

バッテリーの充電状態（SOC）と健全性（SOH）、その推定手法の詳細を解き明かす

Martin Murnane
アナログ・デバイセズ 太陽光発電チーム
Adel Ghazel氏
EBSYS Technology (WEVIOO Group) 最高技術責任者 (CTO)

はじめに

リチウム・イオン・セルをベースとするバッテリー・スタックは、多くのアプリケーションで使用されています。ハイブリッド車（HEV）、電気自動車（EV）、再生可能エネルギー用の蓄電システムなどが代表的な例です。あるいは、グリッド上に配備されるエネルギー貯蔵システムなどでも使われています。その種のシステムは、グリッドの安定化、ピーク・カット、再生可能エネルギーの利用時間のシフトといった目的で用いられます。リチウム・イオン・セルをベースとするバッテリー・スタックを使用するアプリケーションでは、各セルの充電状態（SOC：State of Charge）を測定することが非常に重要です。SOCとは、利用可能な容量（単位はAh）のことであり、定格容量に対するパーセンテージで表します。SOCの値は、バッテリーに蓄積された利用可能なエネルギーの量を評価するための熱力学的な量だと見なすことができます。また、SOCだけでなく、バッテリーの健全性（SOH：State of Health）を推定することも重要です。SOHは、バッテリーが電気エネルギーを貯蔵して供給する能力を、新しいバッテリーと比較する形で表現します。本稿では、バッテリーのSOC/SOHの推定方法について詳細に説明します。その推定に必要な能力を備えた代表的なプロセッサとしては、アナログ・デバイセズの電力制御プロセッサ「ADSP-CM419」が挙げられます。

本稿では、クーロン・カウント技術を利用してSOCとSOHを推定する方法や、推定に使用するアルゴリズムについて説明します。具体的には、クーロン・カウント技術を利用するために必要な測定環境の技術的な仕様を定義した上で、SOC/SOHの推定方法について詳しく解説します。その中で、クーロン・カウント法、電圧法、カルマン・フィルタ法について概観します。また、SOC/SOHの推定を実現する商用ソリューションも紹介します。加えて、SOC/SOHの推定に用いるクラス最高レベルのアルゴリズムの実例も取り上げます。特に、クーロン・カウント向けの強化型アルゴリズム、SOCの推定に向けた汎用アルゴリズム、拡張カルマン・フィルタ・アルゴリズムについて詳しく解説します。更に、SOC/SOHを推定するためのアルゴリズムの評価方法やシミュレーション結果なども紹介します。

SOCの測定原理

バッテリーのSOCの値を決定するのは、バッテリーの種類やそれを使用するアプリケーションに依存する複雑な作業になります。そのため、SOCの値を高い精度で推定するための研究／開発は現在でも盛んに行われています。実際、SOCの値を正確に推定するのは、バッテリー管理（バッテリー・マネージメント）システムの主要な責務だと言えます。SOCの値を高い精度で推定できれば、システムの性能と信頼性を向上し、バッテリーの寿命を延ばすことが可能になります。例えば、バッテリーの過充電／過放電は、バッテリーの内部構造に永続的な損傷を引き起こす可能性があります。SOCを正確に推定できれば、そうした損傷を防止したり、システムの動作の予期せぬ中断を回避したりすることが可能になります。ただ、バッテリーの充放電には複雑な化学的なプロセスや物理的なプロセスが関与します。そのため、様々な動作条件の下でSOCを正確に推定するのは容易ではありません。

一般に、SOCを把握するには、あらゆる動作条件の下で、次に挙げる値を測定する必要があります。すなわち、セルのスタックに流出入する電荷と電流、スタック内の各セルの個々の電圧を、非常に正確かつ同時に測定しなければなりません。それらのデータを基に、監視の対象となるセルのSOCの値を推定します。その際には、事前にロードしておいたセル・パックのデータも併用する必要があります。SOCの計算に必要な追加のデータとしては、セルの温度、測定時にセルが充電中なのか放電中なのかという情報、セルの使用時間（給電を実現するために充放電を行っている期間）、セルのメーカーから入手したその他の関連データなどが挙げられます。メーカーからは、様々な条件の下でリチウム・イオン・セルがどのような性能を発揮するのかを表す特性データを入手できるケースもあります。SOCの推定値を決定したら、その後、システムの稼働中にSOCの更新を続ける作業はそのシステムに委ねられます。必要な一連の作業は、基本的にはセルに流出入する電荷量をカウントする処理に基づいて行われます。この方法では、SOCの初期値を十分な精度で把握できていない場合や、セルの自己放電／リークの影響などが生じた場合には、推定精度が低下する可能性があります。

測定環境の技術的な仕様

本稿では、クーロン・カウントの機能を実現するために開発された評価用プラットフォームをベースとして解説を進めます。そのプラットフォームは、エネルギーの貯蔵に使用される一般的なモジュールのSOC/SOHを推定する用途に向けたものです。バッテリー・モジュールとしては、通常7~8個のリチウム・イオン・セルで構成される24V対応のものを想定しています。このプラットフォームは、以下のような要素で構成されています。

- ▶ ハードウェア・システム：
マイクロコントローラ、インターフェース、ペリフェラルなどを備えています。
- ▶ 組み込みソフトウェア：
SOC/SOHの推定用のアルゴリズムを実装します。
- ▶ PCベースのアプリケーション・ソフトウェア：
システムの構成作業や、データの表示／解析に用いるユーザ・インターフェースとしての役割を担います。

この評価用プラットフォームでは、A/Dコンバータ（ADC）とセンサーを使用して、各セルの電圧とバッテリー・パックの電流／電圧を定期的に測定します。その結果を使用して、SOCの推定に用いるアルゴリズムをリアルタイムで実行します。そのアルゴリズムでは、電圧と電流の実測値、温度センサーによって収集したデータ、PCベースのソフトウェア・アプリケーションによって提供されるデータ（データベースからのコンストラクタの仕様など）を使用します。アルゴリズムの出力はPCで稼働するGUI（Graphical User Interface）に引き渡され、動的な表示とデータベースの更新が実行されます。

SOC/SOHの推定に用いられる手法

通常、SOCとSOHの推定には、以下に示す3つの手法のうちいずれかが使われます。

- ▶ クーロン・カウント法
- ▶ 電圧法
- ▶ カルマン・フィルタ法

これらの手法は、HEV/EVや太陽光発電システムで使われているものも含め、あらゆるバッテリー・システムに適用できます。以下、各手法について説明します。

クーロン・カウント法

クーロン・カウント法は、SOCを計算するための最も一般的な手法だと言えるでしょう。アンペア時カウント法、電流積算法という名称で呼ばれることもあります。クーロン・カウント法では、バッテリーの使用期間（給電を実現するために充放電を行っている期間）にわたり数学的に電流の値を積算します。その結果を使用し、次式によってSOCの値を計算します。

$$SOC = SOC(t_0) + \frac{1}{C_{rated}} \int_{t_0}^{t_0 + \tau} (I_b - I_{loss}) dt \quad (1)$$

