

# MATLAB<sup>®</sup>を用いた リチウム・イオン電池の実践研究

立命館大学 福井正博

# リチウム・イオン電池の用途、 基本特性、性能要求

## EV、続いて定置用蓄電池ブームが来る

### EVブームの背景：急激な環境対応車シフト

中国：環境問題によりEVにシフト（ZEV規制：2020年に100万台）

米国：カリフォルニアZEV規制：2025年に150万台

ドイツ：燃費規制（ZEV規制：2020年に100万台）

→トヨタ、ホンダもEV開発をせざるを得なくなっている

→充電器普及が加速される→EVがより普及推進される

→共同住宅へも充電器設置が進む

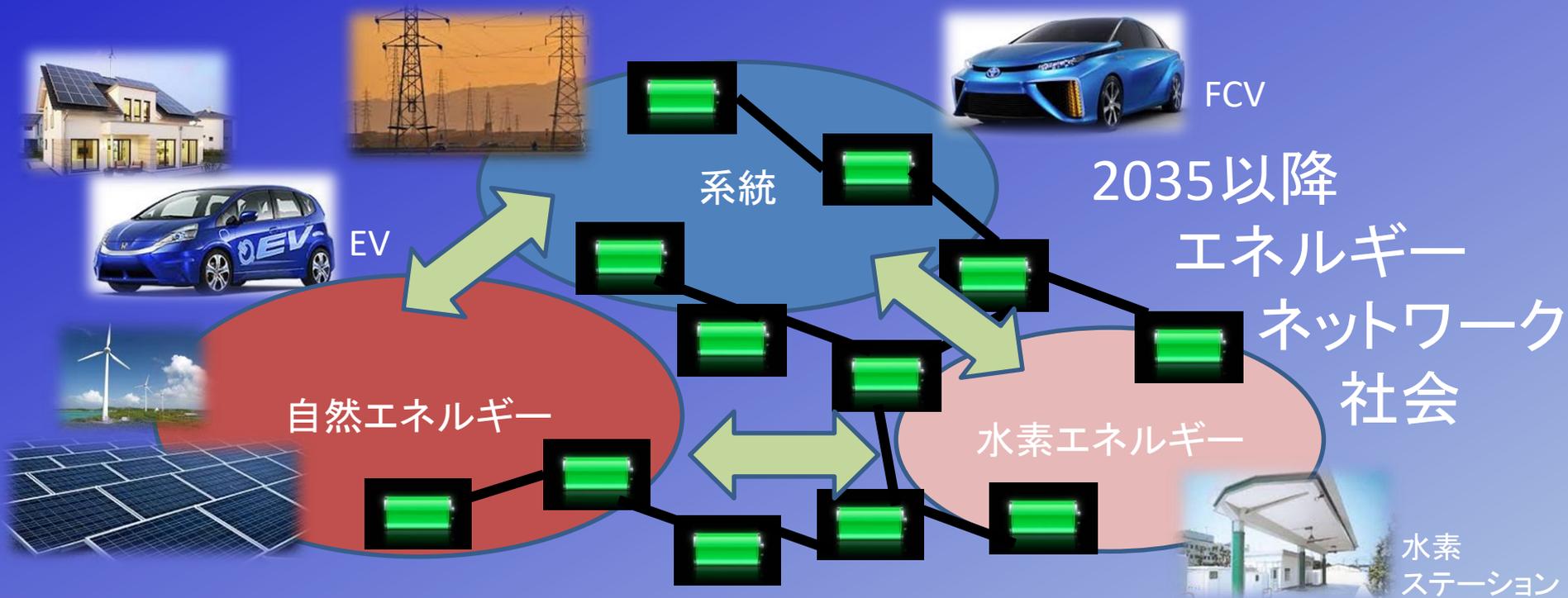
→自動車各社が蓄電池保障に向かう

（5年保証、あるいは10万キロ補償）

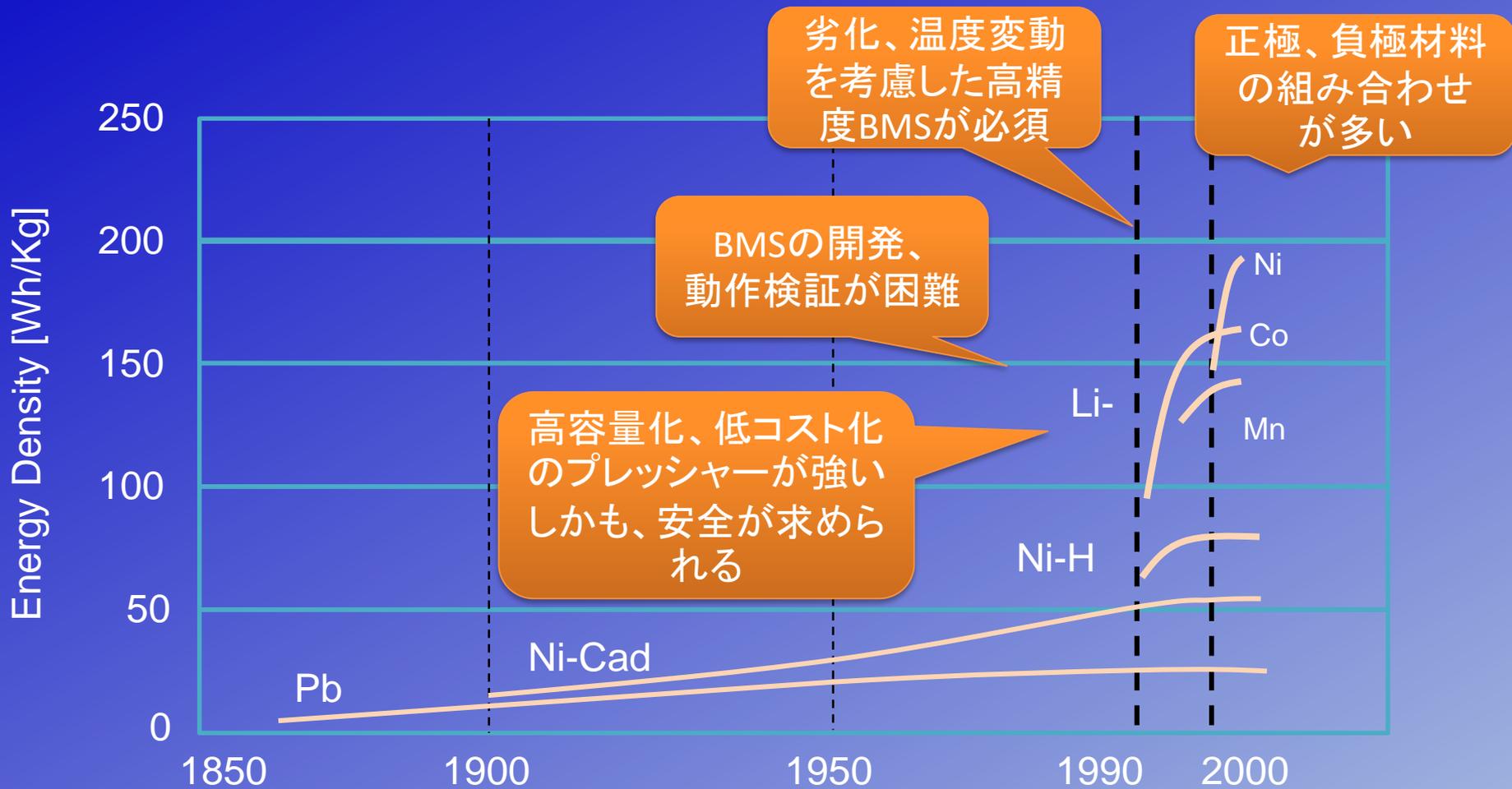
# エネルギーの多極化と電力融通

- V2G:どこで給電するか？
- G2V:どこで充電するか？
- F2V:どこで水素補給するか？
- G2F:どこで水素に変換するか？

- スマートセンサー:  
蓄電池の常時管理
- 最適充放電
- 最適エネ変換(電・蓄・水)

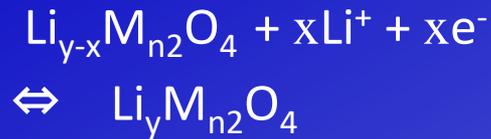


# LiBは夢の蓄電デバイス. しかし。。。

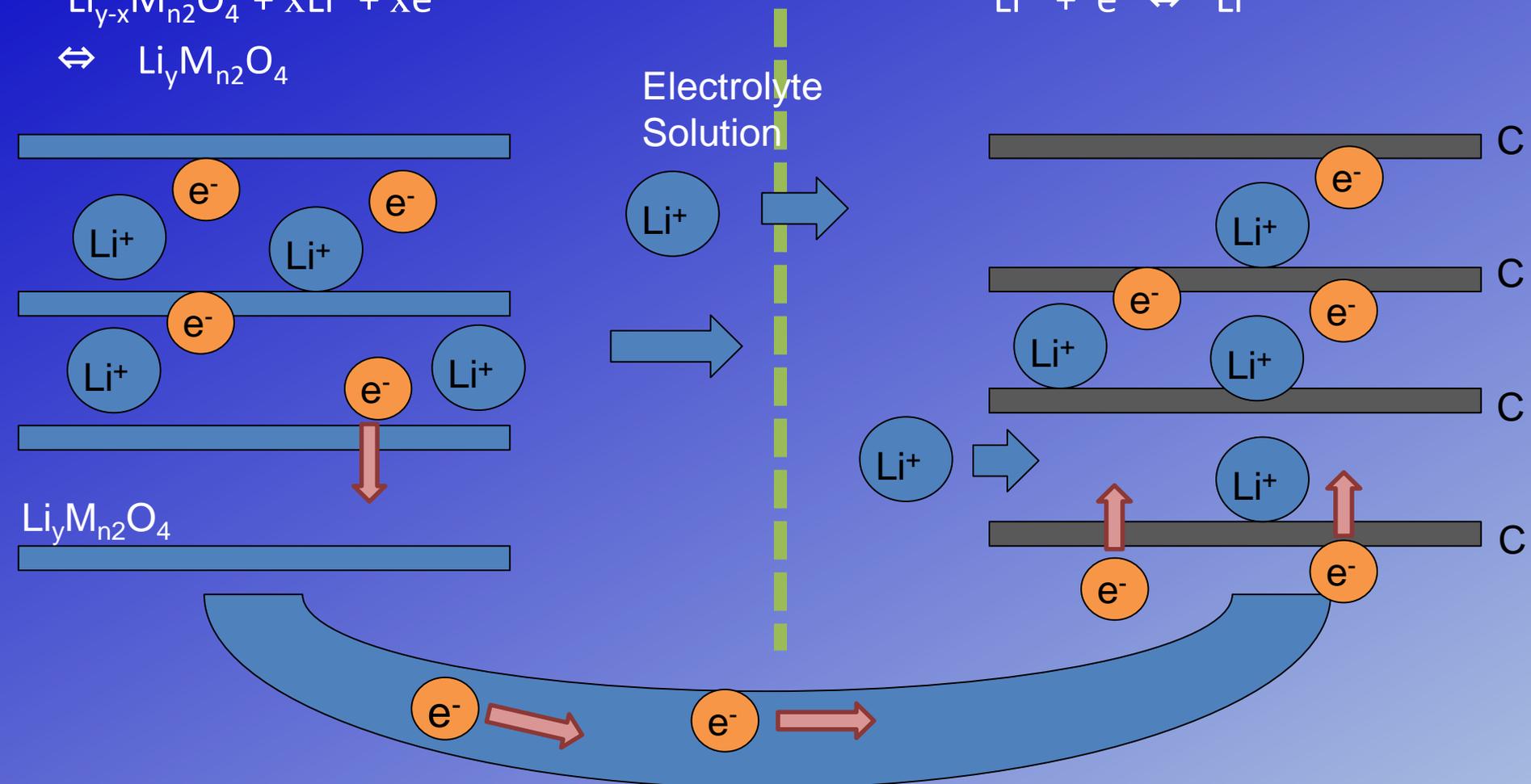


# リチウムイオン蓄電池の仕組み(充電)

Cathode:

