

# 2018/11/05

---

## はじめに

Q: ex2018\_10\_21\_search\_path\_contain\_only\_IMCL4.mなどがどの結果出しに用いられたかはどこで分かる？ A: ex2018\_10\_21\_search\_path\_contain\_only\_IMCL3.mlxなどのように名前で紐づけてあったり, discuss\_2018\_10\_21-\_search\_path\_contain\_only\_IMCL.mlxの中に「コード→結果→議論」のような流れで紐付けられていたりする. mlxファイルを見ることで分かる.

## そもそも送受信フォーカスは正しく機能しているか？

- 問題点
  - 焦点深度が音響境界と一致したときに信号強度がピークをとった事により, 整相加算が適切に機能しているとみなしていたことが問題である. 信号を加算する前に参照点と波形のピークが一致しているかどうかを確認する必要がある.
- 確認方法
  1. IMAT領域を1 pixel分の点として作成した媒質データに対して送受信フォーカスをかけた.
  2. 参照点とRF信号の振幅ピークが一致するかを, 焦点深度を変化させながら, 参照点(赤点), RF信号の振幅ピーク(青点)で逐次示すムービーを作成して目視で確認した.
- 結果
  - IMAT領域に到達する焦点深度において, 参照点とRF信号の信号ピークが良好に一致したことが認められた.

## 受信波面の整う時刻が想定よりも早いことについて

### 1. 受信波面が整っている時刻において

**波形包絡線最大位置と参照点がどれだけの遅延をしているのか確認.**

- 焦点深度が7.3 mmにおいて目視で波形の乱れがほとんどないことを確認.  
(IMCLとIMATの境界位置は奥行方向7.9 mm)
- このときの遅延を確認.
  - 確認されたこと
    - 中央のch50における遅延は, 焦点深度と領域境界との変位によるものとして説明できる.
  - 考察
    - かならずしも境界位置に焦点を合わせないと波面が整わないわけではない.
    - 伝播経路内に混在するIMATにより焦点ずれが生じ, 領域境界に焦点を合わせたときの反射波が歪んだと考えられる.
    - むしろ境界に焦点を合わせないことにより, 媒質の細かな不均一さによる波面歪みの影響が偶然小さくなることもある.
  - 提起できる仮説
    - 境界に焦点を合わせても波面歪みが生じている原因は, 伝播経路中に混在するIMATによるものである.
    - 境界に焦点を合わせずとも, 整っている波面と仮定音速により定まる参照点列とを比較して, 最も当てはまるような仮定音速によりIMCL量が推定できる.

# 2018/11/06

---

## 受信波面の整う時刻が想定よりも早いことについて（続）

### 2. IMCLとIMATの境界を焦点とする条件下で伝播経路内にIMAT領域が混在しているかを確認.

- 伝播経路で囲まれた領域内でのIMAT面積比を確認. 経路の不均質さの目安として.
- 確認方法
  1. 焦点深度に対応した開口径をなすための駆動素子の各位置と焦点位置とを結ぶ線分を音速分布イメージに上書きする. (目視の確認)
  2. 実際にカウントして占有率を確認.
    - 内挿  
焦点と駆動素子のうち最端の素子のインデックスが分かり, x軸方向に何グリッドだけ離れているかの情報があれば, 内挿処理ができる.  
①内挿処理で, 両側端点のインデックスが分かる.  
(2点参照の線形内挿. 非整数に対しては開口中心から下部の駆動素子について floor, 上部の駆動素子についてはceilを用いる.)  
②y軸方向に着目して, その間にあるIMATをカウントする.  
③最後にカウント数を総探索数で割る.
- 結果
  - 焦点深度7.9 mm (IMATとMTとの境界位置) における, 伝播経路で囲まれた領域内 (領域A) でのIMAT面積比は, 2.58 %であった.
- 考察
  - 領域Aにおいて不均質性が生じているために, 波形歪みが生じた可能性がある.
  - そこで, 領域Aに存在するIMATを除去した媒質を用いてRFデータを取得し, 同様に焦点深度7.9 mmでの波形形状を確認する.

