確率・統計処理 & 真値推定！

カルマン・フィルタ入門

リニア・テック　別府 伸耕

linear tec : Nobuyasu Beppu

概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

「カルマン・フィルタ」に関係する理論のピラミッド

ソフトウェア開発
・C言語によるカルマン．フィルタの実装
・C言語による行列計算ライブラリ作成
・PythonによるRiccati方程式Solver作成
・Pythonによる倒立振子シミュレータ作成

ハードウェア開発
・車体のモデル化
・DCモータのパラメータ同定
・センサの特性確認
・マイコンの各種ペリフェラル使用法

現代制御理論
・状態方程式
・可制御性と可観測性
・システムの安定化
・オブザーバ
・最適制御

ディジタル信号処理
・z変換
・ディジタル・フィルタ
・カルマン・フィルタ

力学
力学
・ニュートンの運動方程式
・オイラーの回転運動方程式
・回転座標系におけるコリオリ力

解析力学
・ラグランジュの運動方程式

電気・電子回路理論
受動回路
・LCR回路の基本
・DCモータのモデル化
・RCフィルタの設計

能動回路
・バイポーラ・トランジスタの基本
・トランジスタ回路の設計

数学
微分・積分
・初等関数
・微分
・1変数の積分
・テイラー展開
・偏微分
・2変数の積分

線形代数
・行列の計算
・逆行列
・行列の階数
・固有値
・固有ベクトル
・2次形式

確率・統計
・場合の数
・ベイズの定理
・期待値と分散
・ガウス分布
・中心極限定理
・推定と検定

フーリエ解析
・フーリエ変換
・ラプラス変換

複素関数論
・複素平面
・オイラーの公式

スケジュール

1日目			
時刻	内容	ページ数	補助資料ページ数
10:00 − 11:00	数学：初等関数	46	37
11:00 − 11:10	休憩	−	
11:10 − 12:40	数学：微分・積分	45	75
12:40 − 13:40	昼休み	−	
13:40 − 14:40	数学：線形代数 （最初～行列式）	37	48
14:40 − 14:50	休憩	−	
14:50 − 15:30	数学：線形代数 （連立一次方程式～最後）	18	−
15:30 − 15:40	休憩	−	
15:40 − 17:00	力学	42	67

2日目			
時刻	内容	ページ数	補助資料ページ数
10:00 − 11:00	現代制御理論 （最初～可制御性と可観測性）	30	23
11:00 − 11:10	休憩	−	
11:10 − 12:00	現代制御理論 （安定性～最後）	19	−
12:00 − 12:30	確率・統計（最初～記述統計）	18	29
12:30 − 13:40	昼休み	−	
13:40 − 14:25	確率・統計（確率の基礎～確率変数と確率分布）	27	−
14:25 − 14:35	休憩	−	
14:35 − 15:30	確率・統計 （2次元の確率分布～最後）	29	−
15:30 − 15:40	休憩	−	
15:40 − 17:00	カルマン・フィルタ	46	34

合計ページ数	357	313
合計時間（分）	720	
時間（分）/ページ	2.0	

概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

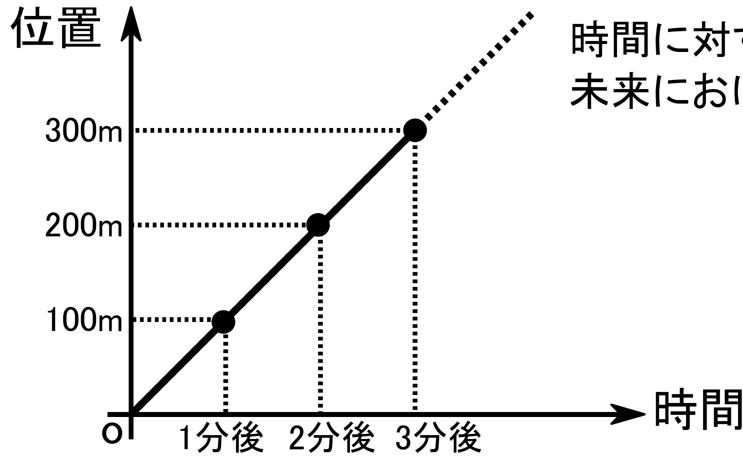
初等関数

名称	代表的な形	主な用途
多項式関数	$a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$	テイラー展開, 伝達関数
有理関数	$\dfrac{a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0}{b_m x^m + b_{m-1} x^{m-1} + \cdots + b_1 x + b_0}$	伝達関数, インピーダンス関数
三角関数	$\sin(x), \cos(x), \tan(x)$	信号波形の表現
逆三角関数	$\sin^{-1}(x), \cos^{-1}(x), \tan^{-1}(x)$	角度の算出
指数関数	a^x, e^x	複素正弦波
対数関数	$\log_a(x), \log(x)$	積和変換 デシベル表示