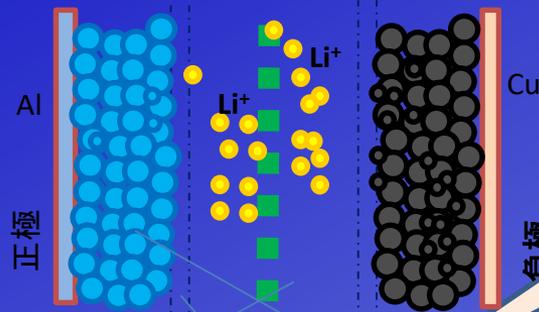


Anode:



# 電池構造と等価回路の関係

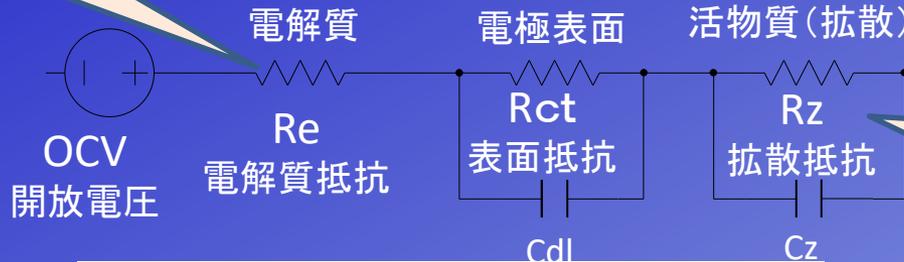
## 電池構造



電解質抵抗  
早い反応

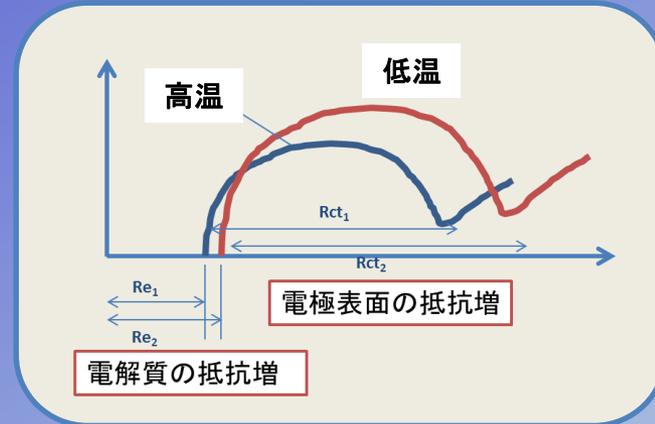
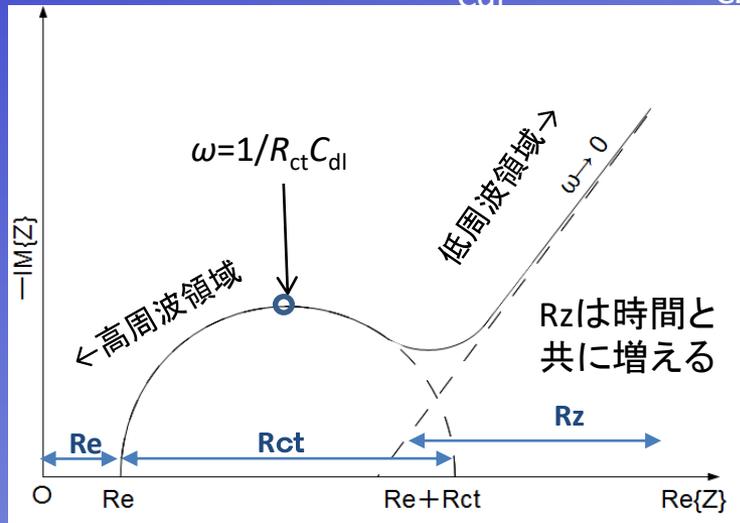
活物質表面抵抗  
早い反応(時定数  
数が数mS)

## 等価回路

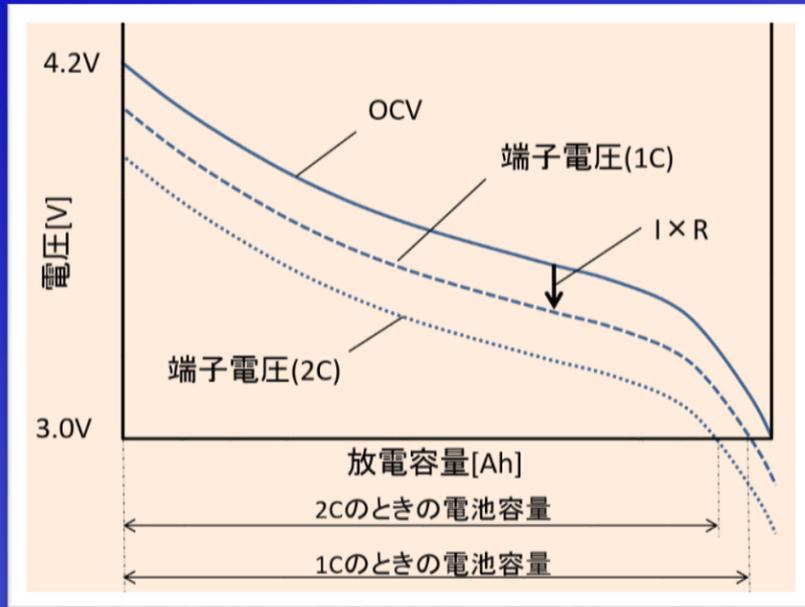


拡散インピーダンス  
遅い反応(時定数が  
30min ~ 2h)

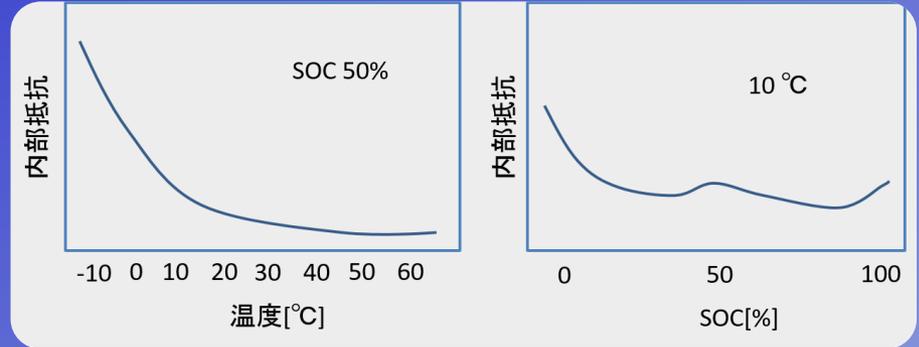
## ナイキスト図 (交流インピーダンス)



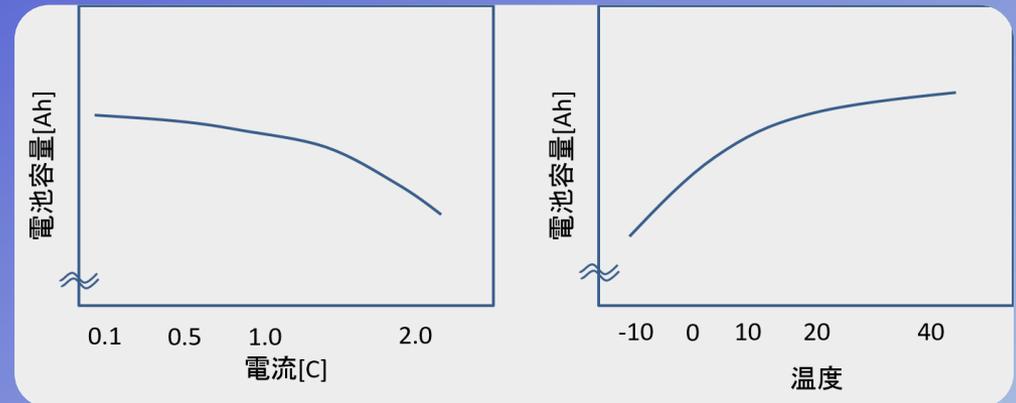
# 放電曲線と電池容量



内部抵抗(内部インピーダンス)は温度、SOC依存



電池容量(FCC)は温度、電流依存



# MATLAB製品を活用した 研究開発事例

(適用例①)

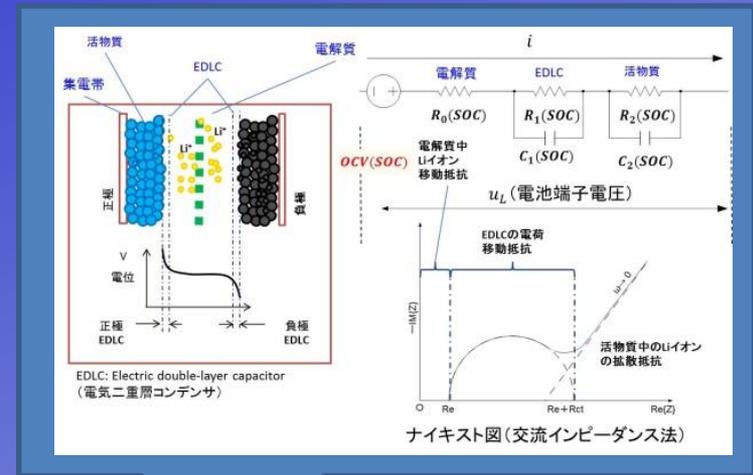
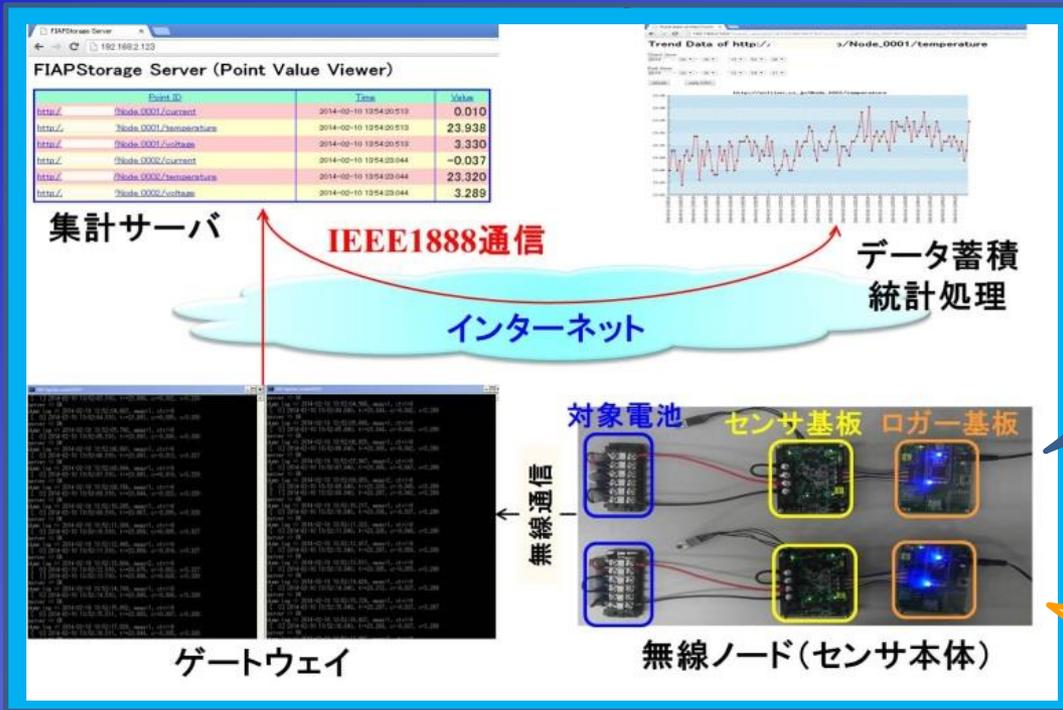
カルマンフィルタを用いた  
残量計アルゴリズム



