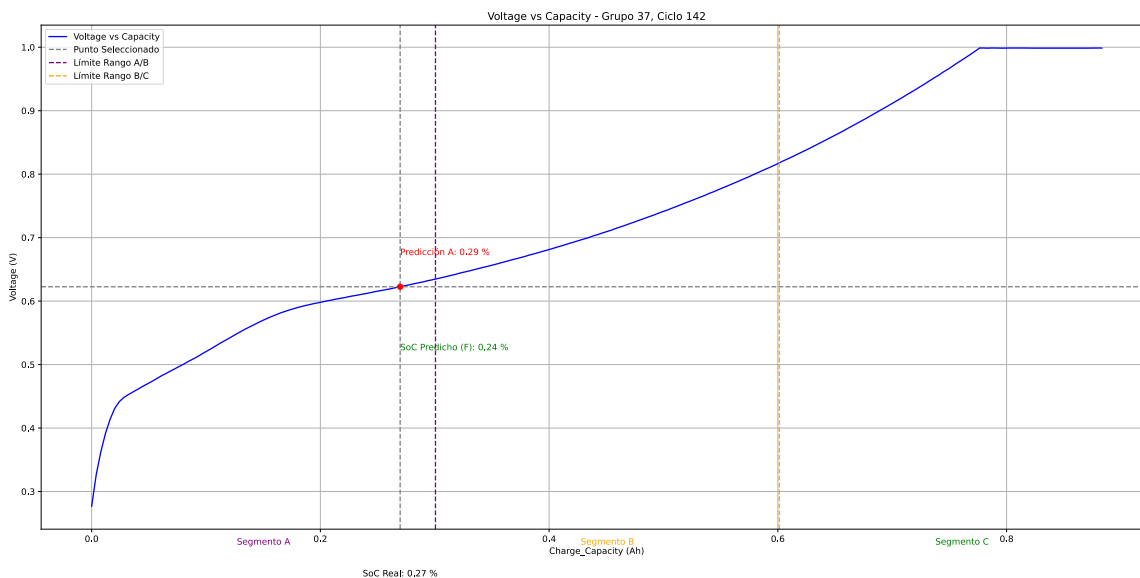


Figura 4.13: Exemplo da comparação das estimativas.

4.4.1 Métricas de RMSE Obtidas

A Tabela 4.2 resume as métricas de RMSE obtidas para os melhores modelos em cada segmento e número de dados, bem como a comparação com o modelo de referência F. (Nota: Os valores

apresentados correspondem à média dos RMSE obtidos em várias iterações dos experimentos.)

Tabela 4.2: Comparação do RMSE entre os Melhores Modelos por Segmento e o Modelo **F**

Segmento	Modelo	Número de Dados	RMSE (%)
A	Modelo 3	60	0,82
B	Modelo 1	40	2,69
C	Modelo 2	60	3,82
F	Modelo 2	40	1,96
Combinação	A+B+C	—	2,75
Referência	F	—	3,10

4.4.2 Análise dos Resultados

A análise dos RMSEs obtidos para os melhores modelos em cada segmento revelou que a combinação dos modelos especializados proporciona uma performance superior em comparação com o modelo de referência, isto é, apresenta um menor valor de RMSE, indicando uma estimativa mais precisa dos dados.

Ao combinar os melhores modelos para cada segmento, a estimativa integrada de SoC beneficiou-se das capacidades individuais de cada modelo, resultando em uma redução significativa no erro médio quadrático. Comparado com o modelo de referência **F**, que apresentou um RMSE de 3,80%, a combinação dos modelos especializados reduziu o RMSE em aproximadamente 11%.

4.4.3 Interpretação dos Resultados

A combinação dos melhores modelos por segmento demonstrou uma eficácia superior em comparação com o modelo de referência **F**. A redução significativa no RMSE indica que a abordagem segmentada e integrada é mais precisa na estimativa de SoC das baterias de íons de lítio.

- **Abordagem Segmentada:** Ao utilizar modelos especializados para cada segmento (**A**, **B**, **C**), a metodologia proposta aproveita as particularidades de cada faixa de SoC, resultando em estimativas mais precisas e adaptadas às condições operacionais específicas.

