認識行動システム論

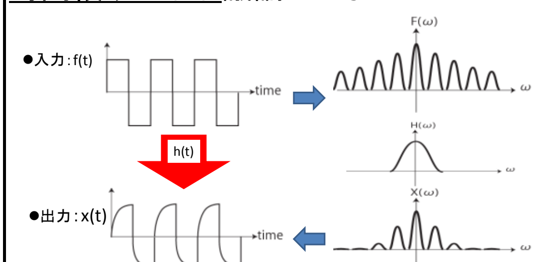
第8回
梶本裕之

日程

12/17 信号処理(相互相関・自己相関)
01/07 古典制御の基礎
01/14 ロボティクス
01/21 画像処理
01/28 期末テスト(授業時間中)

復習：
信号処理の基礎

(復習)周波数領域ではなく, 時間領域のまま議論できないか？



X(ω)＝H(ω)×F(ω)：周波数領域で美しいのは分った.
時間的な現象として何が起きているのか分からない.

(復習)式で考えよう

フーリエ変換

$$F(\omega)=\int_{-\infty}^{\infty} f(t)\exp(-j\omega t)dt$$

逆フーリエ変換

$$f(t)=\int_{-\infty}^{\infty} F(\omega)\exp(j\omega t)d\omega$$

$\boxed{X(\omega)=H(\omega)\times F(\omega)}$

両辺を逆フーリエ変換すれば時間領域の信号に戻る.

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} H(\omega)F(\omega)\exp(j\omega t)d\omega \\ &= \int_{-\infty}^{\infty} H(\omega)\left(\int_{-\infty}^{\infty} f(\tau)\exp(-j\omega\tau)d\tau\right)\exp(j\omega t)d\omega \\ &= \int_{-\infty}^{\infty} f(\tau)\left(\int_{-\infty}^{\infty} H(\omega)\exp(j\omega(t-\tau))d\omega\right)d\tau \\ &= \int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau \end{aligned}$$

$$\int_{-\infty}^{\infty} H(\omega)\exp(j\omega(t-\tau))d\omega = \int_{-\infty}^{\infty}\left(H(\omega)\exp(-j\omega\tau)\right)\exp(j\omega t)d\omega$$

$$\begin{aligned} \int_{-\infty}^{\infty} h(t-\tau)\exp(-j\omega t))dt &= \int_{-\infty}^{\infty} h(t')\exp(-j\omega(t'+\tau))dt' \\ &= \exp(-j\omega\tau)\int_{-\infty}^{\infty} h(t')\exp(-j\omega t'))dt' \\ &= H(\omega)\exp(-j\omega\tau) \end{aligned}$$

逆順の計算もしておく(ふつうはこちら)

$$x(t)=\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau$$

フーリエ変換

$$F(\omega)=\int_{-\infty}^{\infty} f(t)\exp(-j\omega t)dt$$

逆フーリエ変換

$$f(t)=\int_{-\infty}^{\infty} F(\omega)\exp(j\omega t)d\omega$$

両辺をフーリエ変換.

$$\begin{aligned} X(\omega) &= \int_{-\infty}^{\infty}\exp(-j\omega t)dt\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty}\exp(-j\omega(\tau+(t-\tau))dt\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty}\exp(-j\omega\tau)\cdot\exp(-j\omega(t-\tau))dt\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau \\ &= \int_{-\infty}^{\infty} f(\tau)\exp(-j\omega\tau)d\tau\int_{-\infty}^{\infty} h(t-\tau)\exp(-j\omega(t-\tau))dt \\ &= \int_{-\infty}^{\infty} f(\tau)\exp(-j\omega\tau)d\tau\int_{-\infty}^{\infty} h(t')\exp(-j\omega t')dt' \\ &= F(\omega)H(\omega) \end{aligned}$$

(復習)コンボリューション定理

$$X(\omega)=F(\omega)H(\omega)=H(\omega)F(\omega)$$

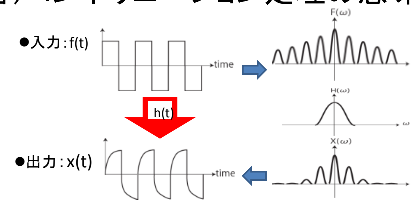
フーリエ逆変換　フーリエ変換

$$x(t)=\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau=\int_{-\infty}^{\infty} h(\tau)f(t-\tau)d\tau$$