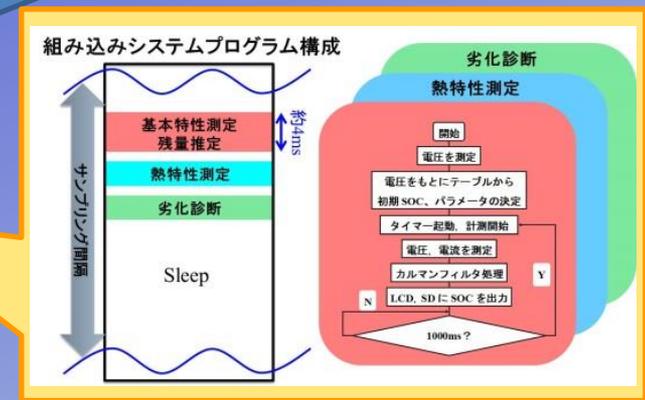
# IoT型蓄電池スマートセンサーの開発

## センサーネットワークによる蓄電池の動的診断

## 蓄電池モデル化技術



## マイコン実装技術

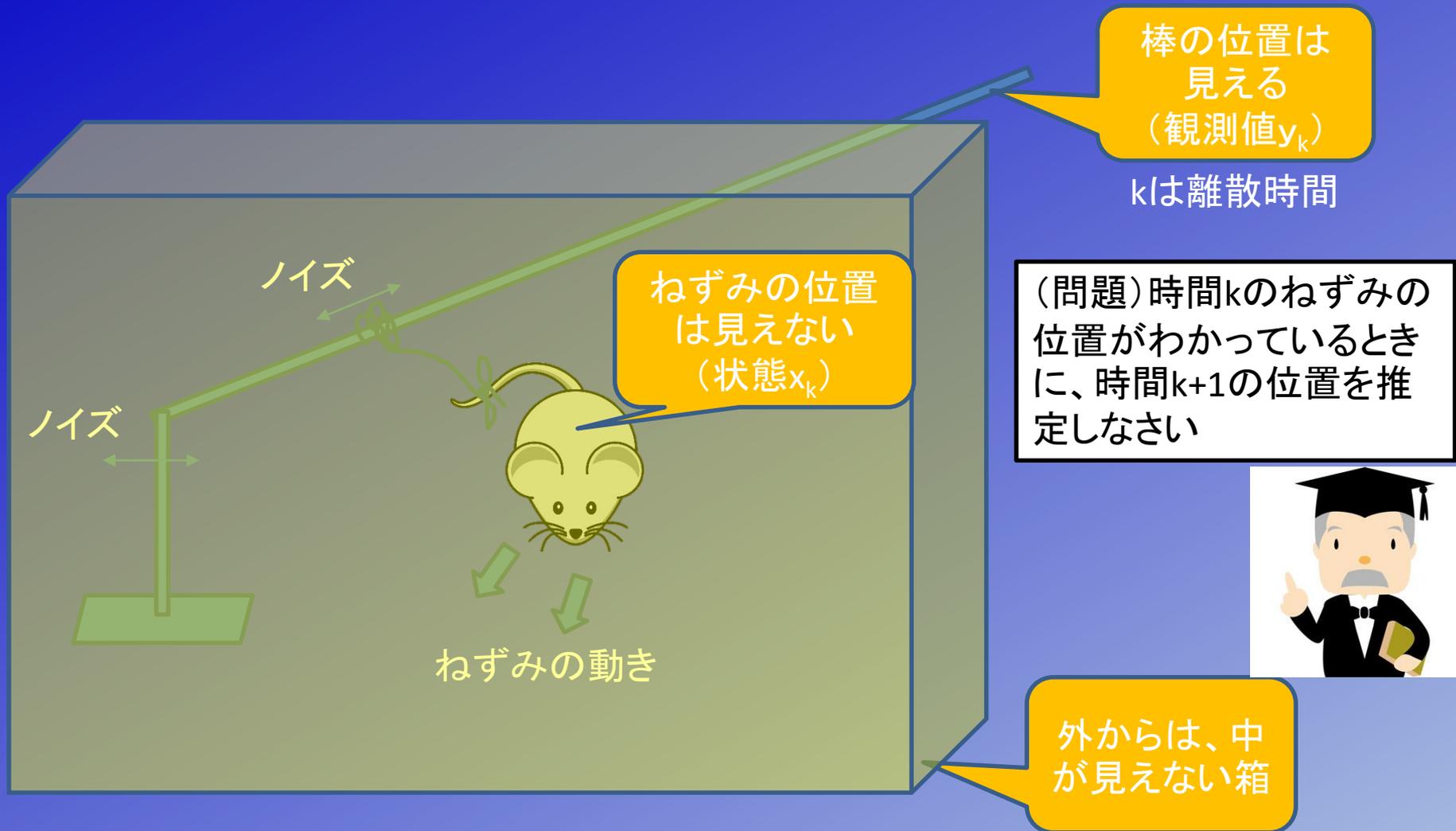


## ネットワーク実装技術

# カルマンフィルタ概説



# (1)カルマンフィルタによる状態予測



# 対象システムの記述

時間k+1での  
ねずみの  
位置

時間kでの  
ねずみの  
位置

外的要因  
(餌のにおいなど)

不確定要因  
(ノイズ)

ねずみは、こ  
の方程式に  
従って動く

$$x_{k+1} = Ax_k + b_u u_k + b \omega_k \quad \text{状態方程式}$$

ねずみの位置  
と棒の位置の  
関係

$$y_k = Cx_k + v_k$$

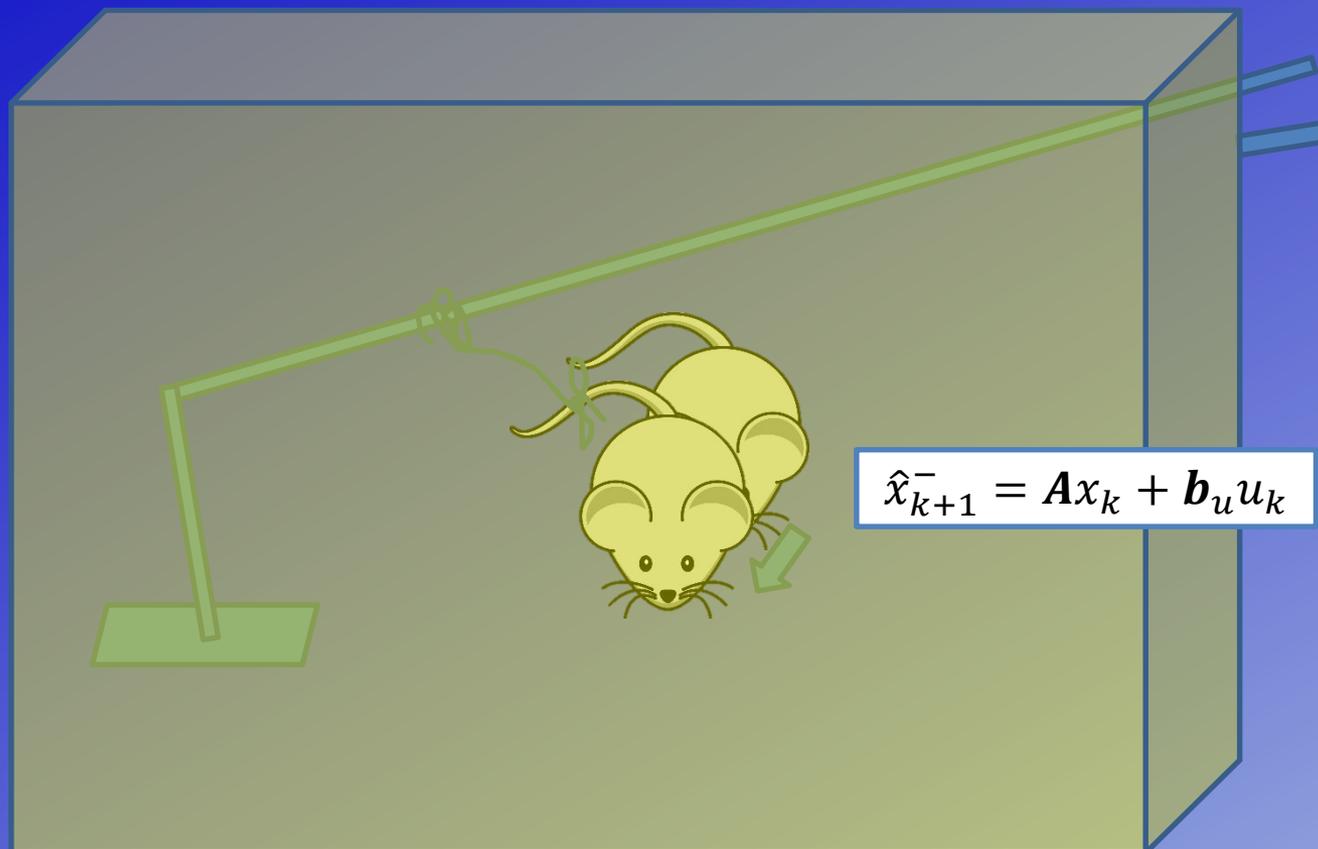
不確定要因  
(観測誤差)

観測方程式

時間kでの  
棒の位置

$\omega_k$  および  $v_k$  は、それぞれ平均が0の正規ノイズ  
従って、 $E[\omega_k]=0$ ,  $E[v_k]=0$

# 予測ステップ:ねずみは状態方程式に従って移動する



# 補正ステップ: 観測値の変化を使って推定精度を上げる

ねずみが  $\hat{x}_{k+1}^-$  いると仮定したときの棒の位置  
(観測値  $\hat{y}_{k+1}^-$ )

実際に見えている棒の位置  
(観測値  $y_{k+1}$ )

棒の位置情報も参考にしてみよう!

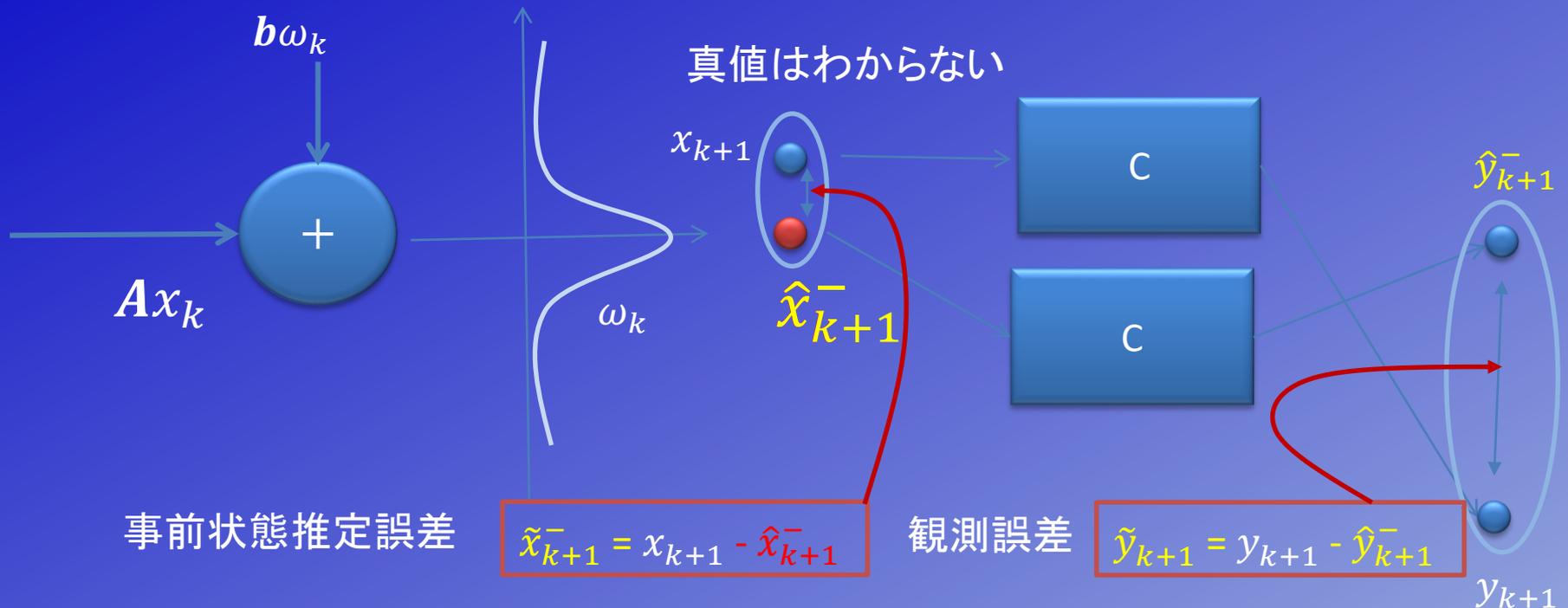
$$\hat{x}_{k+1}^- = Ax_k + b_u u_k$$

$x_{k+1}$

ねずみのいる位置(真値)