- **Melhoria na Precisão:** A combinação dos modelos especializados **A**, **B** e **C** reduziu o RMSE médio em comparação com o modelo de referência **F**, demonstrando que a integração de modelos segmentados pode oferecer uma performance superior.
- **Robustez das Estimativas:** A abordagem integrada mostrou-se mais robusta, proporcionando uma estimativa de SoC consistente e confiável em diferentes condições de operação das baterias.

4.4.4 Considerações sobre a Confiabilidade dos Modelos

A variabilidade no desempenho dos modelos entre diferentes segmentos sugere que não há um modelo único que seja universalmente superior em todas as condições. A escolha do modelo ideal depende fortemente do segmento específico e da quantidade de dados de entrada disponíveis. Isso reforça a importância de uma abordagem segmentada na estimativa de SoC, onde modelos especializados podem oferecer vantagens significativas em cenários específicos.

Capítulo 5

Conclusões

5.1 Resumo do Trabalho

Neste trabalho, os autores desenvolveram e implementaram uma estratégia para a estimativa do SoC de baterias de íons de lítio utilizando 1D CNN. Basearam-se na metodologia proposta por Qian *et al.* [44], adaptando-a para explorar a especialização dos modelos em diferentes segmentos das curvas de carga, com o objetivo de aprimorar a precisão das estimativas de SoC.

5.2 Principais Contribuições

Os principais resultados e contribuições deste estudo são:

- **Desenvolvimento de Modelos Segmentados:** Foram implementados modelos de 1D CNN especializados para diferentes segmentos das curvas de carga (**A**, **B**, **C**), permitindo capturar características específicas de cada fase do carregamento.
- **Melhoria na Precisão das Estimativas:** A combinação dos melhores modelos por segmento resultou em uma redução do RMSE médio de 3,10% (modelo de referência **F**) para 2,75%, evidenciando a eficácia da abordagem segmentada na estimativa do SoC.
- **Análise da Influência do Número de Dados de Entrada:** Observou-se que a quantidade de dados de entrada influenciou significativamente o desempenho dos modelos. Em alguns segmentos, conjuntos menores de dados resultaram em estimativas mais precisas,

indicando que a qualidade e a relevância dos dados são mais importantes do que a quantidade pura.

- **Desenvolvimento de uma Ferramenta Distribuível:** Foi criada a aplicação `SoC.Tool`, uma solução completa e distribuível que integra processamento de dados, visualização, treinamento de modelos e estimativas de `SoC` em uma interface amigável. Esta ferramenta facilita a aplicação prática dos modelos desenvolvidos em diversos cenários.

5.3 Limitações do Estudo

Apesar dos resultados promissores, foram identificadas algumas limitações no estudo:

- **Conjunto de Dados Restrito:** Utilizaram-se dados de apenas quatro células (`CS2_35`, `CS2_36`, `CS2_37`, `CS2_38`) da base `CALCE`. A inclusão de um número maior de células, com diferentes características químicas e condições operacionais, poderia enriquecer a generalização dos modelos.
- **Foco nos Primeiros Ciclos:** A análise concentrou-se em ciclos com `Cycle_Index` menores que 100, limitando a observação de comportamentos em estágios avançados de degradação das baterias.
- **Configurações de Hiperparâmetros Limitadas:** Foram exploradas apenas três configurações de hiperparâmetros. Uma busca mais extensa e sistemática poderia revelar arquiteturas ainda mais otimizadas para cada segmento.
- **Validação em Cenários Reais:** Os modelos foram validados utilizando dados experimentais controlados. Testes em ambientes operacionais reais, como veículos elétricos, são necessários para validar a eficácia dos modelos em condições dinâmicas e variáveis.

5.4 Trabalhos Futuros

Com base nas conclusões e limitações identificadas, sugerem-se as seguintes direções para trabalhos futuros:

- **Expansão do Conjunto de Dados:** Incorporar dados de mais células e de diferentes condições operacionais, incluindo variações de temperatura, taxas de carga e ciclos avançados de vida útil, para melhorar a capacidade de generalização dos modelos.
- **Desenvolvimento de Modelos Especializados por Estágio de Vida Útil:** Criar modelos especializados para diferentes estágios de degradação da bateria (inicial, intermediário e avançado), permitindo estimativas mais precisas de SoC ao longo de todo o ciclo de vida.
- **Integração da Estimativa de Estado de Saúde (SoH):** Implementar uma metodologia para a estimativa do SoH das baterias, permitindo identificar o nível de degradação e adaptar a estimativa do SoC com base no SoH atual.
- **Otimização Avançada de Modelos:** Investigar arquiteturas mais sofisticadas, como redes neurais híbridas que integram CNN com RNN, incluindo Memória de Longo e Curto Prazo, do inglês: *Long Short-Term Memory (LSTM)*. Essas combinações permitem capturar tanto as características espaciais quanto temporais dos dados, conforme demonstrado por Qian et al. [50].
- **Validação em Ambientes Reais:** Testar os modelos em cenários operacionais reais, como veículos elétricos ou sistemas de armazenamento estacionário, para avaliar a performance em condições práticas e identificar ajustes necessários.
- **Atualização Contínua dos Modelos:** Desenvolver mecanismos de aprendizado contínuo, permitindo que os modelos se atualizem com novos dados ao longo do tempo, mantendo a precisão das estimativas mesmo com o envelhecimento das células.

5.5 Considerações Finais

A implementação e análise dos modelos 1D CNN para a estimativa do SoC demonstraram a viabilidade e a eficácia das redes neurais convolucionais na tarefa de monitoramento de baterias de íons de lítio. A abordagem segmentada, com modelos especializados para diferentes partes das curvas de carga, resultou em melhorias significativas na precisão das estimativas, superando a metodologia de referência.

Este estudo contribui para o avanço das técnicas de gerenciamento de baterias, oferecendo uma base sólida para futuras pesquisas que busquem aprimorar a eficiência e a confiabilidade dos sistemas de armazenamento de energia. As direções propostas para trabalhos futuros visam não apenas superar as limitações atuais, mas também explorar novas possibilidades que potencializem ainda mais o desempenho dos modelos de aprendizado de máquina aplicados ao gerenciamento de baterias.

A continuidade deste trabalho, com a implementação das recomendações sugeridas, tem o potencial de impactar positivamente diversas indústrias que dependem de sistemas avançados de armazenamento de energia, contribuindo para soluções mais sustentáveis e eficientes em escala global.

Eventos e Publicações

Os resultados parciais deste estudo foram apresentados no 5º Congresso de Ingenierías y Ciencias Aplicadas de las Tres Fronteras (MEC3F), realizado em setembro de 2024, onde foram discutidos os avanços na modelagem e análise dos segmentos especializados para estimativa do **SoC** de baterias. A apresentação incluiu os principais achados obtidos até o momento, destacando o desempenho das redes especializadas em diferentes segmentos da curva de carga e a comparação com abordagens generalistas.

Além disso, está em andamento a elaboração de um artigo científico intitulado, com previsão de submissão para a revista *Journal of Power Sources*. Esse artigo detalhará a metodologia aplicada, os resultados experimentais e as implicações para o uso sustentável de baterias em sistemas de armazenamento de energia. A expectativa é que a publicação contribua para o avanço no campo de sistemas inteligentes aplicados ao gerenciamento de energia e estimativa de SoC.

Referências Bibliográficas

- [1] G. Plett, *Battery Management Systems, Volume I: Battery Modeling*. Power Engineering, Norwood, MA: Artech House, 2015.
- [2] J. Fleischmann, M. Hanicke, E. Horetsky, D. Ibrahim, S. Jautelat, M. Linder, P. Schaufuss, L. Torscht, and A. van de Rijt, “Battery 2030: Resilient, sustainable, and circular,” *McKinsey Company*, 2023. Colaboração com a Global Battery Alliance.
- [3] R. Fathi, “A guide to li-ion battery research and development,” 2020. Metrohm White Paper.
- [4] A. não informados, “Recent progress and challenges in silicon-based anode materials for lithium-ion batteries,” *Journal of Inorganic Materials*, 2024. Publicado há aproximadamente 1.2 anos; revisa avanços e desafios dos ânodos de silício, incluindo os problemas de expansão volumétrica.
- [5] C. Fu, V. Venturi, Z. Ahmad, A. W. Ells, V. Viswanathan, and B. A. Helms, “Universal chemomechanical design rules for solid-ion conductors to prevent dendrite formation in lithium metal batteries,” *ArXiv preprint arXiv:1901.04910*, 2019.
- [6] A. Manthiram, X. Yu, and S. Wang, “Lithium battery chemistries enabled by solid-state electrolytes,” *Nature Reviews Materials*, vol. 2, no. 4, p. 16103, 2017.
- [7] AM Batteries, “Revolutionising battery production: How dry electrode manufacturing promises to cut costs,” 2024.
- [8] COMSOL Multiphysics, “Lithium-ion battery rate capability.” <http://www.comsol.com/sla>, 2023. Battery Design Module Application Library. COMSOL Software License Agreement 6.0.