### 3. 領域Aに存在するIMATを除去した条件で受信波形形状の時間変動を確認する.

- 結果
  - IMATを除去する前よりもIMATを除去した方が, 反射強度から判別した領域境界に一致する焦点深度での波形形状が整っており, また振幅のコントラストも若干高くなっていた.
  - 一方で, IMATを除去したのにも関わらず, 波形形状の変化がわずかなものであった.
  - また, 反射強度から判別した領域境界は, IMAT除去前では焦点深度7.9 mmの位置であったのが, IMAT除去後では焦点深度8.0 mmの位置へと変動した.
- 考察
  - 点音源で送受信しているため, 全周囲からの反射波を均等に受けている. それによって仮想点音源以外からの波形が混在して波形が乱れているのではないだろうか.
  - 全周囲からの影響を受けないようなIMATの除去をすることによる変化をみることで, その仮説の有用性が確認できると思われる.

# 2018/11/07

---

## IMCL推定方法の二層媒質を用いた原理検証

- 2018/11/06の3.における考察で, 全周囲からの影響を受けないようなIMATの除去を行なうことで, 焦点位置に存在する領域境界からの影響のみを受けるような状況をつくり, 波形形状の変化を観察することが次にやることとして定まった.

- 一方で、そもそも波形形状を評価すること、および、適切な仮定音速を求めることが実際にIMCL推定に繋がらぬのかを確認せずに、仮定ありきで話をすすめてしまっていた。
- そこで、伝搬経路上に確実にIMATが存在しない系のうち、最も簡素と考えられる系として、二層媒質系を用いて、IMCL推定精度を評価する。実際の筋肉・脂肪系に対してはここで得られる精度よりも劣った精度となるはずである。まず、簡素な系で本手法が機能するかどうかを確認することは重要であるとする。
- 実験条件
  - IMATとMTの境界位置が焦点深度7.9 mmに位置するような媒質のうち、単一の層構造をとるような媒質を作成した。
  - 0 - 20 %の範囲で変動させる。
  - [1, 2,..., 20 %]の20通りにする。
  - 仮定音速も20通り。計400回の演算になる。
  - 焦点深さを7.9 mmに固定すれば計算量も大幅に削減できる。
  - RFデータを見たいときはどうすれば良い？
  - ~~400ケース中1ケースだけに着目すれば十分と思う。~~
  - 反射強度から境界面の深度を全ケースに渡って調べる必要がある。[2018/11/08]
    - アルゴリズムのフローが、境界面を検出→参照点の妥当性評価である以上は、反射強度プロファイルを確認する必要がある。
    - 仮定音速が大きく異なると、フォーカスがかからなくなるため、境界面の位置を正確に推定できなくなることが可能性として残ったままの状態では話が飛躍しているように思われる。
    - なるべく配列のサイズを過大にしないようなプログラムに整えていく必要がある。

## ■ 境界面の深度を走査した場合の解析条件

# 2018/11/08

---

## IMCL推定方法の二層媒質を用いた原理検証(続)

- 結果
  - 反射強度プロファイルを参照せずに焦点深さを7.9 mmで固定した場合。
    - 精度良く推定できていることが確認された。
    - 一方で、配列の参照の仕方が不適切であった可能性がデバッグの時点で出てきた。
    - 焦点深さを固定する前の大きい配列を保存しながら行ったIMCL占有率推定プログラムの見直しが必要である。
      - 原理上うまくいきそうという見通しが立ってよかったのは確か。
  - 反射強度から媒質境界を推定。その推定境界位置について参照点とRF信号の正の最大値のインデックスとの距離から適切なIMCL占有率を反映した仮定音速を推定した場合。
    - 計算に時間がかかることが問題であるが、所要時間を控えていなかった。
    - 推定誤差が $\pm 0.5$  %と、非常に高い精度となった。

---

メモ: 【主にアルゴリズムの修正について】

Q: 反射強度が焦点深度0.4 mmで最大となる場合があるが、これはフォーカスがかかっているというよりもむしろ、隣の素子にそのまま音波が透過・到達した事によるものである。よって、反射強度が最大の位置が焦点深度0.4 mmにあるときは、そのピークよりも深部にあるピークを参照させるようにしなければならない

い. peak関数を使って対処できないだろうか.