簡略化のため次のようにも表記される

$$x(t)=f(t)*h(t)=h(t)*f(t)$$

(復習)コンボリューション定理の意味(1)

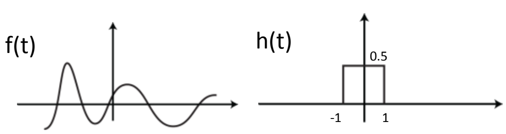


$$x(t)=\int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau \qquad X(\omega)=F(\omega)H(\omega)$$

•h(t)のフーリエ変換がH(ω)であるとする.
•周波数領域でフィルタH(ω)をかけることは,
時間領域では，入力信号x(t)に対する関数h(t)の畳み込み積分(コンボリューション)として表現される.

(復習)コンボリューション定理の意味(2)

$$x(t)=\int_{-\infty}^{\infty} h(\tau)f(t-\tau)d\tau$$



例えば，h(t)=0.5　(-1<t<1)なら,

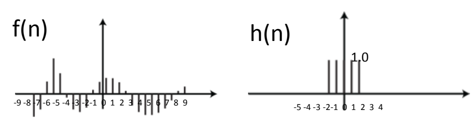
$$x(t)=\int_{-1}^{1} 0.5f(t-\tau)d\tau$$

これは，f(t)を平均化していくフィルタ

(復習)離散化による理解

$$x(t)=\int_{-\infty}^{\infty} h(\tau)f(t-\tau)d\tau \Rightarrow x(n)=\sum_{i=-\infty}^{\infty} h(i)f(n-i)$$

x(n)=...+ h(-4)f(n+4) + h(-3)f(n+3) +...+ h(3)f(n-3) + h(4)f(n-4)+...



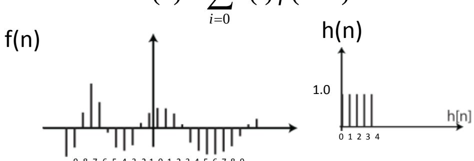
h(n)が，n=-2～2の間だけ1の場合,

x(n)=f(n+2) + f(n+1) + f(n) + f(n-1) + f(n-2)

●この場合，出力xは，入力fの「平均化」になっている.
●つまりこの場合，hは平滑化フィルタである.

(復習)FIRフィルタ

$$x(n)=\sum_{i=0}^{\infty} h(i)f(n-i)$$



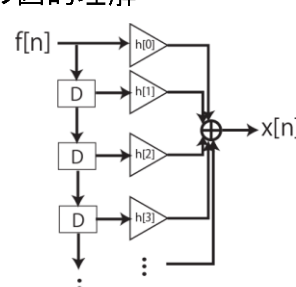
i=0から始める：未来のデータ が使えないことを意味する.
この例は，元データf(n)を，4個平均して出力する.

•未来のデータが使えない例：リアルタイム制御
•先のデータが使える例：画像処理

(復習) FIRフィルタの図的理解

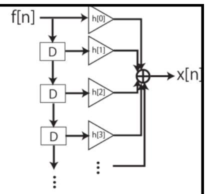
$$x(n)=\sum_{i=0}^{\infty} h(i)f(n-i)$$

D:Delay，遅延器，メモリ.
h[n]：増幅器



FIR＝Finite Impulse Response
個々のインパルス応答を有限個足し合わせたもの.

