

数値計算ソフトウェア

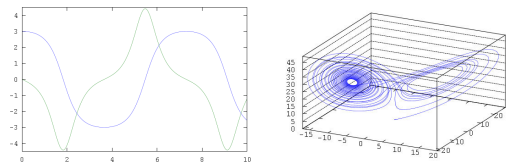
1

背景

機械技術者は様々な力学現象を扱う。この現象を表す解が要求された場合、まずは解析解（厳密解）を求める。しかし、一般には困難である。解析解が得られない場合には、工学的に許容される数値近似解を求めることになる。

目標

様々な数値計算を行ない、汎用数値計算ソフトウェアを活用できるようにする。



Octave

2

数値計算	数値近似解	<ul style="list-style-type: none"> 行列演算 非線型方程式 常微分方程式 統計解析 制御系設計
グラフ表示	可視化	gnuplotなどの外部ソフトウェアを用いて表示
プログラム	バッチ処理	if, for, whileなどの制御構造

代表的な汎用数値計算ソフトウェア

商用	Matlab
非商用	Octave
	Scilab

- 豊富な関数群
- ビジュアル化による直感的理解
- 高度な演算処理

OctaveとMaximaの特徴的な相違点

3

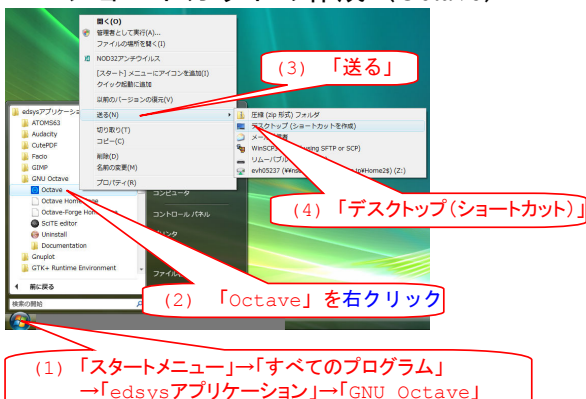
	Octave	Maxima
処理方式	数値計算	数式処理
解	数値近似解	解析解（厳密解）
値の型	実数型	整数型 実数型
グラフ	数値データ	関数
備考	複雑なシステム 大規模なシステム	性質の良い問題

応用数値計算に関する専用ソフトの例

気象予測、衝突解析、応力解析、流体解析、伝熱解析

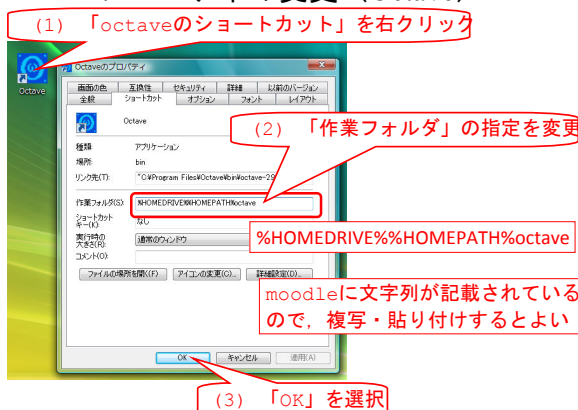
ショートカットの作成 (Octave)

4



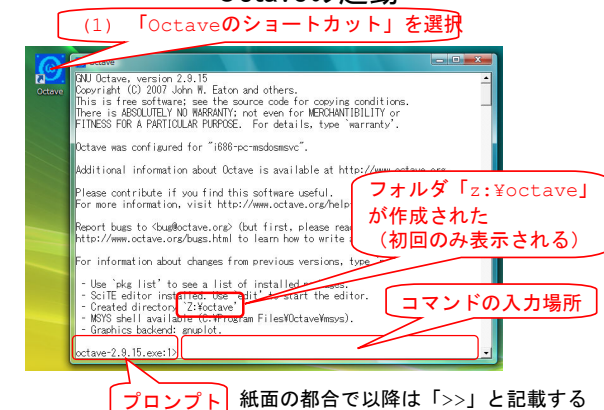
プロパティの変更 (Octave)

5



Octaveの起動

6



Octaveの終了

7

	算術演算子	意味
終了	quit	octaveの終了
	exit	octaveの終了
ヘルプ	help	コマンド等の一覧
	help command	指定commandの解説表示