これらの関数を理解しておけば，エンジニアリングでは困らない．

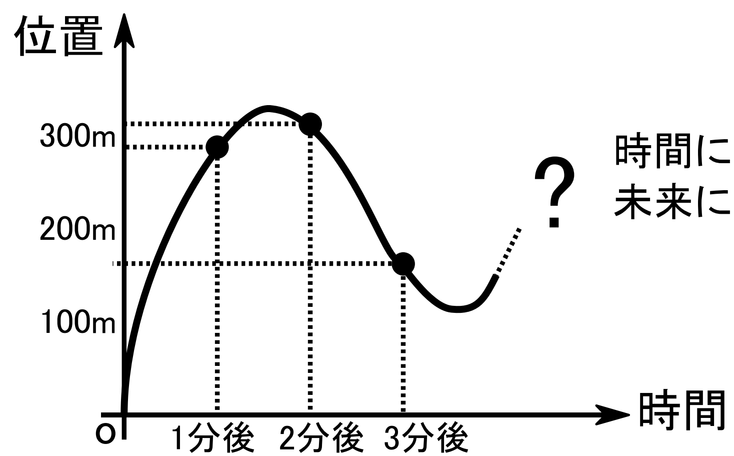
「変化率」に着目する

【変化率が一定】



時間に対する変化率が一定.
未来における位置を予想しやすい.

【変化率が一定ではない】



時間に対する変化率が変動する.
未来における位置を予想しづらい.

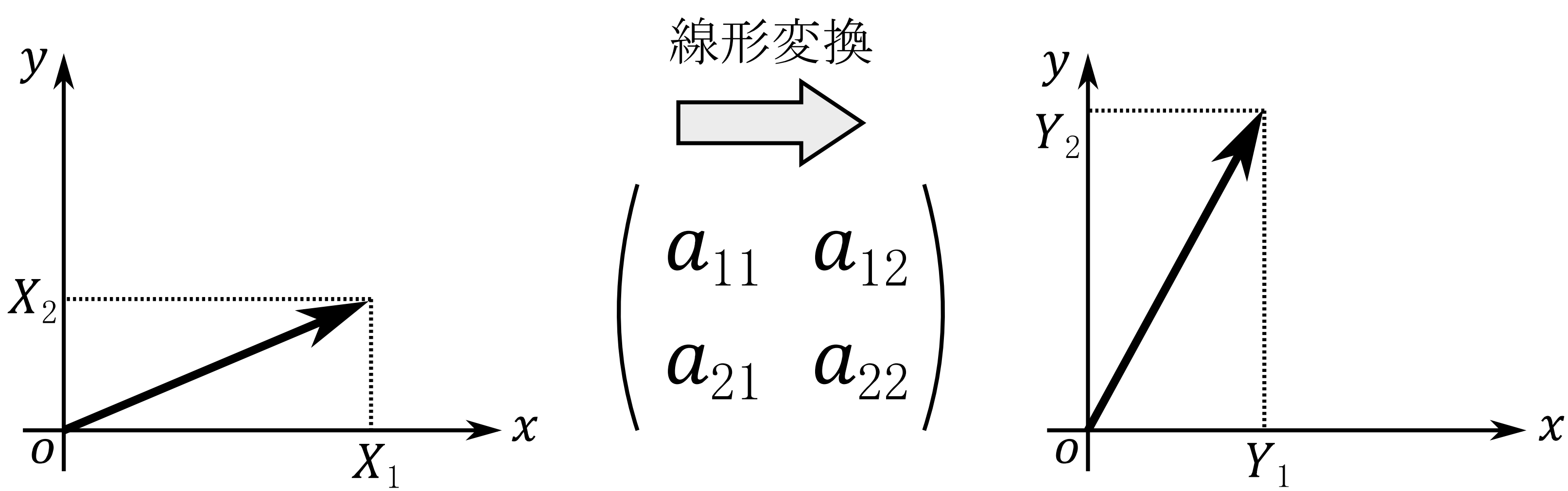
人間は，変化率が一定ならば未来を予測できる.
変化率が一定であるものを「線形」という.