(復習) FIRフィルタと行列



$$x(n)=\sum_{i=0}^{\infty}h(i)\,f(n-i)$$

フィルタが3つのメモリを持つ場合

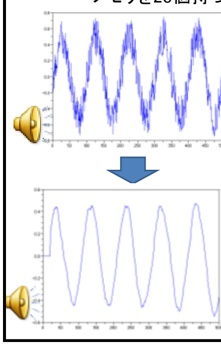
$$x(n)=h(0)\,f(n)+h(1)\,f(n-1)+h(2)\,f(n-2)$$

$x(0)=h(0)f(0)$
$x(1)=h(0)f(1)+h(1)f(0)$
$x(2)=h(0)f(2)+h(1)f(1)+h(2)f(0)$
$x(3)=h(0)f(3)+h(1)f(2)+h(2)f(1)$
$x(4)=h(0)f(4)+h(1)f(3)+h(2)f(2)$
$x(5)=h(0)f(5)+h(1)f(4)+h(2)f(3)$

$$\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\\x(4)\end{bmatrix}=\begin{bmatrix}h(0)&0&0&0&0\\h(1)&h(0)&0&0&0\\h(2)&h(1)&h(0)&0&0\\0&h(2)&h(1)&h(0)&0\\0&0&h(2)&h(1)&h(0)\end{bmatrix}\begin{bmatrix}f(0)\\f(1)\\f(2)\\f(3)\\f(4)\end{bmatrix}$$

(復習)平滑化フィルタの実例

メモリを20個持ったFIRフィルタによって平滑化



Scilabコード例

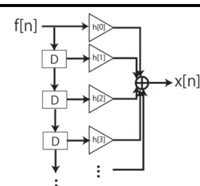
```
time = [0:0.01:100];
//振幅0.5の正弦波に最大振幅0.5のノイズが混入した信号
wave=0.5*sin(time*2*%pi) + 0.5*(rand(time)-0.5);

out=zeros(wave);

//20個を平均する.
for n=20:length(wave),
  for i=0:19,
    out(n)=out(n)+wave(n-i)/20;
  end
end

playsnd(out);
savewave('wave.wav',out);
plot(out(1:500));
```

(復習)エコー



エコー＝時間遅れ信号の重畳.
これはFIRフィルタで実装できる.

Scilabコード例

```
wave = loadwave('aiueo.wav');

out=zeros(wave);

//エコー(1000ステップ前の信号を重畳)
for n=1000:length(wave),
   out(n)=wave(n)+0.9*wave(n-999);
end

playsnd(out,11000);      //11kHzサンプリングで再生
savewave('wave.wav',out,[11000]);
```

原音

1000ステップ前の信号を重畳

1000ステップ前＋2000ステップ前の信号を重畳

沢山重畳

エコーは害

原音

沢山重畳

ゴースト現象

相互相関関数
自己相関関数

エコーキャンセル（テレビだとゴーストリダクション）

FIRフィルタによってエコーの影響を低減することができる．

考え方：
（1）エコー成分のモデルを推定
out(n)=wave(n)+0.5*wave(n-100);
「100ステップ前の信号が0.5のゲインで重畳されている!」

（2）そのモデルに基づき，<u>エコー成分と逆のゲインをかける</u>
out(n)-0.5*out(n-100)
=wave(n)+wave(n-100)-0.5*(wave(n-100)+0.5*wave(n-200))
=wave(n)+0.25*wave(n-200)
⇒エコーが半分に低減！
（3）当然もっと工夫すれば．．．(もっとメモリがあれば)
out(n)-0.5*out(n-100)- 0.25*wave(n-200)
=...=wave(n)+0.125*wave(n-300)
⇒無限にメモリがあれば完璧に消せる．

エコーキャンセル＝エコーの逆行列

$$x(n)=\sum_{i=0}^{\infty} h(i)\,f(n-i)$$

エコーもエコーキャンセルも
FIRフィルタによって表わされる．

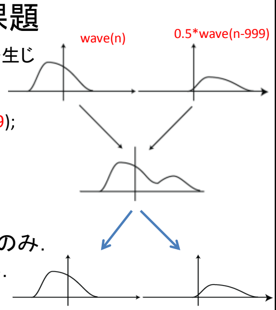
$$\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \end{bmatrix} = \begin{bmatrix} h(0) & 0 & 0 & 0 & 0 \\ h(1) & h(0) & 0 & 0 & 0 \\ h(2) & h(1) & h(0) & 0 & 0 \\ 0 & h(2) & h(1) & h(0) & 0 \\ 0 & 0 & h(2) & h(1) & h(0) \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \\ f(4) \end{bmatrix}$$