ここで、 $SOC(t_0)$ はSOCの初期値、 C_{rated} はバッテリーの定格容量、 I_b はバッテリーの電流、 I_{loss} は損失反応によって消費される電流です。

クーロン・カウント法では、バッテリーに流入する電荷量を積算することによって残容量を計算します。この方法の精度は、主としてバッテリーの電流の測定精度と、SOCの初期値の推定精

度によって決まります。あらかじめ記録されていた容量値か、動作条件から得られた最初の推定値が存在すれば、SOCの値はその使用期間にわたって充放電電流を積算することで算出できます。しかし、バッテリーから放出できる電荷の量は、充放電のサイクルで蓄積された電荷の量よりも常に少なくなります。つまり、充放電の期間には損失が生じるといえます。自己放電に加え、その損失も積算誤差の原因になります。SOCの推定精度を高めるには、そうした要因について考慮しなければなりません。より正確な推定を実現するためには、SOCに関する再キャリブレーションを定期的に行うと共に、電荷を放出可能な容量の減少について考慮する必要があります。

電圧法

SOC（つまり残容量）の値は、制御された条件の下で放電テストを実施することで決定できます。この電圧法では、バッテリーの既知の放電カーブ（電圧対SOC）を使用し、バッテリーの電圧の測定値を、それに相当するSOCの値に変換します。但し、バッテリーの電圧には、その電気化学反応の速度や温度に依存するバッテリーの電流からの大きな影響が及びます。この手法による推定精度を高めるためには、バッテリーの電流に比例する補正項によって電圧の測定値を補正したり、バッテリーのオープン・サーキット電圧（OCV：Open Circuit Voltage）対温度のルックアップ・テーブルを使用したりすることになるでしょう。また、電圧法を実施するためには、バッテリーの電圧範囲が安定している必要があります。このことから、推定機能を実装するのは容易ではなくなります。加えて、実際のアプリケーションで放電テストを実施したとしたら、その後に再充電を行う必要があるでしょう。それには時間がかかるので、この手法はほとんどのアプリケーションには適用できません。電圧法にはもう1つ欠点があります。それは、テストの実行中には、システムの通常動作を中断しなければならないというものです（オフライン法）。この点が、オンライン法であるクーロン・カウント法とは大きく異なります。

カルマン・フィルタ法

カルマン・フィルタは、あらゆる動的システムの内部状態を推定するために用いられるアルゴリズムです。これを使用すれば、バッテリーのSOCを推定することができます。カルマン・フィルタは、1960年に開発された手法です。状態の観測と問題の予測の両方に最適な線形フィルタを適用するための再帰的な解法を提供します。他の推定手法とは異なり、カルマン・フィルタを使用すると、状態を推定した値の動的な誤差範囲が自動的に提供されます。カルマン・フィルタ法では、バッテリー・システムをモデル化する際、状態の記述内に把握したい未知の量（SOCなど）を含めます。それにより、カルマン・フィルタはそれらの値の推定を行います。加えて、それらの推定値の誤差範囲も提示します。その結果、誤差の補正機構を採用しつつ、SOCの値をリアルタイムに予測できるモデル・ベースの状態推定手法を実現することが可能になります。また、カルマン・フィルタは必要に応じて拡張することができます。その拡張カルマン・フィルタを用いれば、SOHをリアルタイムに推定する能力を高められます。特に、拡張カルマン・フィルタは、非線形のバッテリー・システムに対し、線形化のステップを適用する必要がある場合に有用です。カルマン・フィルタは、オンライン法の1つであり、動的に利用できる手法でもあります。但し、バッテリーを的確に表現したモデルを用意する必要があります。また、そのモデルで使用するパラメータを適切に特定しなければなりません。更に、非常に高い演算能力と高精度の初期化も必要です。

SOCの推定方法は他にもたくさん存在します。それらの方法は様々な文献で紹介されています。その一例がインピーダンス分光法です。この手法では、充電、放電の両方に対し、リアルタイムでインピーダンス・アナライザを適用します。そのようにして測定したセルのインピーダンスに基づいて推定を行います。確かに、この手法を使用すればリチウム・イオン・セルのSOC/SOHを推定できます。但し、計測機器による測定をベースとする手法なので、多くのアプリケーションには適用できません。そのため、本稿ではこれ以上この手法には触れないことにします。それ以外にも、電解質の物理的な性質や人工ニューラル・ネットワークに基づく推定手法も存在します。但し、それらの手法はリチウム・イオン・バッテリーには適用できません。

SOC/SOHの推定手法を選択するための方法論

SOCの推定手法として適切なものを選択するにはどうすればよいのでしょうか。そのためには、いくつかの基準を考慮する必要があります。まず、SOC/SOHの推定手法としては、HEV/EVのアプリケーション、再生可能エネルギーを後で使用できるようにするための蓄電システム、グリッド上のエネルギーを貯蔵するためのリチウム・イオン・バッテリーに適用できるものを選択すべきです。また、計算が複雑すぎることなく、高い精度（推定誤差が小さい）が得られるものでなければなりません。なおかつ、リアルタイムの測定が可能なオンライン法を選択する必要があります。加えて、電圧/電流の測定値、温度センサーによって収集されるデータ、PCベースのソフトウェア・アプリケーションによって提供されるデータを使用できる手法を選択しなければなりません。

強化型のクーロン・カウントのアルゴリズム

先述したように、リチウム・イオン・バッテリーのSOC/SOHを推定するための手法としては、クーロン・カウント法が一般的に使われています。但し、同手法にも欠点があります。そこで、その欠点を解消し、推定精度を向上させるためのものとして、クーロン・カウント法を強化するためのアルゴリズムが提案されています。本稿では、それを強化クーロン・カウント・アルゴリズムと呼ぶことにします。強化クーロン・カウント・アルゴリズムでは、SOCの初期値は負荷電圧（充電と放電）またはOCVを基にして取得します。損失については、充電効率と放電効率を考慮することによって補正を施します。使用中のバッテリーが放出可能な最大の容量に対し、動的な再キャリブレーションを実施することにより、バッテリーのSOHも同時に評価することができます。この手法を使用すれば、SOCの値をより正確に推定することが可能になります。