線形変換の具体例（1）

ベクトル "$\boldsymbol{X} = (X_1, X_2)$" を「線形変換」して，新しいベクトル "$\boldsymbol{Y} = (Y_1, Y_2)$" を作る.

$$\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$$

上式の線形変換は，2次元ベクトルを（原点を変えずに）変形する操作 だと見なせる.
これは，一般的な「図形の変形操作」に応用できる.



概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

ニュートンの運動方程式：Newton’s equation of motion

質量が “m (kg)” の物体に “$\boldsymbol{F}$ (N)” の力が印加されたときの運動は，
次の「ニュートンの運動方程式」によって表される．

$$\boxed{m\frac{d^2\boldsymbol{r}(t)}{dt^2} = \boldsymbol{F} \quad \text{あるいは，} \quad m\ddot{\boldsymbol{r}} = \boldsymbol{F}}$$

上式は，次のように解釈できる．

$$\frac{d^2\boldsymbol{r}(t)}{dt^2} = \frac{\boldsymbol{F}}{m}$$

「力」：
加速度を変化させる原因．

「質量」（慣性質量）：
加速度の変化しにくさ．

「加速度」：
物体の未来の挙動に関する情報を持つ．

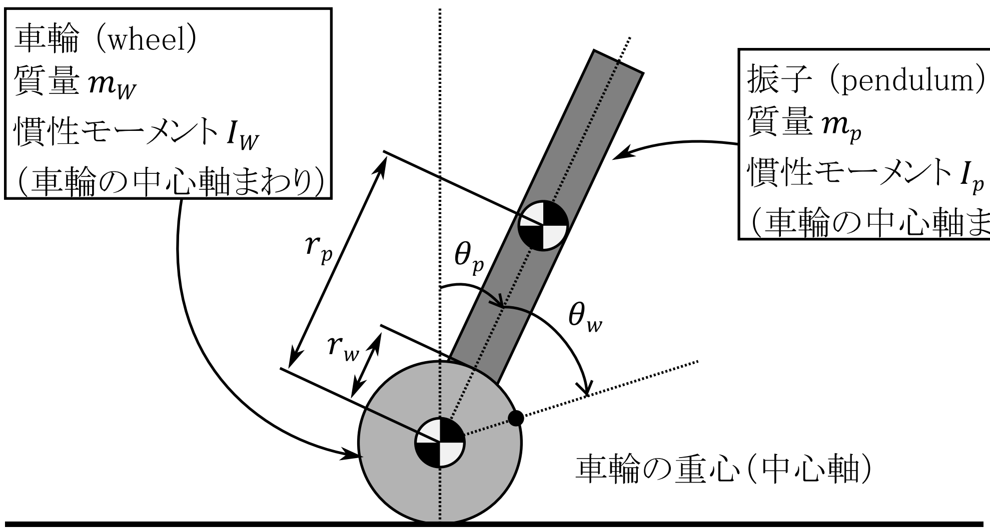
ニュートンの運動方程式は，x，y，zの3方向の式をまとめたものになっている．

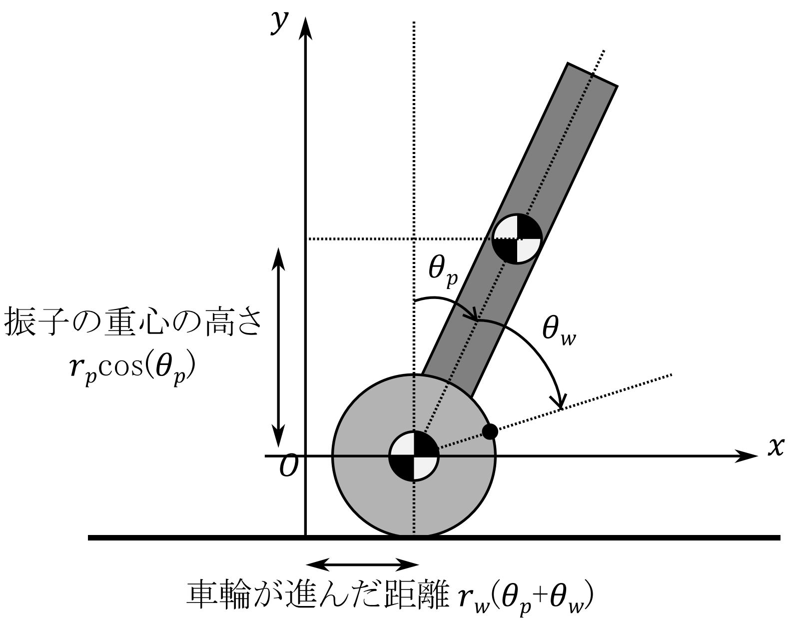
$$\begin{pmatrix} m\dfrac{d^2r_x(t)}{dt^2} \\ m\dfrac{d^2r_y(t)}{dt^2} \\ m\dfrac{d^2r_z(t)}{dt^2} \end{pmatrix} = \begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix}$$