結局，エコーキャンセルとは，
「『エコーの効果を表す行列』の逆行列を掛けて元に戻す」
操作に他ならない．
（実際はそう簡単ではないが．．．）

エコーキャンセルの課題

エコー成分が，どれだけ時間遅れを生じてやってくるのかのモデルを推定

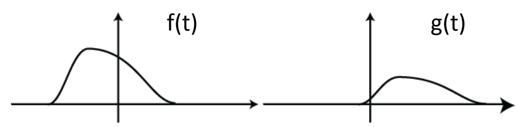
out(n)=wave(n)+0.5*wave(n-999);



＜問題＞
観測できるのは，
エコーの「結果」としてのout(n)のみ．
元の信号はわかっていない．

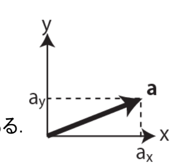
この信号からどのように，
モデルを推定するのか？

より簡単な問題から考えよう



二つの信号が，
●時間的にどれだけずれているのか
●時間のずれを無視したらどれだけ似ているのか
を測定したい．

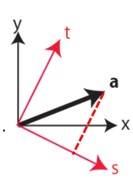
(復習)ベクトル空間と内積



ベクトル **a**=[a_x, a_y] のx成分は？．．．．． a_x

これはベクトル**a**とベクトル **x**=[1,0]との内積である．

a•**x** = [a_x, a_y]•[1,0] = a_x



回転した座標軸，s, tを考える．
ベクトル **a**=[a_x, a_y] の，s 成分は？

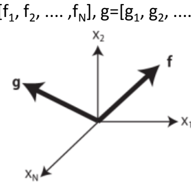
これはベクトル**a**とベクトル **s**=[s_x, s_y]との内積である．

a•**s** = [a_x, a_y]• [s_x, s_y]

内積は，あるベクトルが別のベクトルの成分をどれだけ持つかを表す

（復習）N次元空間では

N次元空間で，二つのベクトル
f=[f_1, f_2, ... ,f_N], g=[g_1, g_2, ... ,g_N] を考える．



内積 **f**•**g**は，ベクトルfの，g軸成分(または逆)を表す．

$$= [f_1 \quad f_2 \quad \cdots \quad f_N] \bullet [g_1 \quad g_2 \quad \cdots \quad g_N]$$

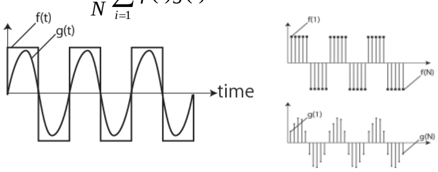
$$= \sum_{i=1}^{N} f_i g_i$$

(復習)波形*f*に波形*g*はどれだけ含まれるか

波形*f*中の，波形*g*の成分

$$= \frac{1}{T}\int_0^T f(t)g(t)dt \quad \text{(連続関数)}$$

$$= \frac{1}{N}\sum_{i=1}^{N} f(i)g(i) \quad \text{(離散化して考えた場合)}$$



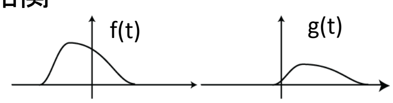
これは二つの波をベクトルと考えた時の内積に他ならない
※内積を連続関数に対して定義

相互相関

＜問題＞
二つの信号が，
●時間的にどれだけずれているのか
●時間のずれを無視したらどれだけ似ているのか
を測定したい．

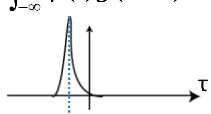
内積を思い出せば，
次の手順で測定すればよいことがわかる
●g(t)をτだけずらしてみる ⇒
●f(t)との内積を取ってみる ⇒
●τを変化させていく．

相互相関



$R_{fg}(\tau)$：二つの関数f(t)，g(t)の，相互相関関数

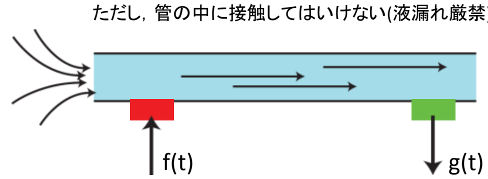
$$R_{fg}(\tau) = \int_{-\infty}^{\infty} f(t)g(t+\tau)dt$$



