

蓄電池における充電制御をおこなうためのコンテキスト指向ソフトウェアアーキテクチャの提案

M2023SE007 並川雄貴

指導教員：野呂昌満

1 はじめに

近年、電気自動車をはじめ多くの機器に蓄電池が用いられている。これらは、充放電を繰り返すことにより必ず劣化するので、電池状態を監視し、安全かつ効率的な制御をおこなうシステムが必要である。これを Battery Management System (以下、BMS と記述) という。蓄電池の中でもリチウムイオン電池は、性能の高さから今後も需要が大きくなると考え、本研究はこれを対象におこなう。

BMS において代表的な非機能要求は、充電時間と電池寿命である。充電時間を短縮することは、機器を素早く再利用できることから利便性の向上に繋がる。一方、充電時間を短縮するために、電池状態を考慮せずに充電電圧を大きくすることは、電池の寿命を縮める要因となる。BMS において、充電時間を短縮することと電池寿命を長期化することは対立する非機能要求 (以下、対立要求と記述) である。BMS における充電制御では、どちらも重要な非機能要求であるので、双方について考慮する必要がある。

本研究では、対立要求を調整するために充電電圧を切り替える。充電電圧の決定には、機械学習技術を用いる。解析的に求められた数理モデルによる決定論的な制御に比べ、電池状態をより詳細に考慮できると考えたらである。機械学習技術を用いる場合のコンセプトドリフト [1] には、オンライン学習を用いることで対応する。コンセプトドリフトとは、入力データと出力すべきデータの関係が時間の経過とともに変化することである。

本研究の目的は、蓄電池における対立要求を調整するソフトウェアアーキテクチャを提案し、妥当性を検証することである。充電電圧の切り替えは強化学習器がおこない、電池状態の変化に応じて動的に制御を更新する。

上述の研究目的より、以下のように研究課題を定義する。

研究課題 1 対立要求を調整するための充電制御をおこなうアーキテクチャの定義

研究課題 2 対立要求の調整と変更容易性の観点からの提案するアーキテクチャの定性的な考察

本研究において提案するアーキテクチャは、充電制御における定量的な評価をおこなうための基盤となるようにコンテキスト指向 [2] に基づいて定義する。提案したアーキテクチャについては、対立要求の調整と変更用意性の観点から定性的な考察をおこなう。

2 関連研究ならびに既存研究

2.1 リチウムイオン電池の劣化要因

リチウムイオン電池の劣化は、充放電できる最大容量が減少することを指す。リチウムイオン電池では、正極

による酸化反応と負極による還元反応により充電がおこなわれる。全体では、リチウムイオンが正極から負極に移動している。負極での還元反応時、電極と電解液の間に SEI(Solid Electrolyte Interphase) という皮膜が形成される。SEI の構成を図 1 に示す [3]。形成された SEI は、負極と電解液との反応を制御する役割があり、電解液の分解を抑制する。一方、この SEI が過剰に生成された場合、電極と電解液との間でリチウムイオンの移動が阻害され、抵抗が大きくなることで、充放電の効率を悪化させる。本研究では、この SEI の成長を最低限に抑えることで電池寿命の長期化を実現する。

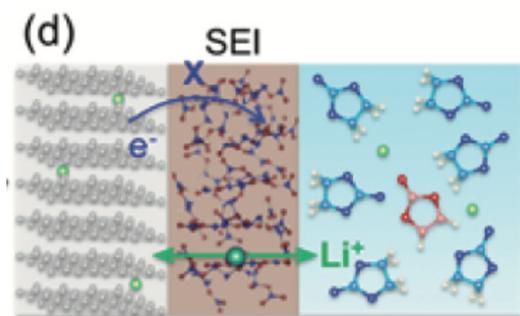


図 1 SEI の構成 ([3] より転載)

2.2 バッテリーマネジメントシステム

BMS は、電池が安全な範囲内で動作するように保護するという安全性とセンサ情報をもとに状態に応じた制御をおこなう効率性を保証するシステムである。

提案するアーキテクチャでは、経過時間毎の電池状態を正確に推定する、充電電圧を電池状態に応じて切り替えるという機能を構造として定義する。電池状態を推定する機能は電池モデル、充電電圧を電池状態に応じて切り替える機能は強化学習器による充電制御により実現する。電池モデルは、電流と電圧の測定値を用いて充電状態と劣化状態として SEI の成長を推定する。強化学習器は、電池モデルからの情報をコンテキストとし、対立要求を調整するために供給する電流を判断する。

