

# フリーソフトを用いたMBD

大野修一

広島大学

Dec. 10, 2019

- Matlab (1984 ver.1.0, 現在 ver. 9.7)
  - <https://jp.mathworks.com/products/matlab.html>
  - <https://ja.wikipedia.org/wiki/MATLAB>
- GNU Octave (1993 ver. 1.0 , 現在 version 5.1.0)
  - <https://www.gnu.org/software/octave/>
  - [https://ja.wikipedia.org/wiki/GNU\\_Octave](https://ja.wikipedia.org/wiki/GNU_Octave)
- Scilab (1994 ver.1.1, 現在 ver. 6.0.2)
  - <http://www.scilab.org/>
  - <https://ja.wikipedia.org/wiki/Scilab>

## 特徴

- Matlab との互換
- Toolbox の関数を部分的に実装
- help は英語
- Simulink に対応するものは無

# インストール

- java runtime environment **が必要**  
<https://java.com/ja/download/>
- ホームページよりダウンロードしてインストール  
<https://www.gnu.org/software/octave/>
- パッケージ (toolbox)  
octave forge <https://octave.sourceforge.io/>

```
pkg install -forge control % インストール  
pkg load control % control のロード  
pkg load all % 全てのロード
```

ロードのコマンドを `startup.m` に書けば起動時に自動的にロード

# Matlab との主な違い

[https://wiki.octave.org/FAQ#How\\_is\\_Octave\\_different\\_from\\_Matlab.3F](https://wiki.octave.org/FAQ#How_is_Octave_different_from_Matlab.3F)

- 行頭に % の他に # で、その行をコメント
- ++, --, +=, \*=, /= などの C 言語の演算子
- ^ だけでなく \*\*
- ~= だけでなく !=
- ' の他に、" を使っても文字列を定義
- xx ... end だけでなく xx ... endxx  
たとえば if ... endif, for ... endfor
- Line continuation
- コマンドラインの ! は、文字列
- 図をセーブしたときの拡張子 .fig ではなく .ofig

## 特徴

- Matlab との非互換  
Matlab へのコンバータがあるが完全ではない
- 独自の Toolbox
- help の主要部分は日本語化

# インストール

- ホームページよりダウンロードしてインストール  
<http://www.scilab.org/>
- モジュール管理 (ATOMS)
- Simulink に似た Xcos

# 起動と初期設定

(起動時に [Scilab 5] が書いてあるロゴが消えない場合は再起動)

- コンソール
- プロンプト
- コマンドライン
- スタートアップファイル

SCIHOME/.scilab および SCIHOME/scilab.ini は, (もし存在する場合,) ファイル SCI/etc/scilab.start

- SCIHOME と SCI

- 1 pwd
- 2 cd SCIHOME
- 3 ls
- 4 cd SCI
- 5 pwd, ls
- 6 cd SCIHOME



- Windows で SciNotes の日本語文字化けは、たとえば「設定」の「フォントを指定」で日本語のフォントを指定
  - コメント
  - スクリプトの拡張子 `sce`, 関数の拡張子 `sci`
- ① 「アプリケーション」の SciNotes
  - ② 「ワーキングディレクトリ」(Working directories) の「ディレクトリを追加」(Add a directory)
  - ③ 「ファイルをオープン」(Open file in), 「ファイルをセーブ」(Save file in)  
フォルダは作成できるが削除できない。
  - ④ `exit`, `start`

主要部分は日本語化

# Scilab と Matlab との違い

- 特殊変数 (4.4 節 : p. 37~) は変更不可  
Scilab の定義済み定数名は、通常、% で始まります。定義済み定数を表 1 に示します。

Table: 特殊定数

変数	意味
%eps	$1 + \%eps = 1$ となる最も大きい値
%inf	無限大
%nan	値なし
%i	虚数単位 $\sqrt{-1}$
%pi	円周率 $\pi = 3.1415927\dots$
%e	ネイピア数 (自然対数の底), $e = 2.7182818\dots$
%t(%T)	真 (true)
%f(%F)	偽 (false)

- 配列の終わり : \$ (p.55) と end(Matlab)

```
-->A=[1,2,3;4,5,6;7,8,9]
```

```
A =
```

```
!   1.   2.   3. !
```

```
!   4.   5.   6. !
```

```
!   7.   8.   9. !
```

```
-->A($)
```

```
ans =
```

```
9.
```

- 変数の入れ替え

```
x=1;y=2;
```

```
[x,y]=(y,x)
```

- 注意

```
-->x= 1./[1 2 3]
```

```
x =
```

```
0.0714286
```

```
0.1428571
```

```
0.2142857
```

```
1
```

```
-->x=1 ./[1 2 3]
```

```
x =
```

```
1. 0.5 0.3333333
```

- 同じ関数名で結果の異なるものの一部

- int8
- max min (p.81)

```
-->max(A),max(A,'c'),max(A,'r')
```

```
ans =
```

```
9.
```

```
ans =
```

```
3.
```

```
6.
```

```
9.
```

```
ans =
```

```
7.
```

```
8.
```


```
9.
```