$R_{fg}(\tau)$が最大の値をとるτ ＝ 元の関数f(t)とg(t)のズレ
（ただし直流成分を取り除いた後）

相互相関の応用：速度計測

管内の流速を正確に測定したい
ただし，管の中に接触してはいけない(液漏れ厳禁)

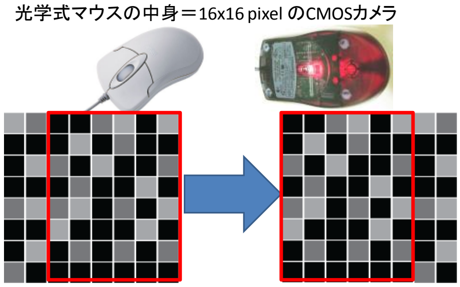


上流に熱源を置き，f(t)でランダムに変動させる．
下流で温度を測定する．g(t)

f(t)とg(t)の相互相関関数が最大となる時間差τが，
水流によって熱が移動するのに要する時間である．

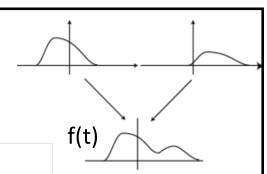
相互相関の応用：速度計測

光学式マウスの中身＝16x16 pixel のCMOSカメラ



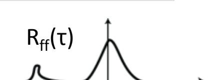
二つの画像(＝2次元関数)同士の相互相関を取ることで
移動量を計測する．自動車の速度計測等にも利用．

自己相関



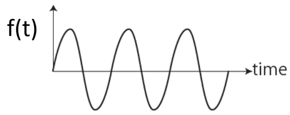
二つの関数f(t)，g(t)の代わりに，
ひとつの関数f(t)の相関を取る．

$$R_{ff}(\tau) =$$

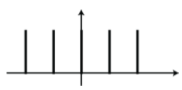


自己相関関数は，
「どれだけずれたら自分自身に近い形になるか」
を表す．
すなわち，エコーを発見していることに他ならない．

周期関数の自己相関

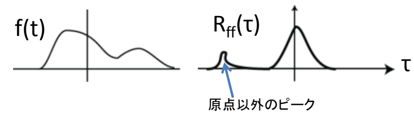


$$R_{ff}(\tau)=\int_{-\infty}^{\infty} f(t)f(t+\tau)dt$$



周期関数の自己相関関数は沢山のピークが出る

自己相関まとめ



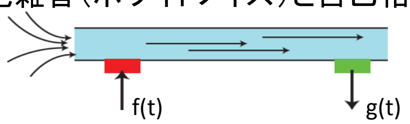
$R_{ff}(\tau)$：自分自身をτずらしたら，自分自身にどれだけ似ているか

●当然，τ＝0で最大値を取る．

●τ≠0で大きな値をとった場合，
その信号にはτの時間遅れ（エコー）成分が含まれている．

●信号に潜む周期性を発見することもできる

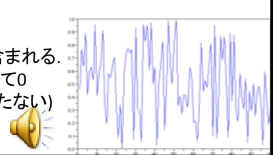
白色雑音（ホワイトノイズ）と自己相関



元の信号f(t)に「周期性」があったら，f(t)とg(t)の相互相関は
沢山の「ニセのピーク」が出てしまう．

元の入力f(t)はなるべく「でたらめ」であることが望ましい⇒白色雑音

＜白色雑音の定義＞
1．あらゆる周波数成分が均等に含まれる．
2．自己相関関数がτ=0以外では全て0
（＝エコー成分，周期性を全く持たない）



白色雑音とM系列

白色雑音をコンピュータで生成するにはどうするか？

M系列（ maximum length code ）：
「自己相関を原点以外0にする」ことに特化した
0,1乱数系列の中で，最も簡単なアルゴリズムで
生成するもの．

