

インタラクティブシステム論 第6回

梶本裕之

Twitter ID kajimoto

ハッシュタグ #ninshiki

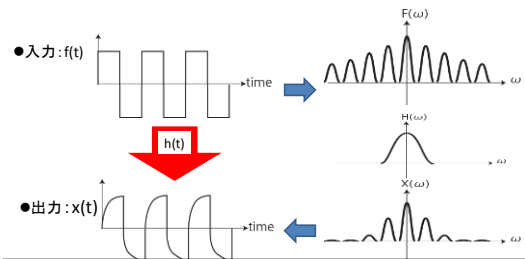
日程

4/9 インタロダクション
4/16 Scilabの紹介(西6号館3階PCルーム)
4/23 出張により休講
4/30 フーリエ変換
5/7 フーリエ変換と線形システム
5/14 信号処理の基礎
5/21 信号処理応用1(相関)
5/28 信号処理応用2(画像処理)
6/4 研究室見学(大学院オープンラボ)
6/11 中間確認テスト
6/18 ラプラス変換
6/25 出張により休講
7/2 古典制御の基礎
7/9 行列
7/16 行列と最小二乗法
7/23 ロボティクス
8/5~11 期末テスト

中間確認テスト

来週、中間テスト用の問題集を配布します。
一度式の導出を覚えることを意図しています。

(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？



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

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

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

両辺を逆フーリエ変換すれば時間領域の信号に戻る。

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

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。

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

(復習)コンボリューション定理

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

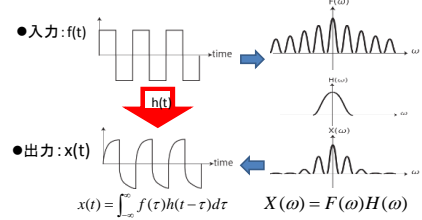
フーリエ逆変換 ↓ ↑ フーリエ変換

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

簡略化のため次のようにも表記される

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

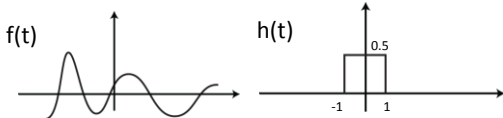
(復習)コンボリューション定理の意味(1)



- $h(t)$ のフーリエ変換が $H(\omega)$ であるとする。
- 周波数領域でフィルタ $H(\omega)$ 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,

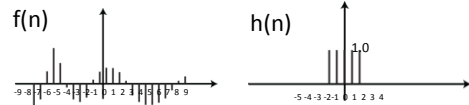
$$x(t) = \int_{-1}^1 0.5f(t-\tau)d\tau$$

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

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

$$x(n) = \dots + h(-4)f(n+4) + h(-3)f(n+3) + \dots + h(3)f(n-3) + h(4)f(n-4) + \dots$$



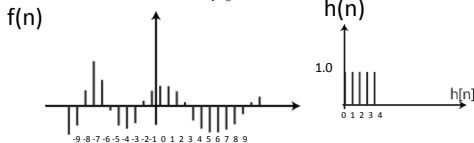
$h(n)$ が, $n=-2 \sim 2$ の間だけ1の場合,

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

- 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

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



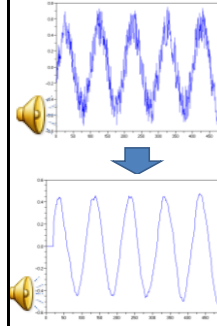
$i=0$ から始める: 未来のデータが使えないことを意味する。
この例は, 元データ $f(n)$ を, 4個平均して出力する。

- 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)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);

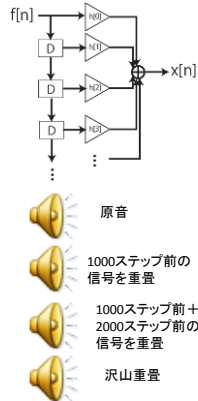
out=zeros(wave);

//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]);
    
