制御系解析・設計支援ツール

A New Interactive CAD based on Window System

東京工業大学　古田 勝久　◯古賀 雅伸

Tokyo Institute of Techonology　K.Furuta and M.Koga

概 要

We developed two useful tools for analyzing and designing control systems which relieve the users of unnecessary tedious or low-level tasks. One of them is very useful a drawing environment, "**Xplot**", which runs under the X Window System and is portable to other machines. Under the Xplot all views of the same data share the common memory area. The other is a simulation language, "**MaTX**", which allows the users to describe matrix and scalar operations equally and to treat real and complex values without distinction.

1　はじめに

近年のＣＡＤシステムの開発に関して、機能の統合化が進む一方[1][2]、システムの複雑さが増し簡単なアルゴリズムの検証にも時間がかかる、ユーザによる機能の追加が難しい、ハードウェアや言語に依存しているため移植が困難であるといった問題が生じている。

著者らは、統合化された豊富な機能を持つＣＡＤシステムの開発を目指す一方、「設計者を支援する。」という制御系解析・設計支援システムの目的を達成するため、解析・設計に要する時間を大幅に短縮できるツール郡を開発した。制御系の解析・設計の際に避けられない大量の行列演算を簡単に行うために「シミュレーション支援言語ＭａＴＸ」を開発した。ＭａＴＸを用いることにより従来の数分の一の時間でシミュレーションを行うことができる。また、シミュレーシン結果を解析するために「シミュレーション結果表示環境Ｘｐｌｏｔ」を開発した。Ｘｐｌｏｔを用いることにより簡単にシミュレーション結果をさまざまな形で表示することができる。同じデータが重複して存在することがなく、１つのデータを複数のグラフが参照するようになっているのでメモリーの使用効率が良い。

2　ＭａＴＸ

2.1　MaTX の特徴

- 紙の上に記述した式をそのままプログラムとして使用することができる。
- 複素数と実数を区別なく扱うことができる。
- 一度記述した式には、値と定義が同時に割り付けられるので、同じ式を何度も記述する必要がない。
- Matrix Editor を用いることにより、インターラクティブに行列を入力・修正することができる。

2.2　MaTX の構成

インタープリタ (matx)

1. 任意の大きさの行列を扱うことができ、変数は、型宣言をする必要がない。
2. スカラーの四則演算と、行列の四則演算は区別する必要がない。
3. 行列を定義する時、変数を成分として使用できる。
4. Ｃ言語と同じ制御フローや関係演算子を使うことができる。

コンパイラー (matc)

Ｃ言語のコードを出力させることができ、ＭａＴＸのソースコードをコメントとして、出力させることができる。一度記述した式には、値と定義が同時に割り付けられるので、同じ式を何度も記述する必要がない。

ライブラリー (matlib)

1. メモリ割り付けを自動的に行なう高レベルのライブラリと、メモリ割り付けはユーザに任されている低レベルのライブラリが用意されている。
2. 関数が自動的に実行列と複素行列を区別して処理する。
3. 行列を名前によって参照することができる。

エディター (mated)

1. 任意の大きさの行列をあつかうことができ、大きさを任意に拡大・縮小できる。
2. カーソルを移動することにより、任意の成分を入力・修正でき, 行列が大き過ぎるときは、自動的にスクロールする。
3. 列成分 (行成分) を削除・移動・複写することができる。

2.3　例題

A, B, Q, R が定値行列で (A, B) が可到達、 $Q = C^T C$ とする時、 (C^T, A) が可観測ならば、 Riccati 代数方程式を満足する正定対称解が存在し、

$$\lim_{k \to \infty} P_k = P$$

となることが知られている。[3]

A 行列が正則である２次の離散時間系

$$x_{k+1} = \begin{bmatrix} 0 & 1 \\ -3 & 4 \end{bmatrix} x_k + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_k \ , \ y_k = \begin{bmatrix} 0 & 1 \end{bmatrix} x_k$$

に対して、 $R = 1$ の場合の最適制御則を求めるために、Riccati 方程式の解 P を求めるプログラムをＭａＴＸを用いて記述例を Fig.1 に示す。

```
func void main()
{
 Matrix A, B, C, Q, R, P, P2;

 A = [[0 1][-3 4]];
 B = [0  1]';
 C = [-2 1];
 Q = C'*C;
 R = I1;
 P = Z2;
 P2 = P;
 while(1) {
  P = Q+A'*P*A-A'*P*B*(R+B'*P*B)~*B'*P*A;
  P = (P' + P)/2;
  if (||P - P2|| < 10E-7) break;
  P2 = P;
 }

 print P;
}
```