デジタル信号処理を前提とした高度なセンシングで
利用されることが多い．

今のところは言葉だけ心にとめましょう．

自己相関とパワースペクトル

自己相関関数をフーリエ変換してみる。

$$R_{ff}(\tau)=\int_{-\infty}^{\infty} f(t)f(t+\tau)dt$$

$$\int_{-\infty}^{\infty} R_{ff}(\tau)e^{-j\omega\tau}d\tau=\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(t)f(t+\tau)dt\right]e^{-j\omega\tau}d\tau$$

$=$

$=$

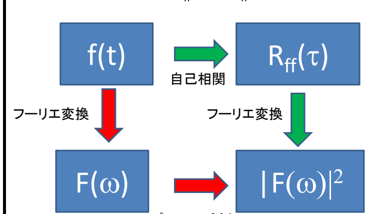
$=$

自己相関とパワースペクトル

$=$

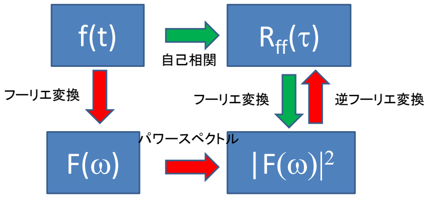
$=$

$$=\|F(\omega)\|^2$$



ウイナー・ヒンチン
の定理

自己相関とパワースペクトル



自己相関は
フーリエ変換⇒パワースペクトル⇒逆フーリエ変換
によって間接的に計算できる

●自己相関を直接計算した時の計算量：$O(n^4)$
●フーリエ変換(FFT)を使った時の計算量：O(nlog(n))
この方が計算が早いため多用される。

信号処理の基礎まとめ

以下の概念を2回で紹介しました．概念は理解しましょう．

- コンボリューション（畳み込み積分）
- FIRフィルタ
- IIRフィルタ
- 周波数的な低域通過フィルタ＝時間的な平滑化フィルタ
- エコー
- エコーキャンセル
- 相互相関
- 自己相関
- 白色雑音
- M系列

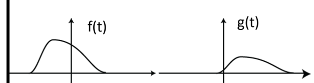
レポート課題：自己相関，相互相関

(1) 適当なwaveファイルに対してエコーを掛けてカラオケのようにする．(前回のレポート)

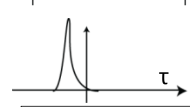
(2)その結果に対して通常の方法で自己相関関数を計算し、確かにエコー分の遅れのところでピークが出ることを確認する(エコーの検出)．

(2)のScilabソースファイルおよび結果のグラフ画像の添付．waveファイルは前回と同じなので添付不要。

（発展トピック）相互相関と検波

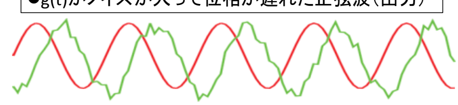


$$R_{fg}(\tau)=\int_{-\infty}^{\infty} f(t)g(t+\tau)dt$$

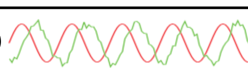


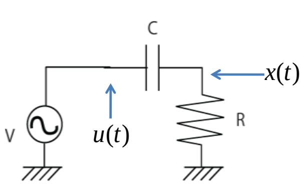
$R_{fg}(\tau)$が最大の値をとるτ
＝元の関数f(t)とg(t)のズレ
（ただし直流成分を取り除いた後）

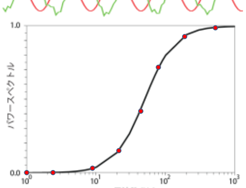
実用上重要なケースの一つ：
●f(t)が正弦波（入力）で，
●g(t)がノイズが入って位相が遅れた正弦波（出力）



正弦波入出力の例（1）

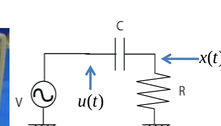




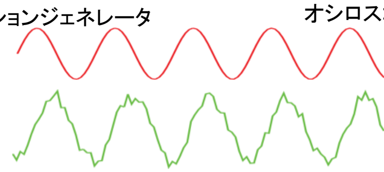


フィルタ回路：入力u(t)，出力x(t)
電子回路，アンテナ，ロボットアーム，etc
フィルタ特性を実測で求める場合
●u(t)=sin(ωt)を**入力**し，
●**出力**を観察する．
ωを変化させることでフィルタ特性を得る．

