

標準 MCAP-CR 型スピーカーシステムシミュレーションプログラム説明書

2012 年 1 月 29 日

鈴木 茂

0. 配布の目的

このプログラムは、バスレフ型スピーカーシステムの動作について技術的な興味を持つ人を対象として配布します。

このプログラムで使用したアルゴリズムは、質点の運動方程式モデルであり、スピーカー振動板の連続体としての動作は考慮していません。このため、モデルの評価には、初級程度の物理知識が必要です。

個人的な感覚での音の良否でしか物事の価値を判断できない人の使用はお断りします。

1. 使用許諾条件

このプログラムの使用者は、以下の条件に同意したものとします。

- i. このプログラムの元になった数式モデル及びアルゴリズムの限界について理解したこと。
- ii. プログラムには未知のバグが含まれることを理解したこと。
- iii. このプログラムに存在するバグ、及び、プログラムを使用した結果によるいかなる損害、不満等についても賠償を請求しないこと。
- iv. このプログラム及びその結果を、商用に使用しないこと。
- v. このプログラム及び、ユーザー自身が改変したプログラムを、著作権者の書面による許可無く再配布しないこと。
- vi. プラットフォームに依存しない一貫性を維持するため、Microsoft Visual C++用のコードに変更して使用しないこと。
- vii. 音の良し悪しの評価には使わないこと。

2. プログラムの機能

このプログラムは、標準 MCAP-CR 型スピーカーシステムの振動板とダクト内の空気塊を質点として、それぞれの変位を計算する機能を有します。

入力は、電気信号ではなく、機械的加振力とし、正弦波、リニアスイープ、ランダム波形での応答を計算します。

計算結果をテキストファイルとして作業フォルダに出力します。

詳細は、プログラムソースコード `mcap-cr_simulator_public_rev01.cpp` を解読してください。

注：プログラムを改変する場合

使用許諾条件の範囲内でのプログラムの改変は自由です。ソースコードには、適宜コメントが付いているので、参考にしてコードを変更してください。

3. 使用方法

このプログラムは、ソースコードの形で配布しているので、ユーザ自身がコンパイルして実行ファイルを作成することが必要です。この点が、通常のソフトウェアと異なりますのでご注意ください。

プログラムは C 言語（一部に C++ 言語の用法を含む）で記述されているため、コンピュータのオペレーションシステムに関係なく、コンパイル出来るシステムであれば使用できます。

ソースコードのコンパイルの確認は、Linux GNU Compiler Collection(GCC)、Windows に GCC を移植した MinGW、Borland C++ Compiler V5.5 を使用して行なっています。

Mac OS は、総合開発環境をセットアップしてください。無償でダウンロードできるそうです。

Windows は、MinGW または Borland C++ Compiler V5.5 をインストールできるバージョンであれば、コンパイラをセットアップ後、使用することが可能と思います。Microsoft Visual C++は、標準言語のまま使用するには適さないので推奨しません。

Free BSD、UNIX や VMS で使用する場合には、おそらくそのままコンパイルして使用することが可能ですが確認はしていないので、ユーザ自身で確認してください。

参考リンク(無償コンパイラの入手先)

GNU Compiler Collection	http://gcc.gnu.org/
MinGW Projects	http://www.mingw.org/

3.1 コンパイラの準備

Linux の場合

GNU C Compiler (GCC) をインストールする。既にインストールされている場合はそのまま使用できます。下記コマンド (rpm 系の場合) 等で確認してください。↵ は Enter キーの押下を表します (以下同様)。

```
$ rpm -qa|grep gcc↵
```

Windows の場合

標準では、C コンパイラはインストールされないの、MinGW(推奨)または、Borland C++ Compiler 等をダウンロードしてインストールします。インストール後にはパスの設定が必要です。エラーがあった場合の対策はインターネットで簡単に検索できますが、MinGW のほうが簡単です。

3.2 プログラムのコンパイル方法

(Linux の場合)