A:

```
[~,ind_max_signal(kk,ll)] =
findpeaks(abs(focal_signal_total(kk,:)), 'Npeaks', 1, 'SortStr', 'descend');
```

これで解決できた.

Q: 正解IMCL占有率, 予測IMCL占有率(これにより仮定音速が求まる)が一致した場合, 媒質の境界面の位置 = 反射強度が最も大きい焦点深度であるはずが, 0.1 mm分ズれていた. これから考えられることは?

A: そもそも受信フォーカスで適用するオフセットが適切ではない. このオフセットはどのように決めたのだったかを見直す.

```
%受信用の参照点算出
for jj = 1:num_echo_receiver
    distance_from_focal_point_all(1,jj) = norm(t_pos(:,jj) - focal_point(:,ii));
    delay_time_all = round(((distance_from_focal_point_all -
focal_depth(1,ii))/v_reference(1,kk))/kgrid.dt); %[sample]
    reference_point(1,jj) = round(delay_time_all(1,jj)+1+
(2*focal_depth(1,ii)/v_reference(1,kk))/kgrid.dt+25);
    %25はfocal_amplitudeを最大にするオフセット.
    reference_point_lowerlimit(1,jj) ...
    = round(delay_time_all(1,jj)*(v_reference(1,kk)/v_muscle)+1+
(2*focal_depth(1,ii)/v_muscle)/kgrid.dt+25-1);
    reference_point_upperlimit(1,jj) ...
    = round(delay_time_all(1,jj)*(v_reference(1,kk)/v_fat)+1+
(2*focal_depth(1,ii)/v_fat)/kgrid.dt+25);
    %どんなに遅延しても早く到達してもこの範囲内に焦点位置からのエコーパルスが入っているで
    だろう上限・下限
end
```

この参照インデックスを,

1. rfデータそのものの整相加算に用いて反射強度を計算, 境界位置を推定.
  2. rfデータに包絡線処理をしてその最大値と参照インデックスとの距離を測っている.
    - 2.が特に致命的なミス. 包絡線処理をすることで, データのピークの位置が変わってしまう. このミスは, どちらもrfデータorその絶対値を用いて議論することで対策できる.
- オフセットの決め方
    - source\_wave.matから, 初期条件での圧力時間プロファイルにおける, RF信号の正の最大値のインデックスが25である. デフォルトでは遅延をかけた後参照しているのは1番目のインデックスであるから, 25-1=24だけオフセットを設けた. 参照インデックスと干渉後のRF信号の正の最大値のインデックスが良好に一致していることを目視で確認した.

Q: 干渉前後で信号周期は変動するはず. 確認したか?

A: 思ったより変動していなかった. 1サンプル程度のズレであった.

# 2018/11/09

## IMCL推定方法の二層媒質を用いた原理検証(続)

- ☒ 反射強度により境界位置を決めているが、誤検出が見られた。これへの対応。
  - ☒ もう一度case26の複雑な構造をした媒質に対してIMCL推定手法を適用する。
    - コードを書き直したことにより、問題点が解決した可能性がある。
      - ☒ 駆動素子数を限定しているのに関わらず、最後に全素子において波形形状評価をしていた可能性がある。これについて確認をする。
- 
- 反射強度による境界位置検出におけるエラーの調査
    - 正解IMCL占有率: 10 %, 予測IMCL占有率: 16 %において誤検出が見られた。このときと、検出が正しく機能した、正解IMCL占有率: 10 %, 予測IMCL占有率: 10 %での結果 (focal\_signal\_total[dim={20,20}])を比較して、上述したピーク関数を用いる推定方法の問題点を明らかにする。
    - 仮定音速が正解音速と大きく異なると、媒質境界位置が0.4 mm~0.5 mmと顕著に外れた位置として検出された。
      - 考えられる原因: 素子間隔は約0.4 mmである。一方で誤検出された媒質境界深度は0.4 mm程度である。すなわち、反射波ではなく透過波を加算した結果であると考えられる。
      - 改善案
        1. 0.4 mm - 0.5 mmの信号強度を参照しないようにする。
          - 問題点として、0.4 mm程度の箇所に領域境界があったときに誤判別してしまう可能性がある。
          - それほどの近距離では正確に音速を予測することがもともと困難であるようなら参照しないことに問題はないはずである。
        2. 反射強度を加味して修正をかける。
  - 確認プロセスにおいて気づいた他の点
    - 信号強度(要修正: 正の値のはずなのに信号振幅に対して用いてしまっている)の絶対値が最大の箇所を探索しているが、整相加算して信号が最大化されるのは正の方向にであるので、絶対値を使うことは適切でない。位相の情報を捨てて振幅の大きさだけに注目しているために推定精度の低下に寄与している可能性がある。この点を改良をして結果を比較する。
      - 例えば、実際のIMCL率よりもIMCL率を多く見積もっているとする。そのとき、仮定音速が小さくなっているのだから、仮定の $l/c$ が大きく見積もられていることになる。このとき、送信フォーカスでの実質の焦点は想定した焦点位置よりも深い箇所に位置することになる。"H:¥result¥2018\_11\_07\_IMCL\_estimation\_principle\_verifiacation¥2018\_11\_09\_focused\_signal\_comparison\_TorF.png"を参照すれば、現に信号強度のピーク位置が奥側にシフトしていることが確認できる。しかしながら、強度の絶対値で境界位置を検出しているため、手前に焦点位置があるものと誤判定してしまっていることが現状の問題点である。そこで、上のようなアルゴリズムの変更を行なう。
      - 差分
 

```
"H:¥experiments¥2018_11_07_IMCL_estimation_principle_verifiacation¥ex2018_11_08_path_only_including_MT_2layers_stroke2.m"
```