技術的な原理

ここでは、バッテリーが放出できる電荷量（バッテリーの残量）を $C_{\text{releasable}}$ として表すことにします。一方、メーカーが提示するバッテリーの定格容量は C_{rated} で表すことにしましょう。この場合のSOCは、次式に示すように、定格容量に対するバッテリーの残量をパーセンテージで表したものになります。

$$SOC = \frac{C_{\text{releasable}}}{C_{\text{rated}}} 100\% \quad (2)$$

ここで、満充電のバッテリーの残量を C_{max} で表すことにします。その値は、定格容量の値とは異なる可能性があります。実際、新しく使用するバッテリーの C_{max} の値は C_{rated} の値とは多少異なります。また、 C_{max} の値は使用時間が経過すると減少していきます。このことから、バッテリーのSOHは以下の式を使って評価されます。

$$SOH = \frac{C_{\text{max}}}{C_{\text{rated}}} 100\% \quad (3)$$

バッテリーが放電している場合の放電深度（DOD : Depth of Discharge）は、 C_{rated} と放電した電荷量（容量）の比をパーセンテージで表したものになります（以下参照）。

$$DOD = \frac{C_{\text{released}}}{C_{\text{rated}}} 100\% \quad (4)$$

ここで、 C_{released} は任意の量の電流によって放電される電荷量です。

充放電を行う際の電流の測定値 I_b を使用すれば、使用期間 τ にわたるDODの差分は次式によって計算できます。

$$\Delta DOD = \frac{-\int_{t_0}^{t_0 + \tau} I_b(t) dt}{C_{\text{rated}}} 100\% \quad (5)$$

ここで、 I_b は充電時には正の値、放電時には負の値になります。時間の経過に伴うDODの積算結果は次式で表されます。

$$DOD(t) = DOD(t_0) + \Delta DOD \quad (6)$$

推定精度を向上させるために η で表される動作効率を考慮すると、DODの式は次のようになります。

$$DOD(t) = DOD(t_0) + \eta \Delta DOD \quad (7)$$

充電時の η は η_c 、放電時の η は η_d で表すことにします。

動作効率とバッテリーの経年劣化を考慮しない場合、SOCは次式で表すことができます。

$$SOC(t) = 100\% - DOD(t) \quad (8)$$

SOHを考慮すると、SOCの値は次式によって推定できることになります。

$$SOC(t) = SOH(t) - DOD(t) \quad (9)$$

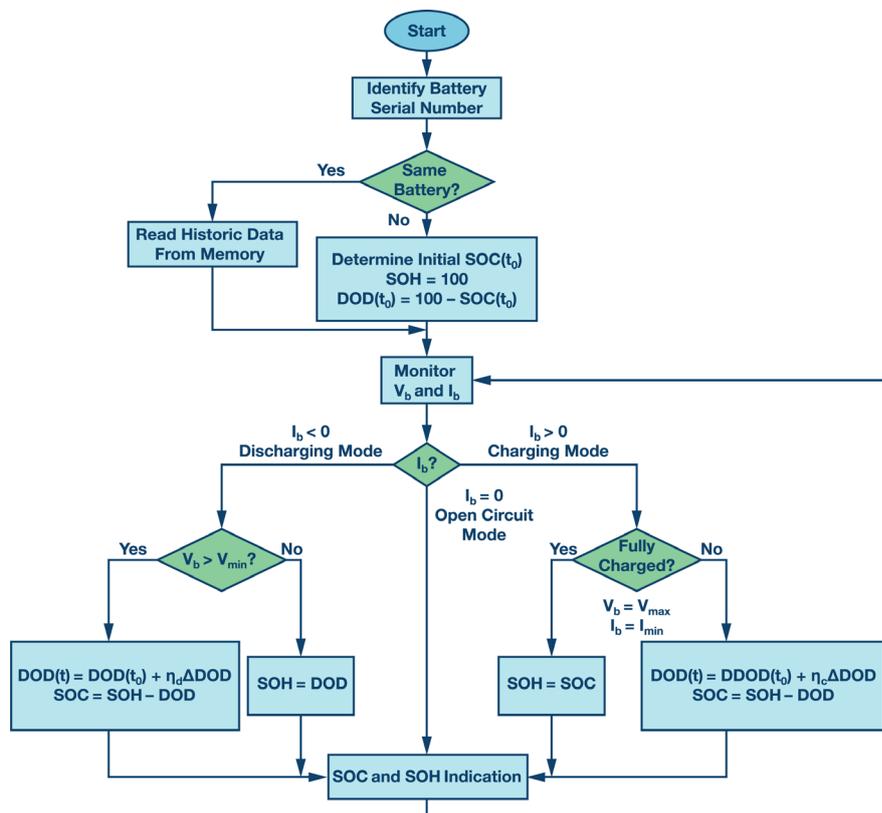


図1. 強化クーロン・カウント・アルゴリズムのフローチャート

図1に示したのは、強化クーロン・カウント・アルゴリズムのフローチャートです。処理を開始する際には、使用中のバッテリーの履歴データをメモリから取得します。新しく使用するバッテリーに関する情報が存在しない場合には、SOHは健全であり、その値は100%になると仮定します。SOCの初期値は、開始条件に応じてオープン・サーキット電圧または負荷電圧をテストすることによって推定します。

推定のプロセスは、バッテリーの電圧 V_b とバッテリーの電流 I_b のモニタリング結果に基づきます。バッテリーの動作モードは、電流の量と方向から把握することができます。放電モードにおいて、DODの値は放出された電荷量を加算することで得られます。充電モードでは、バッテリーに流入して蓄積された電荷量を算出します。充電効率と放電効率を使って補正をかけると、より正確な推定結果が得られます。SOCは、SOHの値からDODの値を差し引くことによって推定できます。バッテリーが、電流が流れないオープン・サーキットの状態である場合、SOCの値は、OCVとSOCの関係から直接取得します。

SOHは、バッテリーが完全に放電した際、または満充電になった際に再評価できるように注意してください。また、バッテリーのメーカーは、使用中のバッテリーの電流と電圧の値を規定しています。バッテリーは、放電中の負荷電圧 V_b が下限値 V_{min} を下回っている場合には完全に放電していることになります。その場合、バッテリーはそれ以上使用できないので、再充電しなければなりません。また、完全に放電した状態における積算DODの値を使ってSOHの値を再評価することで、SOHの再キャリブレーションを実施できます。一方、既に使い始めたバッテリーは、充電中に V_b の値が上限値 V_{max} に達し、 I_b の値が下限値 I_{min} まで減少している場合には完全に充電されています。新たなSOHは、バッテリーに流入した電荷の総和を積算することによって得られます。その値はSOCと等しくなります。実際のアプリケーションでは、完全な充電状態と完全な放電状態はたまにしか発生しま

せん。ただ、バッテリーの完全充電と完全放電を高い頻度で行うと、SOHの推定精度を向上させることができます。

強化クーロン・カウント・アルゴリズムは、EVだけでなく、あらゆる携帯型の機器にも簡単に実装することができます。計算が単純であり、ハードウェアに対する要件が複雑ではないからです。推定誤差は、SOHを再評価した次の動作サイクルにおいて1%まで低減することが可能です。

SOCの初期値の決定

バッテリーは、充電、放電、オープン・サーキットのうちいずれかの状態をとります。通常、バッテリーのメーカーは、定電流/定電圧 (CC-CV) モードで充電する場合のバッテリーの電圧/電流の変化量を規定しています。定電流モードで充電する場合、バッテリーの電圧は徐々に上昇して閾値に達します。バッテリーが定電圧モードで充電されるようになると、充電電流は最初は急速に減少し、その後はゆっくりと減少します。バッテリーが完全に充電されたら、電流は最終的にほぼゼロになります。この充電カーブは、定電流モードの段階のSOCと充電電圧の関係および定電圧モードの段階のSOCと充電電流の関係に変換することができます。これらの関係から充電中のSOCの初期値を推定できます。