# 観測値から推定精度を上げる(事後推定)



$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + g_{k+1} \tilde{y}_{k+1}$$

事後推定

カルマンゲイン

# 拡張カルマンフィルタの処理フロー

## 予測ステップ

$$\hat{x}_{k+1}^- = Ax_k + b_u u_k$$

$$P_{k+1}^- = AP_k A^T + \sigma_w^2 bb^T$$

状態と誤差共分散  
の事前予測

## フィルタリングステップ

$$g_{k+1} = P_{k+1}^- C^T / (C P_{k+1}^- C^T + \sigma_v^2)$$

$$\hat{x}_{k+1} = \hat{x}_{k+1}^- + g_{k+1} (y_{k+1} - \hat{y}_{k+1}^-)$$

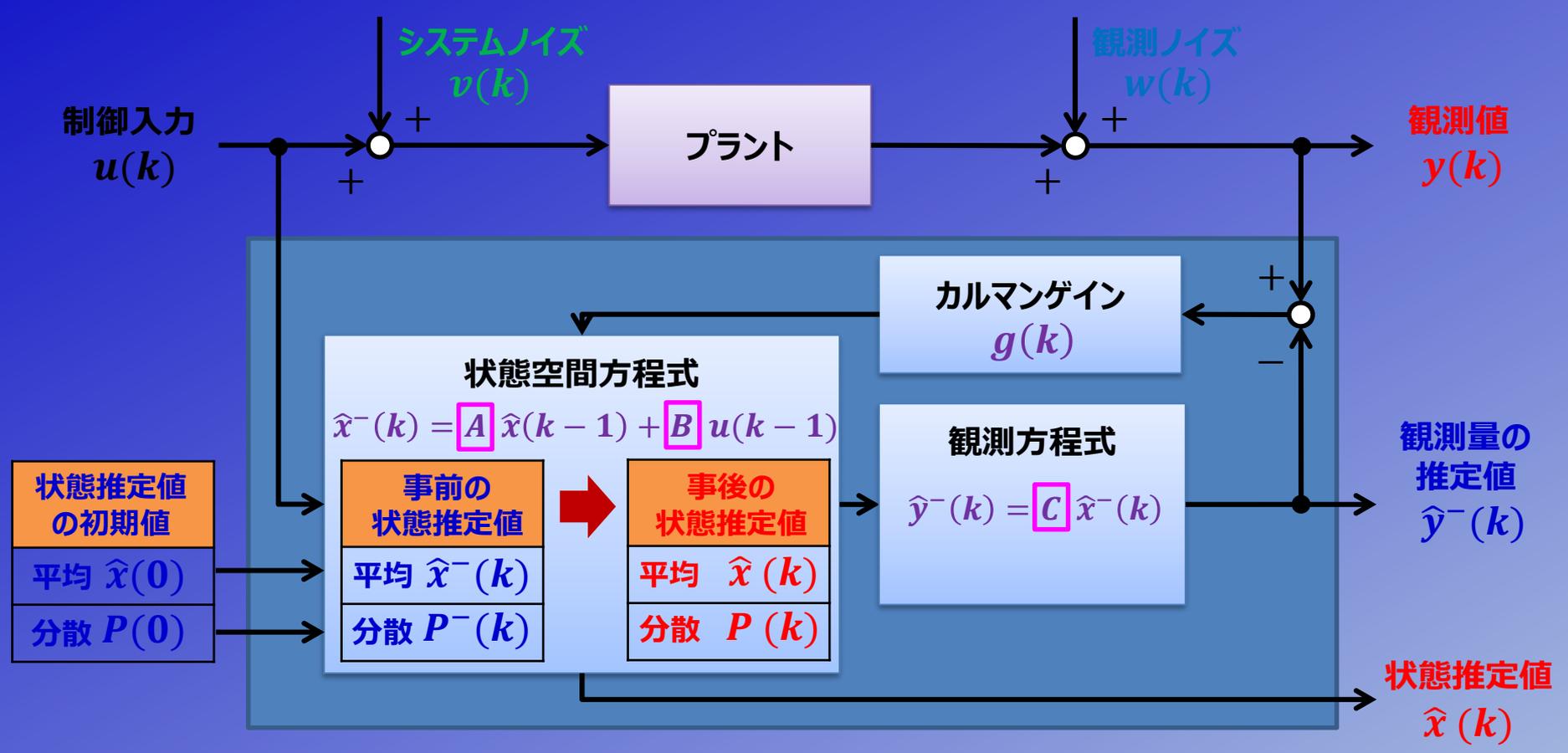
$$P_{k+1} = (I - g_{k+1} C) P_{k+1}^-$$

カルマンゲインの計算

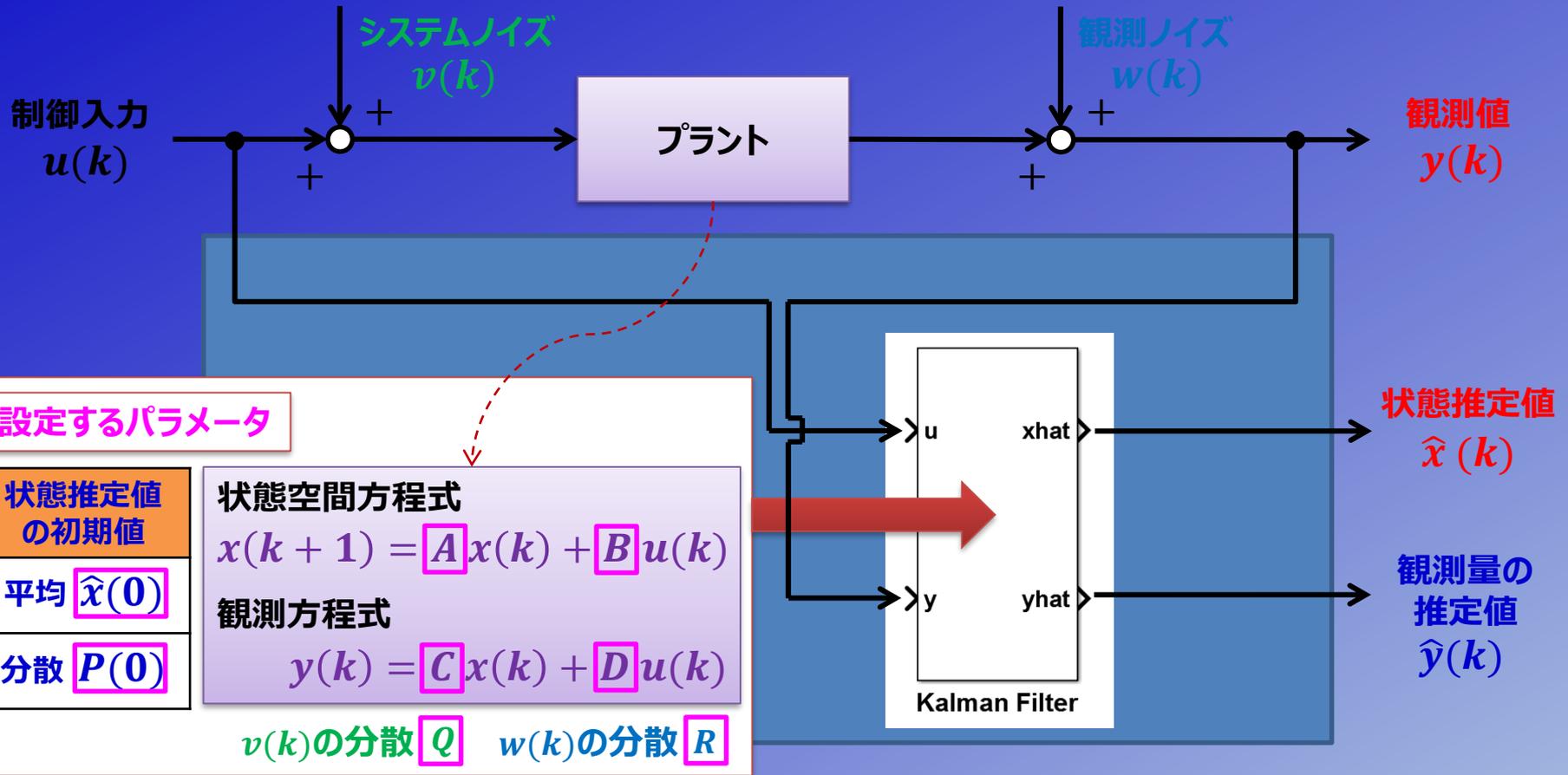
状態と誤差共分散の補正

ねずみの位置  
の補正值

# カルマンフィルタをブロック線図で描くと？

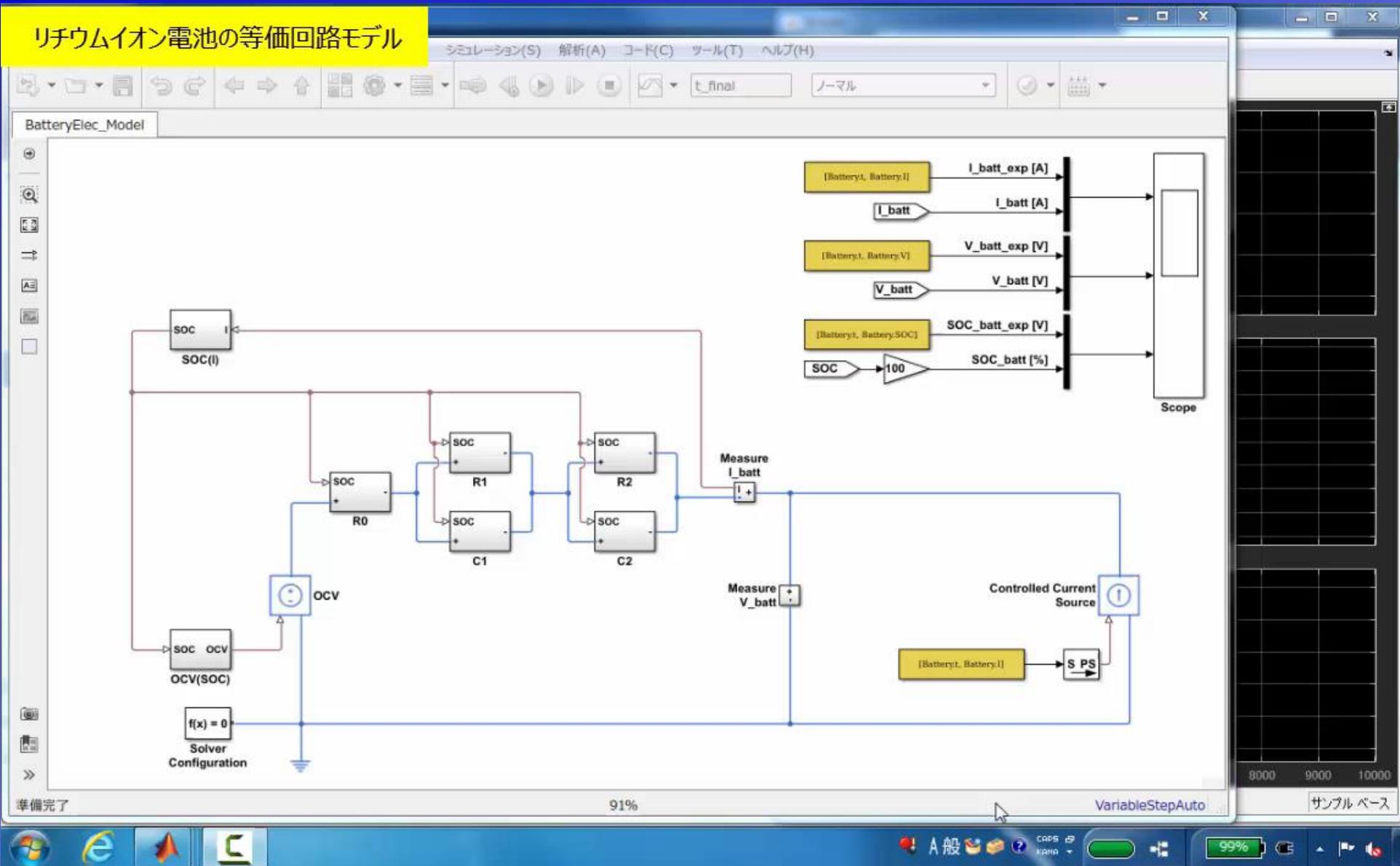


# カルマンフィルタのSimulinkモデル



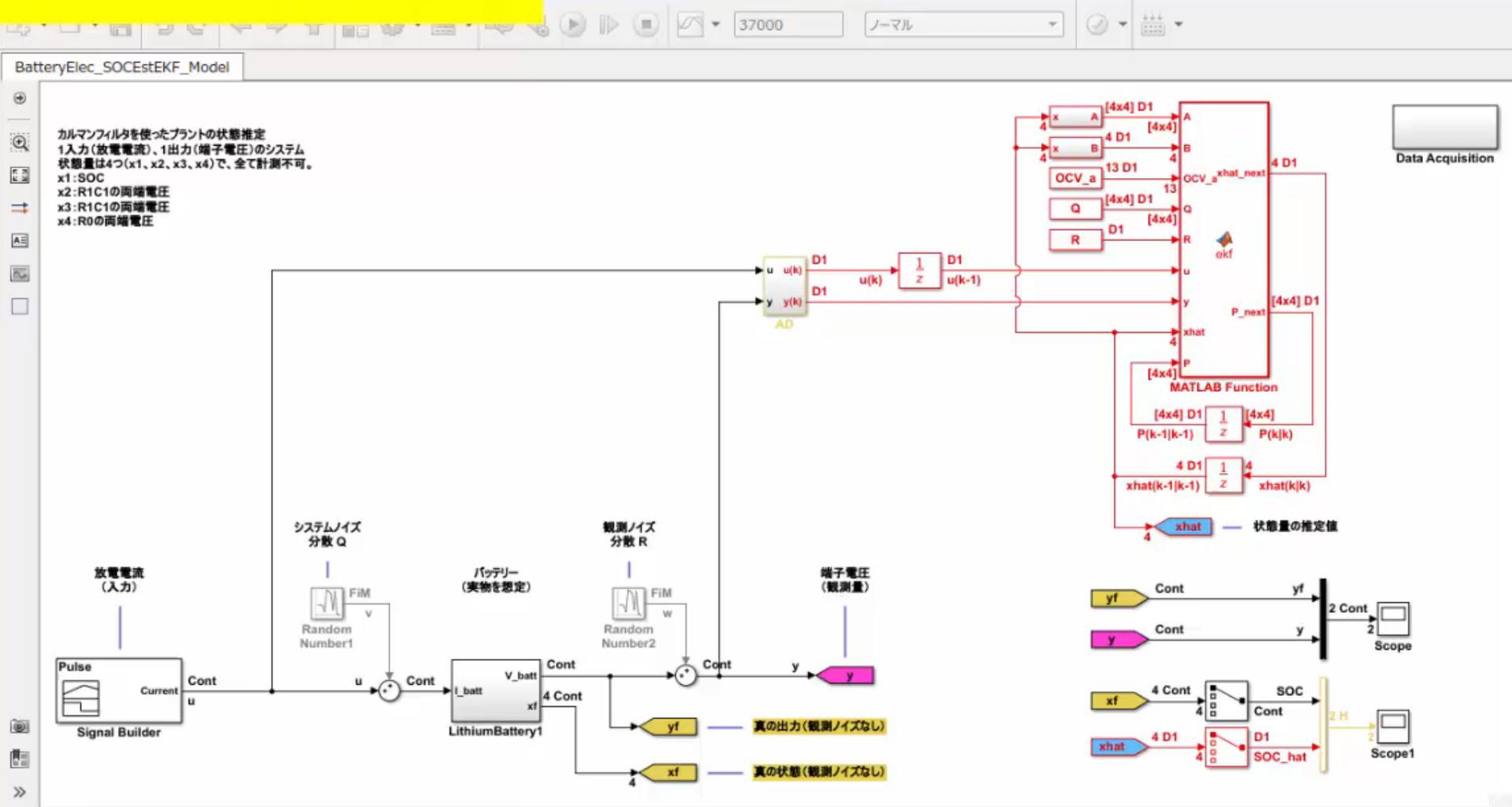
