

ひたすら楽しんで 音響信号解析

— MATLABで学ぶ基礎理論と実装 —

博士（工学） 森勢 将雅【著】

コロナ社

ま え が き

何十年も前から、偉大なる信号処理研究者たちの手によって、数多の書籍が出版されてきた。Webで検索すれば、信号解析や関連トピックを扱う膨大な数の書籍やWebの解説記事が生まれており、どの本が学びやすいか探すだけでも大変であろう。信号処理の書籍は、入門書、おもに洋書で出版される分厚い専門書、その他諸々に分けられる。本書は、「自分で読みたい教科書を作る」という筆者の情熱を出発点に、信号解析の基礎をひたすら楽して習得することを目指して工夫を凝らしている。

フーリエ変換を回避して高速フーリエ変換 (FFT: fast Fourier transform) によるスペクトル解析を習得せよ、という目標は無謀に映るかもしれない。信号解析を習得するために必要な知識は多いため、解析したい信号があるにもかかわらず延々と基礎勉強を続けることは、苦行に感じることだろう。本書は、このような基礎勉強が苦行という懸念を覆し、プログラムにより実践的な内容に絞って習得することを目指している。逆に考えて、先人たちが実装したプログラムを利用すれば信号解析をするだけなら可能である、と割り切って執筆したことが特色である。

本書は、筆者の専門分野である1次元の音響信号解析に特化し、数式をプログラムで実装することに比重を置いた構成を採用した。これは、教科書に記載された数式をどのようにプログラムに落とし込めばよいのかわからずに躓く学生がこれまで多かった、という筆者自身の経験に基づく。数式と同じ結果が得られるよう理論を厳密に実装することは、アルゴリズムによっては容易ではない。そのため、例となるプログラムを数式に合わせて掲載することで、数式とプログラムの対応付けができることを目指している。信号解析の基礎を扱うという都合上、大学学部2,3年生の講義で用いる教科書を想定している。

ひたすら楽するというコンセプトから、本書はオーソドックスなデジタル信号処理の教科書とは異なる道筋で説明を展開することもある。例えば、前述の高速フーリエ変換を習得するためには、背景にある理論として学ぶべきことが多数存在する。おそらく多くの教科書では、フーリエ級数展開を出発点に複素フーリエ級数展開へ進み、そこからフーリエ変換の解説に進む。その後、離散フーリエ変換 (DFT: discrete Fourier transform) を解説してからようやく高速フーリエ変換の説明に至る。場合によっては、アナログ信号処理に関する基礎やラプラス変換、 z 変換が入るかもしれない。一方本書は、例えば「高速フーリエ変換によるスペクトル解析」を初めにゴールとして定める。ゴールに必要な理論に限定して一切の無駄を省いた説明によりただひたすら楽をする、そんな教科書を意識した。楽をするため一般的な書籍に存在する章末問題すら省き、プログラムの書き写しにより基礎的な信号解析を習得できる構成にした。

もう 1 点の特色は、本書で扱うプログラミング言語に MATLAB を採用していることである。本書を執筆している 2020 年であれば、信号解析を学ぶための筆頭候補に挙がるプログラミング言語は Python であろう。Python は deep learning 分野でシェアを伸ばしているプログラミング言語で、信号解析に関するライブラリも豊富にリリースされている。MATLAB を選んだ理由は筆者の好みであるが、別の理由として、やはり楽をするための完成度の高さが挙げられる。MATLAB はフリーソフトではないが、FreeMat や Octave などの MATLAB クローンがフリーで利用できる。本書に記載した例題は、MATLAB 以外では実装されていない関数や乱数を用いる場合を除き、MATLAB (2020a) と FreeMat (v4.2) でほぼ同じ出力が得られることを確認している。信号解析の入門に用いる言語として、たぶん MATLAB が一番楽だと思います。

最後に、本書を執筆する機会を与えてくださった皆様に深謝します。また、相変わらず年中自由気ままな時間に執筆することを許してくれた妻に感謝します。

2020 年 12 月

森勢 将雅

目 次

1. 基礎知識

1.1 離散信号と波形の表示	1
1.2 ベクトルと内積	5
1.3 正 弦 波	6
1.4 複 素 数	8
1.5 オイラーの公式	10
1.6 単位インパルス関数と畳み込み演算	12