バッテリーのメーカーは、バッテリーが様々な電流値で放電する場合の標準的な電圧カーブを提供しています。バッテリーの端子電圧は、使用時間が経過するにつれて低下します。電流量が多いほど端子電圧の低下が速くなり、使用時間が短くなります。このような形で様々な電流値に対応するSOCと放電電圧の関係が得られます。その結果、放電の段階におけるSOCの初期値を推定することが可能になります。

オープン・サーキットの状態については、OCVとSOCの関係が必要になります。バッテリーは、負荷から切り離されるまで様々な値の電流で放電されます。長い休止時間が得られる場合には、SOCの推定にOCVを使用することが可能です。

充電効率と放電効率

バッテリーの充電効率と放電効率は、クーロン効率によって評価できます。クーロン効率は、充電を行う際にバッテリーに流入する電荷量と、放電を行う際にバッテリーから引き出すことのできる電荷量の比として定義されます。なお、充電効率と放電効率の係数は、いくつかのバッテリーのテスト結果の平均値から求めたものである点に注意してください。

テスト済みのすべてのバッテリーは、充電レートと充電時間の積である指定容量まで最大のレート（一定値）で充電されます。続いて、カットオフ電圧まで最小のレート（一定値）で放電します。充電効率は次式で定義されます。

$$\eta_C = \frac{C_{\text{discharge, rate min}}}{C_{\text{charge, rate max}}} \quad (10)$$

放電効率は、1回の放電サイクルにおける C_{max} の値と、2段階で放出される電荷量（容量）の比です。テスト済みのすべてのバッテリーは、完全に充電した後、2段階の電流プロファイルに従って放電します。つまり、最初は指定されたDODまで規定値の電流によって放電され、その後、カットオフ電圧まで最小のレートで放電されるということです。放電効率は次式によって計算します。

$$\eta_d = \frac{I_1 T_1 + I_2 T_2}{C_{\text{max}}} \quad (11)$$

ここで、 I_1 、 I_2 、 T_1 、 T_2 は、それぞれ第1段階と第2段階の放電電流と放電時間です。

SOC向けの汎用アルゴリズム

ここで言うSOC向けの汎用アルゴリズムとは、リチウム・イオン・バッテリーをはじめとするあらゆる種類のバッテリー向けに提案されているアルゴリズムのことです。実際、あらゆるバッテリーに適用することができます。このアルゴリズムでは、周波数領域の線形システム解析を使用します。回路モデルを使用することなく、バッテリーの端子電圧と放電電流のサンプル値に基づいてOCVの値を計算します。OCVとSOCの間のよく知られた対応関係、ある幅の時間ウィンドウの中でSOCは一定であるという仮定、バッテリーは線形／弱非線形のシステムであるという条件に基づき、OCVの値を算出することでSOCを推定します。

数学的な定式化

各時間ウィンドウにおいて、バッテリーの端子電圧 $v(t)$ は次のように分解することができます。

$$\begin{aligned} v(t) &= v_{zi}(t) + v_{zs}(t) \\ v(t) &= v_{zi}(t) + h(t) \times i(t) \end{aligned} \quad (12)$$

ここで、 $v_{zi}(t)$ は放電電流がゼロの場合の端子電圧に対応するゼロ入力応答です。一方、 $v_{zs}(t)$ は、 $i(t)$ が入力、電圧源は短絡、放電電流が存在する場合の端子電圧に対応するゼロ状態応答です。 $h(t)$ は、バッテリーをモデル化した線形システムのインパルス応答です。式(12)の畳み込みの有効性は、線形性に関する仮定に基づいて確立されていることに注意してください。

SOCの値は、時間ウィンドウ $0 \leq t \leq t_w$ 内で抽出されると仮定します。また、 $t < 0$ では放電電流は常にゼロです。これは、 $t = 0$ 以前はバッテリーが負荷から切り離されているとの仮定に基づいています。この仮定は、ウィンドウがシフトされた後には排除されます。この仮定を使用しつつ自己放電の影響を無視すると、ゼロ入力応答は実際にはOCVとなり、以下の式が成り立ちます。

$$v_{zi}(t) = OCV \cdot u(t) \quad 0 \leq t \leq t_w \quad (13)$$

ここで、 $u(t)$ は以下のような単位ステップ関数です。

$$\begin{aligned} u(t) &= 1 \quad \text{if } t \geq 0 \\ u(t) &= 0 \quad \text{if } t < 0 \end{aligned} \quad (14)$$

ここでは、次の関係を満たす $f(t)$ を見いだす必要があります。

$$f(t) \times i(t) = \delta(t) \quad 0 \leq t \leq t_w \quad (15)$$

$\delta(t)$ はディラックのデルタ関数であり、次のように表されます。

$$\begin{aligned} \delta(t) &= 1 \quad \text{if } t = 0 \\ \delta(t) &= 0 \quad \text{if } t \neq 0 \end{aligned} \quad (16)$$

なお、 $f(t)$ が式(15)を満たす必要があるのはウィンドウ内でのみです。

ここで、アルゴリズム1をご覧ください。これは、 $f(t)$ を解くために使用する時間離散型のアルゴリズムです。ここで、 n はウィンドウ内のサンプリング・ポイントの総数です。 t_1 、 t_2 、 \dots 、 t_n はサンプリングに対応する時間ポイントを表しています。鍵になる概念は、サンプルの逆畳み込みを実施するというものです。これは、基本変形を用いて逆行列を求めるプロセスに似たものと言えます。

アルゴリズム1. $f(t)$ の計算に用いるアルゴリズム

```

1: INPUT: Sampled  $i(t_i)$ ,  $0 \leq t_1 < \dots < t_n \leq t_w$ 
2: OUTPUT:  $f(t_i)$ ,  $0 \leq t_1 < \dots < t_n \leq t_w$ 
3: for  $j = 1$  to  $n$  do
4:    $f_{\text{norm}}(t_j) = f(t_j) = \delta(t_j) / i(t_j)$ 
5:    $i_{\text{norm}}(t_j) = i_e(t_j) = i(t_j) / i(t_j)$ 
6: end for
7: for  $i = 2$  to  $n$  do
8:   for  $j = n$  to  $i$  do
9:      $f(t_j) = f(t_j) - f_{\text{norm}}(t_j - 1 + 1) i_e(t_i)$ 
10:     $i_e(t_j) = i_e(t_j) - i_{\text{norm}}(t_j - 1 + 1) i_e(t_i)$ 
11:   end for
12: end for