3 課題解決へのアプローチ

研究課題 1 における対立要求は、充電時間を最小化すること、リチウムイオン電池の劣化要因である SEI の成長を最低限に抑えることと定義する。本研究では、Yadu[4] らが提案した充電制御方法を、コンテキスト指向に基づいたアーキテクチャとして定義する。電池状態を推定する機能は、Naha[5] らと Subramanian[6] らが提案した 2 つのモデルにより実現する。SEI の成長は、式 (1) による

抵抗の増加と式 (2) による電圧と電流の損失によって把握される。

$$R_i = R_f + \Delta R_{SEI} \quad (1)$$

$$I_{SEI} = I_{0,SEI} \exp \left(-\frac{\alpha F}{R_g T} [U_n - U_{SEI}^{ref}] + \sinh^{-1} \left[\frac{I}{I_{0,a}} \right] \right) \quad (2)$$

充電電圧を電池状態に応じて切り替える機能は、強化学習器により実現できると考える。学習に用いる報酬は、式 (3) のように定義される。SEI の成長が予想される場合、電流と電圧が安全な範囲を超えて動作した場合にペナルティを課すことで、これらの状況が起こらない最大の電圧を判断した場合に報酬が最大となるように定義する。

$$R = R_{charge-gain} + R_{qloss} + R_{Istage} + R_{Vstage} \quad (3)$$

コンセプトドリフトへの対応は、コンテキストの情報を機械学習器がオンライン学習することで動的に制御を更新する。コンセプトドリフトは、充放電を繰り返すことで引き起こされる電池状態の変化に応じて最適な充電電圧も変化することを指す。すなわち、コンテキストは、電池状態の変化を考慮できるように定義する必要がある。

研究課題 2 におけるアーキテクチャの妥当性の議論は、対立要求の調整と変更容易性の観点からおこなう。対立要求の調整の観点からは、提案したアーキテクチャにおいて、定量的な議論をおこなうための方法について定性的に議論する。充電時間の最小化については、充電時間と充電量を異なる充電制御と比較することでおこなえる。電極における SEI の成長を最低限に抑えることについては、完全放電状態から最大定格容量まで充電をおこなうことを 1 サイクルとし、規定回数そのサイクルをおこなった場合の SEI の成長を異なる充電制御と比較することでおこなえる。上記のとおり検証をおこなうことで、対立要求の調整について定量的な評価ができると考える。変更容易性の観点からは、BMS において想定される変更要因を挙げ、それらへの対応について議論をする。アーキテクチャが持つ主な機能は、電池状態の測定と推定、充電電圧の切り替えである。これらを実現するコンポーネントの変更要因を想定する。加えて、充電制御では利用者毎に非機能要求の重要度が異なる。この問題については、利用者の充電特性についてもコンテキストとし、報酬関数へ重み付けをおこない、それをオンライン学習することで対応できることを示す。

4 アーキテクチャ

本研究では、充電制御を行うシステムにおいて、扱う概念の構造を示した概念アーキテクチャを定義し、それに個別の利用技術を用いて具体化することで具象アーキテクチャを定義する。以下では、定義した概念アーキテクチャと具象アーキテクチャについて説明する。