 ↓

"H:¥experiments¥2018\_11\_07\_IMCL\_estimation\_principle\_verifiacation¥ex2018\_11\_08\_path\_only\_including\_MT\_2layers\_stroke3.m"

2018/11/13

IMCL推定方法の二層媒質を用いた原理検証(続)

- ☒ homogeneity\_total, max\_signalの可視化
    - ☒ カラーマップの表示
    - ☐ 同じ媒質に対する推定IMCL率の変化に伴うhomogeneity\_totalとmax\_signalのプロファイル表示.
  - ☒ homogeneity\_totalに含まれるInfに対する処理
  - ☐ 経路上のIMATを取り除いたときの推定精度評価
- 
- 今まで行ったIMCL推定アルゴリズムの変更履歴

参照点	検出点	問題点
1. 整相加算	包絡線の最大値	包絡線最大値とRF信号最大値をとる点は一致しない.
2. 整相加算	RF信号絶対値の最大値	整相加算ではRF信号最大値をとる位相を基準に遅延をかけているためRF信号最小値を検出することで計測誤差が生じる.
3. 整相加算	RF信号の最大値	理論上問題なし

- 同じ媒質に対する推定IMCL率の変化に伴うhomogeneity\_totalとmax\_signalのプロファイル表示
  - 確認されたこと
    - 1. homogeneity\_totalのみに着目すればIMCL推定精度が十分でない
    - 2. max\_signal(反射強度最大値)にも着目すればIMCL推定精度が向上する
    - 3. 整相加算における位相と振幅情報の両面から評価していることに近いと考える.
      - もっと考察が必要
- 結果の遷移が明快な動画の作成

```
dim = {medium, estimation};%2
dim2 = {true_depth or false_depth};%1
```

- homogeneity\_totalに含まれるInfに対する処理
  - Infのかわりに最大値を代入

2018/11/16

IMCL推定アルゴリズムの修正

- ☑ 検出点を信号最大値にしてしまうと、ノイズの影響を大きく受ける。そこで、相互相関をとり、時系列的にもっとも送信波形と似た形状の波形がある時刻を割出すことで検出の正確度を担保する。
  - 対象：下の条件で得たデータ  
 "H:¥result¥2018\_11\_07\_IMCL\_estimation\_principle\_verifiacation¥2018\_11\_11\_no\_06\_maxrfsignal\_detect\_boundary\_case26"