```

$f(t)$ を用いると、 $v_f(t) = f(t) \times v(t)$ は次のように計算できます。

$$\begin{aligned} v_f(t) &= f(t) \times v(t) \\ v_f(t) &= f(t) \times [OCV \cdot u(t) + v_{zs}(t)] \\ v_f(t) &= OCV \cdot u_f(t) + f(t) \times v_{zs}(t) \\ v_f(t) &= OCV \cdot u_f(t) + f(t) \times i(t) \times h(t) \\ v_f(t) &= OCV \cdot u_f(t) + \delta(t) \times h(t) \\ v_f(t) &= OCV \cdot u_f(t) + h(t) \quad 0 \leq t \leq t_w \end{aligned} \quad (17)$$

ここで、 $u_f(t) = f(t) \times u(t)$ です。

バッテリーの周波数領域の応答は有限だと考えられます。最終値定理から、次の式が成り立ちます。

$$\lim_{f \rightarrow \infty} h(t) = \lim_{s \rightarrow 0} sH(s) = 0 \quad (18)$$

従って、以下の式が成り立ちます。

$$\lim_{t \rightarrow \infty} \frac{v_f(t)}{u_f(t)} = OCV \quad (19)$$

これは、 t が大きくなると $h(t)$ がゼロに近づき、 $v_f(t)/u_f(t)$ は、現在の時間ウィンドウ内でOCVの良好な近似値を与えるということを示します。

OCVの値を抽出すると、現在の時間ウィンドウ内のシステムのインパルス応答を求めることができます（以下参照）。

$$h(t) = v_f(t) - OCV u_f(t) \quad (20)$$

現在のウィンドウ内でOCVの値の抽出が完了したら、次のウィンドウでもOCVの値を抽出するために同じプロセスを繰り返すことが可能です。

アルゴリズムの実装

アルゴリズム1には、実行時間の面で問題があります。ボトルネックの1つは、 $f(t)$ を求めるために $f(t) \times i(t) = \delta(t)$ を解くステップです。それに続く $v_f(t) = f(t) \times v(t)$ と $u_f(t) = f(t) \times u(t)$ を計算するステップもボトルネックになります。実際には、これら2つのステップは1つにまとめられます。そうすれば、 $f(t)$ を明示的に計算する必要はなくなります。修正後のアルゴリズムはアルゴリズム2のようになります。ここで、 n は1つのウィンドウ内のサンプリング・ポイントの総数です。

アルゴリズム2. 逆畳み込みと畳み込みのステップを組み合わせたアルゴリズム

```

1: INPUT: Sampled  $i(t_1), v(t_1), 0 \leq t_1 < \dots < t_n \leq t_w$ 
2: OUTPUT:  $vf(t_1), uf(t_1), 0 \leq t_1 < \dots < t_n \leq t_w$ 
3: for  $j = 1$  to  $n$  do
4:    $v_{norm}(t_j) = vf(t_j) = v(t_j)/i(t_1)$ 
5:    $u_{norm}(t_j) = uf(t_j) = u(t_j)/i(t_1)$ 
6:    $i_{norm}(t_j) = i_f(t_j) = i(t_j)/i(t_1)$ 
7: end for
8: for  $i = 2$  to  $n$  do
9:   for  $j = n$  to  $i$  do
10:     $v_f(t_j) = v_f(t_j) - v_{norm}(t_j - 1 + 1) i_f(t_1)$ 
11:     $u_f(t_j) = u_f(t_j) - u_{norm}(t_j - 1 + 1) i_f(t_1)$ 
12:     $i_f(t_j) = i_f(t_j) - i_{norm}(t_j - 1 + 1) i_f(t_1)$ 
13:   end for
14: end for

```

OCVの値を抽出したら、SOCの変化をOCVの関数として扱うことにより、SOCの値を推定できます。

このアルゴリズムの時間計算量は $O(n^2)$ です。ここで、 n はサンプルの数を表します。ここで示した方法が有効なものであることを実験によって確認しました。その結果、様々なバッテリーの種類と放電電流に対し、4%未満の誤差でSOCの値をオンラインで抽出できました。

拡張カルマン・フィルタのアルゴリズム

拡張カルマン・フィルタを適用すれば、リチウム・イオン・バッテリー・パックのSOCを直接推定できます。ここでは、バッテリーのOCVとSOCはほぼ線形の関係にあり、それらは周囲温度に依存して変化すると仮定します。この仮定は、実際のバッテリーの挙動に合致しています。バッテリーは、SOCをシステムの状態として定義した非線形システムとしてモデル化できます。そのため、拡張カルマン・フィルタを適用することが可能です。

リチウム・イオン・バッテリーのモデル

図2に示したのは、リチウム・イオン・バッテリー・パックの等価回路モデルです。図中のバルク・コンデンサ C_{cb} はバッテリー・パックの容量を表しています。表面コンデンサ C_{cs} はバッテリーの拡散効果を表現する役割を果たします。抵抗 R_i と R_t は、それぞれ内部抵抗と分極抵抗を表しています。

V_{cb} と V_{cs} は、それぞれバルク・コンデンサと表面コンデンサにかかる電圧を表しています。バッテリー・パックの端子電圧と端子電流は、それぞれ V_0 と I によって表しています。

バッテリーのモデルに必要なパラメータは、実験で取得したデータを基に決定することができます。OCVのテストは、電流パルスを注入し、バッテリーを連続的に放電させることで実施します。

図2のモデルの特性は、以下の各式によって規定されます。

$$\dot{V}_{cb} = \frac{I}{C_{cb}} \quad (21)$$

$$\dot{V}_{cs} = \frac{I}{R_t C_{cs}} V_{cs} + \frac{1}{C_{cs}} I \quad (22)$$

$$V_0 = V_{cb} + V_{cs} + IR_i \quad (23)$$

バッテリーのOCVとSOCの関係は、実際には区分線形になります。そのため、 V_{cb} は次式のように表せます。

$$V_{cb} = kS_{oc} + d \quad (24)$$

ここで、係数 k と d の値は一定ではなく、バッテリーのSOCと周囲温度に依存して変化します。そのため、以下の各式が成り立ちます。

$$\dot{S}_{oc} = \frac{I}{kC_{cb}} \quad (25)$$

$$\dot{V}_{cs} = -\frac{1}{R_t C_{cs}} V_{cs} + \frac{1}{C_{cs}} I \quad (26)$$

$$V_0 = kS_{oc} + V_{cs} + IR_i + d \quad (27)$$

最終的なシステム方程式は、次のようなものになります。

$$\begin{bmatrix} \dot{S}_{oc} \\ \dot{V}_{cs} \end{bmatrix} = \begin{bmatrix} \frac{I}{kC_{cb}} \\ -\frac{1}{R_t C_{cs}} V_{cs} + \frac{1}{C_{cs}} I \end{bmatrix} \quad (28)$$