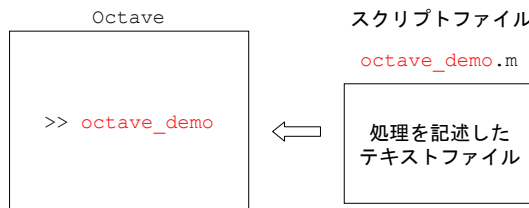
(注意) Octave Windowの強制終了ではエラーが発生する。quitあるいはexitにより終了すること。

demoプログラムの実行

8

demoファイルを実行確認しない。

- moodleの指定場所から、圧縮ファイル「octave_sample.zip」をDownloadし、展開する。
- ファイル「octave_demo.m」「example.m」を、フォルダ「z:\%octave%」に保存する。
- Octaveで「octave_demo」と入力し、実行する。



基本的な処理

- 操作方法
- 演算子
- 関数

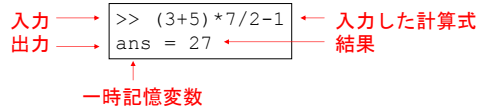
操作方法 (Operation)

10

入力方法の例



結果:



ansを計算に利用できる

```
>> ans * ans
ans = 729
```

大文字と小文字を区別

算術演算子(Arithmetic operator)

11

算術演算子	意味	書式	例	結果
+	和	x+y	5+3	8
-	差	x-y	5-3	2
*	積	x*y	5*3	15
/	商	x/y	5/3	1.6667
^	冪	x^y	5^3	125
**	冪	x**y	5**3	125

Octaveは、整数、小数などの数値を全て倍精度浮動小数点の形式(近似値)で記憶、計算する

注意: 打ち切り誤差, 丸め誤差に注意
計算方法により結果が異なることがある

数学定数 (Constants)

12

定数	意味	値, 備考
pi	円周率	3.14159265358979
e	ネイピア数 (自然対数の底)	2.71828182845905
i, j, I, J	虚数単位	0+1i
eps	計算機の精度	2.22044604925031e-16
realmax	記憶可能な最大実数	1.79769313486232e+308
realmin	記憶可能な最小実数	2.22507385850720e-308

Maximaとは異なり,
「pi」, 「e」に「%」を付けない。

数学関数 (Mathematical Functions)

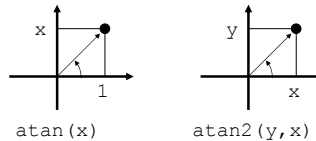
13

書式	意味	計算例	結果
abs(x)	絶対値	abs(-3)	3
sqrt(x)	平方根	sqrt(3^2+4^2)	5
exp(x)	指数関数	exp(2)	7.3891
log(x)	自然対数	log(e^2)	2
log10(x)	常用対数	log10(100)	2
ceil(x)	切り上げ	ceil(2.5)	3
sign(x)	符号	sign(-3)	-1

三角関数 (Trigonometric Functions)

14

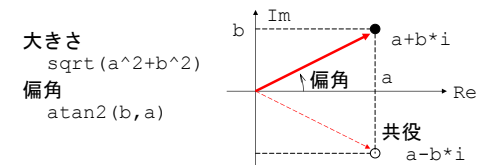
書式	意味	引数	戻り値
sin(x)	正弦	x[rad]	-1~1
cos(x)	余弦	x[rad]	-1~1
tan(x)	正接	x[rad]	inf~-inf
asin(x)	逆正弦	-1~1	-pi/2~pi/2
acos(x)	逆余弦	-1~1	0~pi
atan(x)	逆正接(2象限)	inf~-inf	-pi/2~pi/2
atan2(y,x)	逆正接(4象限)	inf~-inf	-pi~pi



複素数(Complex Number), 複素関数

15

書式	意味	計算例	結果
a+b*i, a+bi	複素数	3+4*i	3+4i
real(z)	実部	real(3+4*i)	3
imag(z)	虚部	imag(3+4*i)	4
abs(z)	大きさ	abs(3+4i)	5
arg(z)	偏角	arg(1+i)	0.78540
conj(z)	共役	conj(3+4*i)	3-4i



変数の定義 (=)