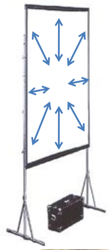
正弦波入出力の例（1）



ファンクションジェネレータ　　オシロスコープ

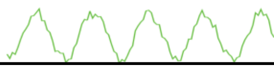


正弦波入出力の例(2)

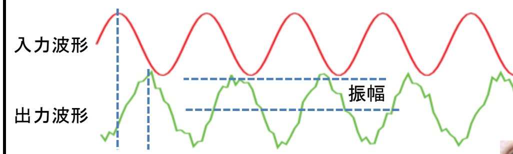


オプティカルフローの速度

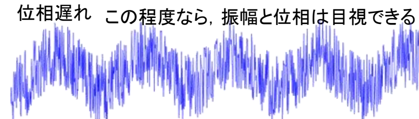
重心動揺



入力と出力の位相差，振幅比を求める



この程度なら，振幅と位相は目視できる



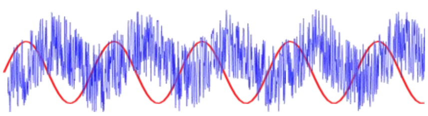
でもたいていこういう残念な感じ．ノイズがひどい

⇒振幅と位相を，どうやって求めたら良い？

方法1：バンドパスフィルタを使う

方法2：相関関数を考える

正弦波による相互相関



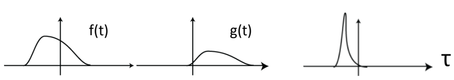
●出力信号f(t)：ノイズだらけの正弦波．位相不明．周波数は既知．

●入力信号g(t)：周波数固定の正弦波

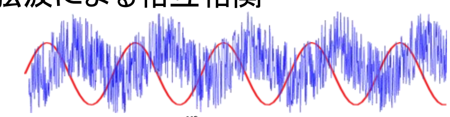
g(t)を動かしていき，ピタリと重なるところを見つければ良い

⇒相互相関の計算にほかならない．

$$R_{fg}(\tau)=\int_{-\infty}^{\infty} f(t)g(t+\tau)dt$$



正弦波による相互相関



$$R_{fg}(\tau)=\int_{-\infty}^{\infty} f(t)g(t+\tau)dt$$

正弦波g(t)を動かす

$$R_{fg}(\tau)=\frac{1}{T}\int_0^T f(t)g(t+\tau)dt$$

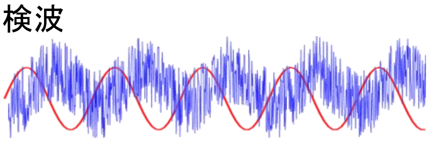
$$=\frac{1}{T}\int_0^T f(t)\boxed{}dt$$

$$=\frac{1}{T}\int_0^T f(t)\boxed{}dt$$

$$=\boxed{}$$

つまり正弦波の場合，積分の計算は2回やるだけでよい！！

直交検波



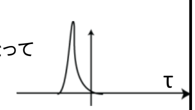
$$R_{fg}(\tau)=\cos(\omega\tau)\frac{1}{T}\int_0^T f(t)\sin(\omega t)dt+\sin(\omega\tau)\frac{1}{T}\int_0^T f(t)\cos(\omega t)dt$$

積分結果をS,Cと書いて

$$R_{fg}(\tau)=S\cos(\omega\tau)+C\sin(\omega\tau)$$

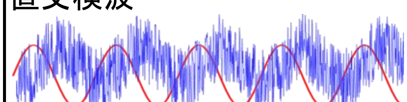
これが最大の値をとる場所τは，微分をとって

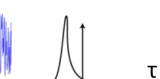
$$\frac{dR_{fg}(\tau)}{d\tau}=\boxed{}=0$$