2. 時間領域での信号解析

2.1 離散信号を扱う上での注意点	17
2.1.1 数式上の辻褄合わせ	17
2.1.2 離散信号固有の性質	19
2.2 信号のパワー	22
2.3 信号の平均時間と持続時間	26
2.3.1 信号の平均時間	28
2.3.2 信号の持続時間	29
2.4 信号の簡単な雑音除去	32
2.4.1 SN 比に基づく信号・雑音の振幅調整	32
2.4.2 移動平均フィルタ	34
2.4.3 メディアンフィルタ	36

3. 離散フーリエ変換の考え方

3.1 正弦波に関する事前知識	39
3.1.1 正弦波を構成するパラメータの定義	40
3.1.2 正弦波を構成するパラメータの推定	42
3.1.3 パラメータ推定のプログラムによる検証	43
3.1.4 周期の拡張	44
3.2 スペクトル解析に関する事前知識	46
3.2.1 正弦波の周波数推定	46
3.2.2 正弦波の重ね合わせ	48
3.3 離散フーリエ変換に向けた事前知識	49
3.3.1 離散信号を用いた場合に対する制約	50
3.3.2 複素数による係数の統合	53
3.3.3 係数を計算する範囲の注意点	55
3.4 離散フーリエ変換	58
3.4.1 離散フーリエ変換の定義	59
3.4.2 高速フーリエ変換によるスペクトル解析	60

4. 高速フーリエ変換によるスペクトル解析

4.1 簡単なスペクトル解析の例	62
4.1.1 振幅スペクトルと位相スペクトルの定義	62
4.1.2 正弦波のスペクトル解析	63
4.1.3 ゼロ埋めによる信号長の調整	64
4.1.4 パーセバルの定理による漏れ誤差の解釈	67
4.2 位相スペクトル解析の例	69
4.2.1 単位インパルス関数を対象にした位相スペクトルの解析	70
4.2.2 位相のアンラップ	72
4.2.3 位相遅延	73

4.2.4 群 遅 延	75
4.3 スペクトルから算出する平均時間と持続時間	78
4.3.1 平均時間の算出	78
4.3.2 持続時間の算出	79
4.3.3 スペクトル解析に基づく解釈	81
4.4 スペクトル重心	82

5. 窓 関 数

5.1 窓関数による信号の切り出し	85
5.1.1 離散フーリエ変換の周期性に起因する問題	85
5.1.2 窓関数による切り出しの数学的な解釈	89
5.1.3 離散信号に対する畳み込み定理	91
5.2 窓関数を導入する意義	93
5.2.1 窓関数の種類	94
5.2.2 窓関数を用いて切り出した信号に基づくスペクトル重心の計算	97
5.3 窓関数の性能	98
5.3.1 メインローブとサイドローブ	98
5.3.2 時間周波数解析のための窓関数	102
5.4 時間周波数解析例で学ぶ時間分解能と周波数分解能	104
5.4.1 チャープ信号の生成とスペクトル解析	104
5.4.2 チャープ信号の時間周波数解析	105
5.4.3 時間分解能と周波数分解能	108
5.5 不確定性原理	110

6. デジタルフィルタ

6.1 線形時不変システム	114
6.1.1 システムを構成する3要素	115
6.1.2 線形性と時不変性	116

6.1.3	線形時不変システムをおもに用いる理由	117
6.2	デジタルフィルタの概要	118
6.2.1	フィルタ処理の数学的解釈	118
6.2.2	フィルタの種類	119
6.2.3	その他のフィルタ	121
6.3	差分方程式から見る2種類のフィルタ	121
6.3.1	差分方程式と FIR フィルタ	122
6.3.2	IIR フィルタ	124
6.3.3	IIR フィルタのスペクトル解析	126
6.3.4	フィルタの接続	129
6.4	フィルタの解析	130
6.4.1	z 変換によるフィルタの安定性解析	130
6.4.2	不安定なフィルタの例	133
6.4.3	フィルタの因果性	133
6.5	窓関数法によるフィルタの設計	134
6.5.1	振幅スペクトルの設計	135
6.5.2	インパルス応答の計算	136
6.5.3	振幅特性の検証	138
6.5.4	窓関数による処理	139
6.5.5	最終的な性能の評価	141
6.5.6	最小位相の例題	142

7. 信号の種類に応じた解析法

7.1	間引きによるダウンサンプリング	146
7.1.1	単純に間引く場合の問題点	146
7.1.2	低域通過フィルタによる高域成分の除去	148
7.1.3	時間軸の補正	150
7.1.4	間引きによる適切なダウンサンプリング	153
7.2	オクターブバンド分析	153
7.2.1	オクターブバンド	153

