

19 ステレオ信号の処理

19.1 ステレオ信号の表現

ステレオ信号とは、2つのチャンネルからなる信号である。この2つのチャンネルの信号がそれぞれ右のスピーカー、左のスピーカーから別々に出て空間的な広がり表現する。MATLABでは、N行2列の行列データで、長さN点のステレオ信号を表現する。また、この形式のデータであれば、soundなどの関数は、ステレオ再生できる。

```
>> fs=8000;
>> t=0:1/fs:1/3;
>> y=sin(2*pi*440*t);
>> y_left=[y zeros(1,8000-length(y))];
>> y_right=[zeros(1,8000-length(y)) y];
>> sound([y_left' y_right'],fs)
```

ヘッドホンで聞けば、それぞれの信号が左右のスピーカーから出ていることが確認できる。

同じ信号を左チャンネルと右チャンネルでレベル差を付けて生成することで、左から右の間の適当な場所に位置付けることができる。この位置のことを定位という。また、左チャンネルと右チャンネルのレベルの差のことを両耳間レベル差 (ILD: Interaural Level Difference) とよぶ。

```
>> y_left=[y z 9/10*y z 1/2*y z 1/10*y z z];
>> y_right=[z z 1/10*y z 1/2*y z 9/10*y z y];
>> sound([y_left' y_right'],fs)
```

19.2 ステレオ信号の操作

CDなどステレオ音源の RIFF 形式 (.wav という拡張子のついたファイル形式) のファイルも MATLAB に読み込むと2列の行列となっていることがわかる。(便宜上、本資料では、ステレオ音源のファイルの名称を song.wav とする)

```
1 >> [y,fs]=wavread('song.wav');
2 >> size(y)
3 ans =
4     13453440         2
5 >> sound(y(:,1),fs)
6 >> sound(y(:,2),fs)
```

4行目の1つ目の数はファイルによって変わること注意。5行目では左チャンネルが、6行目では右チャンネルが再生される。

この曲が、ごく普通のボーカル曲だとすると、メインボーカルは大抵、中央に定位している(ヘッドホンで聞くと頭の真ん中で聞こえる)。中央に定位する音を c とする。この音はILDが0、つまり、右チャンネルにも左チャンネルにも同じように含まれるため中央に聞こえる。

右チャンネルの音声波形は、右チャンネルにしか含まれない音の信号を r とおくと時間波形の重ね合わせなので、 $r+c$ であらわされる。同様に左チャンネルの信号は $l+c$ とあらわすとする。

ここで、左チャンネルの信号から右チャンネルの信号を引く。すると、 $(l+c)-(r+c)=l-r$ となり c が消去される。

この音はどのような音になるか、実際に聞いてみる。

ソースコード 36: 簡単カラオケ

```
>> ll=y(:,1);  
>> rr=y(:,2);  
>> kk=ll-rr;  
>> soundsc(kk,fs)
```

逆にメインボーカルだけを抜くことはできるだろうか? ソースコード 36 のような時間波形の簡単な演算では、なかなか c だけを残すことはできない。

楽曲によっては、オリジナルカラオケが手に入る場合もある。オリジナルカラオケの左チャンネルが前述の l 、右チャンネルが r に近いものであれば、簡単な時間波形の演算で c を得ることができる。

ソースコード 36 で求めた信号 kk は、カラオケに聞こえる。しかし実際には、左チャンネルの演奏と右チャンネルの演奏を足したものではない。したがって、これを使って c を得るのは難しい。

19.3 事例研究: ILD の推定による中央チャンネル分離

ステレオ音源で中央に定位する部分を中央チャンネルとよぶことにする。この中央チャンネルを分離する方法はいろいろと提案されているが、本稿では、“Center Channel Separation based on Spatial Analysis” という論文 ([1] とする) で提案されている手法を紹介する。

この論文では、前述の ILD と同じ ICLD (inter-channel level difference) を利用することで、ある周波数帯域の音が定位する角度を推定しようとしている。ただし、様々な楽器の音がミックスされている音源では、個々の楽器の音のレベル差を直接計算することは難しいので、この論文では、チャンネル間の相互相関 (ICC: inter-channel cross-correlation) を用いて、帯域の定位を推定している。ICC の計算は周波数領域で行なっている ((4),(5) 式) が、この計算方法自体は、この論文の参考文献 [8] で提案されている方法であり、実装するためには、その文献 “Binaural Cue Coding—Part II” ([2] とする) を参照しなければならない。

ここからは、2つの論文を参照するため、[1] の式 (1) は (1-1) のように先頭に 1 を付与し、[2] の式 (1) は (2-1) のように先頭に 2 を付与することにする。

[1] では、中央チャンネル推定のために、帯域ごとに ICC を用いて、その帯域の定位 (角度) を推定し、その角度にしたがって、帯域の信号を、中央 (c) ・左 (l) ・右 (r) チャンネルに分配する。その計算は式 (1-8), 式 (1-9) に示されている。ここで n は帯域の番号であり、 $S_c(n)$ は中央チャンネルの信号、 $S_l(n)$ は左チャンネル、 $S_r(n)$ は右チャンネルの信号である。 $f(1/d)$ は距離の逆数に比例するチャンネル分離のための関数である。 $S_{lt}(n)$ は元のステレオ音源の左チャンネルの n 番目の帯域の信号であり、 $S_{rt}(n)$ は右チャンネルの信号である。

$1/d$ は、変化の傾向としては、ICC と同じようになるため、 $f(1/d)$ は、ICC の値でどの程度中央と両側に分配するかを決めればよい。

そこで、まず ICC について見てみる。ICC は、式 (1-4) で計算される。この式で求められる $\Gamma(n, k)$ は k 番目の時間フレームの n 番目の帯域の周波数領域の ICC である。この Γ は式 (1-5) の左右の信号の帯域ごとの自己相関・相互相関で求められると書かれている。

しかし、これらの式だけでは、周波数領域の相関 Φ はどのように計算するのか、帯域はどのように定義されているのか、式 (1-5) の $\tilde{S}_{i,n}$ をどのように求めるかがわからない。また、 α とは何かもよくわからない。そこで、この部分で引用されている [2] を見てみる。

すると表 1 (TABLE I) より、サンプリング周波数が 32KHz のときに、長さ 1024 で FFT したときに、例えば、1 番目の帯域は、 $A_0, A_0 + 1, \dots, A_1 - 1$ まで、つまり、0, 1 であることがわかる。つまり、1 番目の帯域は、 $0\text{Hz}, 32000/1024 = 31.25\text{Hz}$ に対応している。

Φ の計算は、式 (2-20) で定義されている。ちなみに、 Φ が式 (2-20) のように期待値で定義されるので、式 (2-21) 式の右辺の第 1 項の $\alpha \tilde{S}_{i,n}(k) \tilde{S}_{j,n}^*$ は、 $\alpha E\{\tilde{S}_{i,n}(k) \tilde{S}_{j,n}^*\}$ の間違いであろうと思われる。

ここまでわかれば、MATLAB で実装できるはずである。