- [9] Z. Geng, T. Thiringer, and M. J. Lacey, “Intermittent current interruption method for commercial lithium-ion batteries aging characterization,” *IEEE Transactions on Transportation Electrification*, vol. 8, no. 2, pp. 2985–2994, 2022.
- [10] M. B. Pinson and M. Z. Bazant, “Theory of sei formation in rechargeable batteries: Capacity fade, accelerated aging and lifetime prediction,” *Journal of The Electrochemical Society*, vol. 160, no. 2, pp. A243–A250, 2013.
- [11] K. E. Thomas, J. Newman, and R. M. Darling, “Mathematical modeling of lithium batteries,” in *Advances in Lithium-Ion Batteries*, pp. 345–392, Kluwer Academic Publishers, 2002.
- [12] F. T. F. Doyle, Marc and J. Newman, “Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell,” *Journal of The Electrochemical Society*, vol. 140, no. 6, pp. 1526–1533, 1993.
- [13] M. Doyle and J. Newman, “The use of mathematical modeling in the design of lithium/polymer battery systems,” *Electrochimica Acta*, vol. 40, no. 13, pp. 2191–2196, 1995. International symposium on polymer electrolytes.
- [14] J. Newman and W. Tiedemann, “Porous-electrode theory with battery applications,” *AIChE Journal*, vol. 21, no. 1, pp. 25–41, 1975.
- [15] R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transport Phenomena*. John Wiley & Sons, Inc., 2nd edition ed., 2002.
- [16] T. F. Fuller and J. Newman, “Experimental determination of the transport number of water in nafion 117 membrane,” *Journal of The Electrochemical Society*, vol. 139, no. 5, pp. 1332–1337, 1992.
- [17] J. Newman and N. P. Balsara, *Electrochemical Systems*. The ECS Series of Texts and Monographs, John Wiley & Sons Inc., 4th edition ed., 2021.
- [18] G. Plett, *Battery Management Systems, Volume II: Equivalent-Circuit Methods*. Power Engineering, Norwood, MA: Artech House, 2016.

- [19] A. M. S. P. Manh-Kien Tran, Andre DaCosta and M. Fowler, “Comparative study of equivalent circuit models performance in four common lithium-ion batteries: Lfp, nmc, lmo, nca,” *Batteries*, vol. 7, no. 3, p. 51, 2021.
- [20] G. L. Plett, *Battery Management Systems, Volume I: Battery Modeling*. Norwood, Massachusetts, USA: Artech House Publishers, 2015.
- [21] F. M. González-Longatt, “Circuit based battery models: A review,” in *2nd Iberoamerican Congress of Student of Electrical Engineering (II CIBELEC)*, II CIBELEC, 2006.
- [22] H. Hinz, “Comparison of lithium-ion battery models for simulating storage systems in distributed power generation,” *Inventions*, vol. 4, no. 3, p. 41, 2019.
- [23] L. W. Yao, J. A. Aziz, P. Y. Kong, and N. R. N. Idris, “Modeling of lithium-ion battery using matlab/simulink,” in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1729–1734, 2013.
- [24] H. W. You, J. I. Bae, S. J. Cho, J. M. Lee, and S.-H. Kim, “Analysis of equivalent circuit models in lithium-ion batteries,” *AIP Advances*, vol. 8, p. 125101, 12 2018.
- [25] S. Buller, M. Thele, R. De Doncker, and E. Karden, “Impedance-based simulation models of supercapacitors and li-ion batteries for power electronic applications,” *Industry Applications, IEEE Transactions on*, vol. 41, pp. 742 – 747, 06 2005.
- [26] U. Westerhoff, K. Kurbach, F. Lienesch, and M. Kurrat, “Analysis of lithium-ion battery models based on electrochemical impedance spectroscopy,” *Energy Technology*, vol. 4, no. 12, pp. 1620–1630, 2016.
- [27] M. Ul Hassan, S. Saha, M. E. Haque, S. Islam, A. Mahmud, and N. Mendis, “A comprehensive review of battery state of charge estimation techniques,” *Sustainable Energy Technologies and Assessments*, vol. 54, p. 102801, 2022.
- [28] P. Pillai, S. Sundaresan, P. Kumar, K. R. Pattipati, and B. Balakumar, “Open-circuit voltage models for battery management systems: A review,” *Energies*, vol. 15, no. 18, p. 6803, 2022.