16

定義方法

変数名=値あるいは計算式

```
例 >> x=5
x = 5
>> y=3
y = 3
>> x+y
ans = 8
```

(注) 変数は値を代入しなければ使えない。
Maximaとの特徴的な相違点。

```
>> z
error: `z' undefined ...
```

変数zには値が記憶されていないため、エラー発生

変数の削除 (clear)

指定した変数を削除	clear 変数
全ての変数を削除	clear

数値リスト

- ・ベクトル型
- ・行列型
- ・数値リストの演算

17

数値リスト

18

意味	書式
横ベクトル型	[x1,x2,x3,...,xn]
縦ベクトル型	[x1;x2;x3;...;xn]
行列型	[a11,a12,...,a1n; a21,a22,...,a2n;; am1,am2,...,amn]

区切り	意味
,	要素
;	行
空白	要素
改行	行

```
横ベクトル >> x=[5,2,7,4]
x =
5 2 7 4
```

```
行列 >> A=[1,2,3;4,5,6]
A =
1 2 3
4 5 6
```

```
縦ベクトル >> x=[4;2;3;6]
x =
4
2
3
6
```

数値リストの生成 (増加量指定)

19

横ベクトル型

書式	備考
初期値:増加量:範囲	・ 範囲内で打ち切り ・ 第2項に減少量を指定しても良い
初期値:範囲	増加量は1

処理例

```
>> 1:0.4:3
ans =
    1.0000    1.4000    1.8000    2.2000    2.6000    3.0000
```

```
>> 0:4
ans =
    0    1    2    3    4
```

数値リストの生成 (分割数指定)

20

横ベクトル型

分割形式	書式	備考
等間隔	linspace(a,b,n)	区間 [a,b] をn個に等分割. nの省略時はn=100.
対数	logspace(a,b,n)	区間 [10^a,10^b] について [a,b] をn個に等分割.

(注) 両端を含めてn個

```
>> linspace(0,pi,5)
ans =
    0.000000    0.785400    1.570800    2.356190    3.141590
```

数値リストの生成 (行列型)

21

要素の値	正方行列	m×n行列	例 (m=2, n=3)
全要素が0	zeros(n)	zeros(m,n)	[0,0,0; 0,0,0]
対角要素が1	eye(n)	eye(m,n)	[1,0,0; 0,1,0]
全要素が1	ones(n)	ones(m,n)	[1,1,1; 1,1,1]

対角行列	diag([x1,x2,...,xn])	[x1, 0,..., 0; 0,x2,..., 0; 0, 0,...,xn]
------	----------------------	---

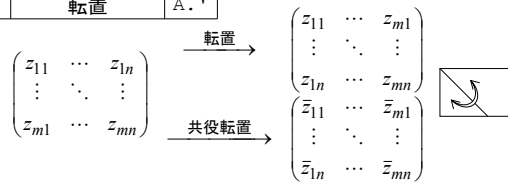
行列の演算

22

演算子	意味	数式
+	和	A+B
-	差	A-B
*	積	A*B
*	スカラーとの積	p*A
^	冪	A^p
'	複素共役転置	A'
.'	転置	A.'

A, B: 行列
p: スカラー

複素数 $z = x + yi$
共役複素数 $\bar{z} = x - yi$



行列の演算関数 (長方形)

23

意味	書式	計算結果
転置	transpose(A)	[1,4; 2,5; 3,6]
行列のサイズ	size(A)	[2,3]
行のサイズ	rows(A)	2
列のサイズ	columns(A)	3
階数	rank(A)	2

```
A=[1,2,3;  
4,5,6]
```

行列の演算関数 (正方行列)

24

意味	書式	
行列式	det(A)	A
逆行列	inv(A)	A ⁻¹
対角要素の和	trace(A)	a ₁₁ + a ₂₂ + ... + a _{nn}
固有値分解	[v, lambda]=eig(A)	Av _i = λ _i v _i (i=1,2,...,n)

[[v₁, ..., v_n], diag[λ₁, ..., λ_n]]

固有値と固有ベクトル

```
A=[1,2;3,4];
[v, lambda]=eig(A)
```

$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$

```
v=
    -0.82456   -0.41597
     0.56577   -0.90938
lambda=
    -0.37228    0.00000
     0.00000    5.37228
```