# (MATLABデモ)リチウムイオン蓄電池等価回路モデル構築

## リチウムイオン電池の等価回路モデル



# (MATLABデモ)カルマンフィルタを用いた残量推定システム

カルマンフィルタを使った蓄電残量推定モデル



準備完了

88%

auto(ode23t)

(適用例②)

リチウムイオン組蓄電池用  
性能・劣化シミュレータの構築

# モデルベースシステム設計

## モデル(計算機上)



電池データ

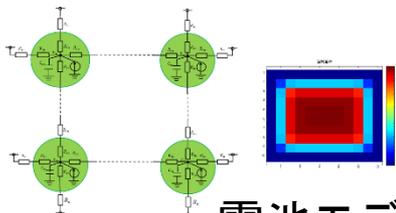


車体走行データ

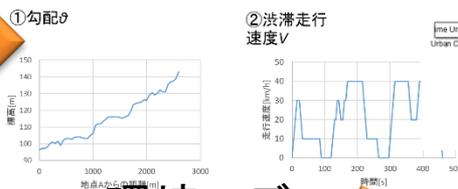


キャリブレーション  
実データ反映

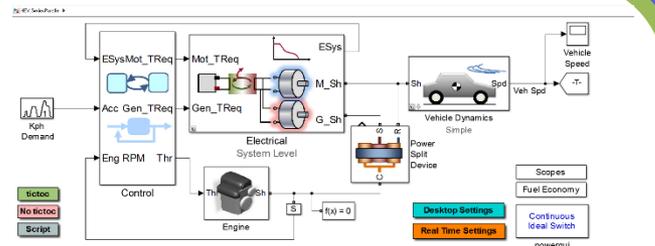
地図データ  
(勾配、カーブ)  
渋滞情報  
(速度、天候)



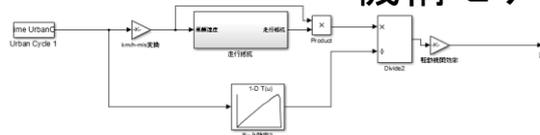
電池モデル



環境モデル



機構モデル

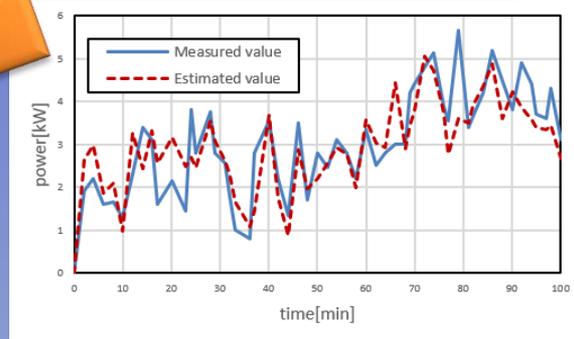


車両モデル

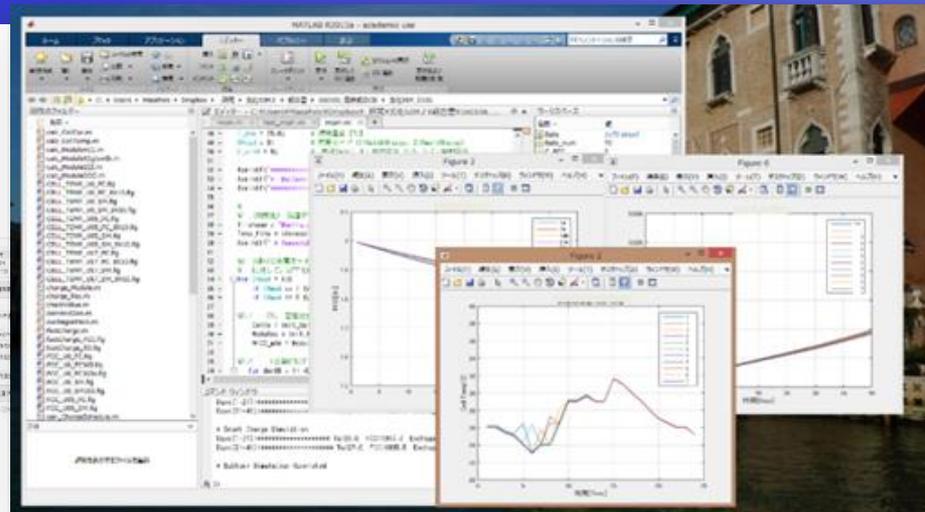
最適化

シミュレーション

実設計への反映、検証



# EV用蓄電池最適化システム



高精度残走行距離メータ

EV用蓄電池  
性能・劣化シミュレータ



実データ収集

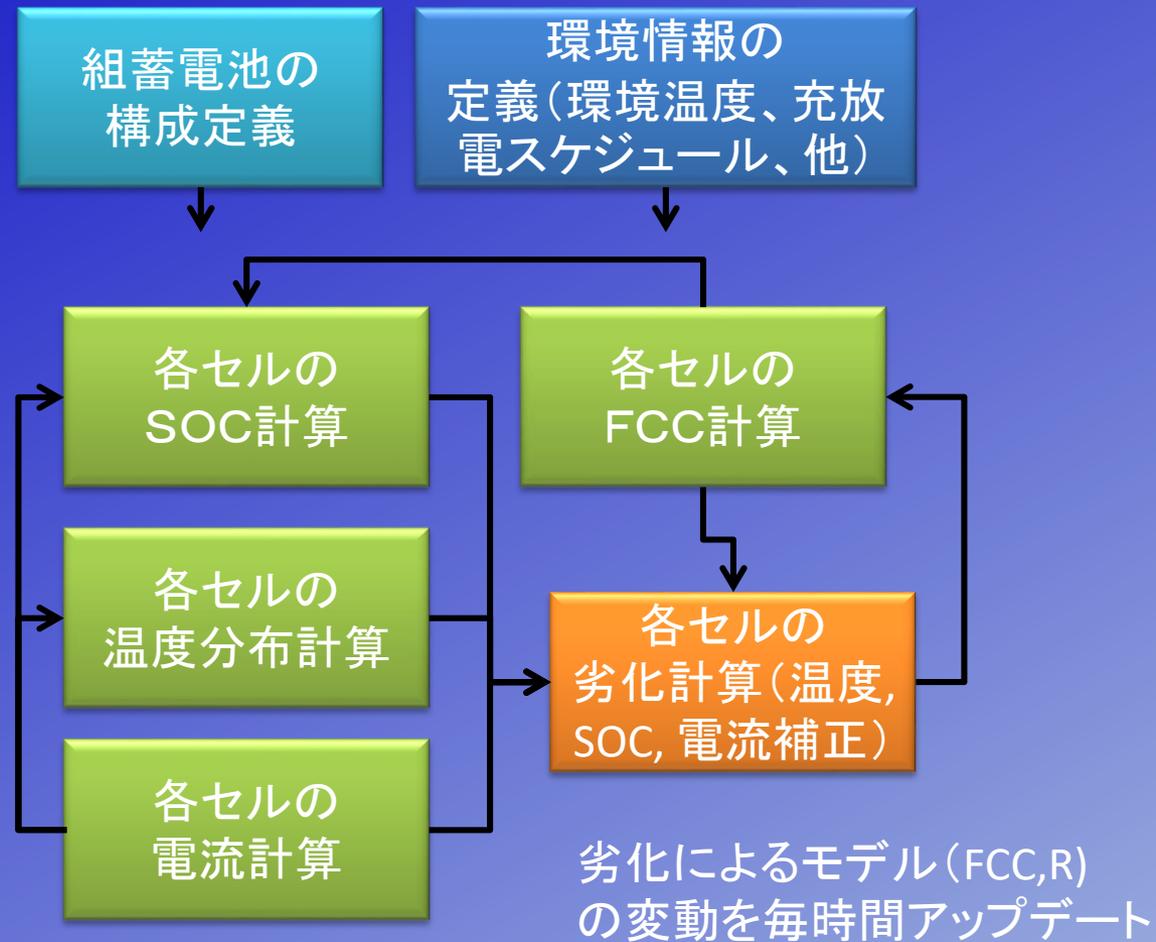


筑波サーキット EVカートレース(2017.10)