$$V_0 = kS_{oc} + V_{cs} + IR_i + d \quad (29)$$

上に示した式によってモデル化されたバッテリー・システムは非線形です。そのため、拡張カルマン・フィルタを適用することにします。

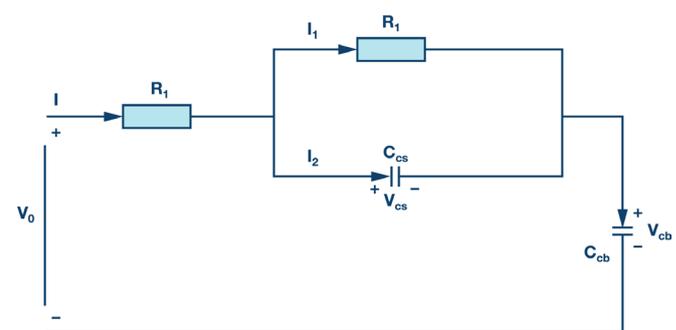


図2. リチウム・イオン・バッテリー・パックの等価回路モデル

拡張カルマン・フィルタの適用

拡張カルマン・フィルタは、カルマン・フィルタを非線形システム向けに拡張したものです。拡張カルマン・フィルタでは、時間ステップごとに線形化プロセスを実行し、非線形システムを線形時変システムで近似します。次に、カルマン・フィルタで線形時変システムを使用すると、真の非線形システム用の拡張カルマン・フィルタが得られます。カルマン・フィルタと同様に、拡張カルマン・フィルタも、入出力の測定値を使用することで真の状態の最小平均2乗誤差の推定値を算出します。その際、プロセス・ノイズとセンサー・ノイズは独立したゼロ平均ガウス・ノイズであると仮定します。

バッテリー・パック・システムに対応する式 (28) と式 (29) では、システムの状態変数は $x_1(t) = S_{oc}$ および $x_2(t) = V_{cs}$ と定義されます。

入力は $u(t) = I$ 、出力は $y(t) = V_0$ と定義します。バッテリー・パック・システムに対応する式 (28) と式 (29) は、次のように書き換えられます。

$$\dot{x} = f(x, u) + w \tag{30}$$

$$y = g(x, u) + v \tag{31}$$

ここで、 $x = [x_1, x_2]^T$ です。

w と v の項は、ランダムな外乱を表しています。それだけでなく、パラメータ d と k の変化によって生じる誤差も表します。 w と v は、それぞれ共分散行列 R と Q を持つ独立したゼロ平均ガウス・ノイズ・プロセスであると仮定します。

関数 $f(x, u)$ と $g(x, u)$ は、次のように表されます。

$$f(x, u) = \begin{bmatrix} \frac{u}{kC_{cb}} \\ -\frac{1}{R_1C_{cs}}x_2 + \frac{1}{C_{cs}}u \end{bmatrix} \tag{32}$$

$$g(x, u) = kx_1 + x_2 + R_1u + d \tag{33}$$

関数 $f(x, u)$ と $g(x, u)$ を、現在の動作点について各サンプルのステップで1次のテイラー級数展開によって線形化します。すると、線形化されたモデルは次式で表されます。

$$\delta\dot{x} = A_k\delta x + B_d\delta u \tag{34}$$

$$\delta y = C_k\delta x + D_k\delta u \tag{35}$$

ここで、 A_k 、 B_k 、 C_k 、 D_k は、以下のようなものになります。

$$A_k = \frac{\delta f(x, u)}{\delta x} = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{1}{R_1C_{cs}} \end{bmatrix}, B_k = \frac{\delta f(x, u)}{\delta u} = \begin{bmatrix} \frac{1}{kC_{cb}} \\ \frac{1}{C_{cs}} \end{bmatrix},$$

$$C_k = \frac{\delta g(x, u)}{\delta x} = [k \quad 1], D_k = \frac{\delta g(x, u)}{\delta u} = R_1$$

式 (34)、式 (35) で表されるモデルは、次のように離散化することができます。

$$x_{k+1} = A_d x_k + B_d u_k \tag{36}$$

$$y_{k+1} = C_d x_k + D_d u_k \tag{37}$$

ここで、 $A_d \doteq E + T_c A_k$ 、 $B_d \doteq T_c B_k$ 、 E は単位行列、 T_c はサンプリング周期、 $C_d \doteq C_k$ 、 $D_d \doteq D_k$ です。

カルマン・フィルタは最適オブザーバであり、その原理は図3のような形で表すことができます。基本的な考え方は、使用するモデルの不確定変数を調整するためにフィードバック系を使用し、出力の推定値と測定値の誤差をリアルタイムに最小限に抑えるというものです。そのようにしてモデルのフィッティングを実施することで、直接的な測定は行えないモデルの物理的なパラメータを観測することが可能になります。補正については、ゲイン・ベクトル K による重み付けを利用します。それにより、フィルタのダイナミクスと性能を補正することが可能になります。ゲインの値は、繰り返しの処理が行われるたびに、状態と測定結果に関する誤差の予測値と不確実性（ノイズ）に基づいて計算されます。フィルタの動的な制御は、状態 Q と測定値 R のノイズ行列の初期化と、誤差共分散行列 P の初期化に基づいて行われます。

図4に示すカルマン・フィルタのアルゴリズムは、2つのフェーズから成ります。1つは行列 P 、 Q 、 R の初期化に関するものです。もう1つは、各サンプリング間隔で2つのステップによって行われる観測に関するフェーズです。このアルゴリズムでは、まず、現在の状態、出力、誤差共分散の値を予測します。次に、物理システムの出力の測定値を使用し、状態の推定に対する補正と誤差共分散の補正を実行します。

このような拡張カルマン・フィルタを適用することにより、リチウム・イオン・バッテリー・パックのSOCの値を推定できます。このアルゴリズムの時間計算量は $O(n^3)$ です。ここで、 n は測定値の数です。上述した拡張カルマン・フィルタをベースとするSOCの推定方法が有効であることを確認するために実験を行いました。その結果、バッテリーのSOCを正確に推定できることが確認できました。また、この方法は、リチウム・イオン・バッテリー・パックのSOHの値の推定にも適用できます。

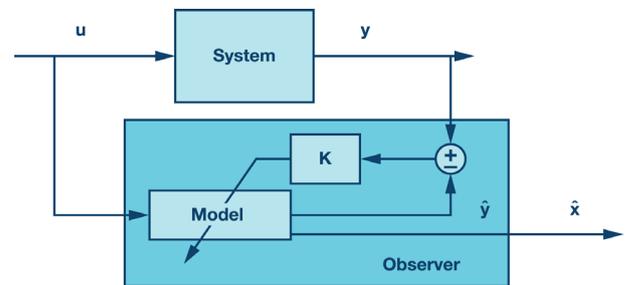


図3. カルマン・フィルタの原理

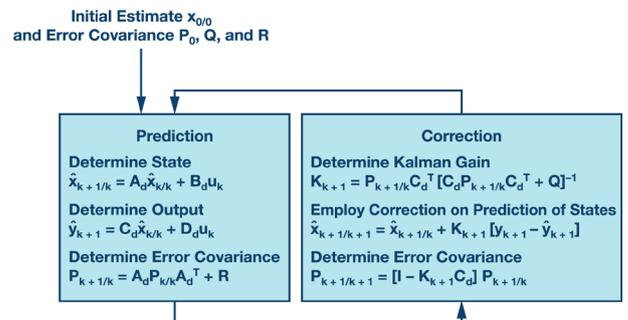


図4. カルマン・フィルタのアルゴリズム

SOC用のアルゴリズムの選択

SOCの推定に用いるものとしては、強化クーロン・カウント・アルゴリズムが有力な選択肢になると考えられます。演算能力、必要な精度、リアルタイム処理に関する制約、システム環境の観点から見て、アプリケーションの要件に適合させる上では同アルゴリズムが最も有利だと言えるでしょう。実際、このアルゴリズムはシンプルなりアルタイムの演算をベースとしており、ハードウェアに関する複雑な制約がありません。複雑さの面では、他のアルゴリズムと比べて明らかに有利でしょう。また、強化クーロン・カウント・アルゴリズムの推定誤差は小さく、アプリケーションで許容可能な精度が得られるはずで。更に、このアルゴリズムはメーカーから提供されるデータ以外に追加の情報を必要としません。