- 以下は全てのディストリビューションで共通です -
解凍したソースコードを、作業ディレクトリにコピーする。
ターミナルを起動し、作業ディレクトリに移動する。
gcc コマンドを入力して、コンパイルする。
- 以下はディストリビューション (GCC のバージョン) により多少異なります -

Vine Linux 6.0 の場合

```
$ gcc -lm -lstdc++ -o mcaps.out mcap-cr_simulator_public_v01.cpp↵
```

ここで、"mcaps.out" は、実行ファイル名で、OS が許す範囲内で好きな名前を付けることができます。
"-o mcaps.out" の部分は省略可能で、その場合には、"a.out" という実行ファイルが出来ます。

OpenSUSE 12.1 の場合

```
$ gcc -lm -o mcap-cr_simulator_public_v01.cpp↵
```

この場合も同様に -o オプションは省略可能です。

(Windows 7 の場合)

解凍したソースコードを、作業ディレクトリにコピーする。
コマンドプロンプトを起動し、作業ディレクトリに移動する。

MinGW を使用してコンパイルする場合

下記コマンドで、コンパイルします。

```
gcc -lm -o mcaps.out mcap-cr_simulator_public_v01.cpp↵
```

Linux の場合と同様、-o オプションは省略可能で、その場合には、"a.exe" という実行ファイルが出来ます。

Borland C++ Compiler の場合

下記コマンドでコンパイルします。

```
bcc32 mcap-cr_simulator_public_v01.cpp↵
```

このとき下記のファイルが出力されますが、*.exe 以外は無くても実行が可能です。

“mcap-cr_simulator_public_rev01.exe”	実行ファイル
“mcap-cr_simulator_public_rev01.obj”	使用しない
“mcap-cr_simulator_public_rev01.tds”	使用しない

3.3 プログラムの実行方法(Linux/Windows 共通)

3.3.1 条件設定ファイルの作成

テキストエディタで、設計条件設定ファイル”input_parameters.txt”を、プログラムの実行ファイルと同じディレクトリに作成してください。付属の例に従い、条件を入力する。このファイルが無ければプログラムを実行しても計算できません。

input_parameters.txt の例	
<p>データの時には、如何なる文字、タブ、改行も含まないこと。 改行コードは何でも良いので、Windows の Notepad でも作成可能。 データ間は 1 個のタブで区切る。2 個のタブが入ると、無のデータを読み込み正しく計算できない。 データには ASCII 文字の数字のみ使用できる。 7 行目以下はプログラムが読み込まないので何でも書ける。サンプルファイルには、データの説明が付いている。</p>	
<pre>7.0 5.47 45 3 10 32 1 0.1 15 10 14 16 21.16 21.16 21.16 15.21 15.21 15.21 50 92 110 120 150 240 0 0 0 0 0 0 0 // -- Below this line does not affect calculation. // 1st row: m0[g], radius of membrane[cm], f0[Hz] // 2nd row: number of chambers(n), cycles, division, thermal condition, amplitude of force // 3rd row: V[0], ..., V[2n] in litre // 4th row: A[1], ..., A[2n] in sq-cm // 5th row: L[1], ..., L[2n] in mm // 6th row: C[0], ..., C[2n] in kg/s</pre>	
1 行目	m0(グラム単位)、実効振動半径(cm 単位)、f0(Hz 単位)
2 行目	副空気室の数(1~12 の整数)、計算するサイクル数、1 周期の分割数、熱力学条件(1.0:等温条件、1.4:断熱条件)、加振力の振幅(ニュートン単位)
3 行目	各空気室の容量(リットル単位)
4 行目	各ダクトの断面積(平方センチメートル単位)
5 行目	各ダクトの有効長(ミリメートル単位)
6 行目	減衰係数(kg/s 単位)