媒質	参照点	検出点	境界位置
腓腹部断面	整相加算	信号最大値	最大振幅

▲検出点が信号最大値なのはおかしい

- 結局相互相関をとってもほとんど違いがなかった。matlabのxcorr関数を使ったがそれが適切でないようにも思えない
- 全焦点深さでの最大信号振幅に着目して境界検出を行っていたが、透過波からの影響をうけるような浅い深さでは信号振幅を参照しないことにより、正しい境界位置を検出できる確率が上がると思われる。
- 東先生に質問してみると、相互相関をとる対象の関数を誤って選択していたことがわかった。
  - RFデータの隣合うチャンネルの時間波形間での相互相関をとり、そのラグを取ることで、曲率をとり、整相加算における参照点プロットがとる曲率と比較することにより、正しい仮定音速を推定するという流れはどうだろうか。

## 2018/11/19

### IMCL推定アルゴリズムの修正

- ☑ 題目のネーミングを考え直す
- 透過波による影響を考慮しないビームフォーミングによる媒質境界判定法について
  - 透過波の影響をうける時間フレームについては参照しないようにする。
  - 受信波形振幅の最大値を用いて検出していたことに問題があった。振幅が大きくなっているのに関わらず、位相が回っていることによって、あたかも振幅が小さく見えてしまう。この状態では信号のピークを正確に判定することが困難になってしまう。そこで、包絡線処理を行なうことで振幅情報だけを取り出す。
  - 確認されたこと
    - 半値幅が設定音速と環境音速との違いで変わりそうか確認してみる
      - 半値幅が確かに設定音速と環境音速の違いが少ない場合において低減されたが、この要因として、設定音速の適切さが支配的なのか、あるいは設定音速と環境音速の誤差が媒質不均質さの補正としての役割を持つことによる効果が支配的なのかこのままでは弁別できない。
    - ところで各設定音速での反射強度を正規化して積算した結果、境界深度がパッキリ見える様になった。
      - 初期値を適当に決めて例えば5条件計測にして境界を見つけられるなら処理速度は上がりそう。境界位置がわずかにシフトしているはずなのでそれも可視化したい。
  - おもったこと

- やはり境界位置を整相加算で見つけてそこから手前の領域の不均一さを評価。続いて不均一さがしきい値を下回っていたら信頼して、環境音速と設定音速との差を波面の曲率によって評価するのがよいとおもう。
- マスク処理をかけないでRFデータを広く使ってやるのがよいと思った。
- 隣合う素子での波形データの相互相関の最大値を評価することで、素子ごとの信号振幅最大値を評価するよりもノイズに強い変位追跡が可能になりそう。
- 波面形状を反映しているRFデータはまるで指紋のようにみえる。

## 2018/11/20

---

### IMCL推定アルゴリズムの修正

- 題目の案
  1. 反射・透過超音波を用いた筋肉内脂肪量推定手法の開発
  2. 対向超音波プローブによる筋肉内脂肪量推定手法の開発
  3. 対向超音波プローブを用いた反射・透過情報による筋肉内脂肪量推定手法の開発➢ 2か3が良いと思った。
- 相互相関を用いた受信波面曲率評価
  - 素子間のRFデータの相互相関をとっているだけなので、RFデータの代表点を算出することはせず、素子間のRFデータの時間シフトを算出する。
  - 仮定参照点との誤差の評価はどのように行なうか
    - 仮定参照点がなす二次関数と、RFデータの代表点がなす二次関数との誤差を評価する。
  - 結果
    - 振幅最大値をとるよりも波面形状評価性能が向上した。
  - 改善点
    - 隣合う素子のRFデータ時間シフトしか見ていない。すなわち、駆動素子の端と端との位置関係などの情報は反映されていない。
    - 隣合う素子だけでなく、多数の参照素子の組合せから時間シフトを推定することで推定精度が向上しないだろうか。
    - しかしながら、隣合う素子のRFデータに連続性があるという洞察から隣合う素子のみを参照しているのに関わらず、波形が大きく変わっているかもしれないRFデータ同士を比較することにどれほどの有用性があるだろうか。
    - 時間シフト推定に採用する素子組合せを相互相関値の程度から決めるというやり方はどうだろうか。

## 2018/11/21

---

### IMCL推定アルゴリズムの修正

- 思ったこと
  - いろいろ場当たり的に手法を変えて考察もせずに検討を繰り返すのはよくない。なぜダメだったのかを考えて改善をするという正攻法が求められている。
    - なぜダメだったのかを考えて：振幅最大値をとることが良くない理由を考えて
    - 改善する：正規化相互相関を用いる。
    - なぜダメだったのかの理由付け：ノイズに弱い