強化クーロン・カウント・アルゴリズムの評価

ここからは、本稿で紹介した強化クーロン・カウント・アルゴリズムの精度と性能の評価について説明します。先述したように、拡張カルマン・フィルタを使用する場合、多くの演算が必要になります。使用するハードウェアに対しては複雑な要件が求められることは明らかです。そのため、SOCの評価を必要とするアプリケーションには適していません。一方、SOC向けの汎用アルゴリズムの評価にはSOC対OCVのカーブの情報が必要です。しかし、その情報はバッテリーのデータシートには記載されていません。SOC向けの汎用アルゴリズムを評価するには、そのカーブの情報を入手することが不可欠です。

以下では、まず強化クーロン・カウント・アルゴリズムの最初の評価ステップについて説明します。そのステップを踏めば、実験によって現実的に得られる電圧／電流の値を処理するために、他の高度なステップを適用することができます。

評価の概要

まず、強化クーロン・カウント・アルゴリズムによって取得したSOCの値を、バッテリーのデータシートに記載されている充放電カーブから取得したSOCの値（実験によって得られた値）と比較します。なお、充放電のカーブは、MATLAB&Simulink（The MathWorks製）のモデルを使って再現することも可能です。MATLAB&Simulinkでは、最も一般的な種類の充電式バッテリー、特にリチウム・イオン・バッテリーを表現できるようにパラメータ化された汎用の動的モデルを使用できます。

シミュレーション結果

まず、シミュレーションを利用して、実装済みの強化クーロン・カウント・アルゴリズムのテストを実施しました。そのシミュレーションでは、充電モード、放電モードを対象としました。それに加え、両モードを組み合わせたケースも対象としました。シミュレーション・ツールとして使用したのはMATLABです。図5において、青色のカーブは、実験で取得したSOCの値（以下、実験値）を表しています。一方、赤色のカーブとしてプロットしたのは、強化クーロン・カウント・アルゴリズムによって推定したSOCの値（以下、推定値）です。

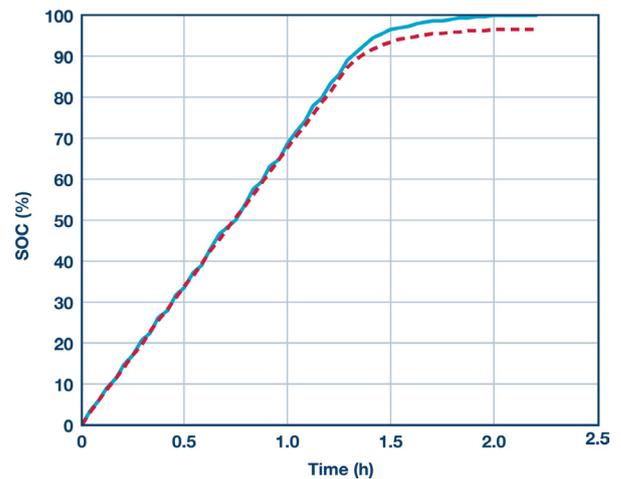


図5.完全充電の段階におけるSOC。実験値と推定値をプロットしています。

充電モード

図5は、完全充電の段階におけるSOCの実験値と推定値を表しています。実験値と推定値の間の最大誤差は、充電段階の最後の時点で約3.5%です。ただ、SOHの再評価を行うと、誤差は大幅に小さくなります。

図6、図7は、定電流モードの段階と定電圧モードの段階のSOCをプロットしたものです。SOCの実験値と推定値が時間の経過に伴ってどのように変化するのが見てとれます。SOHの値を再評価する前にアルゴリズムを実行した場合、最後に得られる最大推定誤差は、定電流モードの段階では2%未満、定電圧モードの段階では1%未満になります。バッテリーが満充電になる際、SOHを再評価する前の推定誤差はアルゴリズムの実行時間に依存して増大することがわかります。また、推定誤差を低減するには、SOCの初期値を正確に決定することが非常に重要です。充電効率を正確に評価することも、SOCの実験値と推定値の間の誤差を低減することにつながります。