7.2.2	オクターブバンド分析の実装	155
7.3	等価騒音レベルの計算	158
7.3.1	正弦波の周波数と知覚する大きさの関係	159
7.3.2	等価騒音レベルの計算法	160
7.4	ウェルチ法	162
7.4.1	振幅スペクトルのばらつき	163
7.4.2	ウェルチ法によるパワースペクトル推定	163
7.4.3	ウェルチ法の効果	164
7.5	周期信号解析に向けた基礎知識	166
7.5.1	周期信号の定義と構成要素	167
7.5.2	周期信号の生成	169
7.6	基本周波数推定	170
7.6.1	自己相関関数の定義	171
7.6.2	解析全体のアルゴリズムとプログラム例	171
7.6.3	推定プログラムの構成	173
7.7	調波間の振幅比	175
7.8	今後の勉強に向けて	177
	引用・参考文献	179
	索引	182

MATLAB は MathWorks, Inc. の登録商標です。本書では、MATLAB およびその他の製品名に TM, ® マークは明記していません。

本書を発行するにあたって、記載内容に誤りがないように可能な限り注意を払いましたが、本書の内容を適用した結果生じたこと、また、適用できなかったことに関して、著者、出版社とも一切の責任は負いませんのでご了承ください。

1. 基礎知識

本書では、数式とプログラムを多数掲載する。プログラムは MATLAB で動作するものであり、例えば以下のように記載する。

```
>> x=1;
```

ここで、>>の後に記載されているプログラムは、読者がコマンドウィンドウに打ち込むことを想定している。本書を読み進めるにあたり、数式を解きながら併せて記載されたプログラムも順次実行することで、数式とプログラムとを対応付けて学べるように工夫した。基本的に各プログラムは単独で動作するようにしているが、記述を短縮するため一部関数化することもある。本書では、MATLAB に関する最低限の知識は備わっていることを前提にしているため、細かい仕様について解説はしない。また、いわゆる MATLAB 的な記載により短いプログラムで高速に動作させることも可能であるが、可読性と数式との対応を意識した実装を優先する。各セクションのタイトルから想像できる内容を理解できていると感じ、MATLAB の知識も十分にある読者は、楽をするため 1 章をスキップしても問題ない。

1.1 離散信号と波形の表示

現実世界に存在する音は連続信号 (analog signal) であり、離散信号 (digital signal) に変換後、計算機に取り込まれる。この操作を **A-D 変換** (Analog-to-Digital conversion や A-D conversion とも) と呼び、A と D は連続信号と離散信号の英語表記に由来する。A-D 変換の結果、計算機上では単なる数字の羅列として音を記録していることになる。例えば、記録された数値が、先頭から順に「1, 3, -5, 2」であったとしよう。これを MATLAB の配列に格納する場

2 1. 基礎知識

合のプログラムは以下となる。

```
>> x=[1,3,-5,2];
```

ここで、各数字は計算機に記録された音響信号の振幅に相当する。末尾にセミコロンを付けているが、MATLAB ではセミコロンを外してもエラーとはならない。セミコロンを外した場合は、計算結果がコマンドウィンドウに表示される。おもにデバッグにおいて、計算結果を逐次確認する際に役立つ仕様である。

現実世界の時刻とプログラム上で格納する配列の添え字については注意が必要なため、本書のコンセプトである数式とプログラムを対応付ける観点からも少し説明を加える。上述の例で、先頭の振幅 1 を記録した時間を考えることにする。「何月何日の何時何分何秒…」という時刻に記録したことは事実だとしても、計算機上でファイル保存する際に、すべての振幅にこのような時刻を記録することはまずない。多くの場合、絶対的な時刻を持たず振幅値のみが順番に保存されることになり、先頭の振幅を原点 (0 秒) における値として扱う。このイメージは、図 1.1 のようになる。

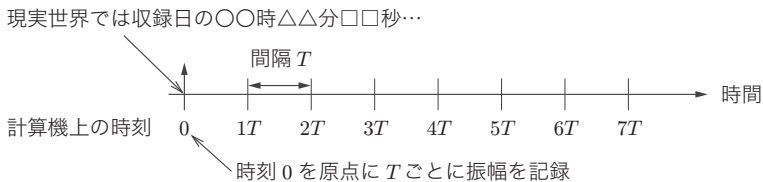


図 1.1 現実世界の時刻と計算機上の時刻との対応付け

今回の例では、振幅値 1 を記録した時刻が原点である。C 言語などのプログラミング言語では、配列の先頭は $x[0]$ であり 0 番目としてアクセスする。一方、MATLAB における配列は先頭が 1 番目であるため、 $x(1)$ としてアクセスする必要がある。MATLAB を使う際には、この差がバグの原因にもなるので特に注意が必要である。なお本書では、上記のように、数式として扱う変数とプログラムとは異なるフォントで記載する。

