

周波数変動状況下における適応交流ノイズキャンセラのスペクトルに現れる特徴

草間 康平

指導教員：戸田 尚宏

1 はじめに

測定される生体電気信号には、外来雑音が多く混入する。特に、商用交流電源に起因する交流雑音は周波数 (50/60Hz) 及びその高調波が生体電気信号に近く、波形を汚す大きな要因となっている。こうし交流雑音は図 1 に示した適応フィルタを用いたノイズキャンセラにより交流雑音を除去することが可能である。しかし、戸田 [1] は適応アルゴリズムの振舞いについて検討し、フィルタ係数の追従が十分である時、測定対象の信号波形自体の存在が自己除去成分と呼ばれる成分を引き起こし、スペクトル上ではノッチとして現れ交流雑音除去性能を悪化させるという問題を指摘した。一方、交流雑音が周波数変動する状況では適応フィルタの追従不足によりスペクトル上でピークが残留すると考えられてきたが、十分に検討されてこなかった。交流雑音除去法の性能評価において、スペクトル上、交流雑音周波数付近の急峻なピークが除去されていることがまず重要である。加えて、適応フィルタを用いる場合においては自己除去成分に相当するノッチが生じていないことが必要である。所望信号が未知である実際の場面においては、雑音除去結果のスペクトルが性能評価指標である。その場合、ノッチもピークも生じていないことが追従不足や自己除去成分の存在しない理想的な状態を示しているのかという重要な疑問に対して現時点では答える理論が存在しない。周波数変動のある状況下ではそれに追従するために適応時間を短く設定する必要があるが、その場合、自己除去成分は必ず発生している。そうであるにも関わらず、もしスペクトル上ノッチもピークも確認できない、滑らかなスペクトルが存在するならば、交流雑音の消し残しが存在し、そのスペクトル上ピークがノッチを打ち消しているに過ぎないと考えられる。このことより、スペクトルのみによる性能評価には大きな問題があることになるが、理論的にこれを証明することは極めて困難である。

そこで、本研究では自己除去成分と追従不足が存在し、時間軸上では誤差がある不完全な状態の適応ノイズキャンセラであってその出力のスペクトル上ノッチもピークも認められない所望信号のスペクトルに近い数値が存在すること、即ちスペクトルのみによる評価の危険性を指摘する。

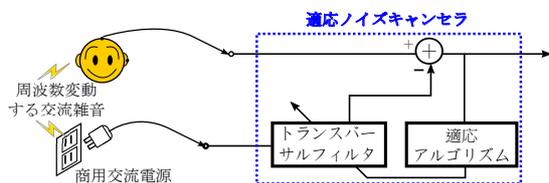


図 1 適応フィルタ

2 適応フィルタの追従度合

ここでは、追従不足の存在を明確にするために、理想的なフィルタ係数と推定されたフィルタ係数を用いて追従度合を定義する。これによって、適応パラメータや周波数変動の大きさによって追従度合を制御でき、追従不足の状態を意図的に設定できるようになる。信号の特性が刻々と変化している状況下でそれに追従するようにアルゴリズムが働く。適応アルゴリズムとして

忘却係数 λ をもつ逐次最小二乗法 [1] を用いた場合、交流雑音の周波数変動する状況では適応時間を短くした場合 (λ が小さい) では図 2(a) に示したようにフィルタ係数の十分な追従により自己除去成分が顕著となりスペクトル上では図 3(a) のようにノッチが生じる。一方、適応アルゴリズムの適応時間が長い場合 (λ が大きい) では図 2(b) に示したようにフィルタ係数の追従不足によりスペクトル上では図 3(b) のようにスペクトルにピークとノッチの両者が生じ、極めて複雑な影響が現れている。