倒立振子の構造の簡略化

「車輪型倒立振子」を，次図のようにモデル化する.
ここでは単純に，「車輪」部分と「振子」部分だけから構成されるものと考える.
変数は次の2つとする.

- “θ_p” 振子部分の垂直方向からの傾き角.
- “θ_w” 車輪の，振子部分に対する回転角.





概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

状態方程式：state equation

一般のn階線形微分方程式で表された線形システムの挙動を表すために，
次のように関数 “$x(t), x^{(1)}(t), \cdots, x^{(n-1)}(t)$” を束ねた「$n$次元ベクトル “$\boldsymbol{x}(t)$”」を考える.

$$\boldsymbol{x}(t) = \begin{pmatrix} x(t) \\ x^{(1)}(t) \\ \vdots \\ x^{(n-1)}(t) \end{pmatrix}$$

n階の線形微分方程式は，係数行列 “A” および “B” と，
時間tの関数をまとめたn次元ベクトル “ $\boldsymbol{u}(t)$” を使って，「1階微分方程式のように」書ける.

$$\boxed{\dot{\boldsymbol{x}}(t) = A\boldsymbol{x}(t) + B\boldsymbol{u}(t)}$$

上式は線形システムを表現する微分方程式の一般形であり「状態方程式」と呼ばれる.
各項の意味は次のとおり.

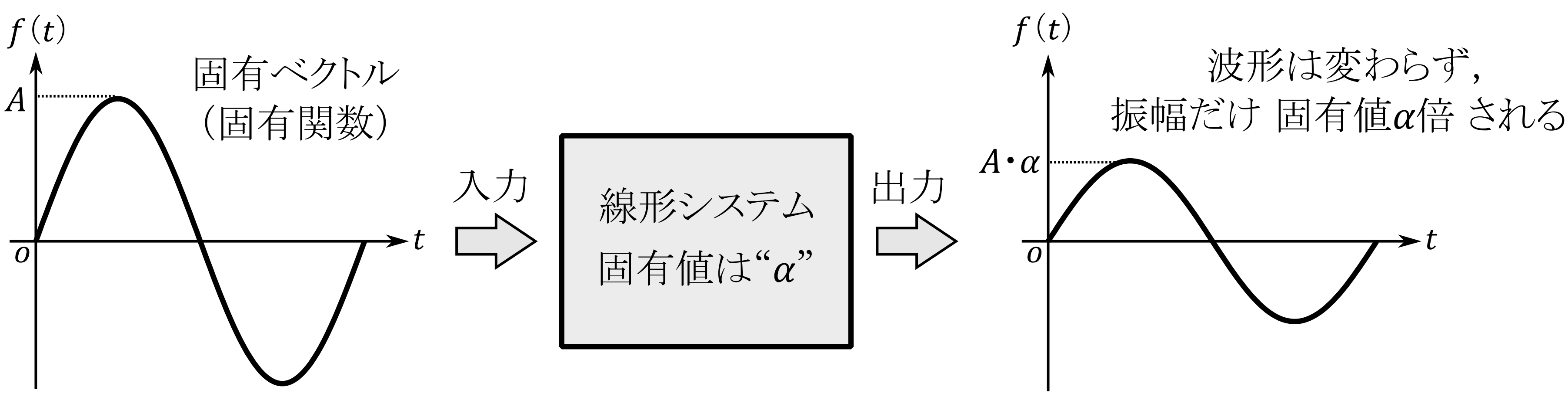
- ベクトル $\boldsymbol{x}(t)$:「状態ベクトル」(state vector). システムの状態を表す.
- ベクトル $\boldsymbol{u}(t)$:「入力ベクトル」(input vector). システムに対する入力を表す.
- 行列 A :「状態行列」(state matrix). システムの挙動を表す.
- 行列 B :「入力行列」(input matrix). 入力信号$\boldsymbol{u}(t)$がシステムに及ぼす影響を表す.