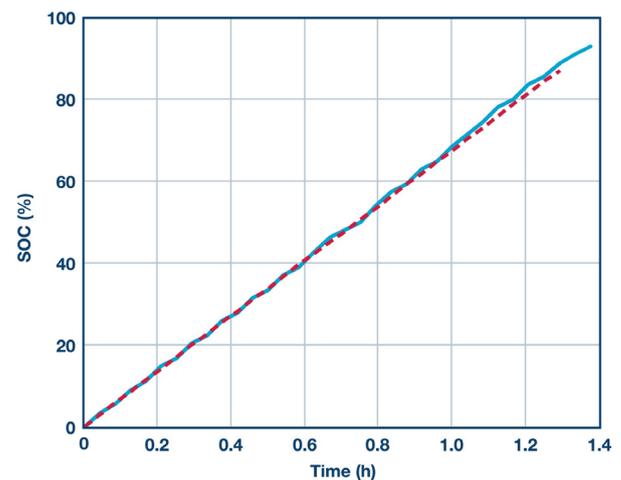


図6. 定電流モードの充電段階におけるSOCの実験値と推定値

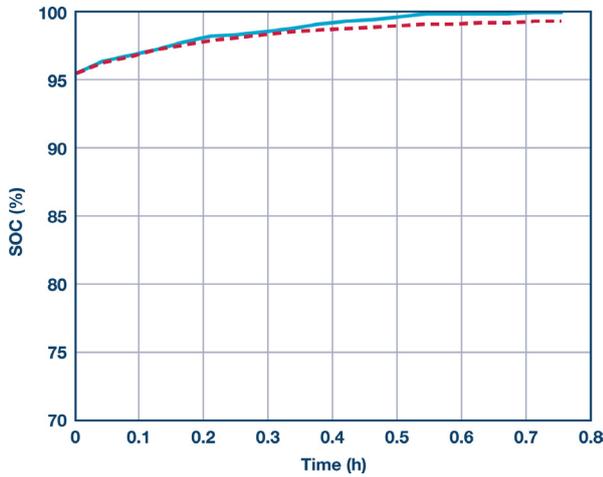


図7. 定電圧モードの充電段階におけるSOCの実験値と推定値

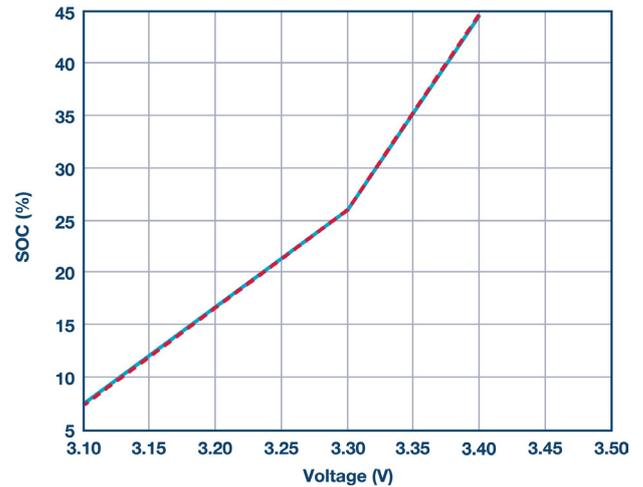


図9. 部分放電の段階におけるSOCの実験値と推定値

放電モード

図8、図9は、完全放電の段階と部分放電の段階におけるSOCの実験値と推定値を示したものです。X軸は、バッテリーの端子電圧を表しています。最大推定誤差は、完全放電の段階では2%以内に収まっており、部分放電の段階ではほぼゼロになっています。SOHの値を再評価する前は、推定値の誤差は、完全放電の段階の最後に最大値に達しています。この誤差は、アルゴリズムの実行時間に依存して増大することがわかります。

充放電を組み合わせた場合についても、強化クーロン・カウント・アルゴリズムの評価を実施しました。この評価では、バッテリー・パックを実際に使用しているときの挙動が再現されます。多くの場合、推定誤差は十分に小さく抑えられます（4%未満）。また、バッテリー・パックの使用を妨げることなく、リアルタイムかつ正確にSOCの値を推定できることが確認できました。

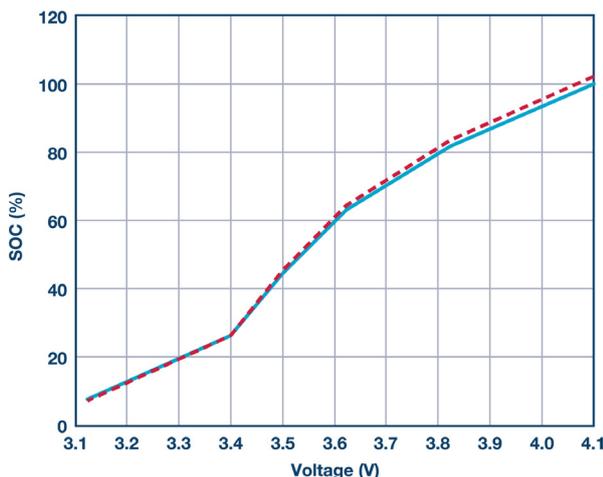


図8. 完全放電の段階におけるSOCの実験値と推定値

参考資料

B.S.Bhangu, P.Bentley, C.M.Bingham, D.A.Stone 「[Nonlinear Observers for Predicting State-of-Charge and State-of-Health of Lead-Acid Batteries for Hybrid-Electric Vehicles](#) (ハイブリッド車用の鉛蓄電池の充電状態／健全性を予測するための非線形オブザーバ)」 IEEE Transactions on Vehicular Technology、2005年

Yi-Ping Chen, Chin-Sien Moo, Kong Soon Ng, Yao-Ching Hsieh 「[Enhanced Coulomb Counting Method for Estimating State-of-Charge and State-of-Health of Lithium-ion Batteries](#) (リチウム・イオン・バッテリーの充電状態／健全性を推定するための強化型クーロン・カウント法)」 Journal of Applied Energy、2009年

Lijin Fang, Fei Zhang, Guangjun Liu 「[A Battery State-of-Charge Estimation Method with Extended Kalman Filter](#) (拡張カルマン・フィルタによるバッテリーの充電状態の推定方法)」 IEEE/ASME International Conference on Advanced Intelligent Mechatronics、2008年

Lei He, Bingjun Xiao, Yiyu Shi 「[A Universal State-of-Charge Algorithm for Batteries](#) (バッテリーの充電状態を推定するための汎用アルゴリズム)」 47th IEEE Design Automation Conference "DAC '10"、2010年

Andreas Jossen, Marion Perrin, Sabine Piller 「[Methods for State-of-Charge Determination and Their Applications](#) (充電状態の決定方法とその応用)」 Journal of Power Sources、2001年

著者について

Martin Murnaneは、アナログ・デバイセズ（アイルランド リムリック）の太陽光発電チームに所属しています。以前は車載チームに所属していました。アナログ・デバイセズに入社する前は、エネルギー・リサイクル・システム向けのアプリケーションの開発（Schaffner Systems）、Windowsベースのアプリケーション・ソフトウェア/データベースの開発（Dell Computers）、ストレイン・ゲージ技術を使用する製品の開発（BMS）などに携わっていました。リムリック大学で電子工学の修士号と経営管理学の修士号を取得しています。

Adel Ghazel氏は、WEVIOOホールディングの一部門であるEBSYS（Embedded Systems Technology）の最高技術責任者（CTO）です。2001年からアナログ・デバイセズと連携し、ワイヤレス通信/電力線通信、動画の解析、エネルギー効率に関連する革新的な組み込みアプリケーションの開発に取り組んでいます。1997年より、IEEEのシニア・メンバー。現在は、チュニジアのカルタージュ大学/SUP'COMで電気通信工学の教授を務めています。また、グリーン・スマート通信システム（GRESKOM）研究所の所長、ミン・テレコム大学（フランス）工学部の客員教授でもあります。

オンライン・サポート・コミュニティ



アナログ・デバイセズのオンライン・サポート・コミュニティに参加すれば、各種の分野を専門とする技術者との連携を図ることができます。難易度の高い設計上の問題について問い合わせを行ったり、FAQを参照したり、ディスカッションに参加したりすることが可能です。

ez.analog.com にアクセス

* 英語版技術記事は [こちら](#) よりご覧いただけます。

アナログ・デバイセズ株式会社

本社 〒105-6891 東京都港区海岸1-16-1 ニューピア竹芝サウスタワービル 10F
大阪営業所 〒532-0003 大阪府大阪市淀川区宮原3-5-36 新大阪トラストタワー 10F
名古屋営業所 〒451-6038 愛知県名古屋市中区牛島町6-1 名古屋ルーセントタワー 38F

©2017 Analog Devices, Inc. All rights reserved.
本誌記載の商標および登録商標は、各社の所有に属します。
Ahead of What's Possible はアナログ・デバイセズの商標です。

www.analog.com/jp

TA15421-0-2/17

 **ANALOG
DEVICES**
AHEAD OF WHAT'S POSSIBLE™