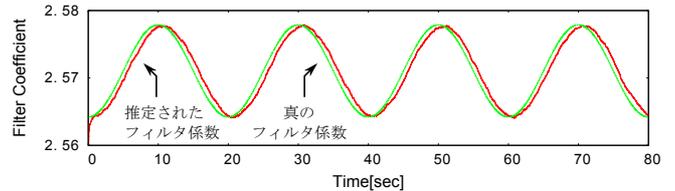
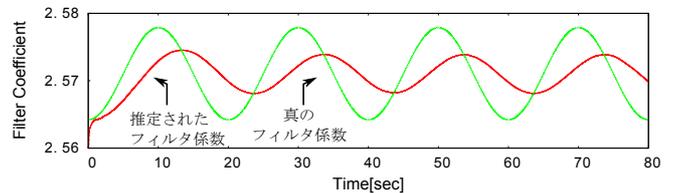
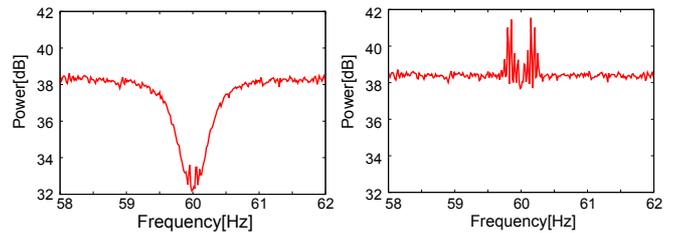
(a) λ が小さい場合(b) λ が大きい場合

図 2 フィルタ係数の追従が十分



(a) 追従が十分

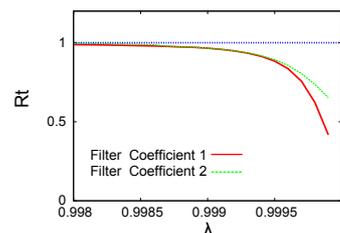
(b) 追従が不十分

図 3 フィルタ係数の追従不足

そこで、このフィルタ係数を用いて、以下のように追従度合 R_t を定義する。 R_t は 1 に近い程よく追従していると言える。

$$R_t = \frac{\sum_{k=1}^N |\hat{h}(k)|}{\sum_{k=1}^N |h^*(k)|} \quad (k = 1, \dots, N-1, N) \quad (1)$$

ここで、 $\hat{h}(k)$ を推定されたフィルタ係数、 $h^*(k)$ を理想的なフィルタ係数、 N を 1 周期分のデータ点数を表す。図 4 は横軸に適応パラメータ λ 、縦軸に各フィルタ係数の追従度合 R_t を定義評価したものである。適応パラメータ λ が 1 に近いほど、単調に追従が悪くなっていることが明らかである。

図 4 λ における追従度合

3 スペクトルへの影響

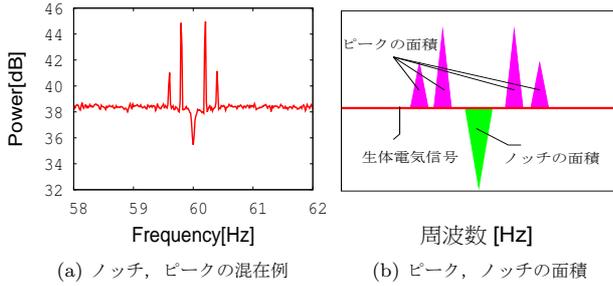


図5 スペクトル評価基準

ここでは、一般的な商用交流電源由来の周波数の揺らぎは $0.2[\text{Hz}]$ 以内と保障されていることから、変動幅 ω_d をその値に設定した上で、数値シミュレーションによりスペクトルへの影響を検討する。また、スペクトルへの影響として、図 5(a) に示したように、実際にノッチやピークの混在があるため、5(b) に示したように所望信号のパワースペクトルより上に突出しているピーク領域の面積と下に突出しているノッチ領域の面積の総和 D_f を定義する。この D_f が極小となるように試行を繰り返し、 $(\lambda^*, \omega_s^*) = (0.9997, 0.02)$ を得た。図 6 は ω_s を固定した上で、 λ を横軸に D_f を示したものである。また、図中○印の $\lambda = 0.9997$ の時に真のスペクトルの差が最も小さいことを表し、 (λ^*, ω_s^*) は極小であると考えられる。この λ でのノイズキャンセラの出力のパワースペクトルは図 7 に示すようなピークもノッチも存在しない滑らかなスペクトルとなっている。