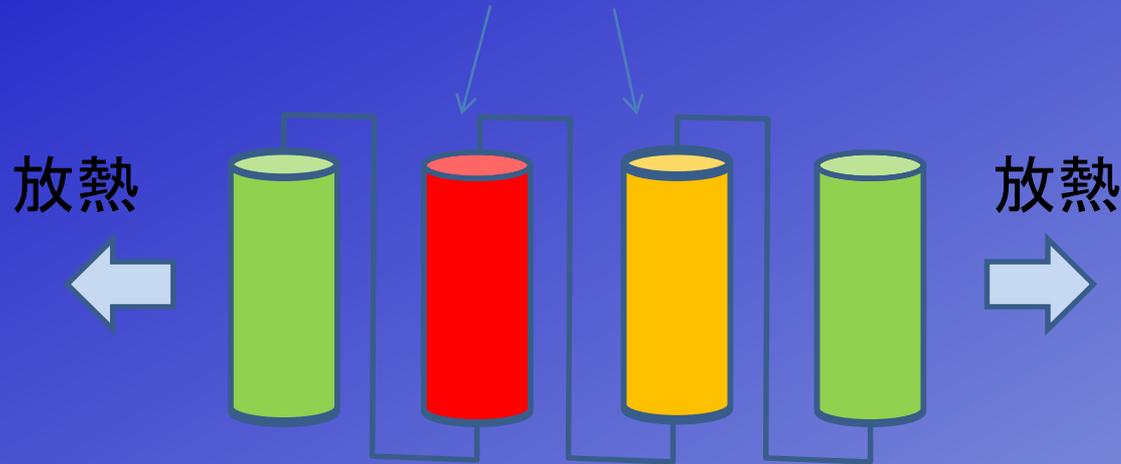
LIB部門2位  
(学生1位)

# 組蓄電池用劣化シミュレータの構成



# 組電池の温度管理

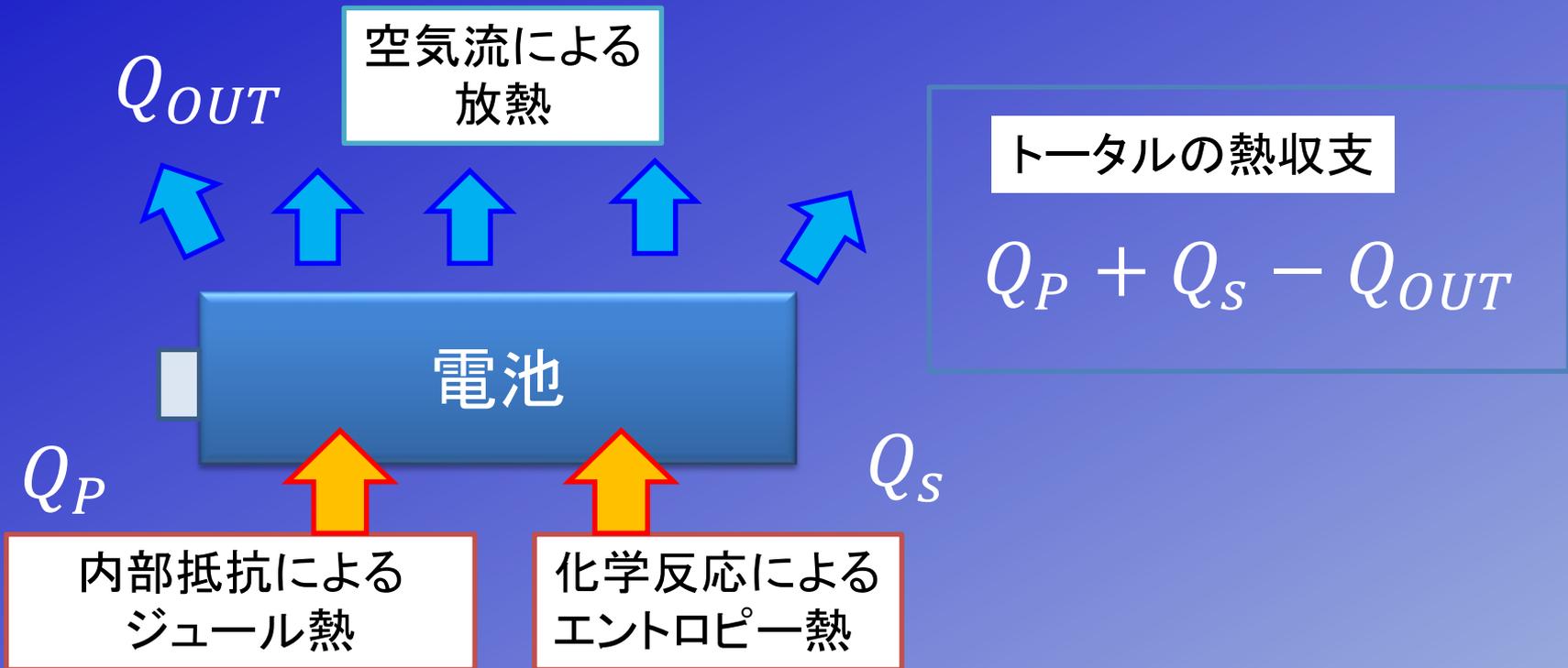
劣化しやすい(バランスが崩れる要因となる)



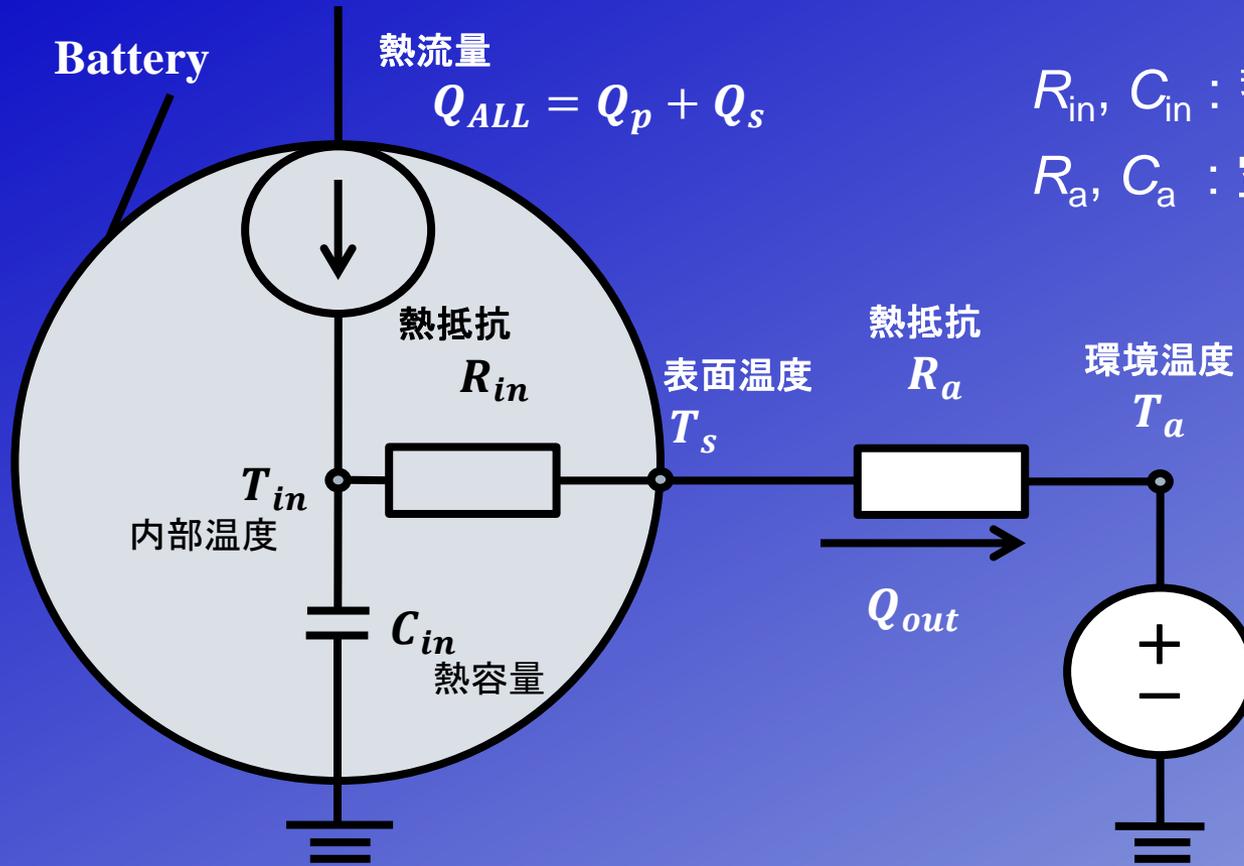
充放電による発熱  
内部ほど熱く、周辺ほど温度が低い

温度、SOC、容量、内部抵抗のばらつきは、性能低下や劣化加速の要因となる

# 発熱と放熱



# 熱回路による表現

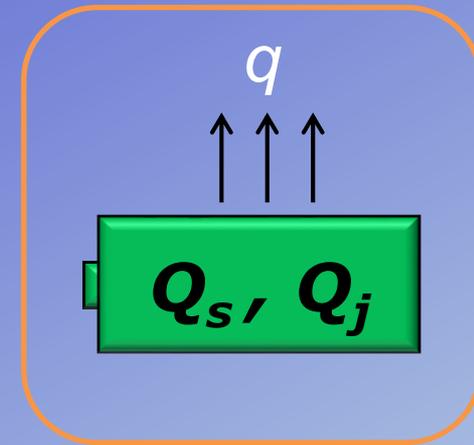


$R_{in}, C_{in}$  : 蓄電池の熱抵抗、熱容量

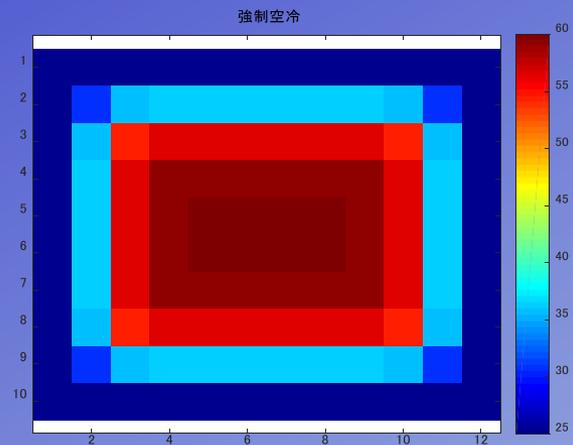
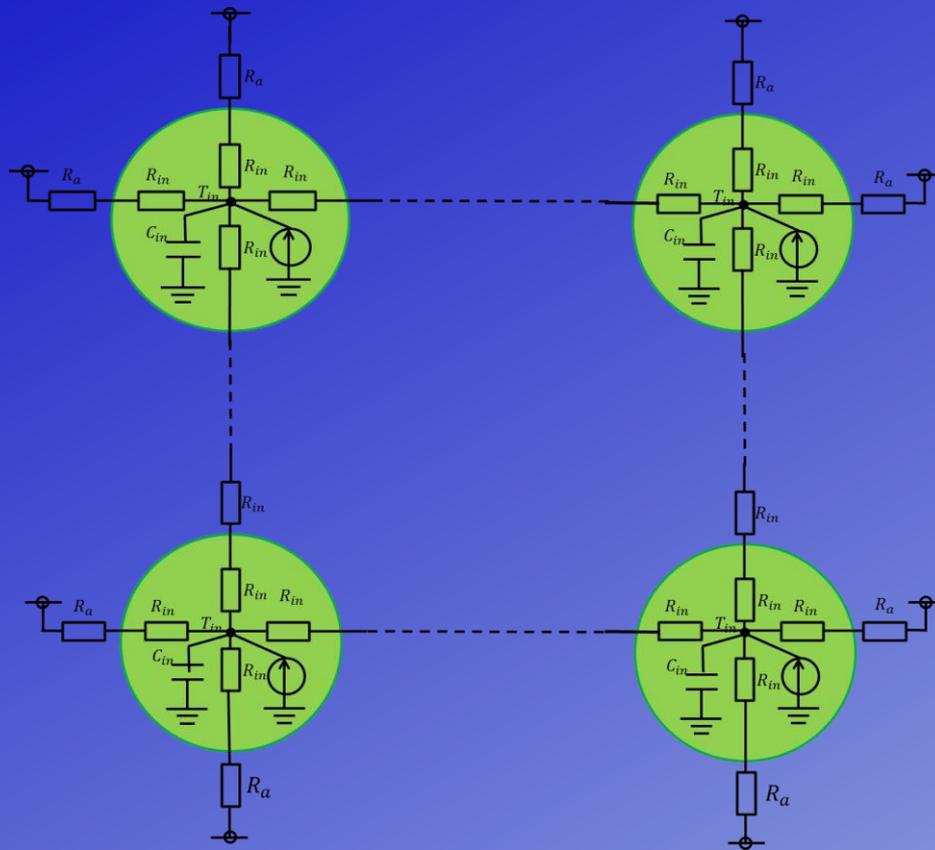
$R_a, C_a$  : 空気の熱抵抗、熱容量