Fig. 1 Example Program

3 Ｘｐｌｏｔ

3.1 Xplot の構成

CAD システムを開発するために、今回は X Window システムを採用した。本研究では、新しく **Canvas Widget Class** と、**Data Information Widget Class** という Meta Class を作成する。前者のサブクラスとして、1D-Plot Widget Class, 2D-Plot Widget Class, 3D-Plot Widget Class, SemiLog Widget Class 後者のサブクラスとして、1D Data Information Widget Class, 2D Data Information Widget Class 3D Data Information Widget Class, SemiLog Data Information Widget Class を作成した。

Canvas Widget Class

Canvas Widget Class は、サイズを任意に変更することができ、隠れていた部分が再び現われた時、再表示を行なう。ウィンドウ内の任意の場所に Line, Box, Cirle, FilledBox, FilledCircle, Comment 等のオブジェクトを描くことができ、不必要になったオブジェクトはイレーザーによって削除することができる。

Data Information Widget Class

このクラスは、様々な情報の表示や、ユーザーによる入力のために使われる。データ表示部は、 Data Label 部と Data Value 部に分けてテーブル状に、管理されている。データが多過ぎて、ウィンドウ内にデータを表示できないときは、データ表示部が自動的に Scroll Window になり、 Scroll Bar をマウスで動かすことにより任意のデータを表示できる。

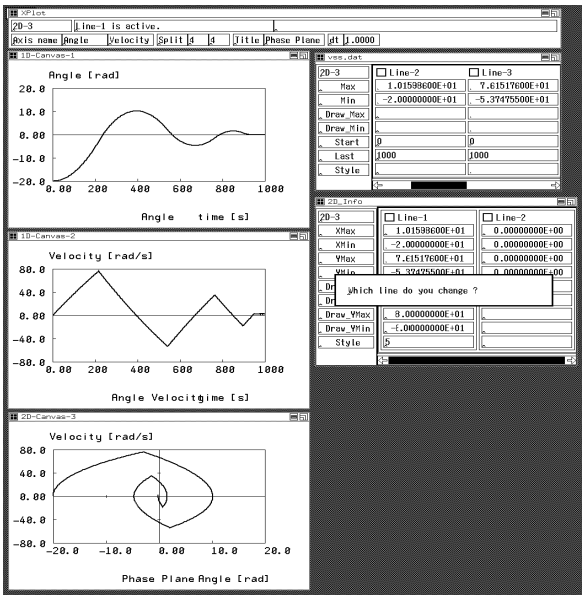

Fig. 2 1D-Plot Widget, 2D-Plot Widget

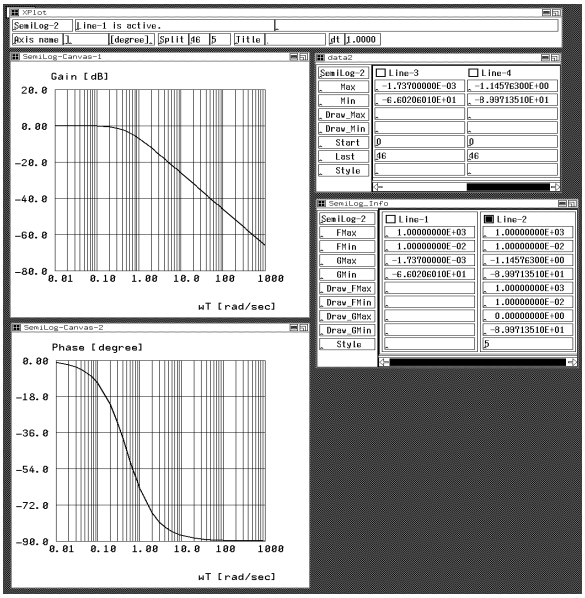

Fig. 3 SemiLog Widget

3.2 Xplot の概観

Xplot の概観を Fig.2, Fig.3 に示す。

4 おわりに

「設計者を支援する。」という制御系解析設計支援システムの目的を達成するため、解析設計に要する時間を大幅に短縮できるツールの開発を行った。制御系の解析設計の際に避けられない大量の行列演算を簡単に行うために「シミュレーション支援言語ＭａＴＸ」を開発した。また、シミュレーション結果を解析するために「シミュレーション結果表示環境Ｘｐｌｏｔ」を開発した。

5 参考文献

[1] 古田 勝久: 制御系ＣＡＤの現状と動向, 電気学会雑誌,106-2,pp.105-112,(1986)

[2] 赤掘 一郎, 原 辰次: オブジェクト指向に基づく制御系の解析・設計支援システム, 計測自動制御学会論文集,Vol.24, No.5, pp.506-513, (1989)

[3] 古田 勝久: ディジタルコントロール,131-143, (1989)