```



エコーは害



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。

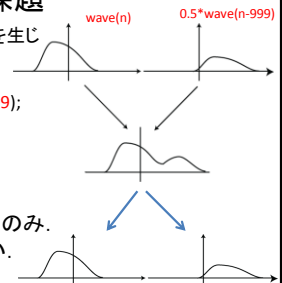
考え方:

- 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!」
- そのモデルに基づき、エコー成分と逆のゲインをかける
 $out(n)-0.5*out(n-100)$
 $=wave(n)+0.5*wave(n-100)-0.5*(wave(n-100)+0.5*wave(n-200))$
 $=wave(n)+0.25*wave(n-200)$
 ⇒エコーが半分に低減!
- 当然もつと工夫すれば... (もつとメモリがあれば)
 $out(n)-0.5*out(n-100)-0.25*out(n-200)$
 $=...=wave(n)+0.125*wave(n-300)$
 ⇒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

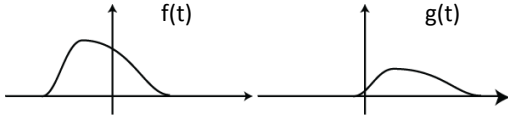
エコー成分が、どれだけ時間遅れを生じてやってくるかのモデルを推定

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



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。

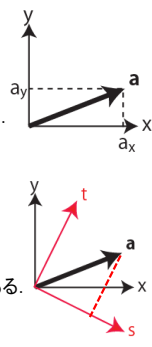
$$a \cdot x = [a_x, a_y] \cdot [1, 0] = a_x$$

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。

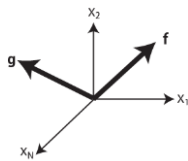
$$a \cdot s = [a_x, a_y] \cdot [s_x, s_y]$$

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 $f=[f_1, f_2, \dots, f_N], g=[g_1, g_2, \dots, g_N]$ を考える。



内積 $f \cdot g$ は、ベクトルfの, g軸成分(または逆)を表す。

$$= [f_1 \ f_2 \ \dots \ f_N] \cdot [g_1 \ g_2 \ \dots \ g_N]$$

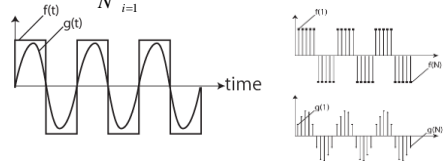
$$= \sum_{i=1}^N f_i g_i$$

(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

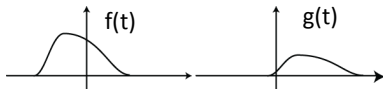
$$= \frac{1}{T} \int_0^T f(t)g(t)dt \quad (\text{連続関数})$$

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



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

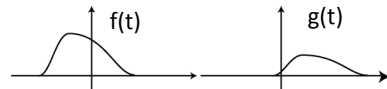


- <問題>
二つの信号が、
- 時間的にどれだけずれているのか
 - 時間のずれを無視したらどれだけ似ているのかを測定したい。

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

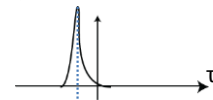
- g(t)をτだけずらしてみる ⇒
- f(t)との内積を取ってみる ⇒
- τを変化させていく。

相互相関



$R_{fg}(\tau)$: 二つの関数f(t), g(t)の, 相互相関関数

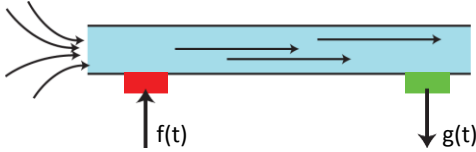
$$R_{fg}(\tau) = \int_{-\infty}^{\infty} f(t)g(t+\tau)dt$$



$R_{fg}(\tau)$ が最大の値をとる $\tau =$ 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)

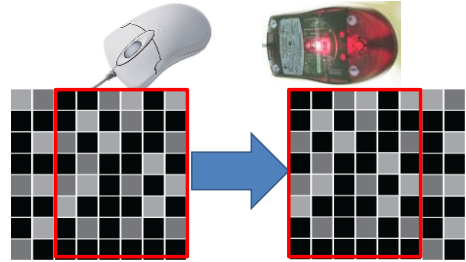


上流に熱源を置き、 $f(t)$ でランダムに変動させる。
下流で温度を測定する。 $g(t)$

$f(t)$ 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。

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

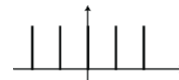


自己相関関数は、
「どれだけずれたら自分自身に近い形になるか」
を表す。
すなわち、**エコー**を発見していることに他ならない。

周期関数の自己相関

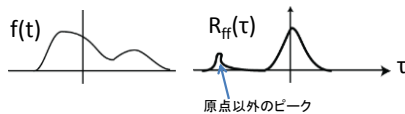


$$R_{ff}(\tau) = \int_{-\infty}^{\infty} f(t)f(t+\tau)dt$$



周期関数の自己相関関数は沢山のピークが出る

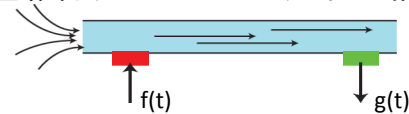
自己相関まとめ



$R_{ff}(\tau)$: 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、 $\tau=0$ で最大値を取る。
- $\tau \neq 0$ 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

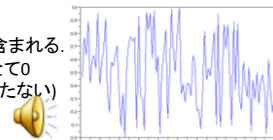


元の信号 $f(t)$ に「周期性」があったら、 $f(t)$ と $g(t)$ の相互相関は
沢山の「ニセのピーク」が出てしまう。

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 $\tau=0$ 以外では全て0
(=エコー成分、周期性を全く持たない!)



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。

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

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

$$=$$

$$=$$

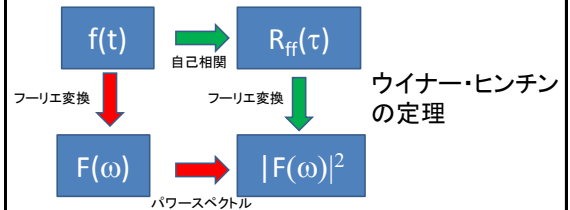
$$=$$

自己相関とパワースペクトル

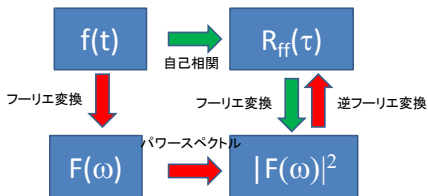
$$=$$

$$=$$

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



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: $O(n^2)$
 - フーリエ変換(FFT)を使った時の計算量: $O(n \log(n))$
- この方が計算が早いので多用される。

レポート課題1: 自己相関

- (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) その結果に対して定義通り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)。
- (3) ウィナー・ヒンチンの定理を用い、パワースペクトルの逆フーリエ変換によって同じ結果が出ることを確認する

(2)(3)のScilabソースファイルおよび結果のグラフ画像の添付. waveファイルは添付不要。(2)(3)の処理にかかった時間についてコメント

レポート課題: ヒント

- (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)