- gsort
- fft

- 図はデフォルトで上書き (p.124)  
図を上書きしないモードに変更するには  
-->da=gda();da.auto\_clear = 'on';  
とします. 逆に, 上書きモードに変更するには  
-->da=gda();da.auto\_clear = 'off';  
とします. なお, 上の da は変数であり, 別の名前でもかまいません.
- 新しい Figure Window は, scf(n)

Xcos は GUI を用いてシステムを定義しシミュレーションを行うことができる Scilab アプリケーション

## Xcos の起動

- Scilab コンソールメニューの「アプリケーション」の「Xcos」を指定
- アイコン  をクリック

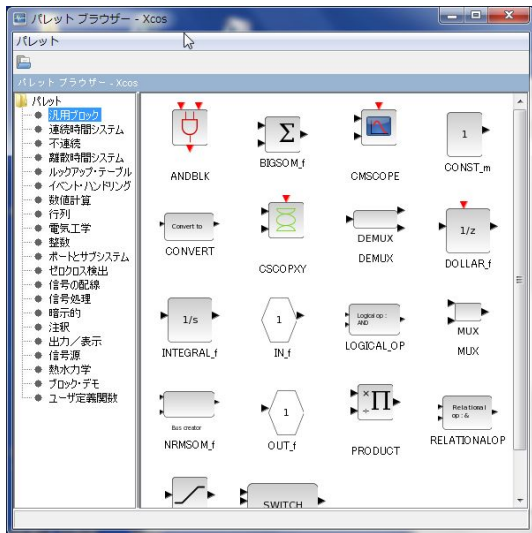
通常、画面左側パレットブラウザ、右側エディタ



# Xcos と Simulink との主な違い

- パレットブラウザー (Simulink ライブラリブラウザー)
- アクティベーションの概念
  - 上側にアクティベーション入力を持つブロックが存在
  - 下側にアクティベーション出力を持つブロックが存在
- ブロックを右クリックすると Block help (Simulink ダブルクリック)
- ブロックの接続方法
- 引き出し点の作成方法
- Modelica が利用可

# パレットブラウザ



# ブロック

入力ポート, 出力ポート, アクティベーション入力ポート, アクティベーション出力ポート

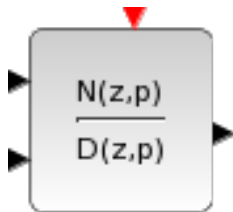
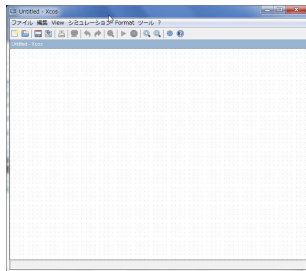


Figure: DLRADAPT\_f

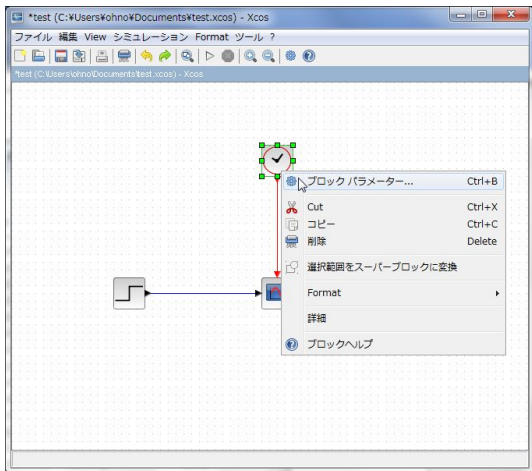
# エディタ



- 1 パレットブラウザでブロック上で左クリックし、そのままダイアグラムへドラッグしドロップ
- 2 パレットブラウザでブロック上で右クリックすると、「追加」のプルダウンメニューがあるのでコピーしたいダイアグラムを選択