線型連立方程式

25

次の線型連立方程式の解を求めなさい。

$$\begin{cases} x+2y=3 \\ 3x+4y=7 \end{cases}$$

行列を用いて、係数と未知数を分離する。

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 7 \end{pmatrix} \quad Ax = b$$

逆行列を用いて解を求める

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}^{-1} \begin{pmatrix} 3 \\ 7 \end{pmatrix} = \begin{pmatrix} -2 & 1 \\ 3/2 & -1/2 \end{pmatrix} \begin{pmatrix} 3 \\ 7 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad x = A^{-1}b$$

```
A=[1,2;3,4];b=[3;7];
x=inv(A)*b
```

$x = \begin{pmatrix} 1.00000 \\ 1.00000 \end{pmatrix}$

リストの積, 追加

26

処理	式	結果
ベクトルの積	u*v'	32
	u'*v	[4, 5, 6; 8,10,12; 12,15,18]
ベクトルの追加	[u,w]	[1,2,3,4,5,6]
	[u;w]	[1,2,3; 4,5,6]
拡大係数行列	[A,b]	[1,2,5; 3,4,6]

```
u=[1,2,3]
v=[4,5,6]
```

```
A=[1,2;3,4]
b=[5;6]
```

要素毎の演算

27

演算子	意味	数式	計算結果
.*	積	C=A.*B	cij=aij*bij
./	商	C=A./B	cij=aij/bij
.^	冪	C=A.^B	cij=aij^bij
f()	関数	C=f(A)	cij=f(aij)

A, Bは
同サイズの
数値リスト

計算例

```
>> x=[3,4,5,6], y=[8,6,4,2]
x =
     3     4     5     6
y =
     8     6     4     2

>> x .* y
ans =
    24    24    20    12

>> x ./ y
ans =
    0.37500    0.66667    1.25000    3.00000
```

```
A=[3,4;5,6];
B=[8,6;4,2];
A = (3 4)
    (5 6)
B = (8 6)
    (4 2)
```

要素毎の関数演算

```
>> sqrt(A)
ans =
    1.7321    2.0000
    2.2361    2.4495

>> sin(B)
ans =
    0.98936   -0.27942
   -0.75680    0.90930
```

要素毎の積

```
>> sqrt(A) .* sin(B)
ans =
    1.71362   -0.55883
   -1.69226    2.22731
```

(参考) 通常の行列の積

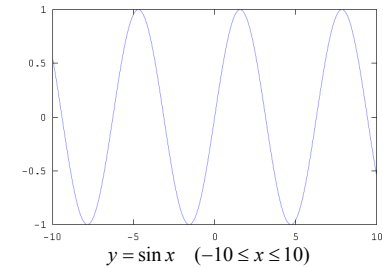
```
>> sqrt(A) * sin(B)
ans =
    0.20001    1.33463
    0.35849    1.60252
```

グラフ

- ・ 2次元曲線 (plot)
- ・ 3次元曲線 (plot3)
- ・ 3次元網目 (mesh)

```
2次元平面内に曲線を描画する
plot([x1,...,xn],[y1,...,yn])
```

```
x=-10:0.1:10;
plot(x, sin(x));
```

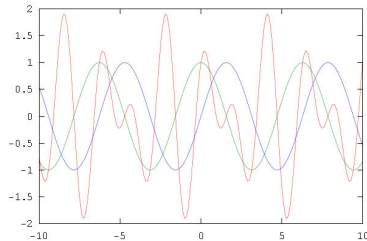


2次元グラフ (重ねて描画)

```
plot(x_list,[y1_list;...;ym_list])
```

```
x=-10:0.1:10;
plot(x,[sin(x);cos(x);sin(2*x)+cos(3*x)]);
```

$$y = \begin{cases} \sin x \\ \cos x \\ \sin 2x + \cos 3x \end{cases} \quad (-10 \leq x \leq 10)$$



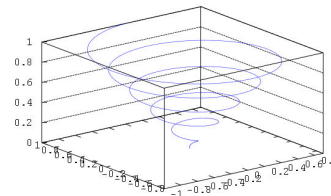
3次元グラフ (plot3)

3次元空間内に曲線を描画する

```
plot3([x1,...,xn],[y1,...,yn],[z1,...,zn])
```

```
t=linspace(0,10,300);
plot3(t.*sin(pi*t),t.*cos(pi*t),t);
```