計算アルゴリズムの内容と各パラメータの意味は、下記文書等を参照してください。

http://mcap.web.fc2.com/documents/MCAP008J_S-MCAP-CR_simulation_r0.pdf

但し、上記の文書後にプログラムの改変が行われているので、アルゴリズムの確認のみに使用してください。

注

計算するサイクル数は、多くとも 10 以下とすることをお勧めします。多すぎるとデータの後処理に時間がかかります。

1 周期の分割数は、大きくても 128(7 ビット)以下にすることをお勧めします。通常は 32(5 ビット)で十分です。分割数を大きくすると、ある程度計算誤差を減らすことは出来ませんが、扱うファイルのサイズが増え、計算後の解析に支障をきたします。

3.3.2 プログラムの実行

プログラムのコンパイルが完了し、実行ファイルが出来た場合には、実行ファイルと条件設定ファイルをプラットフォーム(OS)が同じ別のPCにコピーして使用することが出来ます。

プログラムを実行するためには、ターミナル(Windowsの場合はコマンドプロンプト)を開き、プログラムの実行ファイルが存在するフォルダに移動します。

Linux の場合(実行ファイル名を”mcaps.out”とする)

\$./mcaps.out

Windows の場合(実行ファイル名を”mcaps.exe”とする)

mcaps

解析のオプション

解析には、以下のオプションがあります。

(1) vector sum option

結果をフーリエ変換用プログラムで更に解析する場合に、各変位の和に、内部ダクトの変位を加算するかしないかの選択をします。ここで、0を入力すると、全てのダクトの変位を加算し、1を入力すると、内部ダクトの変位を加算しません。ダクトの中の空気塊が完全に質点の場合は1を選択すべきですが、空気の場合は、内部ダクトの変位による音波も外部ダクトを通して出てくるため、0を選択することにも意味はあります。

このオプションは、フーリエ変換用の出力(dftv.txt)にのみ適用されます。

```
ssuzuki@linux-qo0c:~/simulator> ll
total 48
-rwxr--r-- 1 ssuzuki users 462 1月 28 16:00 input_parameters.txt
-rwxr--r-- 1 ssuzuki users 13765 1月 28 16:45 mcaps-cr_simulator_public_rev01
.cpp
-rwxr-xr-x 1 ssuzuki users 25786 1月 28 17:43 mcaps.out
ssuzuki@linux-qo0c:~/simulator> ./mcaps.out
[0] values and parameters calculation done!
Choose vector sum option for later Fourier Transform analysis.
      0: All ducts, 1: ignore internal ducts
Your choice = 0
```

(2) 入力信号選択

単一周波数の正弦波、リニアスイープ、ランダム波形を選択出来ます。

0: 単一周波数の正弦波

1: リニアスイープ

2: ランダム波形

Choose Sinusoidal, Sweep or Random

Sweep option may take long to calculate.

0:Sinusoidal, 1:Linear Sweep, 2:Random

Your choice = 0

0:Sinusoidal

周波数を[Hz]単位で数字のみ入力し、Enter キーを押してください。その後計算を実行し、計算が終わったところでプログラムが終了します。

1:Linear Sweep

この後、プログラムの指示に従って、スイープの完了周波数[Hz]とスイープ速度[Hz/s]を入力してくだ

さい。

下記を参考に必要な数字を入力してください。

Maximum Frequency: 100 から 1000 の範囲に限定されます。通常は 300Hz で十分です。この範囲外の数字を入力すると自動でリミットがかかります。

Sweep Rate: 10～200 の範囲に限定され、この範囲外の数字を入力するとリミットがかかります。最大周波数とスイープ速度を同じ数字にするよう推奨されますが、これは、後のフーリエ解析(今回の配布範囲外)において、解像度を 1Hz にするためなので、それ以外でも実行には支障がありません。

注: スイープ速度を遅くすると、プログラムの実行に時間がかかり、出力ファイルのサイズが大きくなります。出力ファイルのサイズが大きすぎると、普通のソフトウェアでは処理することが出来なくなります。

2:Random

このオプションを選ぶと、サンプリング速度(100×1 周期の分割数)でランダムな信号を与えます。1 周期の分割数は、3.3.1 の条件設定ファイルで指定した値になります。この後は、計算が自動で始まり、必要な処理を終えたところで終了します。