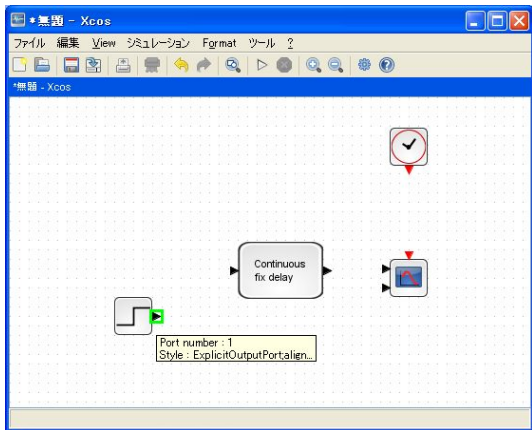
# ブロックパラメータ

ブロックを右クリックすると、パラメータ変更



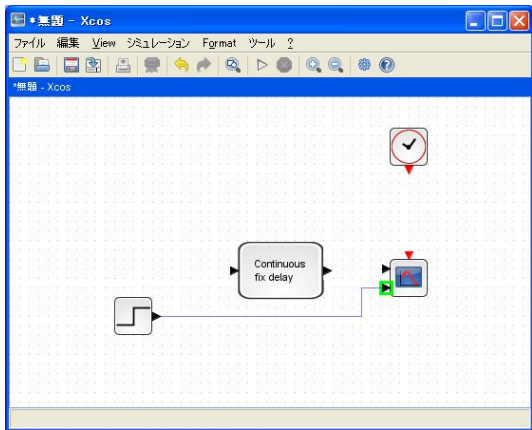
# ブロックの接続方法

ポートを右クリックし、緑の□から右クリックしたままドラッグ



# ブロックの接続方法

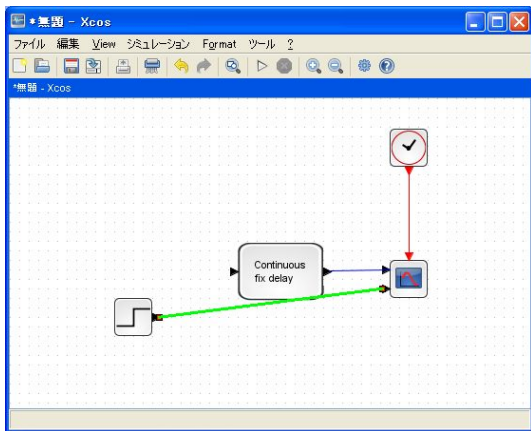
接続したいポートまでドラッグし、緑の□でドロップままだらッグ



折点が必要な場所ではクリック (既存の線に折点を追加したい場合はその場でダブルクリック)

# 引き出し点の作成方法1

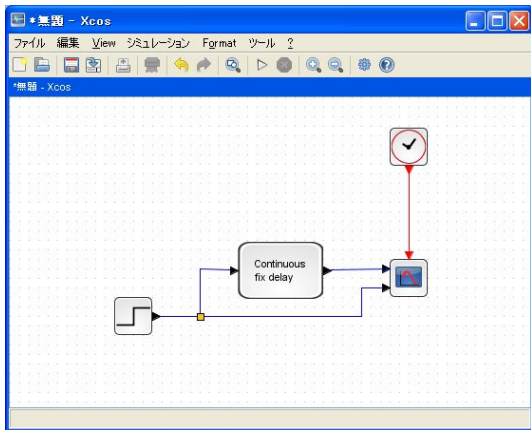
引き出し点をとりたい線をマウスで指定すると線が緑に変化



クリックで引き出しを開始し、そのままドラッグしポートでドロップで接続



## 必要に応じて整形



# 引き出し点の作成方法2

ポートから引き出し点を取りたい線をマウスで指定すると線が緑に変化. クリックで引き出し点を作成. 必要に応じて整形.

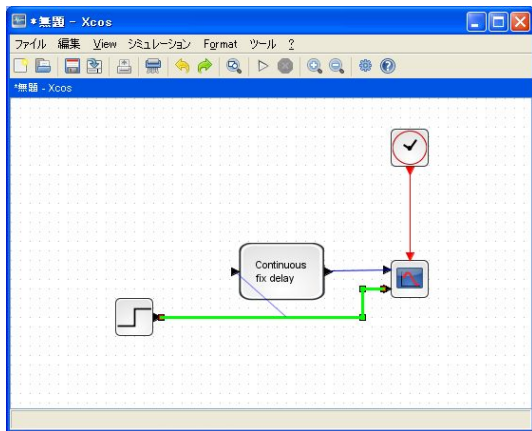


Figure: 引き出し点の作成方法2

- ダイアグラムは保存可能
- ダイアグラムには拡張子 `xcos`
- 「エクスポート」はダイアグラムを指定する画像フォーマットで出力

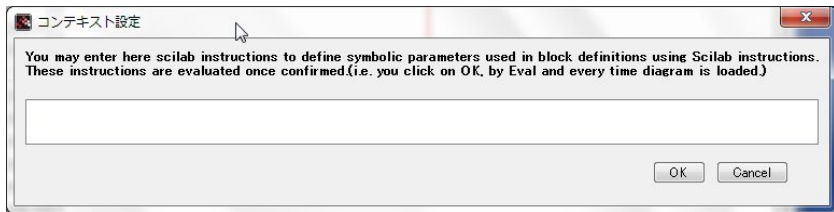
# シミュレーション

- シミュレーション「開始」
- シミュレーション「停止」
- 「設定」シミュレーションの終了時間，誤差，時間間隔



# コンテキスト

各ブロックから参照できる変数を設定。たとえば、ブロックの共通パラメータとして利用。



# スーパーブロック

複数のブロックから構成されたブロック

- 左クリック・ドラックで範囲を選択
- 右クリックあるいは編集メニューで「スーパーブロックへのセレクション」

- <https://www.modelica.org/>
- 非営利国際組織の Modelica 協会
- Xcos では Modelica を用いるブロック (Modelica Block)
- Fortran または C++ コンパイラ