要素毎の積
に注意



$$(x, y, z) = (t \sin \pi t, t \cos \pi t, t) \quad (0 \leq t \leq 10)$$

3次元グラフ (mesh)

網目状に描画する

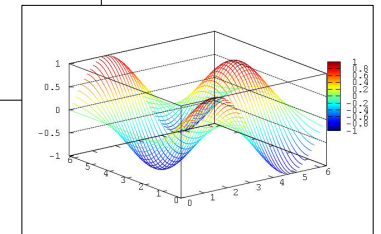
```
mesh(
[x1,...,xn],[y1,...,ym],
[z11,...,z1n;...;zml,...,zmn]
)
```

座標の意味

	[x1,...,xn]
[y1,	[z11,...,z1n;
...];
ym]	zml,...,zmn]

$$z_{ij} = f(x_j, y_i)$$

```
x=y=linspace(0,2*pi,50);
sx=sin(x);
cy=cos(y);
z=cy' * sx;
mesh(x,y,z);
```



$$z = \sin x \cos y \quad (0 \leq x \leq 2\pi, 0 \leq y \leq 2\pi)$$

作業フォルダ

意味	例
path	探索フォルダの一覧
addpath(path)	探索フォルダへpathを追加
rmpath(path)	探索フォルダからpathを削除
genpath(path)	pathのサブフォルダを抽出
dir	作業フォルダのファイル一覧
cd	作業フォルダをユーザーフォルダへ変更
cd dir	作業フォルダをdirへ変更

スクリプトファイル

	注意点
拡張子	m
保存場所	現在の作業フォルダ, あるいはpathで一覧されるフォルダ
記載内容	対話形式で入力する手順と同じ内容. ただし, 最初にダミー命令を記載する.
実行方法	Octave上でファイル名を入力する.

スクリプトファイルの例
example.m

```
1;
x = 0:0.1:10;
y = sin(x);
plot(x,y)
```

← ダミー命令

対話形式で入力する手順と同じ

入出力関数

書式	意味	使用方法	
disp(arg)	画面表示	disp(x)	値xの表示
		disp('msg')	文字列の表示
input('msg')	キーボード入力	a=input('msg')	値の入力

表示桁数 (format)

```
デフォルトの表示桁数 { >> pi
                        pi = 3.1416
表示桁数の変更         >> format long
                        >> pi
内部の演算桁数         pi = 3.14159265358979
表示設定を戻す         >> format
```

有効数字は15桁程度

命令の区切り

37

区切り	意味	結果の画面表示
改行	次の行に分ける場合	する
,	1行に続ける場合	する
;	1行に続ける場合	しない

改行の場合

```
>> x=5
x = 5
>> y=3
y = 3
>> x+y
ans = 8
```

「,」の場合

```
>> x=5,y=3,x+y
x = 5
y = 3
ans = 8
```

各処理の結果が
画面表示される

「;」の場合

```
>> x=5;
>> y=3;
>> x+y;
```

```
>> x+5;y=3;x+y;
```

画面表示が抑制される

Octave

38

管理者	John W. Eaton
対応OS	Linux, Windows, MacOS,...
記述言語	C++, Standard Template Library (STL)
ライセンス	GNU GPL

1988年頃	化学反応装置に関する学部用教科書と共に使用するソフトウェアを意図して, James B. Rawlings, Jhon W. Eatonにより作成された
1992年	より柔軟な数値計算ツールとして本格的に開発
1994年	Ver1.0がリリースされる
1997年	GNU GPLに移行し, <u>自由に使える</u> ようになる.
現在	現実的な問題を解くために, 多様な分野の講義で利用されている

<http://www.gnu.org/software/octave/>

参考文献

39

- (1) John W. Easton: GNU Octave Manual
- (2) Octave Quick Reference
- (3) 北本: Octaveを用いた数値計算入門, ピアソン・エデュケーション
- (4) 吉田和信: Octaveによる動的システムシミュレーション入門, <http://www.ecs.shimane-u.ac.jp/~kyoshida/octave.htm>
- (5) 松田七美男: Octaveの概要, <http://ayapin.film.s.dendai.ac.jp/~matuda/TeX/lecture.html>
- (6) 北野正雄: 関数電卓とOctaveのすすめ, <http://ocw.kyoto-u.ac.jp/jp/common/course24/pdf/octave.pdf>