- これについての議論をしている論文や教科書をあたる。時間はかけずに参照程度に。
- 改善するために正規化相互相関を導入した理由：たくさんの情報を総合しているためノイズに強い
- これについても言及している教科書等をあたる。時間はかけずに。
- 相互相関の改良へのヒント
  - 位相限定相互相関
    - 信号をフーリエ変換して、位相情報のみを取り出して相関をとっている。
    - ここで問題なのは、位相が回れば相関値が簡単にあがる。つまり、半波長の移動量しか正確にトラッキングできないということ。
    - 振幅情報も交えて見たほうがいいに決まっている。
    - これが有効に機能するのは、構造固有の空間周波数を持った画像データ等に対してである。
  - コヒーレンシーという指標がある。
    - 相互相関のフーリエ変換した形式のものである。
    - 説明しているウェブページがあったので参考にする。 [http://www.eri.u-tokyo.ac.jp/people/knishida/Lecture/Timeseries\\_cross.pdf](http://www.eri.u-tokyo.ac.jp/people/knishida/Lecture/Timeseries_cross.pdf)

## 2018/11/22

---

### IMCL推定アルゴリズムの修正

- 調べてわかったこと
  - 相互相関関数は、クロススペクトルの逆フーリエ変換である。
  - クロススペクトルの逆フーリエ変換として相互相関を評価するときには、信号対雑音比が良好な帯域で重み付けを多めに行って逆フーリエ変換をすることによって、位相の傾きをなるべく保つようにしたい。(一般化相互相関関数)
  - RFデータの帯域と送信波形の帯域は似通っているので、帯域を適切に選んで重み付けするという処理は有用ではない。
- やらうと思ったこと
  - ch間のシフト量は精度良く分かったのだから、それを利用して波面形状評価を行ってやれば良い。すなわち参照点列のch間でのシフト量と検出点のch間でのシフト量を比べて上げれば良い。評価関数は平均二乗誤差等で。
  - 結局設定音速誤差と境界検出深度誤差との関係を可視化する必要がある。
  - 実際にブロック肉を用いて波形先頭到達時間等を求めるときには、一般化相互相関関数が役に立つかもしれない。

## 2018/11/26

---

### IMCL推定アルゴリズムの修正

- やったこと
  - 水平方向にセクタスキャンする
    - 境界と駆動アレイ間の不均質さが最も少ない送受信条件をもとめる
    - その条件で最も信号強度の大きい設定音速をもとめる

- 具体的なフローチャートの策定
- これらの情報を可視化するためのコード作成・修正

## 2018/11/27

---

### IMCL推定アルゴリズムの妥当性検証

- やったこと
  - "H:¥experiments¥2018\_11\_25\_IMCL\_estimation\_boundary\_search\_re¥ex2018\_11\_25\_path\_only\_including\_MT\_stroke5\_case26.m"のリファクタリング
  - 上のスクリプトからの結果を更にわかりやすく可視化する
    - 境界位置検出精度がひと目で分かるようにする
      - 媒質音速分布イメージの上に, 検出境界位置をプロットする
      - IMCL占有率1 ~ 20 %の場合で実施する.
      - アニメーションにできるとさらに良い.
  - 解析条件の決定的なミス
    - IMCL率が適正でなかった.
      - IMCL濃度がEMCL占有領域の関数になっていた.
      - 全体の脂肪率推定のために, 全体の脂肪率を, EMCL率が変わっても一定に保たせたかったためにIMCL濃度を適応的に変動させていた.
    - IMCL率を適正にして再度解析を回す.
  - ミスはなかった.
    - 上のことを考慮してコードを回していたみたいだ.
  - その他の細かなミス
    - t\_posのsingle型では, 厳密に0.4 mm間隔で配置されていないようだ.
    - 駆動素子の判定が完全ではなかった.
    - そこで, round(a,4)を用いて小数点以下3桁以降は落とすようにした.
    - "H:¥experiments¥2018\_11\_25\_IMCL\_estimation\_boundary\_search\_re¥ex2018\_11\_27\_path\_only\_including\_MT\_stroke6\_case26.m"に反映した.
  - アニメーションを作った.