システムの安定判別法 （離散時間システム）

入力を常に “$\boldsymbol{u}_k = \mathbf{0}$” とした離散時間システム “$\boldsymbol{x}_{k+1} = A_d \boldsymbol{x}_k$” の安定性について，
次の定理が成り立つ．

> 離散時間システムが安定であるための必要十分条件は，
> 行列 “A_d” のすべての固有値の絶対値が “1” より小さいことである．

※ 次図は「線形システム」と「固有値」の直感的なイメージ．



“$\boldsymbol{x}_{k+1} = A_d \boldsymbol{x}_k$” という関係式は，前時刻の出力を入力に入れ続ける様子を意味する．

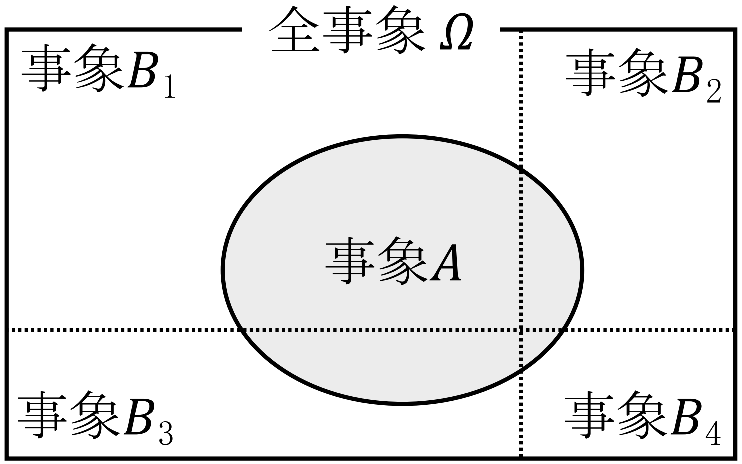
概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

ベイズの定理: Bayes' theorem

次図のように全事象Ωが事象B_1, B_2, B_3, B_4に分割されており，互いに「排反」だとする.
このとき，次式が成り立つ.

$$\Omega = B_1 \cup B_2 \cup B_3 \cup B_4$$



ここで，別の「事象A」を考える.
事象Aが起こった前提で事象B_i（$1 \leq i \leq 4$）が起こる条件付き確率"$P(B_i|A)$"は次のとおり.
ただし，「乗法定理」より"$P(B_i \cap A) = P(B_i|A) \cdot P(A) = P(A|B_i) \cdot P(B_i)$"が成り立つ.

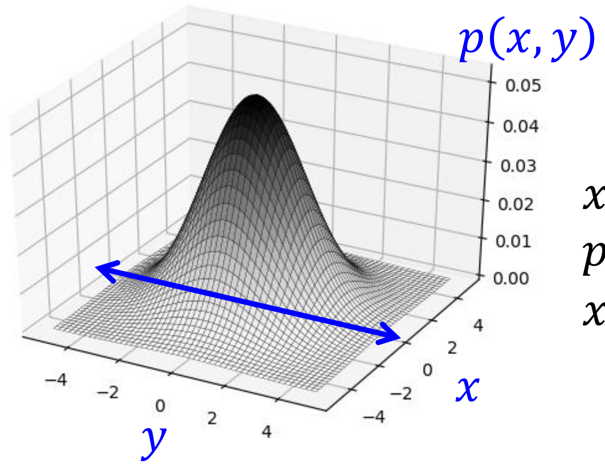
この式を「ベイズの定理」という.（実質，条件付き確率の定義にすぎない.）

$$P(B_i|A) = \frac{P(B_i \cap A)}{P(A)} = \frac{P(A|B_i) \cdot P(B_i)}{P(A)}$$

周辺確率密度関数: marginal probability density function

確率変数 XおよびYに関する同時確率密度関数 “$p_{X,Y}(x,y)$” を考える.
$p_{X,Y}(x,y)$を確率変数Yについて積分すると,
確率変数 X に関する単独の確率密度関数 “$p_X(x)$” が得られる.
この関数 $p_X(x)$ のことを, 確率変数Xに関する「周辺確率密度関数」という.

$$p_X(x) = \int_{-\infty}^{\infty} p_{X,Y}(x,y)dy$$



$x = 0$の位置で,
$p_{X,Y}(x,y)$の値をy方向にすべて足し合わせれば,
$x = 0$をとる確率“$p_X(x=0)$”が得られる.