4.1 概念アーキテクチャ

提案する概念アーキテクチャを図 2 に示す。アーキテクチャは、コンテキスト指向に基づいて定義し、充電電

圧の大きさをコンテキストに応じて切り替えることで対立要求の調整を実現する。

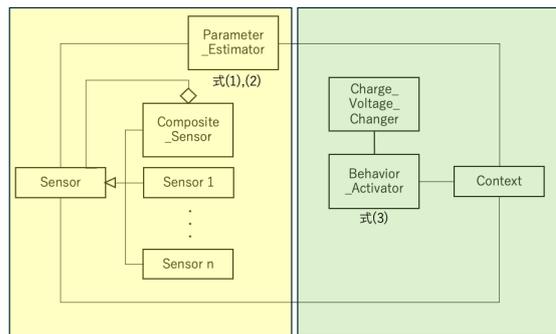


図 2 概念アーキテクチャ

黄色で着色した部分は電池モデルであり、センサから得た情報とそれらをもとに推定した値をコンテキストとして定義する。緑色で着色した部分は強化学習コンポーネントであり、コンテキストの情報をもとに充電電圧を動的に切り替える。最適な充電制御は、電池状態と利用者毎に異なるので、コンテキストはそれらの情報を含むように定義する必要がある。センサにより測定される情報とそれらをもとに電池モデルが推定した情報をコンテキストとして定義した。電池状態の推定は Parameter Estimation が式 (1) と式 (2)、電流積算法を用いておこなう。最適な充電電流は、式 (3) を用いて Behavior Activator が決定し、Charge Voltage Changer が電圧の制御をおこなう。Sensor の構成は、より性能の良いセンサの導入や考慮するパラメータの増加により変更されることが考えられるので Composite パターンを用いる。新たなセンサを追加は、それをサブクラスとして追加するだけでよい。

提案するアーキテクチャでは、コンテキスト指向の概念を導入している。充電時の最適な制御は、電池の状態により変化する。加えて、充電時間と電池寿命のどちらの非機能要求が重要となるかについても、利用者の行動特性によって変化する。電池状態に応じた充電制御を実現するために Context の情報をもとに、強化学習コンポーネントがオンライン学習し、ポリシーの更新をおこなう。このように定義することで、状態に応じた振舞いを実現する。

4.2 具象アーキテクチャ

具象アーキテクチャを図 3 に示す。BMS の運用時に使用する機械学習器、センサ、モデル、コンテキストを記述することで図 2 のアーキテクチャを具体化した。

センサは、電流計と電圧計に相当する Ammeter と Voltmeter を用いる。これらの測定値をもとにパラメータの推定は、Simple_Battery_Model と Hybrid_ROM_Model がおこなう。Simple_Battery_Model は、電流と電圧から抵抗を推定する。Hybrid_ROM_Model は、Simple_Battery_Model からの推定値とセンサの測定値から、充電量と SEI の成長を推定する。充電電流の決定をおこなう強化学習器は、Reinforcement_Learner に相当する。Context は、Hybrid_ROM_Model、Sensor から送

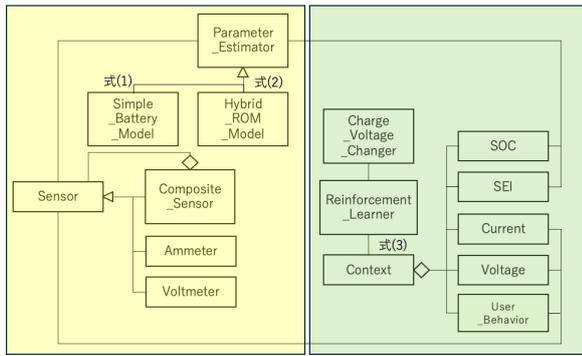


図 3 具象アーキテクチャ

られる情報と利用者の重視する非機能要求を判断する情報を持つものと定義した。Reinforcement_Learner が Context の情報から最適な充電電流の選択をおこない、Charge_Voltage_Changer が充電電圧を制御する。

5 考察

本研究で提案したアーキテクチャは、3つの機能を実現するように構成した。実現する機能毎に色付けをおこなったものが図4である。それぞれが実現する機能は赤色がセンサとして電池から測定可能な情報を得る。青色はセンサで測定できないパラメータを推定する。紫色はコンテキストから充電電圧の決定と制御の更新をおこなう。

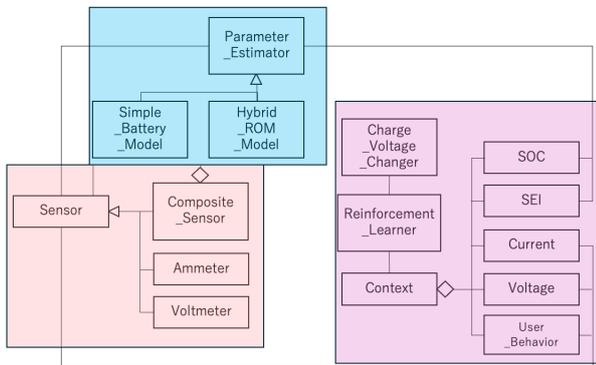


図 4 実現する機能毎に分割した具象アーキテクチャ

5.1 対立要求の調整の観点からのアーキテクチャの考察

対立要求の調整の観点からのアーキテクチャの考察は、本研究において提案したアーキテクチャを用いることで、対立要求を調整を定量的に検証できることを示す。シミュレーションソフトウェアを用いた場合、アーキテクチャに基づいて設計されたBMSのシミュレーションをおこなう手順についてまとめる。