熱容量

$$C_{in} \frac{d(T_{in} - T_s)}{dt} = Q_s + Q_j - q$$



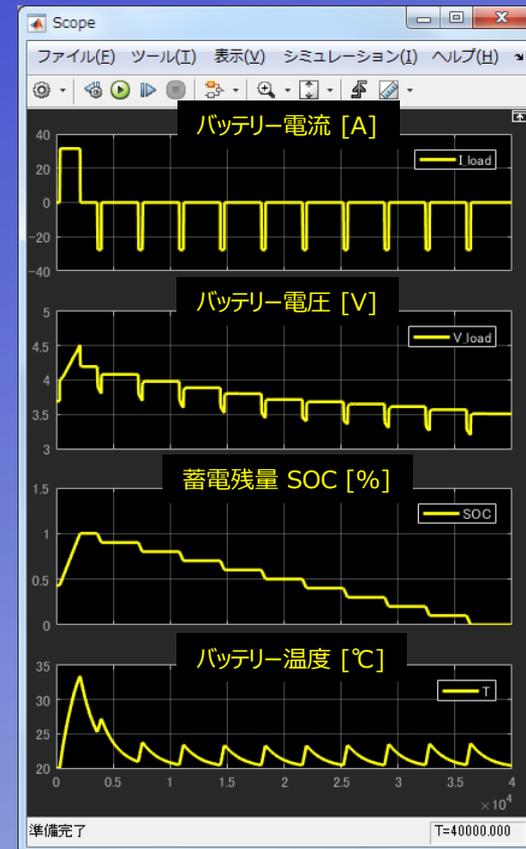
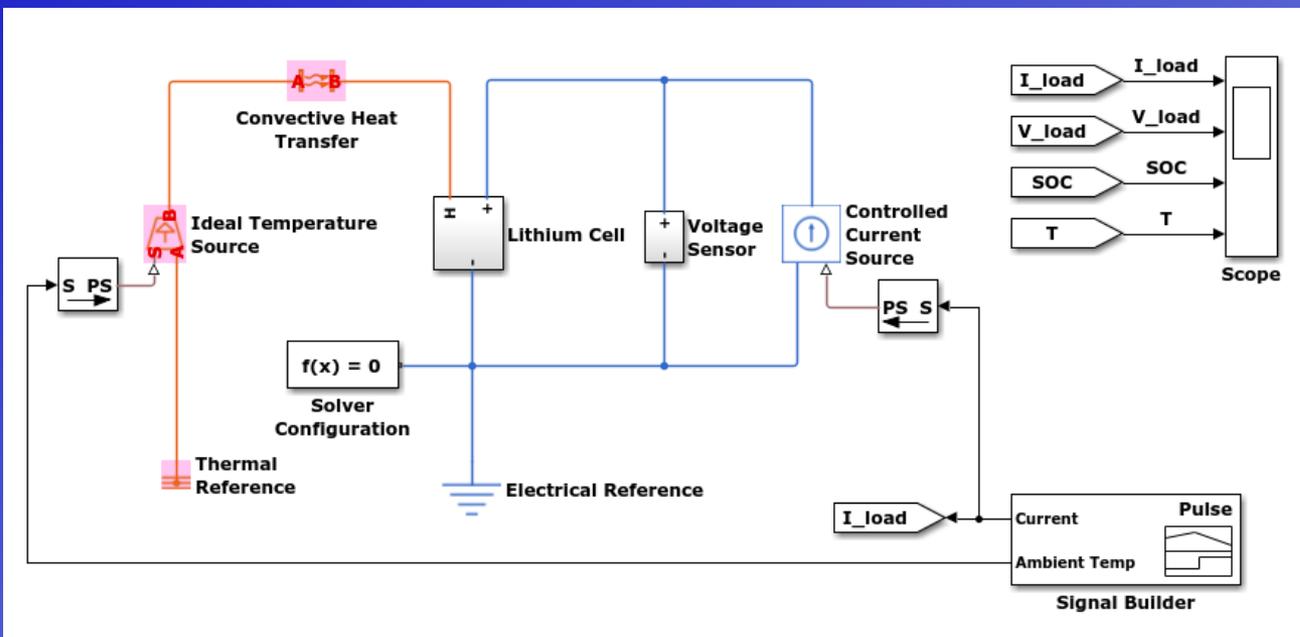
# 熱回路による表現(組電池)



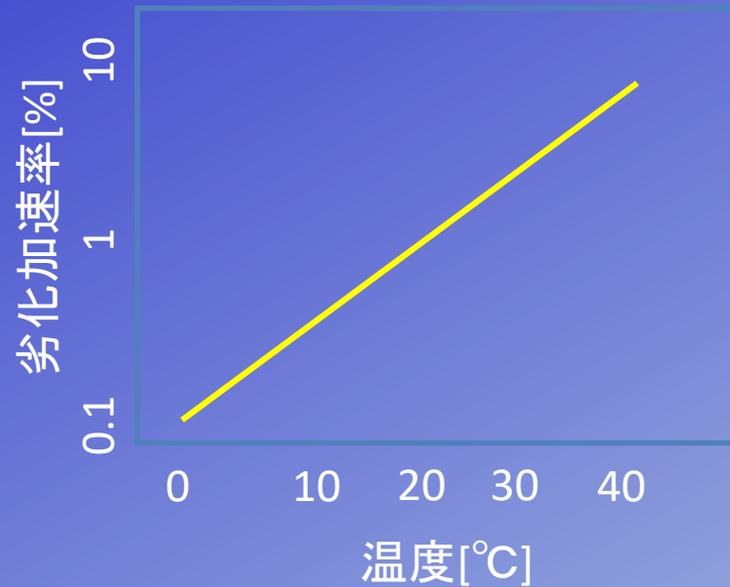
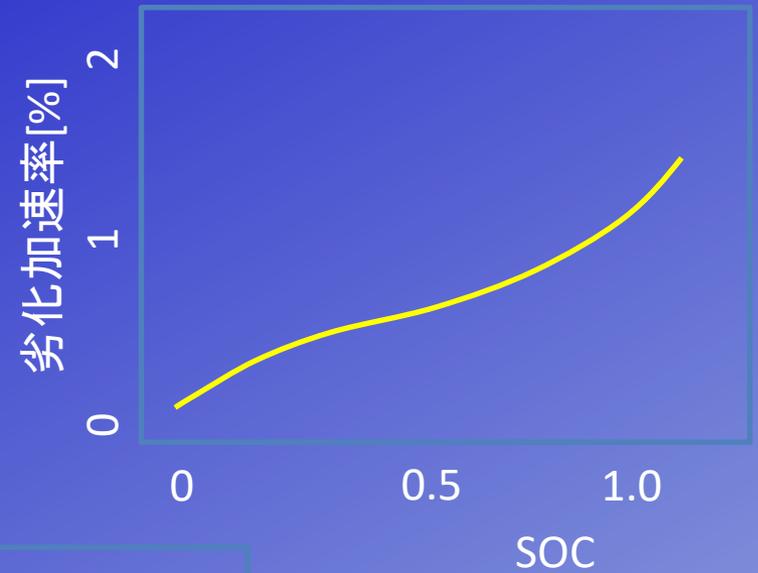
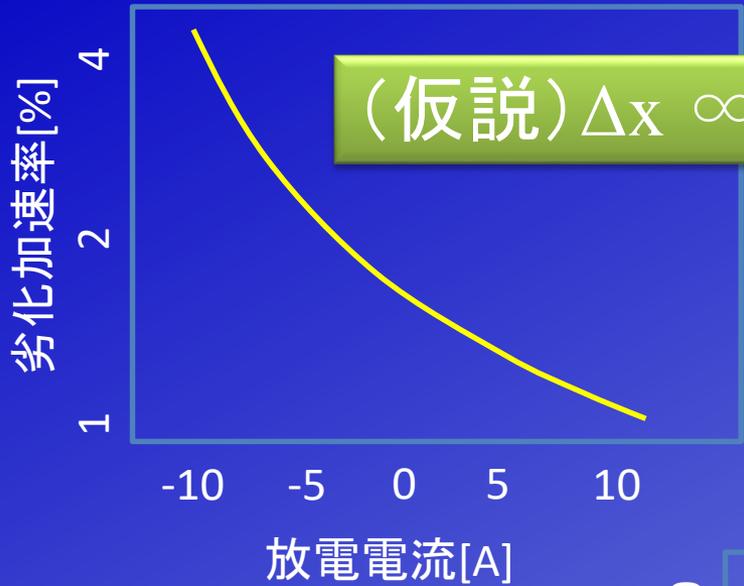
# バッテリーの等価回路モデル(電気、熱)

Simscape

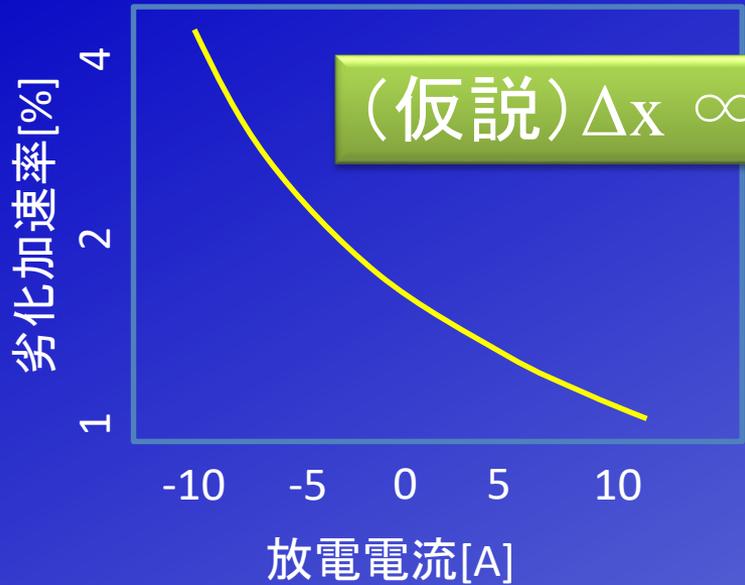
電気系と熱系の連成解析が可能。  
外部環境の温度、バッテリー内部の電力損失に伴う発熱の解析が可能。