同様に, 同時確率密度関数 $p_{X,Y}(x,y)$を確率変数Xで積分すれば,
確率変数Yに関する周辺確率密度関数 “$p_Y(y)$” が得られる.

$$p_Y(y) = \int_{-\infty}^{\infty} p_{X,Y}(x,y)dx$$

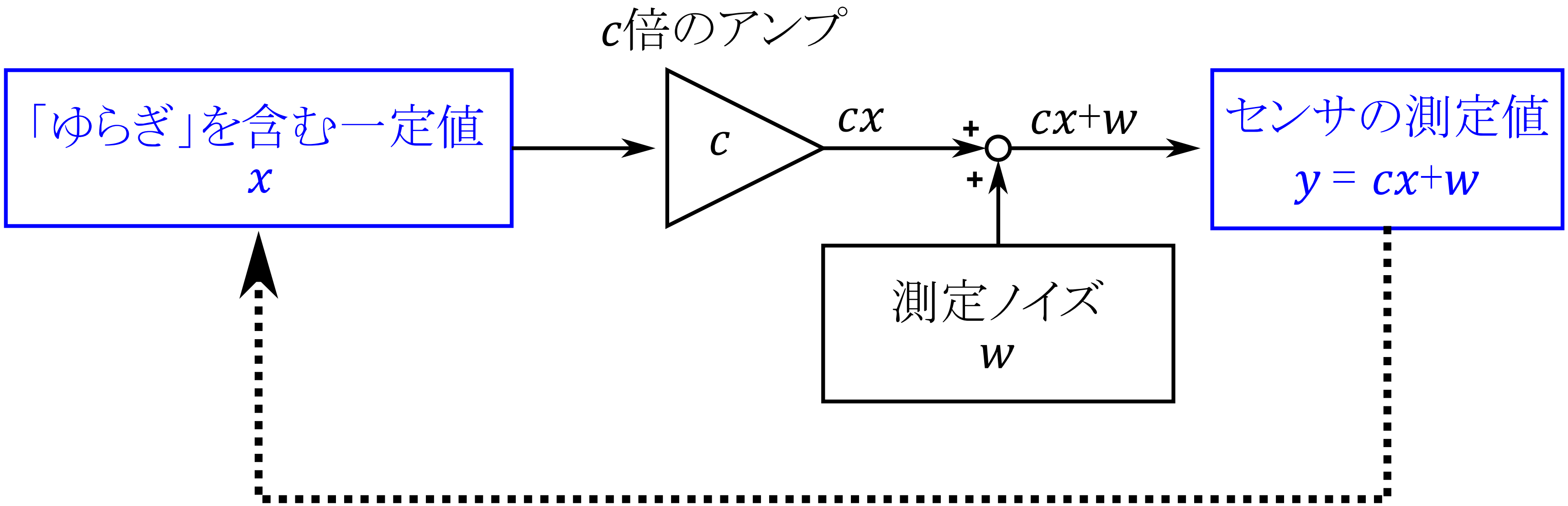
概要

1. 数学
2. 力学
3. 現代制御理論
4. 確率・統計
5. カルマン・フィルタ

測定系のモデル

何らかの，一定値を保つように制御されているパラメータ“x”を考える.
このxは，時々刻々と変化している（ゆらいでいる）とする.
この「真値x」を，雑音を含む測定系で測定した値を「測定値y」とする.

これ以降，「測定値y」をもとにして「真値x」の推定値を算出する方法を考える.