1. 提案したアーキテクチャに基づいた充電制御をおこなうプロトタイプを用いて未使用のリチウムイオン電池を充電する。
2. 1と同様の条件下でCCCV充電方式を用いた充電をおこなう。

3. 1と2の充放電を規定回数おこなった結果を比較する。

上記の手順で検証をおこなうことで、提案したアーキテクチャが対立要求を調整する充電制御がおこなえたかを定量的に比較することができる。モデルフリーな充電制御であるCCCV充電方式は、静的な充電制御であり、1度定義された充電制御は変更されることがない。一方、提案したアーキテクチャに基づいて作成されたシステムは、紫色の部分により充放電サイクルを繰り返すことでContextとして定義された電池状態をReinforcement_Learnerがオンライン学習し、制御を動的に更新する。すなわち、システムがコンセプトドリフトへ対応できているかを示すことができる。対立要求である充電時間の最小化は、充電をおこなった時間と充電量を比較し、SEIの成長を最低限に抑えられたかは、充放電サイクル数と電池に流れるSEI電流の総和を比較することで検証がおこなえる。ここでの充電量とSEI電流の総和は、青色の部分のParameter_Estimationにより把握され、赤色の部分Sensorからの情報を基に推定される。提案したアーキテクチャは、SEIの成長を電池モデルにより考慮するので、SEIの成長による抵抗の増加が比較対象よりも小さいことが期待できる。これは、同等の電圧を流した場合において、より大きな電流を電池に流すことができ、充電時間を短縮できると考える。提案したアーキテクチャにおいて、比較対象よりも小さい充電時間において大きい充電量が得られ、同じ充電サイクル数においてSEI電流の総和が小さければ対立要求の調整ができたと考えられる。

5.2 アーキテクチャが提供する変更容易性

提案したアーキテクチャに基づいてBMSを設計した場合に提供する変更容易性は、想定される変更シナリオをあげて議論する。変更シナリオは、充電制御システムの主要な機能に基づいて考察する。主要な機能は、図4のとおりである。それぞれ変更が想定される部分は赤色の部分がsensor、青色の部分がSimple_Battery_ModelとHybrid_ROM_Model、紫色の部分がcontextとReinforcement_Learnerである。これらのコンポーネントが変更されるシナリオを以下にまとめる。

- 電池の状態を測定する新たなセンサの追加もしくは、変更
- 推定精度と計算可能性を兼ね備えた電池モデルへの変更
- 強化学習コンポーネントの報酬関数の変更

電池状態を測定するセンサの追加は、Sensorにコンボジットパターンを用いているので、新たに測定するセンサをサブクラスとして追加するだけで対応できる。例として、電池温度や外部環境温度などの新たなSensorからの測定値を用いることが想定される。これは新たなセンサとして温度計に相当するThermometerをSensorのサブクラスに追加するだけで良い。

電池モデルの変更については、入力に変更がない場合、Simple_Battery_ModelとHybrid_ROM_Modelをより優れた電池モデルと置き換えるだけで良いと考える。入力

に変更がある場合、先述のとおり新たなセンサを Sensor のサブクラスとして追加するだけで良い。出力に変更がある場合、新しく得られる電池状態の推定値を Context へ追加することで対応できる。

強化学習コンポーネントの報酬関数の変更は、Context と Reinforcement Learner の間の関連を変更する。1つ目と2つ目の変更シナリオのように新たなセンサや電池モデルが追加された場合、測定値もしくは推定値を Context に新たなパラメータとして追加する。この新たなパラメータを報酬関数に組み込むことで実現できる。例として、電池温度の上昇量を推定値として追加する場合を想定する。Context に温度上昇における推定値を追加し、報酬関数へ充電による電池温度の上昇が安全範囲内を超えた際にペナルティを与えることで対応できる。