図 1.1 で示したように、A-D 変換では、振幅の記録を特定の時間間隔 T ごとに実施している。この時間間隔のことを標準化周期 (sampling period) と呼

ぶ。この逆数、つまり 1 秒当たりの振幅値の数に相当する値が標準化周波数 (sampling frequency でサンプリング周波数ともいう) である。振幅を 10 ms ごとに記録している場合、標準化周期は 10 ms、標準化周波数は 100 Hz となる。

波形を表示する際は、横軸の単位を秒などの時間にすることが望ましい。MATLAB で波形を表示する代表的な方法は `plot` 関数であり

```
>> plot(x);
```

と記載する。plot 関数では、引数が一つの場合、横軸を 1 から 4 サンプルと見なした数字で表示する。以下のプログラムでは、標準化周波数を 10 Hz に設定し、横軸を秒を単位とした時間軸として波形を表示する。

```
>> fs=10;
>> t=(0:3)/fs;
>> plot(t,x);
```

`fs` は標準化周波数に対応し、また、この変数名は慣習的に標準化周波数を記述するために用いられる。`t` は n 番目の振幅が何秒の値であるかを記録する配列である。この例では、振幅の先頭を原点として扱っており、先頭の振幅から時刻 0, 0.1, 0.2, 0.3 秒に対応する。図 1.2 は、横軸をサンプルにした場合と、秒にした場合との表示の比較である。今回のプログラムでは省略しているが、横軸

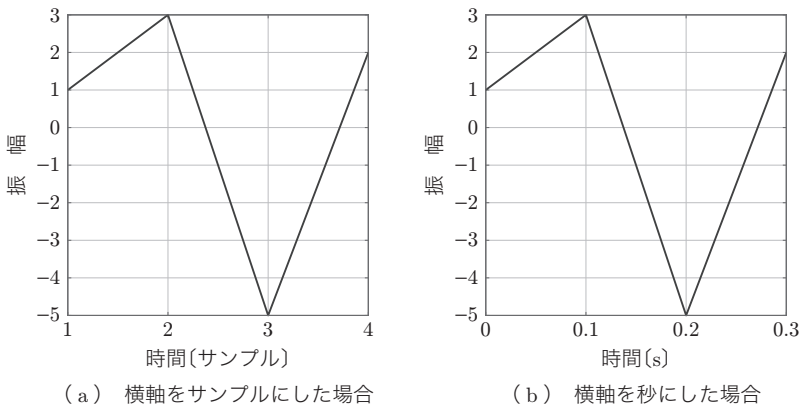


図 1.2 横軸の単位をサンプルと秒にして表示した 4 サンプルの振幅

4 1. 基礎知識

と縦軸のラベルは、それぞれ `xlabel` 関数と `ylabel` 関数により表示している。

余談であるが、MATLAB でプロットした図のフォントのデフォルトの大きさは、2 カラムの論文に掲載することを想定すると小さい。フォントサイズは毎回修正することもできるが、MATLAB 起動時に自動的に実行されるスクリプトである `startup.m` を活用すれば、デフォルトのフォントサイズを変更できる。具体的には、MATLAB 起動時に

```
>> edit startup.m
```

とコマンドを打ち、エディタで

```
set(0,'DefaultAxesFontSize',14);  
set(0,'DefaultTextFontSize',14);
```

と打ち込んで保存する。コマンドウィンドウに入力しないため、`>>` は付していない。その後、MATLAB を再起動すれば、図のフォントサイズがつねに 14 で表示される。この処理は、1 回行えば以後恒久的に利用できる。MATLAB 起動時のフォルダを変更した場合は、同様の処理を再度行えばよい。

波形を表示する際には、横軸は時間（秒やミリ秒）、縦軸は振幅にすることが推奨される。振幅は、A-D 変換時のマイクボリュームによって同じ大きさの音でも離散信号としては異なる値になるため、単位を記載しない。具体的には

```
>> xlabel('Time (s)');  
>> ylabel('Amplitude');
```