これ以外の入力を与える場合には、コードを追加してコンパイルし直してください。

3.3.3 計算結果の確認

結果の出力

計算の結果、作業フォルダ内に、表1に示すファイルが出力されます。

表1 計算結果の出力ファイル

出力ファイル名	詳細	備考
x_vectors.csv	時刻[s],加振力[N],振動板変位[mm],ダクト変位[mm]...の順にテキストとして出力されます。	
v_vectors.csv	時刻[s],加振力[N],振動板速度[m/s],ダクト変位速度[m/s]...の順にテキストとして出力されます。	
normalized_x_vectors.csv	時刻[s],変位の加重和[mm],振動板の変位[mm],ダクト変位を振動板面積との比で補正した値[mm],...がテキストとして出力されます。	正規化は下記の処理としています。
normalized_v_vectors.csv	時刻[s],速度の加重和[m/s],振動板の速度[m/s],ダクト変位速度を振動板面積との比で補正した値[m/s],...がテキストとして出力されます。	正規化は下記の処理としています。
matrices.csv	標準剛性行列、剛性行列、途中計算のための行列が出力されます。	結果に疑問がある場合に確認します。
dftv.txt	1 行目にデータ数、2 行目に時間の解像度、3 行目以降に速度の正規化された補正和を出力します。 vector sum option の結果が反映されます。	フーリエ解析に使用します。

正規化の処理方法は次式の通りです。正確に表現すれば正規化ではありませんが、振動板面積を基準として、各ダクトの変位及び速度の効果を補正しています。

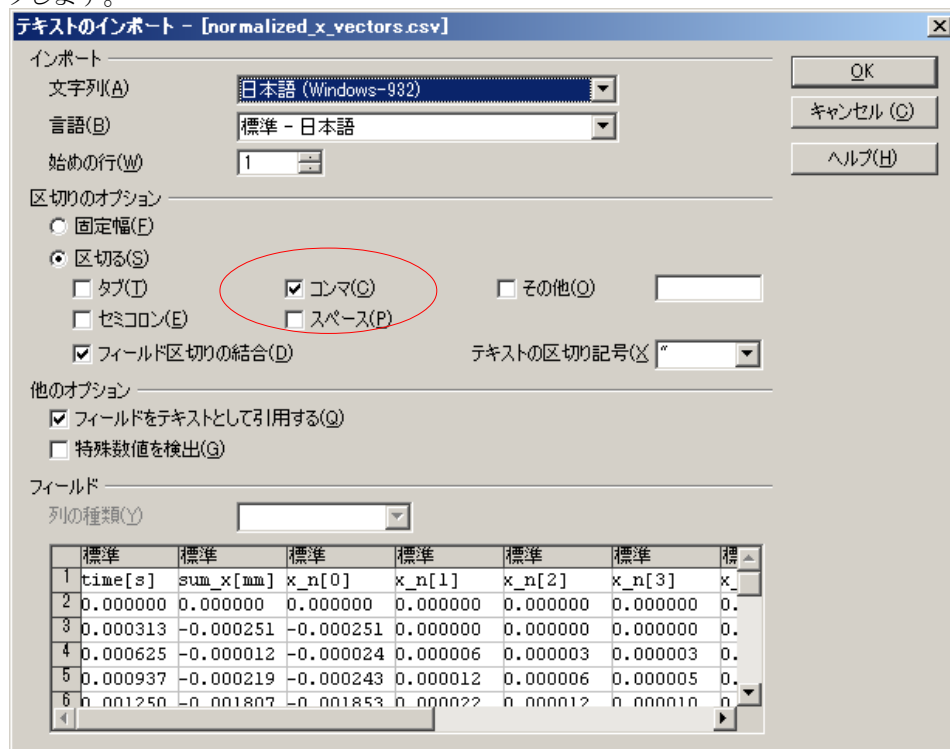
$$x_j^* = r_j x_j = \frac{a_j}{a_0} x_j \quad \text{変位の正規化} \quad v_j^* = r_j v_j = \frac{a_j}{a_0} v_j \quad \text{速度の正規化}$$