```
//waveファイルのロード
wave = loadwave('●●●.wav');

//1000ステップごとのエコーを10回重ねる例
out=[];
for i=0:9,
    out=out+[zeros(1,1000*i), wave, [zeros(1,9000-1000*i)]];
end
```

レポート課題: ヒント

- (2) その結果に対して定義通り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)。

```
time=length(out);
auto_correlation=zeros(1,time);

for tau=1:time,
    auto_correlation(tau)= [ ];
end

plot(auto_correlation);
```

自己相関の定義式から、[zeros(1,tau), out] と [out, zeros(1,tau)] の内積を取れば良い。(ベクトルの内積はどう書き表せるか?)

レポート課題: ヒント

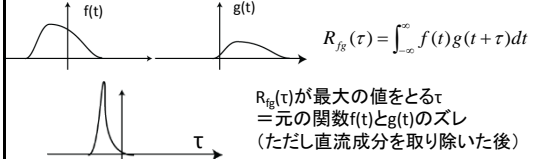
(3) ウィナー・ヒンチンの定理を用い、パワースペクトルの逆フーリエ変換によって同じ結果が出ることを確認する

```
//フーリエ変換
fourier = [ ] //過去のレポート課題を参照
//パワースペクトル
power_spec = [ ] //過去のレポート課題を参照
//自己相関
auto_correlation = [ ] //逆フーリエ変換