とする。英語で記載する際、単語と括弧や単位との間に半角スペースを空ける必要があることに注意しよう。本書では図のキャプションを日本語にしているが、論文に記載することを意識してここでは英語で記載している。

最後に、本書における数式の表現方法について記載する。A-D 変換前の信号は $x(t)$ とし、A-D 変換後の信号は $x[n]$ と表すことにする。この二つの信号は

$$x[n] = x(nT) \quad (1.1)$$

という関係で結びつく。角括弧は、括弧の中身が整数の場合に使うことで離散

信号であることを明確に示す。また、離散信号の表現では、一部、数列を表現する際に用いる表記として x_n を用いることがある。この使い分けはほかの教科書等との整合性を意識しており、本書において数学的な意味での差はない。

1.2 ベクトルと内積

ある MATLAB の変数が N サンプルの振幅の時系列を持つ場合、 N 次元のベクトル (vector) であると表現する。MATLAB では行列演算を簡単にできるように、配列では横ベクトル (row vector) と縦ベクトル (column vector) を区別する。例えば、以下のプログラムにおいて、 \mathbf{x} は横ベクトル、 \mathbf{y} は縦ベクトルとして格納される。

```
>> x=[1,3,-5,2];
>> y=[1;3;-5;2];
```

二つのベクトルの次元数が同じであれば、つぎの式により内積 (inner product) が計算できる。プログラムと数式ではフォントが変わっているが、同じアルファベットであるものはプログラムと数式が対応する。

$$\vec{x} \cdot \vec{y} = \sum_{n=0}^{N-1} x[n]y[n] \quad (1.2)$$

ここで、 $\vec{x} = (x[0], x[1], x[2], \dots, x[N-1])$ とし、 \vec{y} も同様に N 次元のベクトルとする。この数式では行列の縦横は区別せず、 $x[n]$ はベクトルの n 番目の要素を参照している。 \vec{x} の記載は教科書により異なることもあるが、本書では上記の書き方を用いる。また、内積は以下の式で定義されることもある。

$$\vec{x} \cdot \vec{y} = |\vec{x}| |\vec{y}| \cos \theta \quad (1.3)$$

表記の差はあれど意味するところは同じであるため、見かけの記述に惑わされずに理解してほしい。

内積を MATLAB で計算する場合、いくつかの選択肢がある。一番直接的な方法は、for 文を使い総和記号をそのまま実装するやり方であろう。ここでは、

索引

【あ】	
アンチエイリアシング フィルタ	51
安定性	130
【い】	
位相 ——のアンラップ	40 72
位相スペクトル	62
位相遅延	73
移動平均フィルタ	34
因果性	133
インパルス応答	118
【う】	
ウィーナー・ヒンチンの定理	171
ウェルチ法	162
【え】	
エネルギー	28
【お】	
オイラーの公式	10
オクターブ	153
オクターブバンド分析	82, 153
折り返し	52, 148
【か】	
ガウス関数	113
角周波数	20
加算器	115
カットオフ周波数	119

【き】	
基音	168
基本周期	168
基本周波数	168
極	132
虚数単位	9
虚部	8
【く】	
矩形窓	94
群遅延	69, 75, 148
【け】	
減衰域	120
【こ】	
高域通過フィルタ	119
高速フーリエ変換	39, 60
高調波	168
コーシー・シュワルツの 不等式	112
【さ】	
最小位相	137
最小位相応答	142
サイドローブ	98
サイドローブレベル	99
差分方程式	122
三角関数	6
【し】	
時間周波数解析	102
時間分解能	104
自己相関	171

自己相関法	171
持続時間	26, 78, 79, 110
実部	8
時不変システム	116
時不変性	116
時変システム	117
周期信号	40, 166
収束	130
周波数分解能	104
純音	6
巡回畳み込み	92
循環畳み込み	93
乗算器	115
振幅	40
振幅スペクトル	62
【す】	
スペクトル	59
スペクトル解析	10, 45
スペクトル重心	82, 97, 101
スペクトル包絡	175
スペクトル漏れ	94
スペクトログラム	103
【せ】	
正規乱数	24
正弦	6
正弦波	6
ゼロ位相	136
ゼロ埋め	60, 64, 87, 103
ゼロ点	132
遷移域	120
全域通過フィルタ	121
線形システム	116
線形時不変システム	114