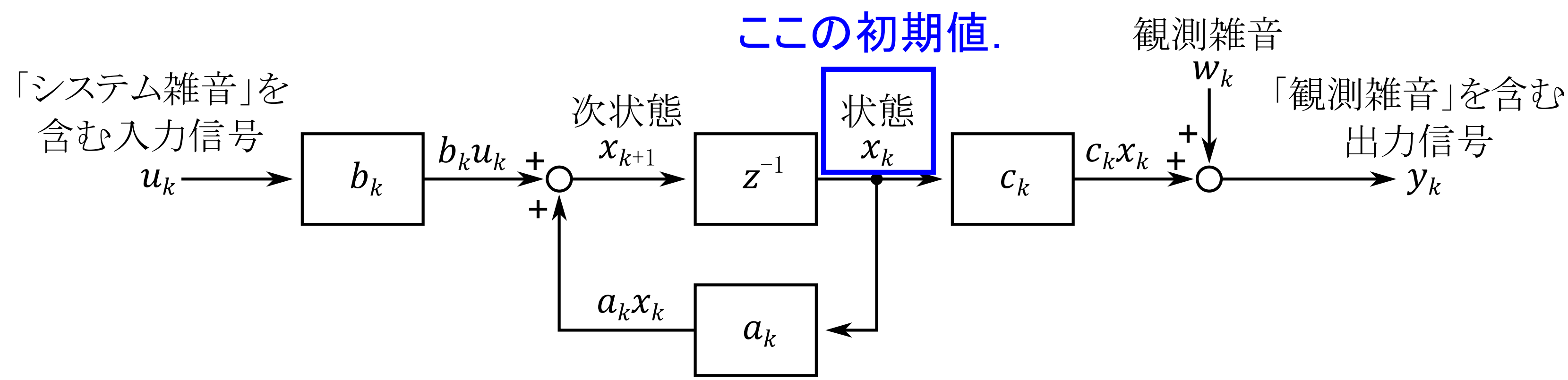
センサの測定値yから，もとの値xを推定したい.

システムの不確定要素（1）

準備として，システムにおける不確定要素を「確率変数」として表しておく.

システムの初期状態“x_0”は，期待値が“$\bar{x}_0$”，分散が“$\sigma_{x_0}^2$”の正規分布に従うと仮定する.
初期状態x_0は設計者が決めるものだが，そこには何らかの誤差が生じうる.
よって，設計者が決めた設計値を「期待値$\bar{x}_0$」とし，その確かさの程度を「分散$\sigma_{x_0}^2$」で表す.

$$p_{X_0}(x_0) = \frac{1}{\sqrt{2\pi\sigma_{x_0}^2}} e^{-\frac{(x_0-\bar{x}_0)^2}{2\sigma_{x_0}^2}}$$



カルマン・フィルタの処理の流れ

時間の流れ

時刻 0	測定値の取得	測定値 y_0
	測定値にもとづく補正	現時点の推定値 $\hat{x}_0$　分散 $\sigma_0{}^2$
	状態方程式にもとづく予測	次時点の推定値 $\tilde{x}_1$　分散 $\sigma'_1{}^2$
時刻 1	測定値の取得	測定値 y_1
	測定値にもとづく補正	現時点の推定値 $\hat{x}_1$　分散 $\sigma_1{}^2$
	状態方程式にもとづく予測	次時点の推定値 $\tilde{x}_2$　分散 $\sigma'_2{}^2$
時刻 2	測定値の取得	測定値 y_2
	測定値にもとづく補正	現時点の推定値 $\hat{x}_2$　分散 $\sigma_2{}^2$
	状態方程式にもとづく予測	次時点の推定値 $\tilde{x}_3$　分散 $\sigma'_3{}^2$

$$\sigma_0^2 = \left(\sigma_{x_0}^{-2} + c_0^2 \sigma_{w_0}^{-2}\right)^{-1}$$

$$\hat{x}_0 = \bar{x}_0 + c_0 \sigma_{w_0}^{-2} \sigma_0^2 (y_0 - c_0 \bar{x}_0 - \bar{w}_0)$$

$$\sigma_1'^2 = a_0^2 \sigma_0^2 + b_0^2 \sigma_{u_0}^2$$

$$\tilde{x}_1 = a_0 \hat{x}_0 + b_0 \bar{u}_0$$

$$\sigma_1^2 = \left(\sigma_1'^{-2} + c_1^2 \sigma_{w_1}^{-2}\right)^{-1}$$

$$\hat{x}_1 = \tilde{x}_1 + c_1 \sigma_{w_1}^{-2} \sigma_1^2 (y_1 - c_1 \tilde{x}_1 - \bar{w}_1)$$

$$\sigma_2'^2 = a_1^2 \sigma_1^2 + b_1^2 \sigma_{u_1}^2$$

$$\tilde{x}_2 = a_1 \hat{x}_1 + b_1 \bar{u}_1$$

直前の1ステップ分のデータだけを
次時点の推定に利用する．
よって，メモリを節約できる．

※前時刻までの情報は
次時点の予測値に引き継がれる．