$\boxed{}$ ∴ $\boxed{}$ ⇒位相遅れτが求まった

直交検波





$\therefore \arctan\left(\frac{C}{S}\right)=\omega\tau$ ⇒位相遅れτが求まった

このときの相関値は，

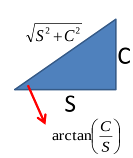
$$R_{fg}(\tau)=S\cos(\omega\tau)+C\sin(\omega\tau)$$

$$=S\cos\left(\boxed{}\right)+C\sin\left(\boxed{}\right)$$

$$=S\boxed{}+C\boxed{}$$

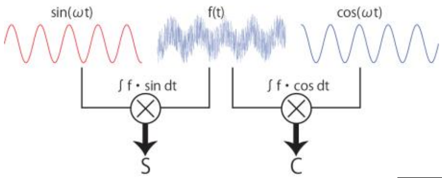
$$=\boxed{}$$

$=\boxed{}$ ⇒振幅に相当する量



直交検波まとめ

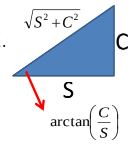
周波数ωがわかっているノイズだらけの信号f(t)があったとき，
振幅と位相は次のように計算できる．



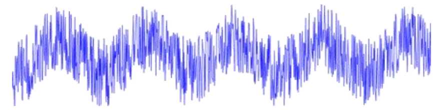
（1）sin(ωt)とcos(ωt)との内積をとり，S，Cとおく．

（2）位相遅れはarctan(C/S)となる．

（3）振幅は√(S²+C²)に比例する．



直交検波：もっと数式で理解する



問題を定式化

信号f(t)が，

f(t) =

と仮定できるとする．周波数ωはわかっている．

計測データから，振幅Aと，位相ずれφを求めるには？

直交検波：さらに数式で理解する

信号にsin(ωt)をかけ，積分する（＝内積をとる）

積分時間Tは充分長い．

$$S = \frac{1}{T}\int_0^T \left(A\sin(\omega t+\phi)+noise(t)\right)\sin(\omega t)dt$$

$$= \frac{1}{T}\int_0^T \qquad + \int_0^T$$

$$= \frac{1}{T}\int_0^T$$

$$=$$

$$=$$

$$=$$

直交検波：さらに数式で理解する

信号にcos(ωt)をかけ，積分する（＝内積をとる）

積分時間Tは充分長い．

$$C = \frac{1}{T}\int_0^T \left(A\sin(\omega t+\phi)+noise(t)\right)\sin(\omega t)dt$$

$$= \frac{1}{T}\int_0^T A\sin(\omega t+\phi)\sin(\omega t)dt + \int_0^T \cancel{noise(t)\cos(\omega t)}dt$$

$$= \frac{1}{T}\int_0^T A\left(\sin(\omega t)\cos(\phi)+\cos(\omega t)\sin(\phi)\right)\cos(\omega t)dt$$

$$= A\cos(\phi)\frac{1}{T}\int_0^T \cancel{\sin(\omega t)\cos(\omega t)}dt + A\sin(\phi)\frac{1}{T}\int_0^T \cos^2(\omega t)dt$$

$$= A\sin(\phi)\frac{1}{T}\int_0^T \frac{1+\cancel{\cos(2\omega t)}}{2}dt$$

$$=$$

直交検波：さらに数式で理解する

$$S = \frac{A}{2}\cos(\phi) \qquad C = \frac{A}{2}\sin(\phi)$$

この二つの結果から，

位相差

$$\frac{C}{S} =$$

$$\phi =$$

振幅

$$S^2+C^2 = \frac{A^2}{4}\left(\cos^2(\phi)+\sin^2(\phi)\right)$$

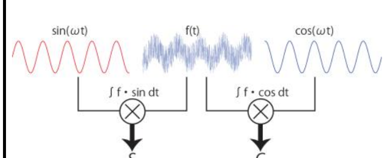
$$=$$

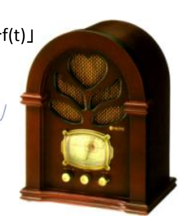
$$A =$$

位相差と振幅が求まった

（参考）AMラジオの同期検波

ラジオの信号は，典型的な
「周波数ωがわかっているノイズだらけの信号f(t)」





復調方式の一つ：「同期検波」

直交検波をリアルタイムに行い，ノイズに隠れた真の信号を得る．

今ではPC上で計算可能（ソフトウエア無線技術）

（※通信の分野では「直交検波」という用語は別の意味で使うので注意）