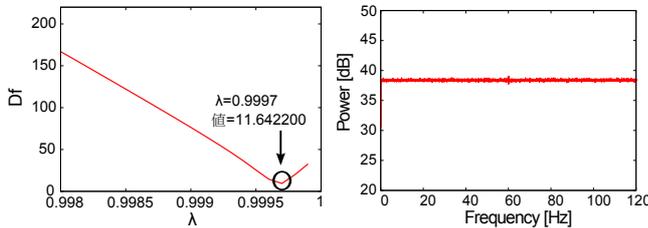


図6 λ における D_f の変化

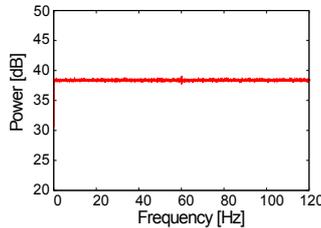


図7 周波数変動がある状況下でのパワースペクトル

周波数変動下及び、 $0 < \lambda < 1$ の状況下では、スペクトル上のノッチ、すなわち自己除去成分が必ず存在していることが理論上わかっている。しかしながら、平坦なスペクトル得られたということはノッチを打ち消すピーク、すなわち追従不足がある筈である。このことにより、この $\lambda = 0.9997$ では見掛け上交流雑音除去を最も良く除去できているように考えられる。

4 所望信号との誤差

ここでは前節のことを客観的に示すために、真の所望信号との誤差を求める。図 8 は周波数が変動する状況下と変動がない状況下、それぞれにおいて、 λ を種々変化した際の時間軸上で交流雑音除去後の信号と所望信号を二乗誤差で評価したものである。周波数変動状況下において、仮にスペクトルが滑らかであった $\lambda = 0.9997$ の時に追従不足も自己除去成分もない理想的な点であるとすると、これ以上に低い誤差は存在しない筈である。しかし、図 8 より $\lambda = 0.9997$ では周波数の変動ない状況下にお

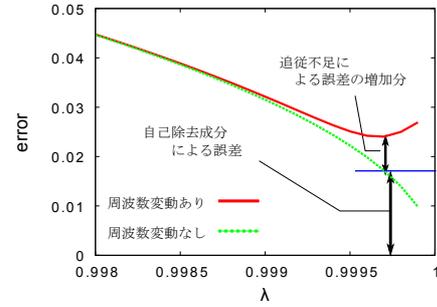


図8 λ における所望信号との誤差

ける誤差は、変動がある状況下における誤差よりも低いことが明らかである。さらに、図 9 に示したように周波数変動がない状況下での $\lambda = 0.9997$ の時のパワースペクトルには明らかに深いノッチが現れており、時間軸上での所望信号との誤差が小さいにも拘わらず、スペクトル上では周波数変動のない場合の方が、図 7 の変動のある場合よりも悪い状況であると判断される。また、これらの事は周波数の変動の速さを変化させた上でも同様な結果が得られた。

これらの結果より、スペクトルのみで完全な交流雑音除去ができていないことを判断するのは誤りであるという問題を指摘することができた。

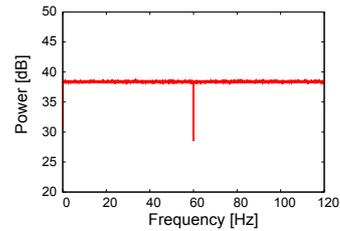


図9 周波数変動がない状況下でのパワースペクトル

5 おわりに

周波数変動状況下において、フィルタ係数の追従が不十分の時は、スペクトル上でピークとノッチの両方が存在し複雑な影響があることがわかった。また、適応フィルタの追従不足と、自己除去成分の両者が存在し、誤差が大きい状態であるにも拘わらず、パワースペクトル上ノッチもピークも認められない状態が存在することが明らかとなった。このことにより、交流雑音の周波数が変動する状況下で、スペクトルのみで完全な交流雑音除去ができていないことを判断するのは誤りであるという問題を指摘した。即ち、交流雑音除去が完全ならば、スペクトルは滑らかであるが、逆にスペクトルが滑らかな事をもって、完全な交流雑音除去がなされているとは言えないことを指摘した。この事はこれまで全く議論されてこなかった点である。

今後の課題としては、スペクトルに代わる基準の創出、及び周波数変動状況下に対応した適応フィルタの構築が必要となる。その際、非因果的な係数平滑化 [2] が有効になると考えられる。

参考文献

- [1] 戸田尚宏, “パルス状信号に混入する交流雑音の適応フィルタによる除去法とそのフラッシュ光網膜電位図への適用”, 電子情報通信学会論文誌 D, Vol.J94-D, No.10pp.1685-1695, 2011.
- [2] 長谷川雅之, “振幅変動状況下における非因果的平滑化を導入した適応ノイズキャンセラによる交流雑音除去”, 愛知県立大学大学院情報科学研究科 修士論文, 2014.