グラフで表示する方法

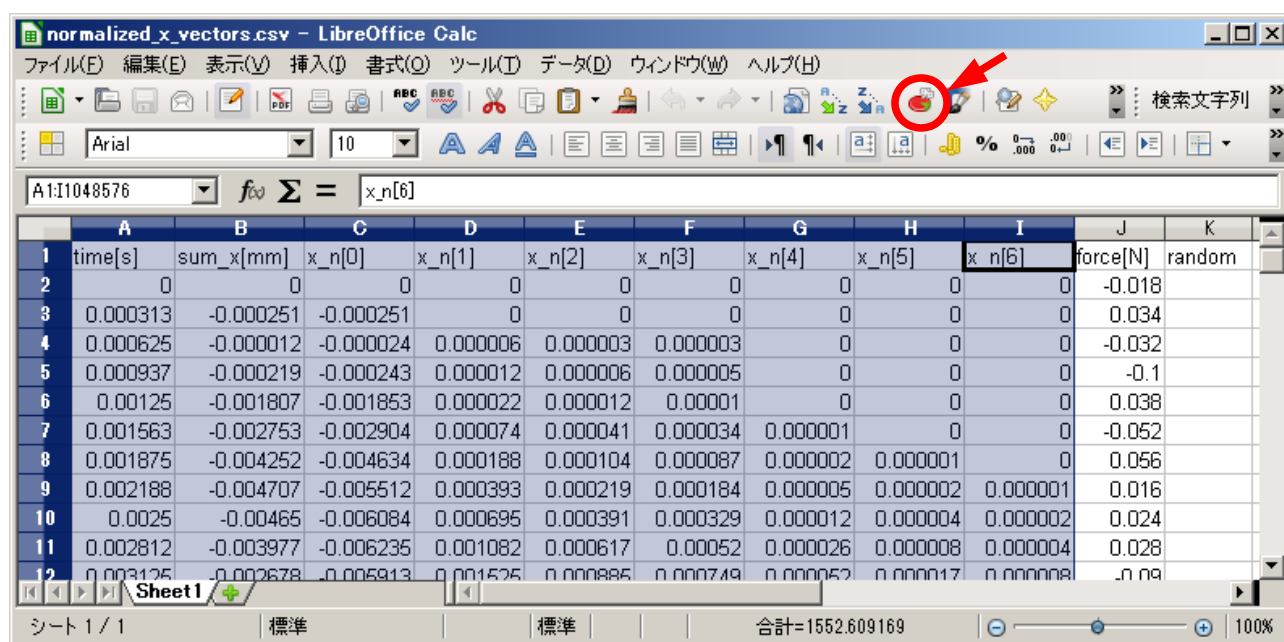
このプログラムにはグラフで表示する機能はありません。LibreOffice Calc 等を使用してグラフ表示をしてください。