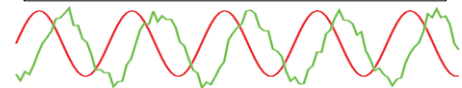
plot(auto_correlation);
```

※(参考)(2)(3)の結果の微妙な違いは、(3)が**信号の無限繰り返しを仮定している**ために生じる。逆に(2)もそう仮定すれば(3)と全く同じ結果が得られる

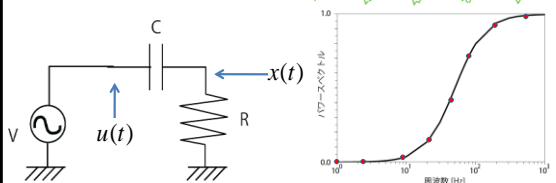
(発展トピック) 相互相関と検波



- 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)

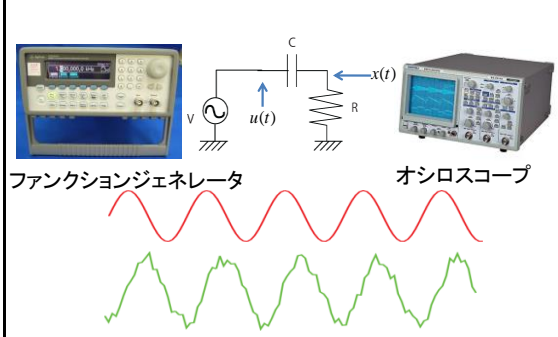


正弦波入出力の例(1)



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)



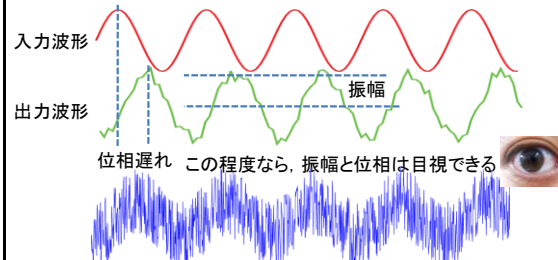
ファンクションジェネレータ オシロスコープ

正弦波入出力の例(2)



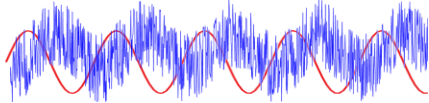
オプティカルフローの速度
 重心動揺

入力と出力の位相差, 振幅比を求める



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

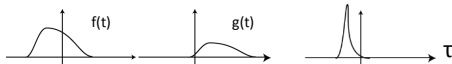
正弦波による相互相関



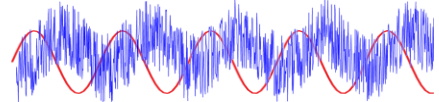
- 出力信号f(t): ノイズだらけの正弦波. 位相不明. 周波数は既知.
- 入力信号g(t): 周波数固定の正弦波

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

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

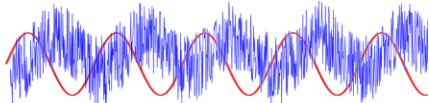
$$= \frac{1}{T} \int_0^T$$

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

$$=$$

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



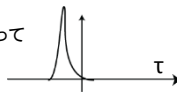
$$R_{fg}(\tau) = \cos(\omega\tau) \frac{1}{T} \int_0^T f(t) \sin(\omega t) dt + \sin(\omega\tau) \frac{1}{T} \int_0^T f(t) \cos(\omega t) dt$$

積分結果をS,Cと書いて

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

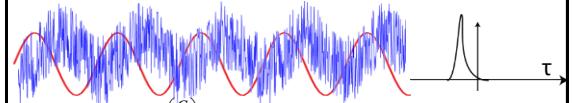
これが最大の値をとる場所τは, 微分をとって

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



⇒位相遅れτが求まった

直交検波



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

このときの相関値は,

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

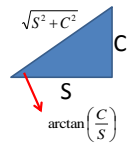
$$= S \cos(\dots) + C \sin(\dots)$$

$$= S \dots + C \dots$$

$$=$$

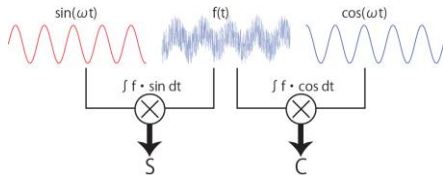
$$=$$

⇒振幅に相当する量



直交検波まとめ

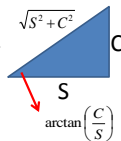
周波数ωがわかっているノイズだらけの信号f(t)があったとき, 振幅と位相は次のように計算できる.



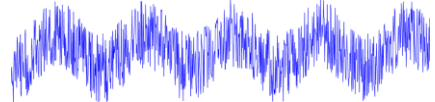
(1) sin(ωt)とcos(ωt)との内積をとり, S, Cとおく.

(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,

$$f(t) = \dots$$

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 $\sin(\omega t)$ をかけ、積分する(=内積をとる)
積分時間 T は充分長い.

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

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

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

$$= \dots$$

$$= \dots$$

$$= \dots$$

直交検波:さらに数式で理解する

信号に $\cos(\omega t)$ をかけ、積分する(=内積をとる)
積分時間 T は充分長い.

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

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

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

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

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

$$= \dots$$

直交検波:さらに数式で理解する

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

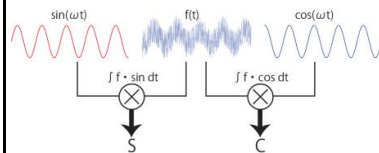
この二つの結果から,

<p>位相差</p> $\frac{C}{S} = \dots$ $\phi = \dots$	<p>振幅</p> $S^2 + C^2 = \frac{A^2}{4} (\cos^2(\phi) + \sin^2(\phi))$ $= \dots$ $A = \dots$
---	---

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」

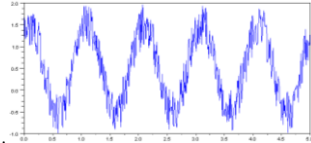
直交検波をリアルタイムに行い、ノイズに隠れた真の信号を得る。
今ではPC上で計算可能(ソフトウェア無線技術)

レポート課題2:

ノイズ入り正弦波の位相を調べる

次のScilabソースで表されるノイズ入り正弦波の位相遅れを求め、妥当性をコメントせよ。

```
//ノイズ入り正弦波
t=[0:0.01:5];//時刻
f=1.0;//周波数
amp=0.5;//振幅
phi=0.3*pi;//位相遅れ
y=sin(2*pi*f*t+phi) + rand(t);
plot(t,y);
```



```
S =  //yとsinの内積
C =  //yとcosの内積
ans_phi=atan(C,S) //検出された位相遅れ
```