- [29] C. Fan, M. D. Higgins, and W. D. Widanage, “Real-time state of charge estimation of electrochemical model for lithium-ion battery,” in *2018 IEEE Transportation Electrification Conference and Expo (ITEC)*, pp. 217–224, IEEE, 2018.
- [30] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson, 2015.
- [31] J. Waring, C. Lindvall, and R. Umeton, “Automated machine learning: Review of the state-of-the-art and opportunities for healthcare,” *Artificial Intelligence in Medicine*, vol. 104, p. 101822, 2020.
- [32] V. Jain, “Everything you need to know about activation functions in deep learning models,” *Towards Data Science*, 2019.
- [33] S. Sharma, “Activation functions in neural networks,” *Towards Data Science*, 2017.
- [34] “How to use softmax function for multiclass classification,” *Turing*, 2024.
- [35] J. Brownlee, “How to choose loss functions when training deep learning neural networks,” *Machine Learning Mastery*, 2020.
- [36] J. Brownlee, “Difference between a batch and an epoch in a neural network,” *Machine Learning Mastery*, 2022.
- [37] H. Zulkifli, “Understanding learning rates and how it improves performance in deep learning,” *Towards Data Science*, 2018.
- [38] M. Grübler, “Entendendo o funcionamento de uma rede neural artificial,” *Medium*, 2018.
- [39] M. Nielsen, *How the backpropagation algorithm works*. 2019.
- [40] IBM, “What is overfitting?,” *IBM*, 2024.
- [41] M.-F. Ng, J. Zhao, Q. Yan, G. J. Conduit, and Z. W. Seh, “Predicting the state of charge and health of batteries using data-driven machine learning,” *J. Power Sources*, 2019. DOI: <https://doi.org/10.1038/s42256-020-0156-7>.
- [42] R. R. Richardson, C. R. Birkl, M. A. Osborne, and D. A. Howey, “Gaussian process regression for in situ capacity estimation of lithium-ion batteries,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 127–138, 2019.

- [43] K. A. Severson, P. M. Attia, N. Jin, N. Perkins, B. Jiang, Z. Yang, M. H. Chen, M. Aykol, P. K. Herring, D. Fraggedakis, *et al.*, “Data-driven prediction of battery cycle life before capacity degradation,” *Nature Energy*, vol. 4, no. 5, pp. 383–391, 2019.
- [44] C. Qian, B. Xu, L. Chang, B. Sun, Q. Feng, D. Yang, Y. Ren, and Z. Wang, “Convolutional neural network based capacity estimation using random segments of the charging curves for lithium-ion batteries,” *Energy*, vol. 227, p. 120333, 2021.
- [45] P. Vibhuti, “Automating estimation of battery state of charge using experimental data in matlab,” *Journal La Multiapp*, vol. 1, no. 4, pp. 28–32, 2020.
- [46] Z. Cui, L. Wang, Q. Li, and K. Wang, “A comprehensive review on the state of charge estimation for lithium-ion battery based on neural network,” *International Journal of Energy Research*, vol. 46, no. 5, pp. 5423–5440, 2022.
- [47] M. Ragone, V. Yurkiv, A. Ramasubramanian, B. Kashir, and F. Mashayek, “Data driven estimation of electric vehicle battery state-of-charge informed by automotive simulations and multi-physics modeling,” *Journal of Power Sources*, vol. 483, p. 229108, 2021.
- [48] G. dos Reis, C. Strange, M. Yadav, and S. Li, “Lithium-ion battery data and where to find it,” *Energy and AI*, vol. 5, p. 100081, 2021.
- [49] C. Qian, B. Xu, L. Chang, B. Sun, Q. Feng, D. Yang, Y. Ren, and Z. Wang, “Convolutional neural network based capacity estimation using random segments of the charging curves for lithium-ion batteries,” *Energy*, vol. 227, p. 120333, 2021.
- [50] C. Qian, H. Guan, B. Xu, Q. Xia, B. Sun, Y. Ren, and Z. Wang, “A cnn-sam-lstm hybrid neural network for multi-state estimation of lithium-ion batteries under dynamical operating conditions,” *Energy*, vol. 294, p. 130764, 2024.