# 保存劣化の補正係数



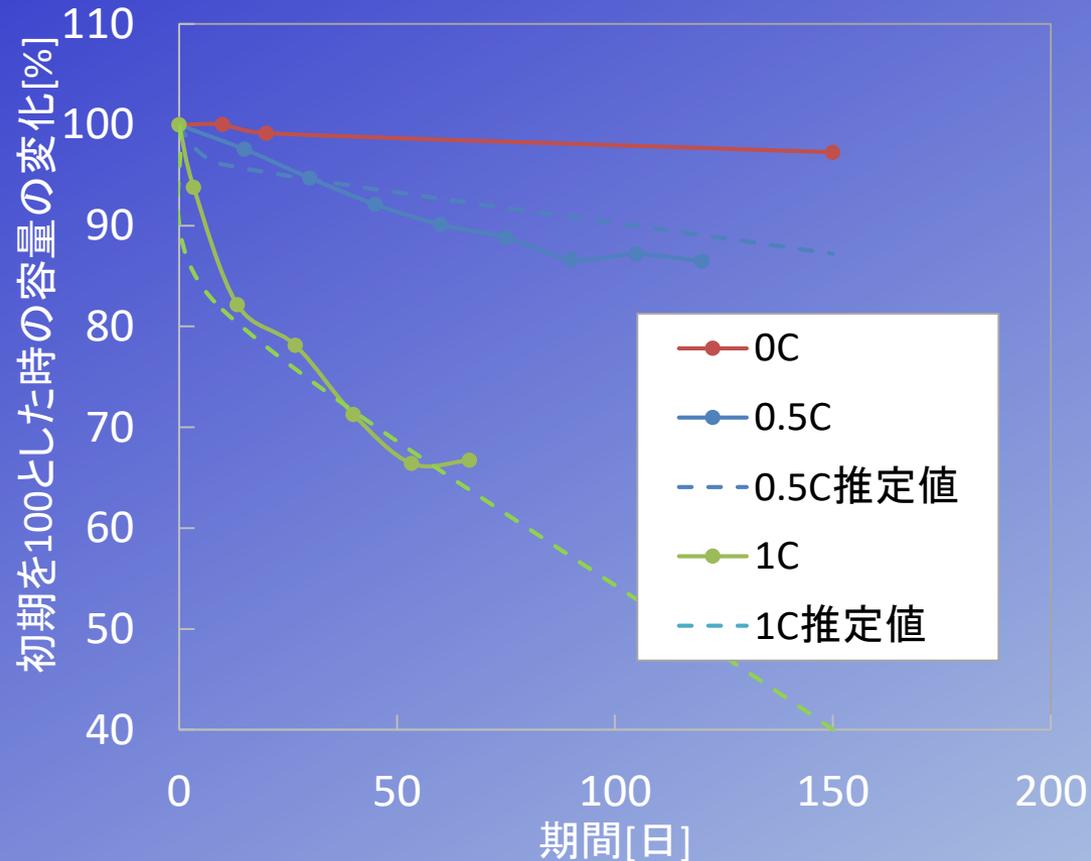
# キャリブレーションによるフィッティング



劣化モデル

フィッティング

実測定



# MATLAB/ Simulinkの歩き方

- 試作: とりあえず試してみるのが簡単  
とりあえずSimulinkで組み立て、工夫したいところはMATLABでコード作成
- 教育: 初学者にイメージを伝えるのが容易  
数式だけだと敷居が高くても、簡単な動作をつけると理解しやすい
- コミュニケーション: 共同研究者との共通理解、  
技術トランスファーが容易
- 自動化: 組込みシステムへの自動アップロードと  
実動作の確認ができる(設計TATの短縮)

# まとめ

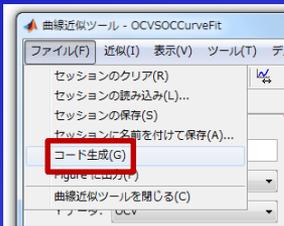
- 蓄電池を中心としたEVやスマートグリッドシステムへのMATLAB活用事例を紹介した
- MATLABは電卓やEXCEL並みの使いやすさでシステムレベルのモデル化、シミュレーション、フィティング、最適化を有機的にこなす
- 化学系(電池)、熱系(温度解析)、電気系(回路)の混在するシステムのモデル化、解析に有効である
- 今後は、MATLABからスマートセンサープログラム(パラメータ)の自動アップデートに取り組む

# 付録

# OCV-SOC特性(実験データ)を多項式で近似

Curve Fitting Toolbox

GUI上で多項式近似の推定作業を一度行うと、その作業を一つの関数の形でMATLABコードに自動変換することが可能。異なるSOC、OCVの実験データを使って同じ多項式近似の作業をしたいときはそれを自動化することが可能。



OCVとSOCの関係を表す実験データを設定する。

The screenshot shows the 'Curve Fitting Toolbox' interface. The 'File' menu is open, highlighting 'Code Generation'. The main window shows 'OCV-SOC特性' as the data name. The 'X Data' is set to 'SOC' and 'Y Data' is set to 'OCV'. The polynomial equation is defined as  $y = a_0 + a_1x + a_2x^2 + a_3x^3$ . The 'Results' pane shows the general model  $f(x) = a_0 + a_1x + a_2x^2 + a_3x^3$  and the estimated coefficients:  $a_0 = 3$ ,  $a_1 = 4.517$ ,  $a_2 = 2.876e+05$ ,  $a_3 = -9.596e+04$ . A plot shows the experimental data points (dots) and the fitted polynomial curve (line).

フィッティングされた多項式の係数の推定結果を確認する。

実験データにフィッティングする多項式の数式を定義する。

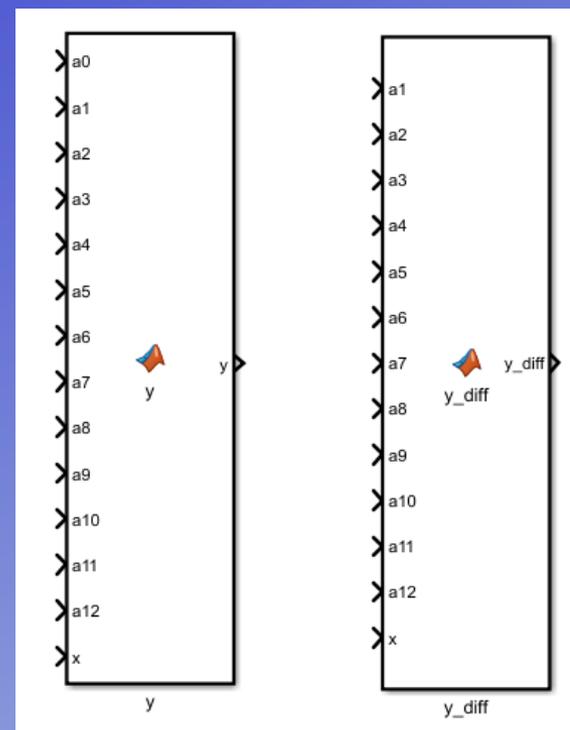
実験データ (・) とフィッティングされた多項式 (—) を重ね合わせたグラフで確認する。

# OCV-SOCの多項式の数式処理(微分など)

Symbolic Math Toolbox

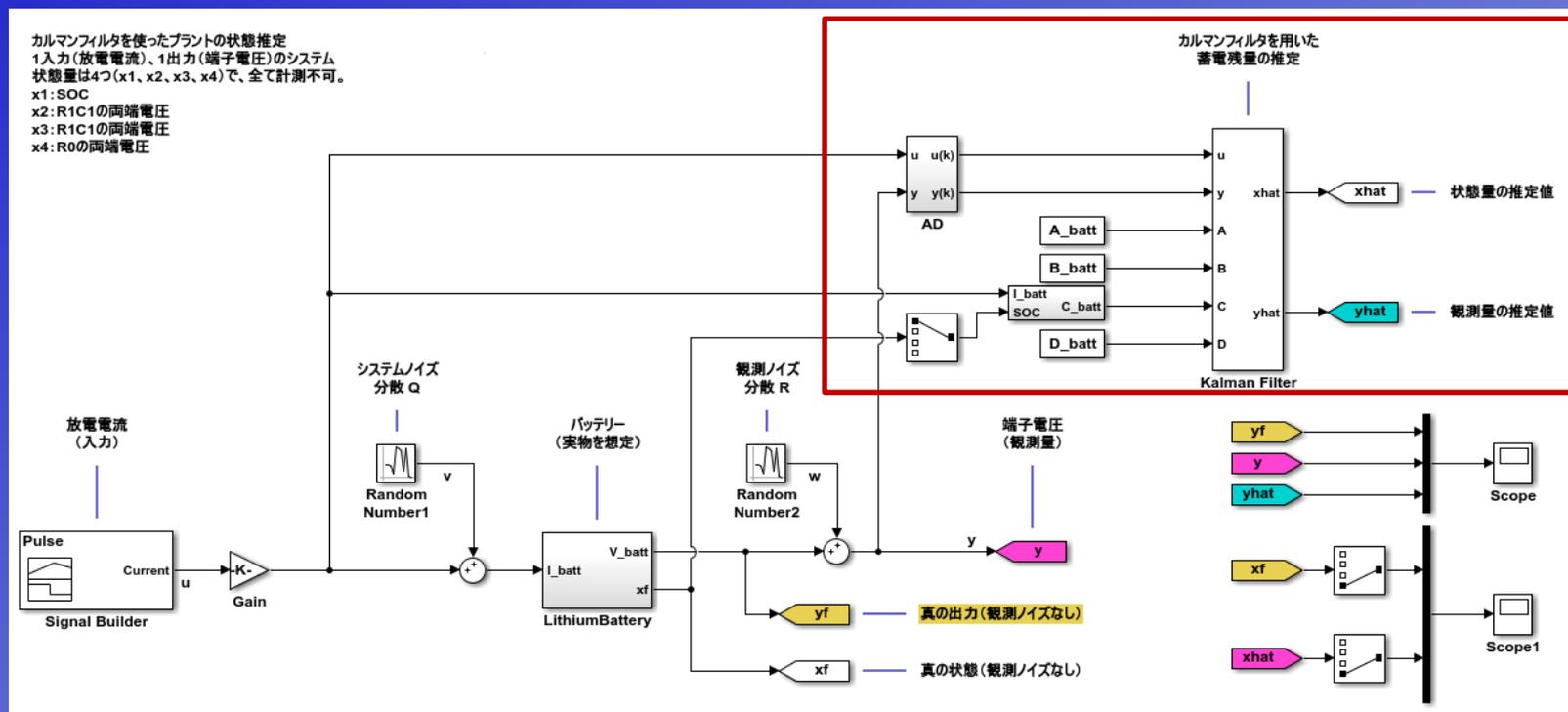
MATLAB (数値解析) で、Symbolic Math Toolboxを使うことで数式処理をすることも可能。  
また、数式処理により得られた数式を、Simulink (ブロック線図) で使うためにMATLAB Functionブロックに自動変換することも可能。

```
1 %% OCV-SOC特性 (実験データ) の多項式を微分した数式を算出
2 % 出力: y = OCV, 変数: x = SOC
3 % y = a0 + a1*x + a2*x^2 + a3*x^3 + a4*x^4 + a5*x^5 ...
4 %       + a6*x^6 + a7*x^7 + a8*x^8 + a9*x^9 + a10*x^10 ...
5 %       + a11*x^11 + a12*x^12;
6
7 %% 係数、変数
8 syms a0 a1 a2 a3 a4 a5 a6 a7 a8 a9 a10 a11 a12
9 syms x
10
11 %% 多項式
12 y = a0 + a1*x + a2*x^2 + a3*x^3 + a4*x^4 + a5*x^5 ...
13     + a6*x^6 + a7*x^7 + a8*x^8 + a9*x^9 + a10*x^10 ...
14     + a11*x^11 + a12*x^12;
15
16 %% 多項式の微分
17 y_diff = diff(y);
18
19 %% MATLAB Functionブロックにインポート
20 new_system('my_system');
21 open_system('my_system');
22 matlabFunctionBlock('my_system/y', y);
23 matlabFunctionBlock('my_system/y_diff', y_diff);
```



# カルマンフィルタを用いた蓄電残量の推定

この部分は、ExcelやMATLABで作られたロジックがあれば、そのロジックをSimulink環境にMATLAB Functionブロック等を使い、組み込むことも可能。



# 熱ライブラリ (Thermal)

Simscape

## 要素



Convective Heat Transfer

熱伝達



Conductive Heat Transfer

熱伝導



Radiative Heat Transfer

熱放射



Thermal Mass

熱容量



Perfect Insulator

断熱



Thermal Reference

温度の基準

## センサ



Ideal Heat Flow Sensor

熱流センサ



Ideal Temperature Sensor

温度センサ

## 信号源



Ideal Heat Flow Source

熱流源



Ideal Temperature Source

温度源

# 推奨ツール構成

- 基本環境
  - MATLAB、Simulink、Stateflow
- 曲線・曲面近似
  - Curve Fitting Toolbox
- 数式処理
  - Symbolic Math Toolbox
- 物理モデリング(電気、熱)
  - Simscape
- パラメータ最適化
  - Simulink Design Optimization、Optimization Toolbox
- カルマンフィルタ、逐次最小二乗法
  - Control System Toolbox、Simulink Control Design
  - System Identification Toolbox



Accelerating the pace of engineering and science

© 2017 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.