LibreOffice Calc を使用する例

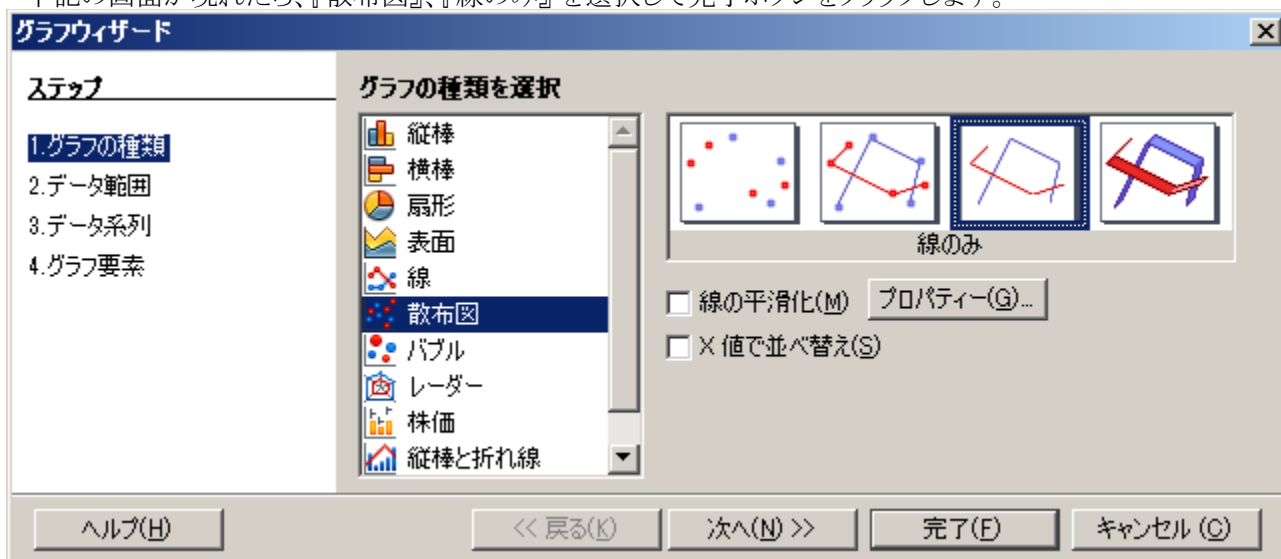
エクスプローラ(Windows の場合)で、作業フォルダを開き、normalized_v_vectors.csv をダブルクリックします。ソフトウェアが起動し、下記の画面が表示されたら、区切り文字に『コンマ(C)』を選択して OK ボタンをクリックします。



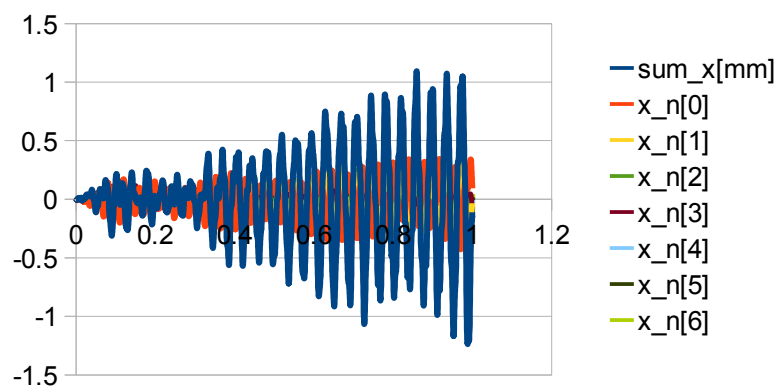
下記の画面が現れたら必要な行を選択し、グラフボタンをクリックします。このとき、時系列で表示するためには、必ず A 列を選択してください。



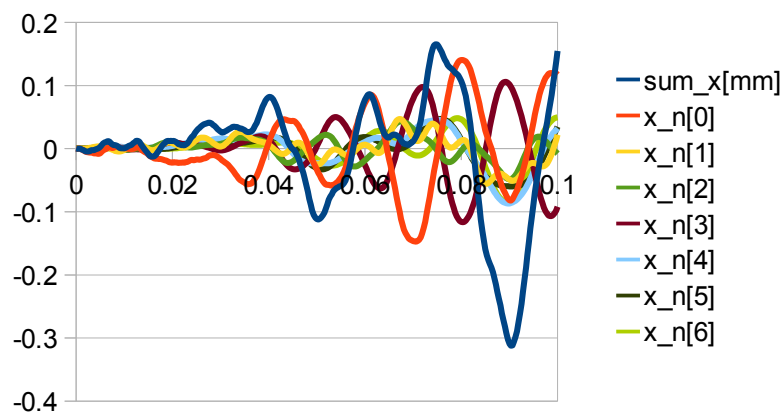
下記の画面が現れたら、『散布図』、『線のみ』を選択して完了ボタンをクリックします。



最初に、下記のような時系列グラフが現れます。



このままでは見にくいので、X 軸を適当なレンジに設定して見たい部分を表示してください。



詳細は、LibreOffice のヘルプ等を参照してください。

プログラムのバグを発見した場合には、下記までレポートをください(義務ではありません)。
mcapspeakers@gmail.com

以上