5.3 利用者毎に異なる充電特性への対応

充電制御では、利用者毎に非機能要求の重要度が異なるという問題がある。リチウムイオン電池を用いた機器の例として、電気自動車を挙げる。電気自動車の充電は、充電能力に応じて以下の3つのレベルが定義されている。

1. レベル1：120(V), 12(A) の電源を用いた充電
2. レベル2：240(V), 最大 80(A) の電源を用いた充電
3. レベル3：480(V) 以上の電源を用いた充電

これらは、レベル1が低速充電、レベル2とレベル3は急速充電と定義されている。充電能力は電圧と電流の積でもとまる電力を用い、 $120(V) \times 12(A) = 1440(W)$ を基準とすることで、利用者がおこなった充電がどちらかを判断できる。

提案したアーキテクチャに基づいた充電制御を実現した場合、Context として定義した User_Behavior が急速充電回数と低速充電回数を保持することで Reinforcement Learner の報酬関数に重み付けをおこなう。重み付けは、急速充電を多くおこなう利用者に電池寿命の長期化を目的とする R_{gloss} の影響を小さくすることで、より大きい充電電圧を選択する制御をおこなう。対照的に、低速充電を多くおこなう利用者へは、選択した充電電流の大きさから得られる報酬 $R_{charge-gain}$ を小さくすることで、より SEI の成長を抑制する制御をおこなう。双方の非機能要求を重視する利用者へは、重み付けはおこなわない。これをオンライン学習することで利用者毎に重要度の異なる非機能要求に対応できると考える。

6 おわりに

6.1 まとめ

本研究では、蓄電池の充電時間を短縮する、電池寿命を長期化するという対立する要求を調整するソフトウェアアーキテクチャを提案した。本研究では、対立要求を調整する充電制御をおこなうために、充電電圧の調整に深層強化学習を用いた。

提案したアーキテクチャの妥当性を検証するために対立要求の調整の観点と変更用意性の観点から定性的な議論をおこなった。対立要求の調整の観点からは、提案したアーキテクチャを用いることで定量的な考察をおこなえ

ることを定性的に示した。コンセプトドリフトへは、オンライン学習をおこなうことで対応できることについても議論をおこなった。変更用意性の観点からは、提案したアーキテクチャの主な機能からユースケースを想定し、変更要因に対応できることを保証した。加えて、利用者毎に異なる充電特性については、強化学習器に用いる報酬関数へ重み付けをおこなうことで対応した。本アーキテクチャは利用者毎に重要度の異なる非機能要求を考慮できることを示した。

6.2 今後の課題

今後の課題は、以下の通りである。

- 提案したアーキテクチャのプロトタイプの実装
- シミュレーションソフトウェアを用いた、アーキテクチャの妥当性検証

プロトタイプ的设计は、提案したアーキテクチャに基づき、電池モデルと強化学習器を作成し、学習をおこなう。電池モデルは、強化学習器が学習をおこなうためのデータ生成と実装時に電池状態を把握するために用いる。

ソフトウェアを用いて、作成したプロトタイプによりシミュレーションをおこなうことで、対立要求の調整について定量的な検証をおこなう。考察で記述した方法で検証をおこなうことで、提案したアーキテクチャの有効性について議論する。

参考文献

- [1] Gama, João, et al. "A survey on concept drift adaptation," ACM computing surveys (CSUR), Vol.46.4, pp.1-37, 2014.
- [2] 紙名哲生. "文脈指向プログラミングの要素技術と展望." コンピュータソフトウェア, Vol.31.1, pp.3-13, 2014.
- [3] 館山佳尚. "リチウムイオン電池 Solid Electrolyte Interphase (SEI) に関する第一原理計算研究," Journal of Computer Chemistry Japan, Vol.18.1, pp.18-28, 2019.
- [4] Yadu, Ankit, et al. "On-Device Personalized Charging Strategy With an Aging Model for Lithium-Ion Batteries Using Deep Reinforcement Learning," IEEE Transactions on Automation Science and Engineering, Vol.21.4, pp.5037-5047, 2023.
- [5] Naha, Arunava, et al. "An incremental voltage difference based technique for online state of health estimation of li-ion batteries," Scientific reports, 2020.
- [6] Swernath, Subramanian, et al. "A Real Time Adaptive Charging Approach for Cycle Life Extension of Li Ion Cells," Journal of The Electrochemical Society, Vol.168.5, 2021.