線形性	116	転置	6	平均時間	26, 78
【そ】		【と】		ベクトル	5
騒音レベル	158	等価騒音レベル	158	【ほ】	
相対パワー	63	等ラウドネスレベル曲線	158	ホワイトノイズ	23
阻止域	120	【な】		【ま】	
【た】		ナイキスト周波数	52	窓関数	67, 85
帯域阻止フィルタ	120	内積	5, 43	窓関数長	103
帯域通過フィルタ	120	【は】		窓関数法	121, 134
ダウンサンプリング	146	【は】		【め】	
畳み込み	13, 118	倍音	168	メインローブ	98
畳み込み積分	13	パーセバルの定理	67, 111	メディアンフィルタ	36
畳み込み定理	91, 119	発散	130	【も】	
畳み込み和	13	ハニング窓	94	漏れ誤差	67
縦ベクトル	5	ハミング窓	94	【よ】	
単位インパルス関数	12, 70	パワースペクトル	63	余弦	6
短時間フーリエ変換	102	【ひ】		横ベクトル	5
【ち】		非周期信号	167	【ら】	
遅延素子	115	非線形システム	116	ラウドネス	158
チャープ信号	104	標本化周期	2	ラジアン	7
調波間振幅比	176	標本化周波数	3	【り】	
調波構造	167	【ふ】		離散周波数番号	59
直線位相	73, 137	不確定性原理	110	離散信号	1
直線畳み込み	93	複素共役	55	離散フーリエ変換	59
直流成分	23	複素数	8	離散フーリエ変換	39, 59
直交関数列	47	複素正弦波	12	リップル	138
【つ】		複素平面	8	両対数グラフ	165
通過域	120	ブラックマン窓	94	【れ】	
【て】		プリエコー	142	連続信号	1
低域通過フィルタ	119	フーリエ変換	39, 75		
デジタルフィルタ	114	フレームシフト	103		
ディラックのデルタ関数	12	ブロック図	114		
デシベル	22	【へ】			
伝達関数	119	平滑化	34		

【A】		abs 関数	9	atan2 関数	44
A 特性	160	A-D 変換	1	axis 関数	108
		angle 関数	9, 71		

	[B]		[I]	set 関数	22
blackman 関数	95	ifft 関数	136	sin 関数	7
	[C]	IIR フィルタ	124	sinc 関数	90
ceil 関数	66	imag 関数	9	size 関数	6
colorbar 関数	108	imagesc 関数	107	SN 比	32
colormap 関数	107			sound 関数	7
conj 関数	55	[L]		soundsc 関数	8
conv 関数	14	legend 関数	97	startup.m	4
cumsum 関数	157	loglog 関数	165	stem 関数	87
	[E]			sum 関数	6
exp 関数	10	[M]			
	[F]	mean 関数	23, 165	[T]	
fft 関数	60, 63, 64	median 関数	36	title 関数	21
fftshift 関数	99			[U]	
FIR フィルタ	122	[O]		unwrap 関数	72
floor 関数	66	ones 関数	135		
fprintf 関数	29			[X]	
	[G]	[P]		xlabel 関数	4
grid 関数	21	plot 関数	3	xlim 関数	22
	[H]	[R]		[Y]	
hamming 関数	95	rand 関数	37	ylabel 関数	4
hanning 関数	95	randi 関数	37	ylim 関数	22
		randn 関数	24		
		real 関数	9	[Z]	
		round 関数	37	z 変換	123, 126
				zeros 関数	61
		[S]			
		semilogx 関数	158		

— 著者略歴 —

2002年 釧路工業高等専門学校情報工学科卒業
2004年 和歌山大学システム工学部デザイン情報学科卒業
2006年 和歌山大学大学院システム工学研究科博士前期課程修了
2008年 和歌山大学大学院システム工学研究科博士後期課程修了（短期修了），博士（工学）
2008年 関西学院大学博士研究員
2009年 立命館大学助教
2013年 山梨大学特任助教
2017年 山梨大学准教授
2019年 明治大学准教授
現在に至る

ひたすら楽して音響信号解析

— MATLAB で学ぶ基礎理論と実装 —

Beginning Acoustic Signal Analysis

— Fundamental Theory and Its Implementation with MATLAB —

© Masanori Morise 2021

2021年2月12日 初版第1刷発行



検印省略

著者 森 勢 将 雅
発行者 株式会社 コロナ社
代表者 牛来 真也
印刷所 三美印刷株式会社
製本所 有限会社 愛千製本所

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社
CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844 · 電話 (03) 3941-3131(代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-00939-2 C3055 Printed in Japan

(松岡)



< 出版者著作権管理機構 委託出版物 >

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつと事前に、出版者著作権管理機構（電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp）の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。